CHAPTER 5
VALIDATION AND EVALUATION OF THE METHODOLOGY WITH SIMULATIONS

5.1 INTRODUCTION

A simulation study was done in which the proposed closed-loop identification methodology, see Section 4.8, was successfully implemented. In this study a multivariable plant, controlled by an MPC controller, was identified from simulated closed-loop data. In order to evaluate the consistency of the identification methodology, the plant was identified for different settings in the controller, as well as for different added disturbances [46, 47]. Different methods for ensuring informative experiments were also considered.

In this chapter this simulation is described. First of all, in Section 5.2, the set-up, regarding the type of plant used, the chosen controller settings and all the cases considered, are discussed. Section 5.3 follows this discussion with an explanation of how each of the identification steps is implemented in MATLAB.

The obtained models are validated making use of the chosen methods described in Section 4.6. The validation process is also described in Section 5.3 in terms of the MATLAB commands. In Section 5.4 the expected validation results are discussed and the obtained results are given and also discussed. Finally, it is concluded in Section 5.5 that the proposed methodology can deliver consistent and satisfactory identified models of MIMO plants, controlled by MPC controllers.

5.2 SIMULATION SET-UP

5.2.1 Plant

The plant, which was used in the simulation, is a linear multivariable plant with two inputs and two outputs. The closed-loop configuration of the plant and controller is shown in Fig. 5.1. This plant, also given in Eqn. (5.1), is a benchmark example used in many of the MPC toolbox examples [29]. This is a scaled down version of what can be found in industry and it aids in demonstrating the basic identification steps of the proposed methodology for identifying multivariable plants, controlled by MPC controllers.

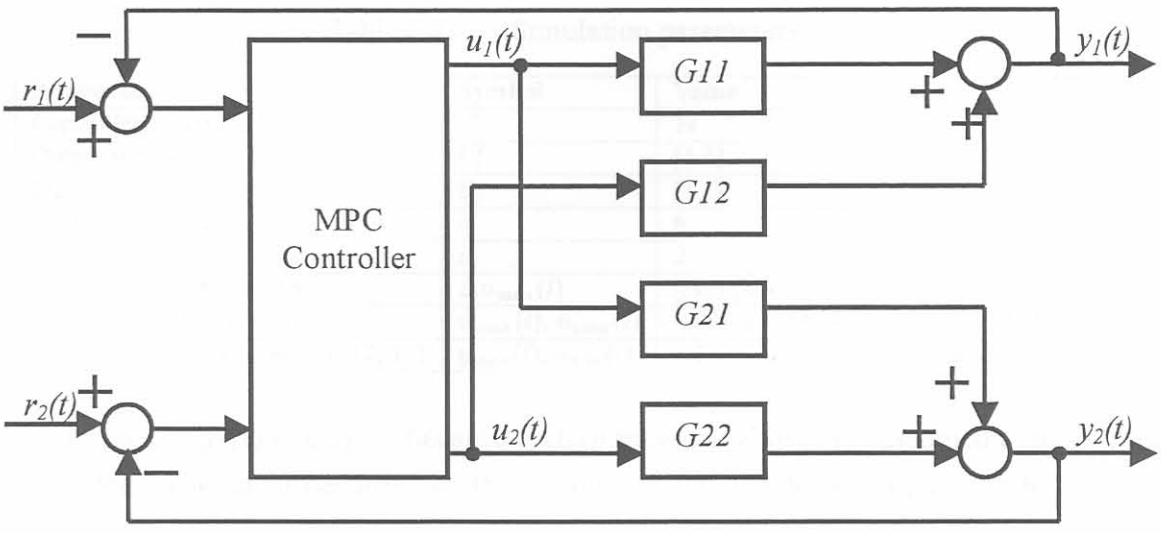

Figure 5.1: Closed-loop configuration of a multivariable plant controlled by an MPC controller.

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21.0s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \tag{5.1}$$

Since the MPC toolbox requires discrete-time models, the discrete step response model of the plant was obtained with the MPC toolbox function *tfd2step,* using a sampling time of 1min. The discretised MIMO transfer function was approximated with 90 step response coefficients.

5.2.2 Controller

For open-loop stable plants, the nominal stability of the closed-loop system depends on the prediction horizon h, the number of control moves i and the weighting matrices Γ_l^y and Γ_l^u. No precise conditions on h, i, Γ_l^y and Γ_l^u exist that guarantee closed-loop stability. In general, decreasing i relative to h makes the control action less aggressive and tends to stabilise a system. For $h = \infty$, nominal stability of the closed-loop system is guaranteed for any finite i, and time-varying input and output weights. More commonly Γ_l^u is used as a tuning parameter. Increasing Γ_l^u always has the effect of making the control action less aggressive [29].

In general, one must choose the horizons and weights by trial-and-error, using simulations to judge the effectiveness of these choices. In this simulation equal weightings were chosen

Table 5.1: Simulation parameters.

Parameter	Symbol	Value
Controller execution time	T	1s
Output weights	Γ_l^y	[1 1]
Input weights	Γ_l^u	[1 1]
Prediction horizon	h	6
Control moves	i	2
Saturation limits (cases 6, 9)	$\Delta u_{\max}(l)$	$\lvert\Delta u_1(k+l)\rvert \leq 0.1,\ \lvert\Delta u_2(k+l)\rvert \leq 0.05$
Input constraints (case 10)	$u_{\max}(l), u_{\max}(l)$	$-0.5 \leq u_i(k+l) \leq 0.5,\ i = 1,2$
Output constraints (cases 11, 12, 13)	$y_{\max}(l), y_{\min}(l)$	$-1.5 \leq \hat{y}_i(k+l) \leq 1.5,\ i = 1,2$

for $u_1(t)$, $u_2(t)$, $y_1(t)$ and $y_2(t)$, because each of these variables were assumed to be of equal importance. The equal penalties on the outputs reflects the desire to track both set-points accurately.

Because there is delay in the system, i was chosen to be smaller than h. This also makes the control action less aggressive and ensures stability. Because there was no *ringing*, i.e. damped oscillation, in the manipulated variables for the chosen parameters, it was not necessary to implement *blocking*. When blocking is implemented $\Delta u_i(t)$ is kept zero for a number of steps, e.g. $\Delta u_i(t)$ is kept zero for first two steps $u(k+2) = u(k+1) = u(k)$ [29].

Table 5.1 gives the final parameters chosen for the MPC controller. The MPC toolbox function *mpccon* was used for the design of the unconstrained controllers and the function *cmpc* was used for the design of the constrained controllers. These functions make use of quadratic optimisation, see Eqn. (2.1). When the function *mpccon* is used, the quadratic programme (QP) problem is solved analytically and this results in a linear controller. The *cmpc* function solves the QP problem iteratively, which results in a nonlinear controller. Furthermore, the default state estimate was used to calculate the controller gain matrix, since this results in a DMC controller [29].

5.2.3 Simulation Scenarios

Data sets from the following different cases, summarised in Table 5.2, were used for identification. In most of these cases, structured tests were performed by adding external test signals to the reference inputs. These signals are discussed in Section 5.3.1. In the other cases no reference inputs were added, i.e. $r_i(t) = 0$. Either no disturbance was added, or a unit disturbance was added at $t = 1$ and removed at $t = 2$ (pulse disturbance), or a constant disturbance of one unit was added from $t = 1$ onwards (step disturbance).

Unconstrained Control Law:

- case 1: no disturbances to the system,
- case 2: pulse disturbance added to $u_1(1)$,
- case 3: output pulse disturbance added to $y_1(1)$,
- case 4: step disturbance added to $u_1(1)$,
- case 5: output step disturbance added to $y_1(1)$,
- case 6: saturation limits on the manipulated variables,
- case 7: no disturbances to the system, **but with** $r_i(t) = 0$,
- case 8: output step disturbance added to $y_1(1)$, **but with** $r_i(t) = 0$,
- case 9: saturation limits on the manipulated variables and output step disturbance added to $y_1(1)$, **but with** $r_i(t) = 0$ **and the output inter-sampled**,

Constrained Control Law:

- case 10: enforced hard bounds on the manipulated variables
- case 11: enforced hard bounds on the output variables,
- case 12: enforced hard bounds on the output variables, **but with** $r_i(t) = 0$, and
- case 13: enforced hard bounds on the output variables with output step disturbance added to $y_1(1)$, **but with** $r_i(t) = 0$.

Note that the hard bounds are fundamentally different from the saturation limits. The hard bounds are defined relative to the beginning of the prediction horizon, which moves as the simulation progresses. Therefore, at each sampling period k, the hard constraints apply to a block of calculated moves that begins at sampling period k and extends for the duration of the input horizon i. The saturation constraints, on the other hand, are relative to the fixed point, $t = 0$, the start of the simulation [29].

The function *mpcsim* was used to simulate the different closed-loop responses for the unconstrained cases 1-9. In cases 10-13, where hard bounds were implemented, the function *cmpc* was used to generate the controllers and to simulated the closed-loop responses, because it determines optimal changes of the manipulated variables subject to constraints [29]. In all the cases a simulation time of 90s was used.

Cases 1-6 and 10-11 are used to evaluate the consistency of the proposed identification methodology. In these cases the plant was identified for different settings in the controller

Table 5.2: Case scenarios.

Case	Uncon-strained	Saturation Limits	Constrained	Input Disturbance	Output Disturbance	PE Reference Input	Inter-Sampled
1	√					√	
2	√			√		√	
3	√				√	√	
4	√			√		√	
5	√				√	√	
6	√	√				√	
7	√						
8	√				√		
9	√	√			√		√
10			√ (input)			√	
11			√ (output)			√	
12			√ (output)				
13			√ (output)		√		

(cases 1, 6, 10, 11), as well as for different added disturbances (cases 1-5). In these cases, **structured tests** were performed by adding external test signals to the reference inputs. These structured tests ensured good SNRs and PE reference signal, which in turn ensured identifiability. Here, again, SNR refers to the ratio between the noise and the plant input signal.

In cases 7-9, and 12-13 no structured tests were performed. The reference signals were zero and thus not PE. Here other methods to ensure identifiability are considered. In cases 7 and 8, no identifiability condition was satisfied: references were not PE; controllers were linear; and outputs were not inter-sampled. In case 9 the outputs were inter-sampled and the plant was thus identifiable. In case 12 and 13 the plant was also identifiable, since the controllers were constrained and thus nonlinear. In case 13 a disturbance was added, to evaluate the influence of the SNR with nonlinear feedback.

5.3 Implementation of the Methodology

The proposed methodology, as described in Section 4.8, was implemented in MATLAB. This implementation is discussed in terms of the five SID steps, as well as the methodology validation step.

5.3.1 Experiment Design

Signals to be Measured: The manipulated variables, $u_1(t)$ and $u_2(t)$, as well as the output variables, $y_1(t)$ and $y_2(t)$, were measured. These values were obtained as outputs of the *mpcsim* and *cmpc* functions.

Sampling Time: A sampling time of 1min, which is equal to the execution time of the controller, was used in cases 1-8 and 10-13.

The standard MPC toolbox function *mpcsim* was modified to allow inter-sampling of the output. This inter-sampling was implemented in case 9.

Excitation Signals: In cases 1-6 and 10-11 **structured tests** were performed by adding external test signals to the reference inputs, $r_1(t)$ and $r_2(t)$, to guarantee informative data. For each of these structured test cases two different trials were run where different test signals were used. In the first trial $r_1(t)$ was stepped at time zero and $r_2(t)$ was kept constant for 45s after which it was also stepped.

In the second trial a multivariable PRBS signal was generated making use of the MATLAB function *idinput*. The uncorrelated PRBS signals were added to the reference inputs. The period of these PRBS signals was taken as a tenth of the slowest time constant, which resulted in a period of 2min. The PRBS signals gave slightly better results than only step references. Therefore, the results obtained in the second trail of each case are discussed. A representation of the PE reference signals is given in Fig. 5.2. Some typical resulting input and output signals are given in Fig. 5.3.

In cases 7-9 and 12-13, where other methods to ensure identifiability are considered, **no structured tests** were performed and the reference inputs were kept at zero. In case 7, with the linear controller and no disturbances, the resulting inputs and outputs were zero. In cases 8, 9 and 13 where there were added disturbances, all the inputs and outputs look similar to Fig. 5.4. For case 12, with the nonlinear controller and no disturbances, the inputs and outputs are shown in Fig. 5.5.

5.3.2 Data Collection

Collection: The data collection was straightforward, since these values are the outputs that were generated by the *mpcsim* and *cmpc* functions.

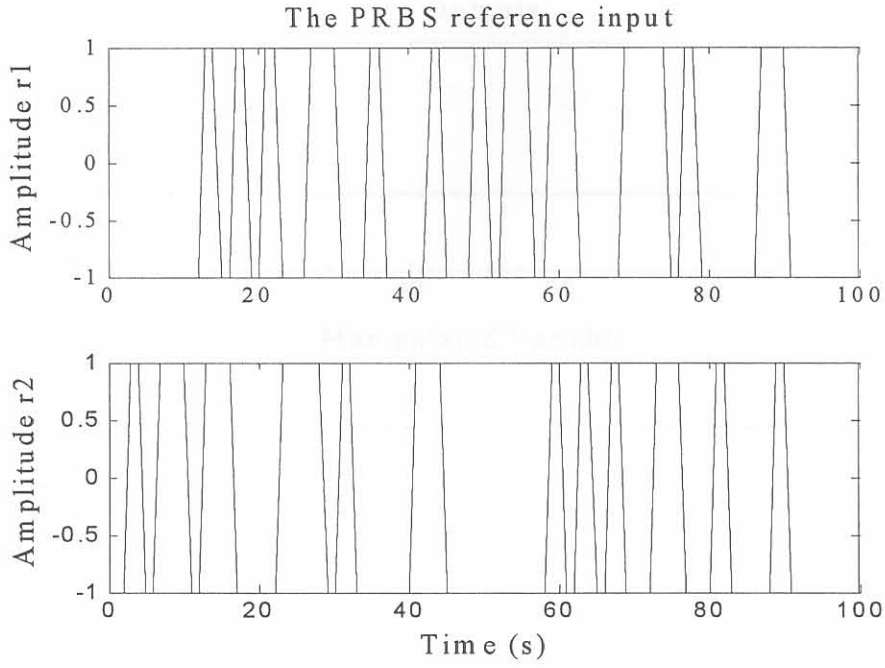

Figure 5.2: The persistently exciting PRBS reference signals used in trial 2.

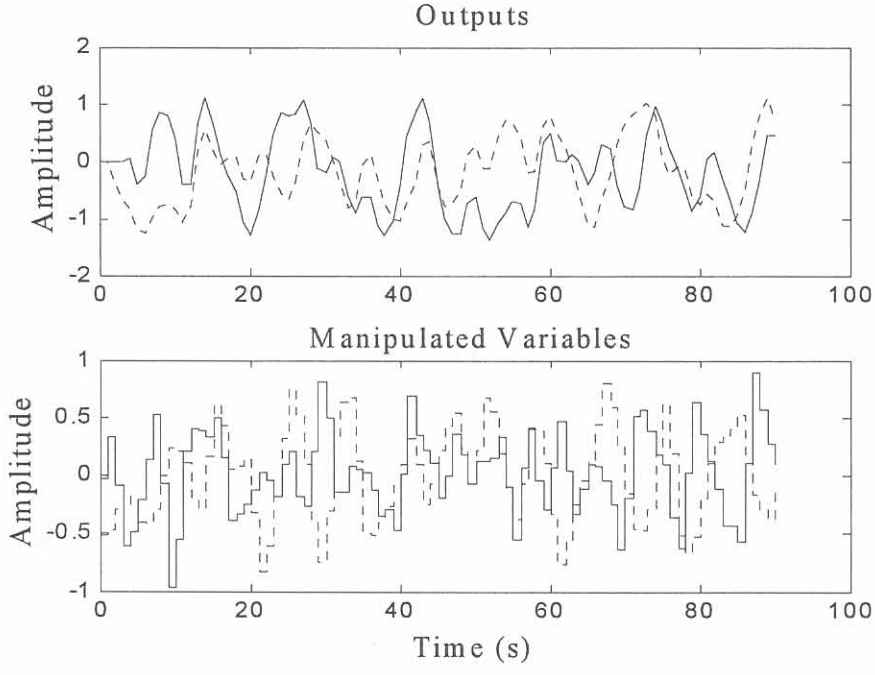

Figure 5.3: The resulting inputs and outputs in case 11 for a PRBS reference input. The dotted lines represent u_1 and y_1 and the solid lines represent u_2 and y_2.

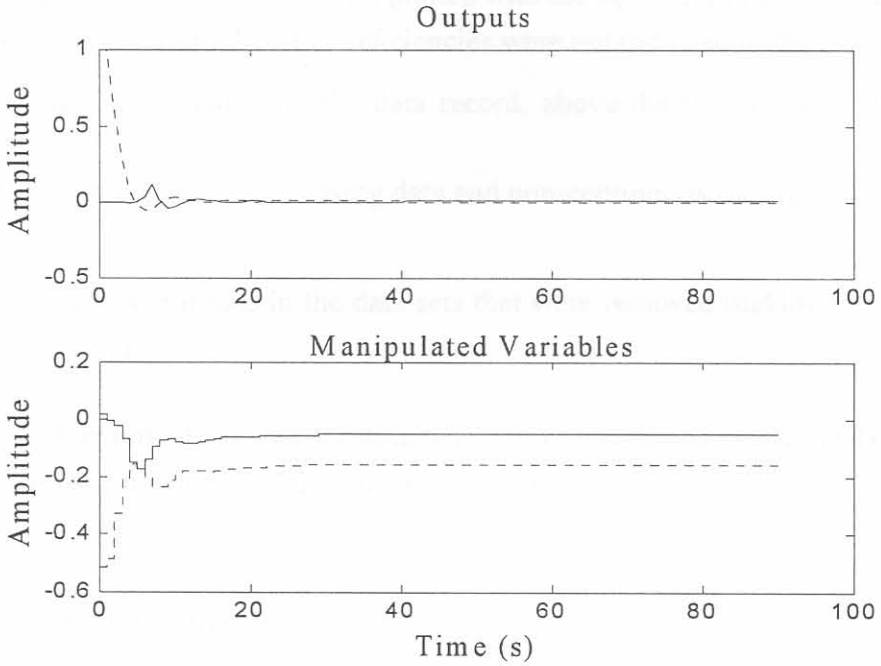

Figure 5.4: The resulting input and output signals in case 13 with $r_i(t) = 0$ and a disturbance. The dotted lines represent u_1 and y_1 and the solid lines represent u_2 and y_2.

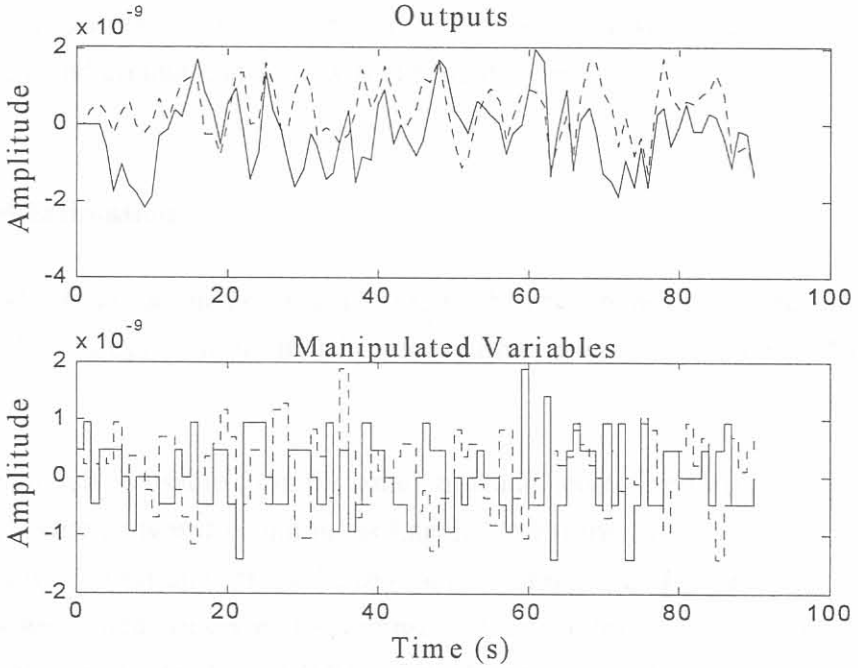

Figure 5.5: The resulting input and output signals in case 12 with $r_i(t) = 0$. The dotted lines are u_1 and y_1 and the solid lines are u_2 and y_2.

Preprocessing: The data sets were first plotted with the *idplot* command in order to inspect them for deficiencies. The following deficiencies were not included in the simulation:

- high frequency disturbances in the data record, above the frequency of interest to the system dynamics, and
- occasional burst and outliers, missing data and non-continuous data records.

However, there were trends in the data sets that were removed making use of the MATLAB function *dtrend*.

Time Delay: The time delays were estimated from visual inspection of the data inputs and outputs, as well as knowledge of the true plant and the previous model.

5.3.3 Model Structure Selection

Type of Structure: The multivariable ARX type model structure was used.

Order Selection: Since a plant model is necessary to design an MPC controller it was assumed that an old model already existed, which gave an indication of the model order. Therefore, the known order, namely first-order, of the model was used and it was not necessary to estimate and compare models with different orders.

5.3.4 Model Estimation

The LSE PEM estimation method was used to fit the chosen models to the estimation range of the data. The multiple-output models were estimated with the standard MATLAB SITB command, *idarx*.

To get the models in usable form for simulation and controller design, the discrete identified *th* models were convert to continuous Laplace transform models. First the ARX difference equation was determined from the *th* models making use of the *th2arx* command. The ARX models were then converted to continuous Laplace transfer functions with the *d2cm* command. The Zero-Order-Hold (ZOH) method, together with a sampling time of 1s and 0.5s for the inter-sampled model, were used for the transformation from discrete-time to continuous-time.

5.3.5 Model Validation

The standard validation methods, together with a measured closed-loop validation data set, not used for model estimation, were used. The validation data set is a part of the data set obtained from the simulation in case 6 where PRBS reference inputs were used.

Simulation and Prediction: The *compare* command was used to compute, for each identified model, both the *pure simulated* output ($\hat{y}_\infty(t \mid m)$) and the 6-step ahead predicted output ($\hat{y}_6(t \mid m)$). The command was also used to determine the numerical values of fit $J_k(m)$ between these outputs and the measured outputs (simulated "true" output). In MATLAB this value of fit is represented as a percentage. The higher the percentage, the better the fit.

Residual Analysis: The *resid* command was used to calculate and display the auto-correlation function of the residuals (test for whiteness), as well as the cross-correlation between the residuals and the plant input (test for independence). The residuals were also plotted for a simple visual inspection.

Model Reduction: Since first-order models were estimated in this simulation, the models were not inspected for order reduction.

5.3.6 Methodology Validation

In the methodology validation step, the proposed closed-loop SID methodology is validated.

Visual Time and Frequency Domain Comparison with the Open-Loop Identified Model: The *step* command was used to plot the step responses and the *impulse* command was used to plot the impulse responses for each of the SISO transfer functions of the different identified models, including the open-loop identified model. The obtained plots are visually compared. For the frequency domain comparison the *bode* command was used to plot both the amplitude and phase responses for each of the SISO transfer functions, of the different identified models, including the open-loop identified model. These responses are also visually compared.

Simulation and Prediction Fit Comparison with the Open-Loop Identified Model: The *compare* command was used to determine the percentage of fit for the pure simulated and 6-step ahead predicted outputs of the model identified in open-loop. These values are compared with those obtained for the closed-loop identified models.

Residual Comparison with the Open-Loop Identified Model: Again, the *resid* command was used to calculate and display the auto-correlation function of the residuals, as well as the cross-correlation between the residuals and the plant input for the model identified from open-loop data. These functions are visually compared with the ones obtained for the closed-loop identified models.

Comparison of Both Models with the True Model: Equation (**??**) was used to determine the numerical value of fit in the frequency domain, *freqfit,* between the true model and the open-loop identified model, as well as with each of the closed-loop identified models. For each model, the total *freqfit* value was taken as the sum of the *freqfit* values of each SISO transfer function. The obtained numerical values are compared. In the same manner, Eqn. (4.11) was used to determine the numerical value of fit, *stepfit,* for the step responses of the models. Large values indicate unsatisfactory models.

Closed-Loop System Examination: Again, the discrete step response models of the identified models were obtained with the MPC toolbox function *tfd2step* using a sampling time of 1s. The transfer functions were approximated with 90 step response coefficients. The *mpccon* command was then used to design the new controllers. These controllers were used to control the true plant. The *mpcsim* command was used to simulate the closed-loop responses.

Furthermore, the *mpccl* command was used to determine the closed-loop models. The commands *smpcpole* and *max* were used to determine the maximum poles in order to see if the closed-loop systems are stable. The *smpcpole* command computes all the discrete poles and the *max* command then determines which pole has the largest absolute value. If the maximum absolute value is equal to or smaller than one, all the poles are on or inside the unit circle.

5.4 Validation Results

In this section the expected validation results are discussed and the obtained validation results are then shown and also discussed.

5.4.1 Expected Results

It was expected that, irrespective of the constraints and added disturbances, satisfactory models would be identified in all the cases where structured tests were performed that ensured

PE reference inputs and good SNRs. The direct closed-loop SID approach was used together with the PEM estimation method and this estimation method works if the data set is informative and the model set contains the true system, irrespective of correlation with the additive output noise. Informative data sets were ensured with the PE reference inputs and it was also known that the chosen model set (first-order ARX) contained the "true" model. Therefore, it was a reasonable assumption that these models would describe the plant accurately and that the controllers designed from these models would be able to control the plant.

It was also expected that if the noise model of the ARX structure is not an accurate description of the true noise model, the identified models might contain a bias. A possible bad fit in the low frequency regions was also expected, since the ARX model structure penalises the high-frequency errors more than low-frequency errors [44].

A possible difference between the models identified from open-loop and closed-loop data was also expected, since the frequency weighting for these two types of models are very different [11] and the plant may exhibit different dynamics under presence of the controller than in open-loop [6]. For the same reason it was expected that the closed-loop identified model might even produce a better controller, as the input weighting in this case favours the cross-over frequency region relevant for the controller [48]. Therefore, it was also be expected that inaccurate models in the low and high frequency regions could still result in good controllers. A bias in these frequency regions could therefore be irrelevant.

In general, unsatisfactory models were expected in the cases where structured tests were not performed, since in these cases PE reference signals did not enure identifiability and good SNRs were also not ensured. Also, for the inter-sampling method that ensures identifiability an imprecise model was expected. The reason being that, although the system was identifiable, it is shown in Section 4.4 that with $r_i(t) = 0$ a large variance can still result. However, for a constrained control law and good SNRs the identification of an accurate and precise model was expected, since a nonlinear controller ensures identifiability.

5.4.2 Obtained Results

In all cases where structured tests with PE reference signals and good SNRs were used, the identified models correspond very well with the model obtained from open-loop data and ensured good controller performance. The only case where no structured test was performed and a good model was still identified, is case 12. Here the controller was constrained and no disturbance was added to the system. The following cases delivered satisfactory models:

- case 1,
- case 2,
- case 3,
- case 4,
- case 5,
- case 6,
- case 10,
- case 11, and
- case 12 ($r_i(t) = 0$ and constrained controller).

The models identified in the above-mentioned cases are very similar and, consequently, the obtained validation results also look very similar. These satisfactory results are classified as **class A** results. The results from all of these cases cannot be shown. Therefore, in each section, the results of only one or two of these cases are given as a representation of all **class A** results.

All the cases that delivered unsatisfactory results had $r_i(t) = 0$. These cases are:

- case 7 (with no disturbance),
- case 8 (with disturbance),
- case 9 (output inter-sampled), and
- case 13 (constrained controller with disturbance).

In case 7, with the linear controller and no disturbances, the resulting inputs and outputs were zero and, therefore, no model could be identified. Thus, no validation results are given for this case. The results obtained in cases 8, 9 and 13 are unsatisfactory and also similar. These unsatisfactory results are classified as **class B** results and, in each section, the results of only one or two of these cases are given as a representation of all **class B** results.

5.4.2.1 Simulation and Prediction

In Figs. 5.6 and 5.7 the pure simulated and 6-step ahead predicted output for the model identified in case 11 are compared with the measured output of the validation data set. In the simulation the *measured* output is the output computed with the *true* model, while the simulated and predicted outputs are computed with the *identified* model. These figures show that,

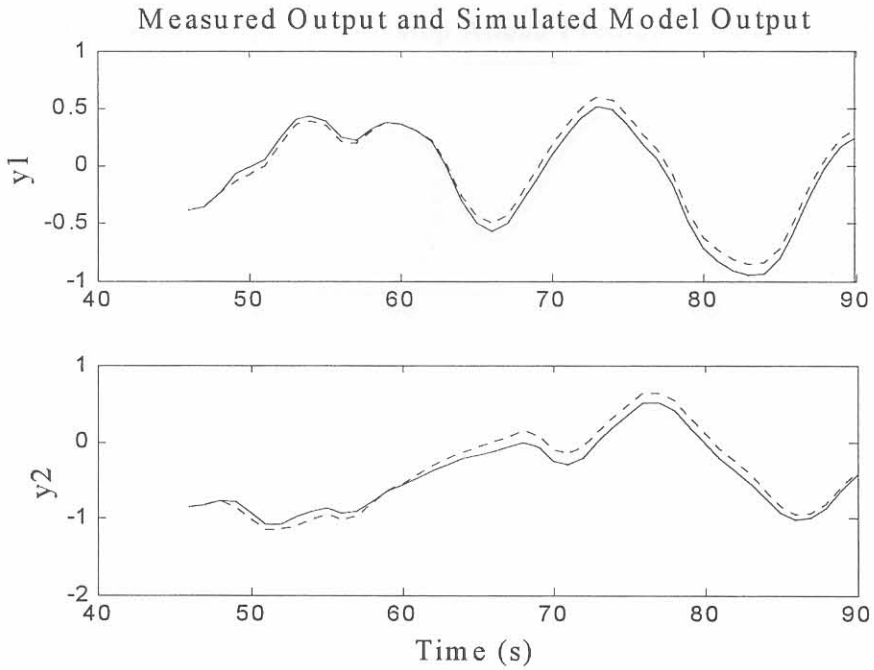

Figure 5.6: The measured (solid line) and pure simulated output (dotted line) for the model identified in case 11.

although the 6-step ahead prediction is better than the pure simulation, both the predicted outputs and pure simulated outputs follow the true output closely. These are representative of the **class A** results.

The computed numerical values of fit for the models identified in all the different cases are shown in Figs. 5.8, 5.9, 5.10 and 5.11. All the **class A** models are capable of reproducing the validation data with an average of 81.3% fit for pure simulation and 89.6% for 6-step ahead prediction. These figures show that all the **class B** cases obtain much smaller percentages of fit.

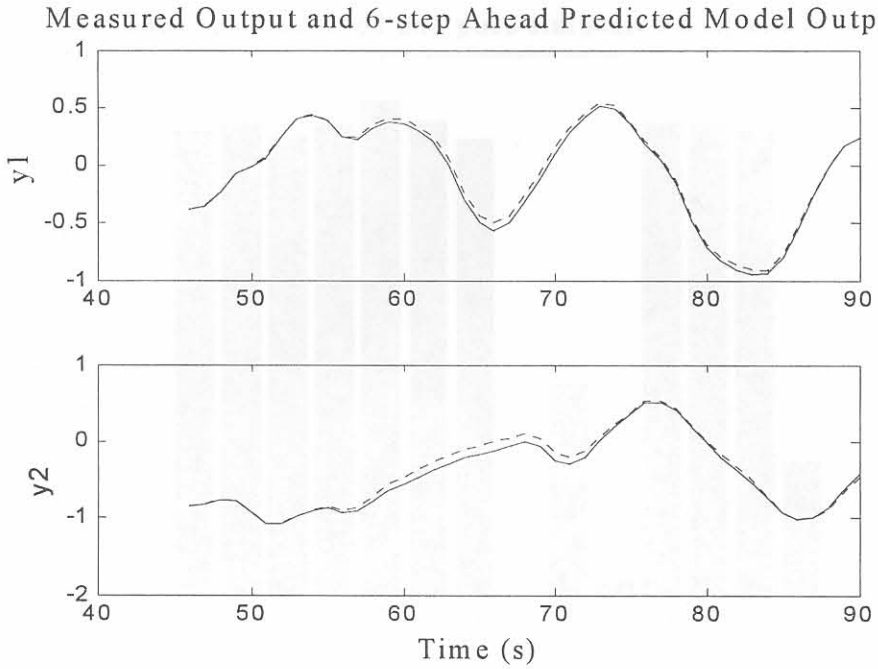

Figure 5.7: The measured (solid line) and 6-step ahead predicted output (dotted line) for the model identified in case 11.

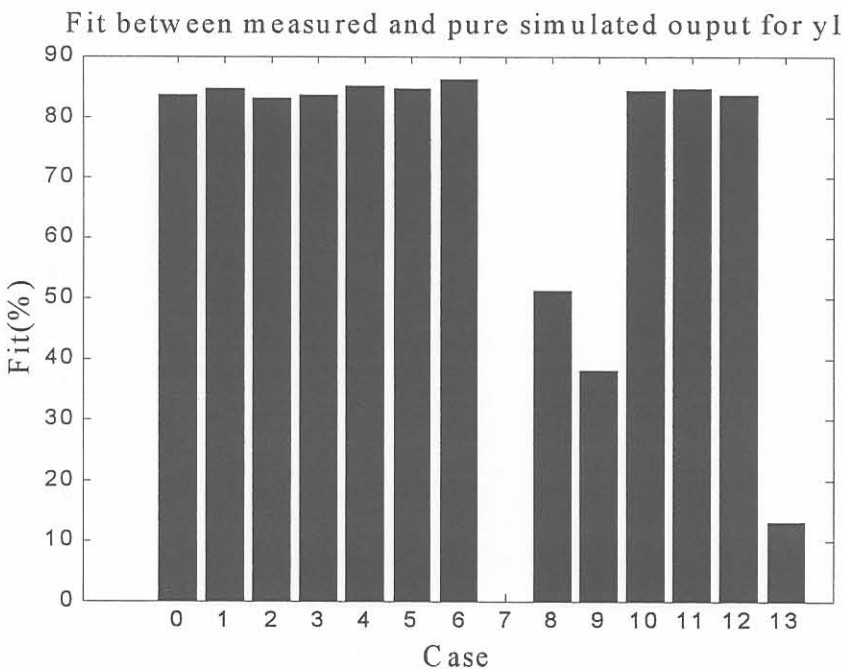

Figure 5.8: Fit between measured and pure simulated output y_1 for different cases and for the model identified from open-loop data (case 0).

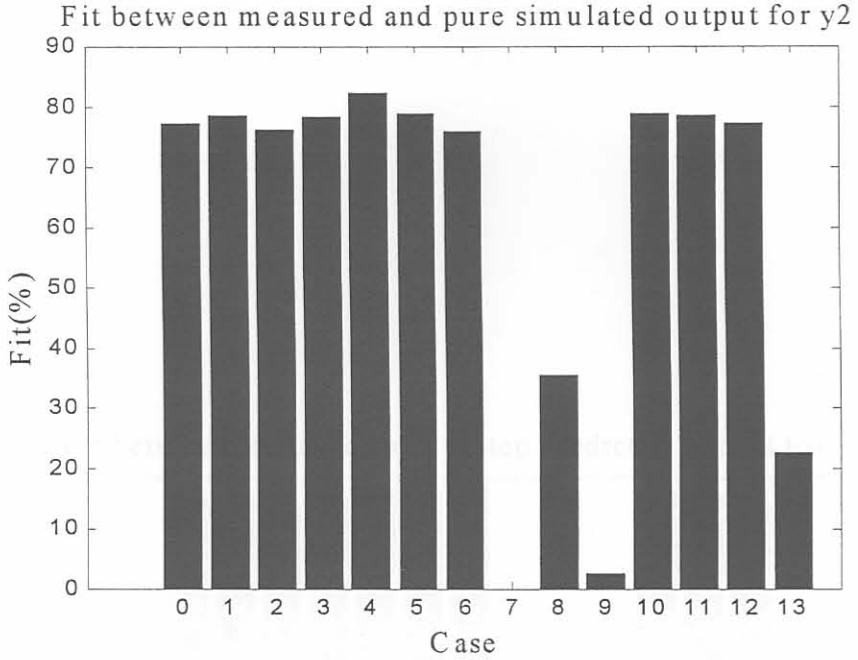

Figure 5.9: Fit between measured and pure simulated output y_2 for different cases and for the model identified from open-loop data (case 0).

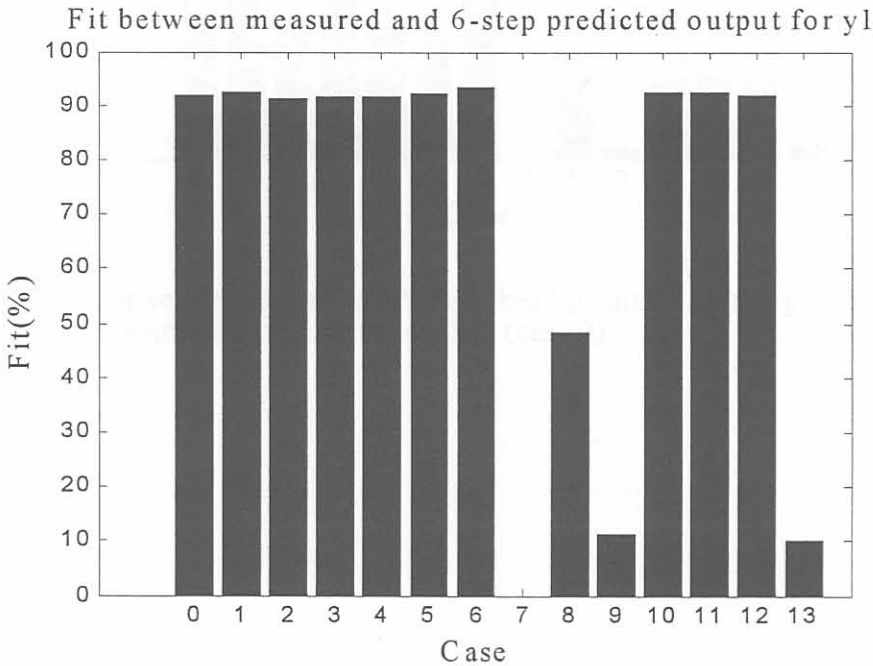

Figure 5.10: Fit between measured and 6-step ahead predicted output y_1 for different cases and for the model identified from open-loop data (case 0).

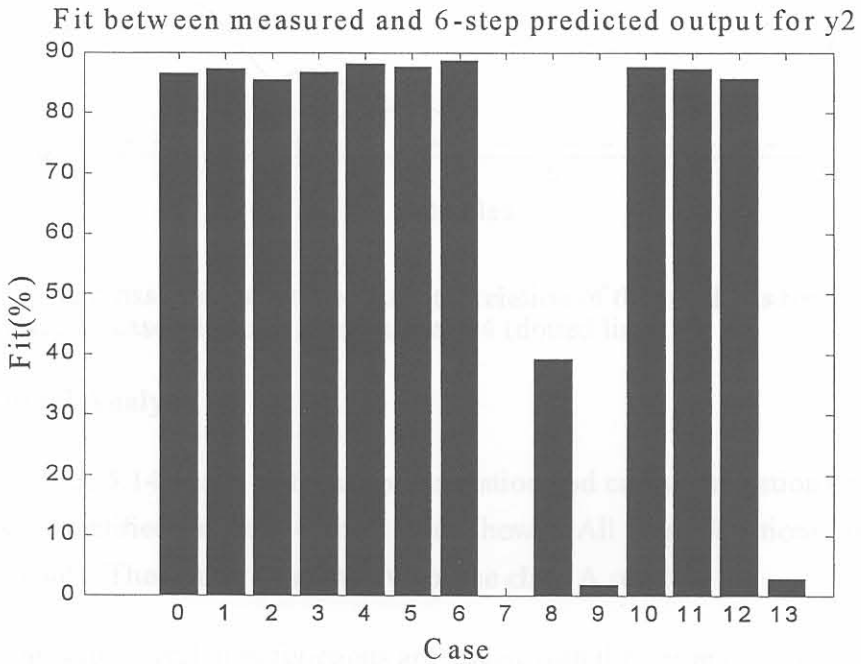

Figure 5.11: Fit between measured and 6-step ahead predicted output y_1 for different cases and for the model identified from open-loop data (case 0).

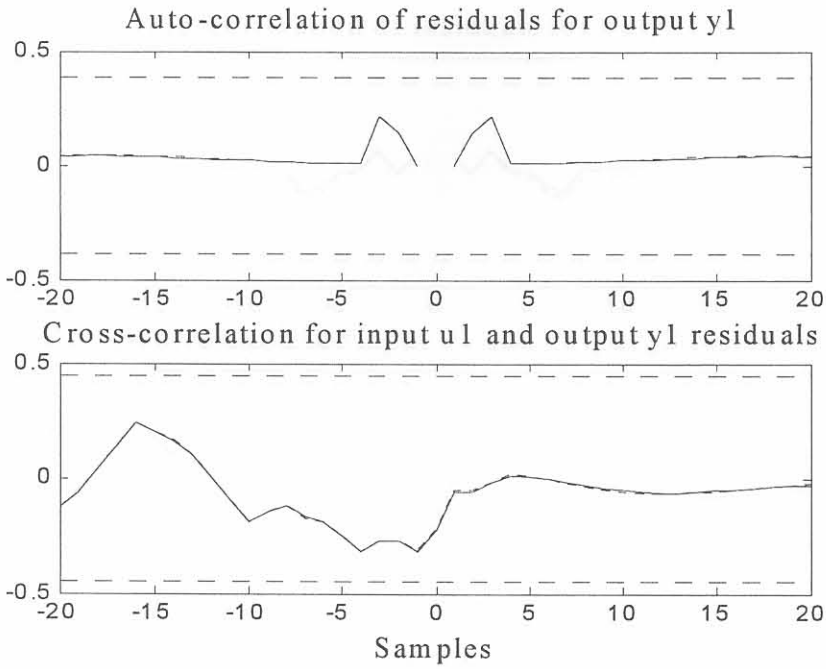

Figure 5.12: The cross-correlation and auto-correlation of the residuals for y_1 and u_1 of the models identified in case 11 (solid line) and case 4 (dotted line).

5.4.2.2 Residual Analysis

In Figs. 5.12, 5.13, 5.14 and 5.15 the auto-correlation and cross-correlation of the residuals for the models identified in cases 4 and 11 are shown. All these functions stay within the confidence bounds. These are representative of the **class A** results.

Note that the auto-correlation functions are scaled with the variance. Therefore, the value at lag = 0 is 1. However, this is not indicated to keep the scale of the figures between 0.5 and -0.5 which is the interesting region [36].

In Figs. 5.16, 5.17, 5.18 and 5.19 the auto-correlation and cross-correlation of the residuals for the models identified in cases 9 and 13 are shown. These functions do not always stay within the confidence bounds, but go outside the bounds. These are representative of the **class B** results.

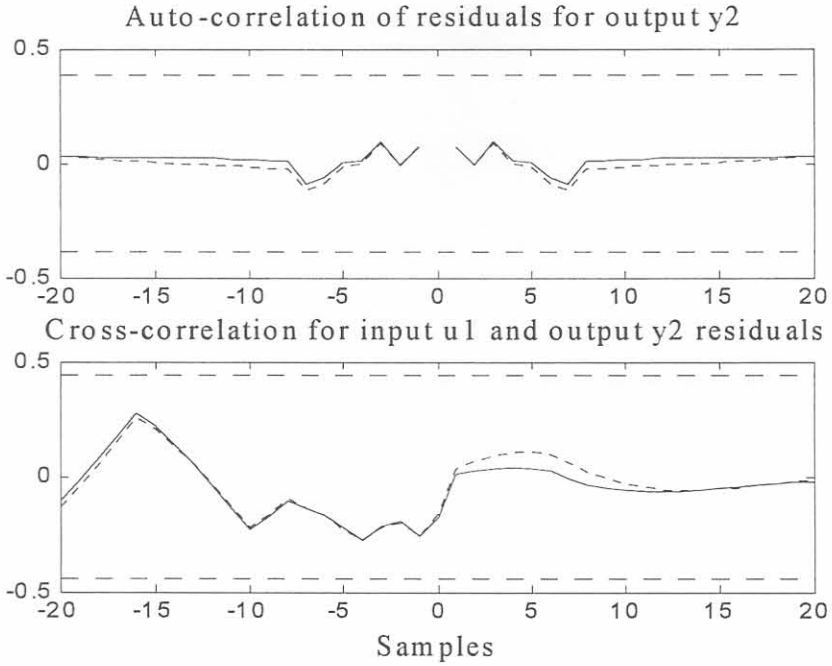

Figure 5.13: The cross-correlation and auto-correlation of the residuals for y_2 and u_1 of the models identified in case 11 (solid line) and case 4 (dotted line).

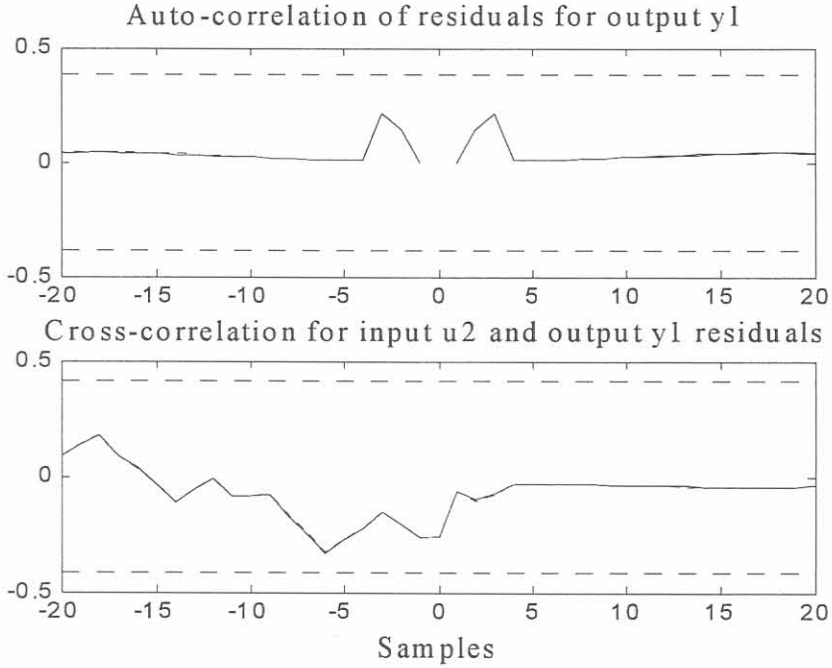

Figure 5.14: The cross-correlation and auto-correlation of the residuals for y_1 and u_2 of the models identified in case 11 (solid line) and case 4 (dotted line).

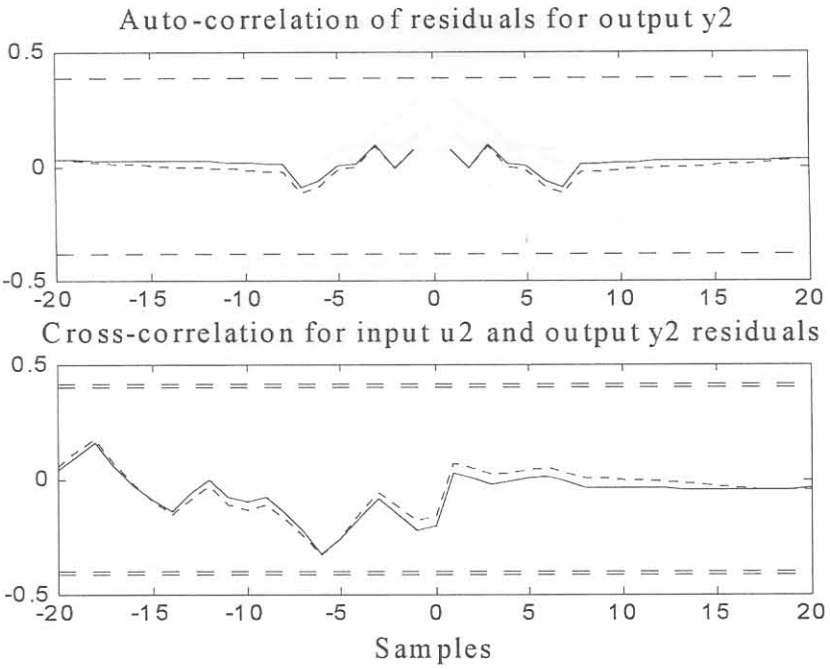

Figure 5.15: The cross-correlation and auto-correlation of the residuals for y_2 and u_2 of the models identified in case 11 (solid line) and case 4 (dotted line).

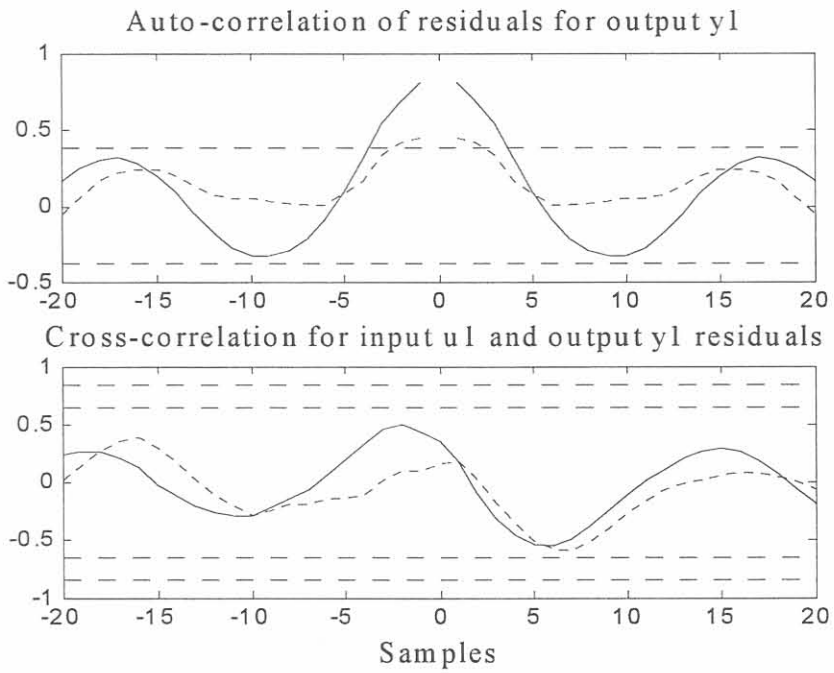

Figure 5.16: The cross-correlation and auto-correlation of the residuals for y_1 and u_1 of the models identified in case 13 (solid line) and case 9 (dotted line).

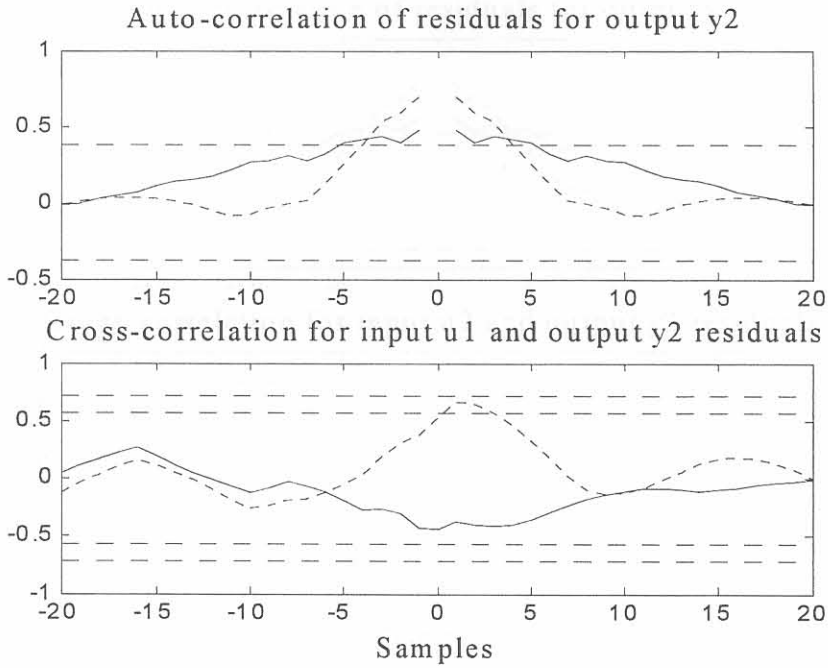

Figure 5.17: The cross-correlation and auto-correlation of the residuals for y_2 and u_1 of the models identified in case 13 (solid line) and case 9 (dotted line).

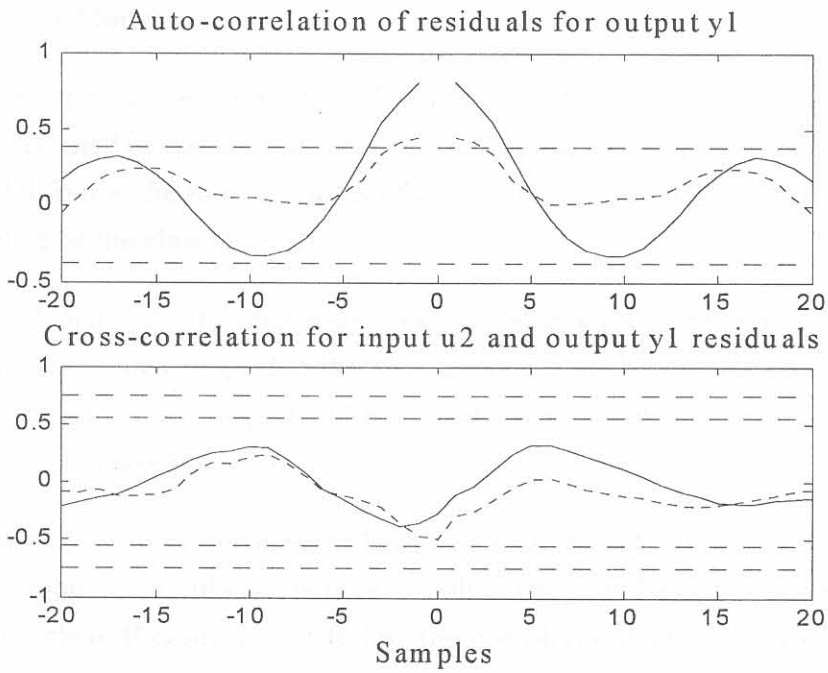

Figure 5.18: The cross-correlation and auto-correlation of the residuals for y_1 and u_2 of the models identified in case 13 (solid line) and case 9 (dotted line).

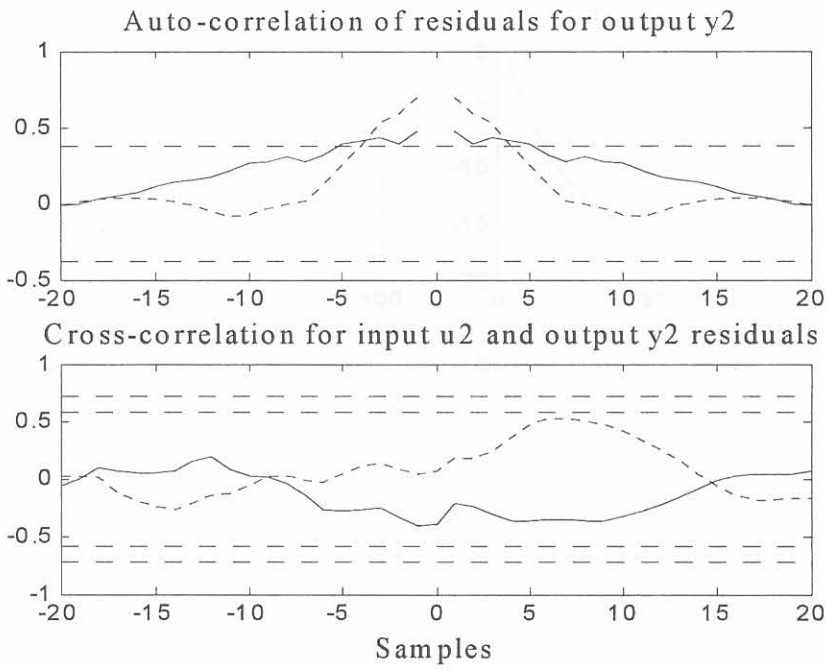

Figure 5.19: The cross-correlation and auto-correlation of the residuals for y_2 and u_2 of the models identified in case 13 (solid line) and case 9 (dotted line).

5.4.2.3 Visual Time and Frequency Domain Comparison with the Open-Loop Identified Model

In Fig. 5.20 the step responses of the model identified from the open-loop data are compared with a model identified in case 3. This figure shows that the step responses of the closed-loop identified model follow the step responses of the open-loop identified model closely. These are representative of the **class A** results.

In Figs. 5.21 and 5.22 the step responses for the models identified in cases 8 and 9 are shown. These figures show that the step responses of the models identified in these cases do not follow the open-loop identified model's step responses in Fig. 5.20. These are representative of the **class B** results.

The impulse responses of the models identified in the **class A** cases also follow the open-loop identified model's impulse responses closely. The impulse responses of the models identified in the **class B** cases do not follow the open-loop identified model's impulse responses.

In Figs. 5.23, 5.24, 5.25 and 5.26 the model identified from open-loop data and the model identified in case 6 are compared in the frequency domain. The frequency responses of the

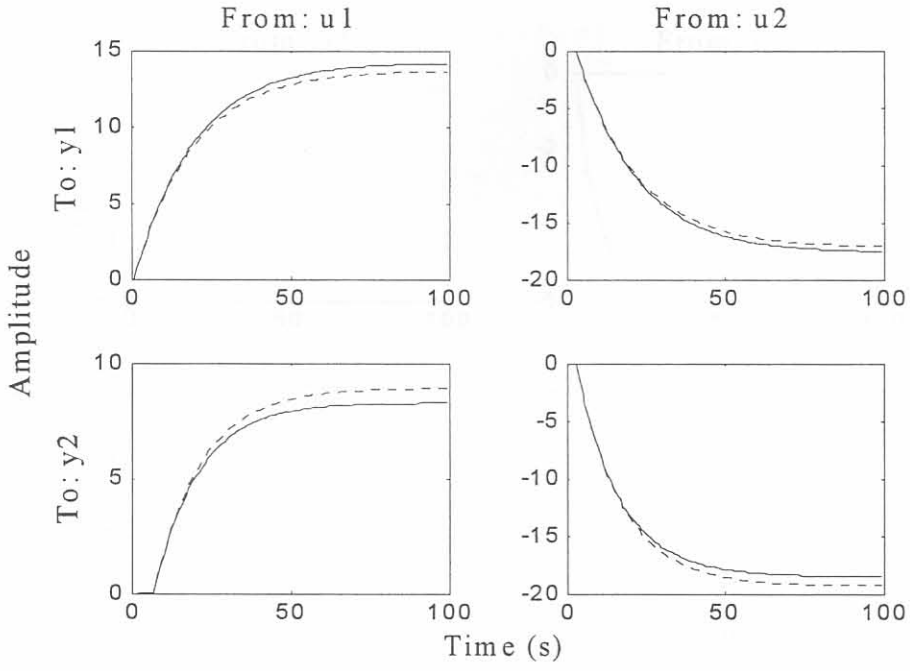

Figure 5.20: The step responses for the model identified from open-loop data (solid line) and the model identified in case 3 (dotted line).

closed-loop identified model agree very well with that of the open-loop identified model in the low, high and cross-over frequency regions. These are representative of the **class A** results.

In Figs. 5.27, 5.28, 5.29 and 5.30 the frequency responses of the open-loop identified model is compared with that of the model identified in case 13. The plots show that this closed-loop identified model does not compare very well with the open-loop identified model in the frequency domain. This is also the case for the other **class B** models.

5.4.2.4 Simulation and Prediction Fit Comparison with the Open-Loop Identified Model

The computed numerical values of fit for the models identified in all the different cases are compared with the numerical value of fit of the model identified from open-loop data in Figs. 5.8, 5.9, 5.10 and 5.11. The open-loop identified model is indicated with the label *open*. The graphs show that the percentages of fit in all the **class A** cases are comparable to the open-loop identified model's percentages of fit, while the percentages of fit in all the **class B** cases are much smaller.

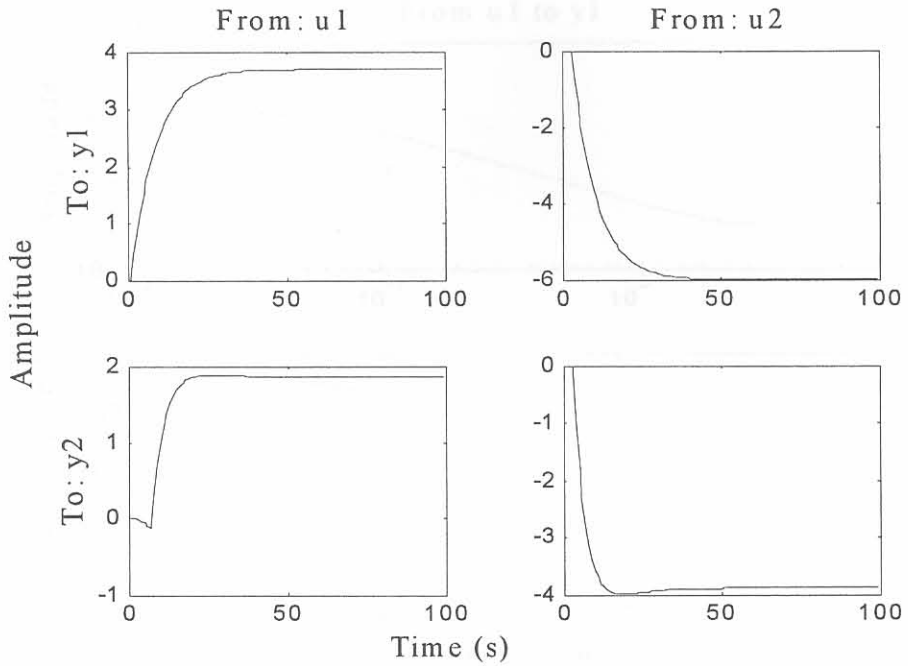

Figure 5.21: The step responses for the model identified in case 8.

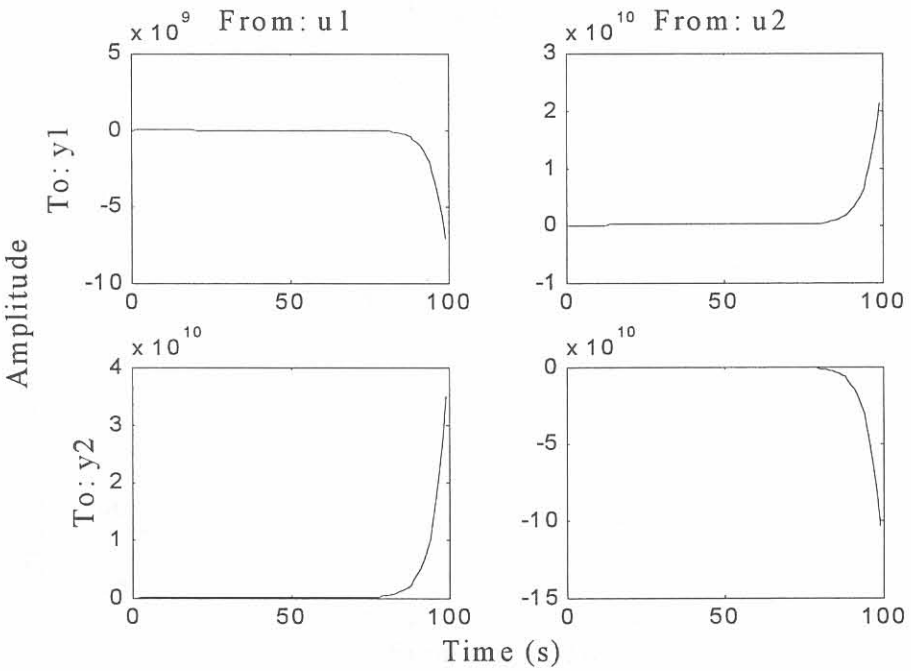

Figure 5.22: The step responses for the model identified in case 9.

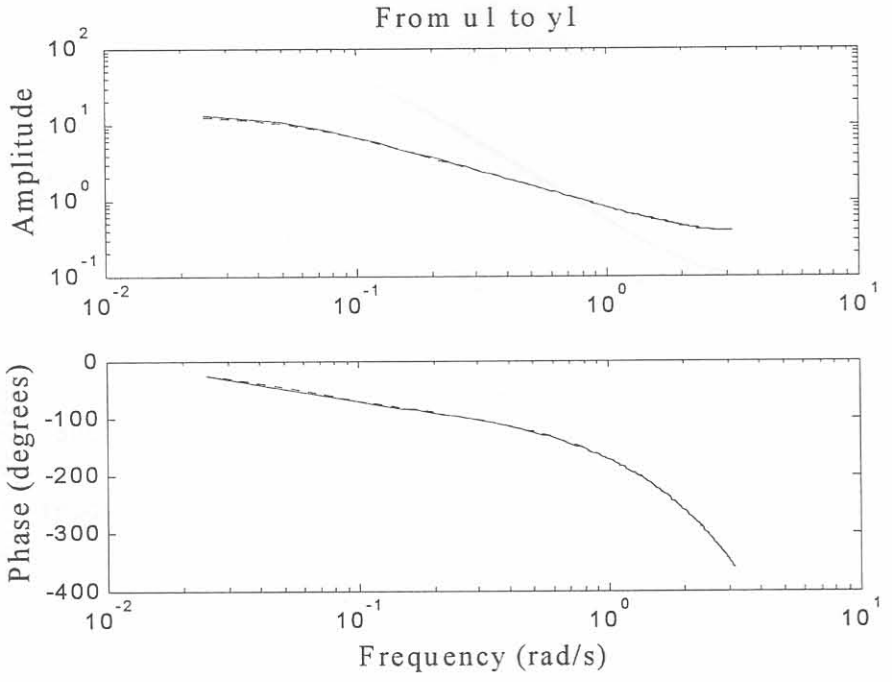

Figure 5.23: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 6 (dotted line).

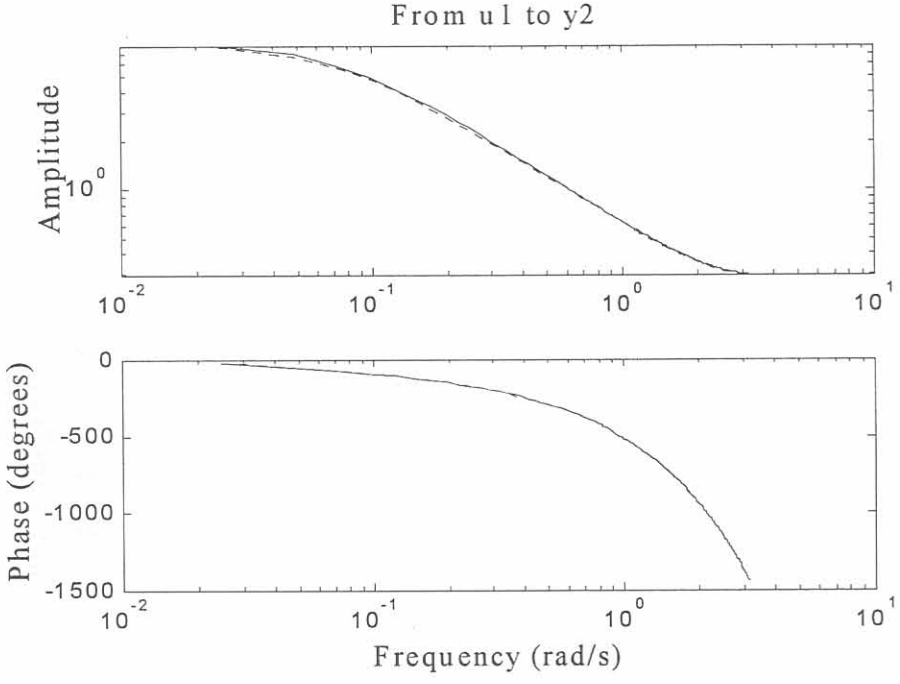

Figure 5.24: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 6 (dotted line).

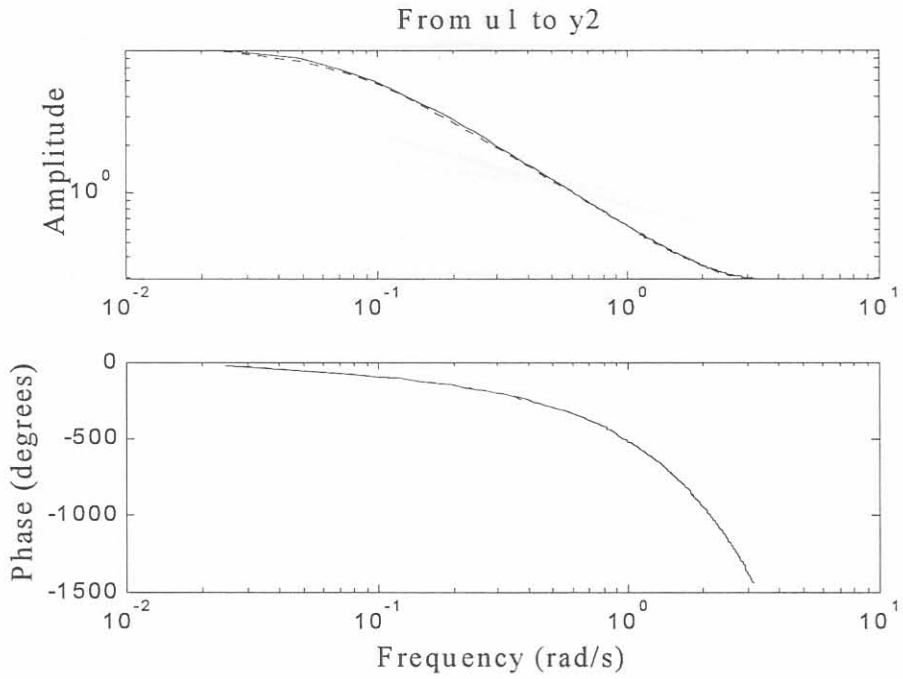

Figure 5.25: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 6 (dotted line).

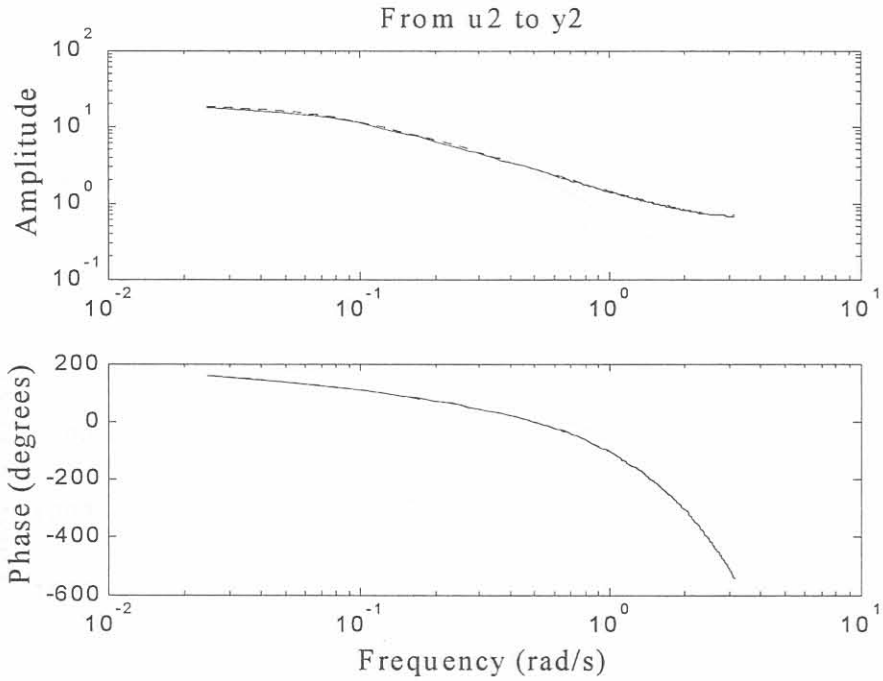

Figure 5.26: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 6 (dotted line).

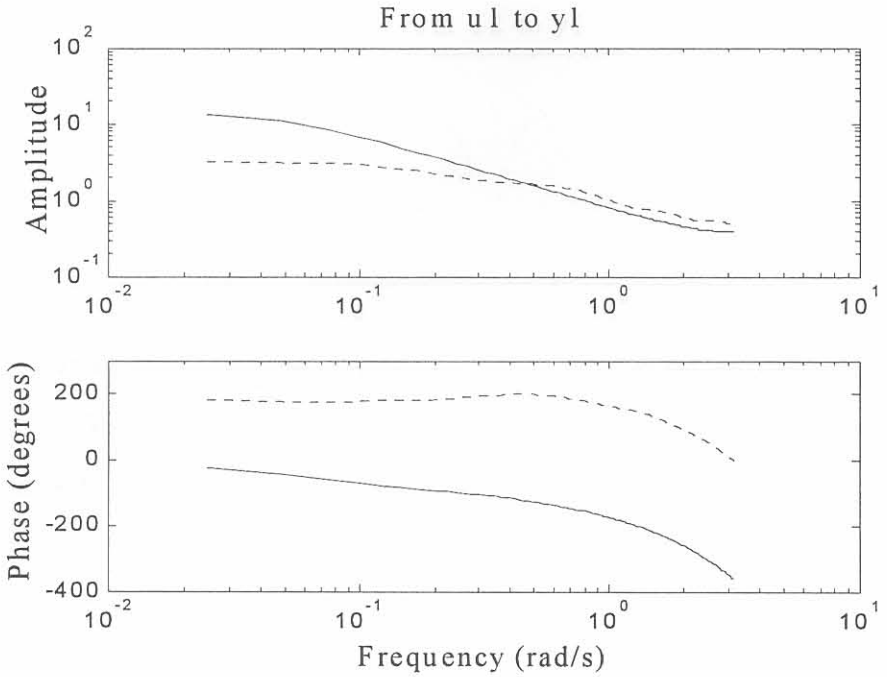

Figure 5.27: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 13 (dotted line).

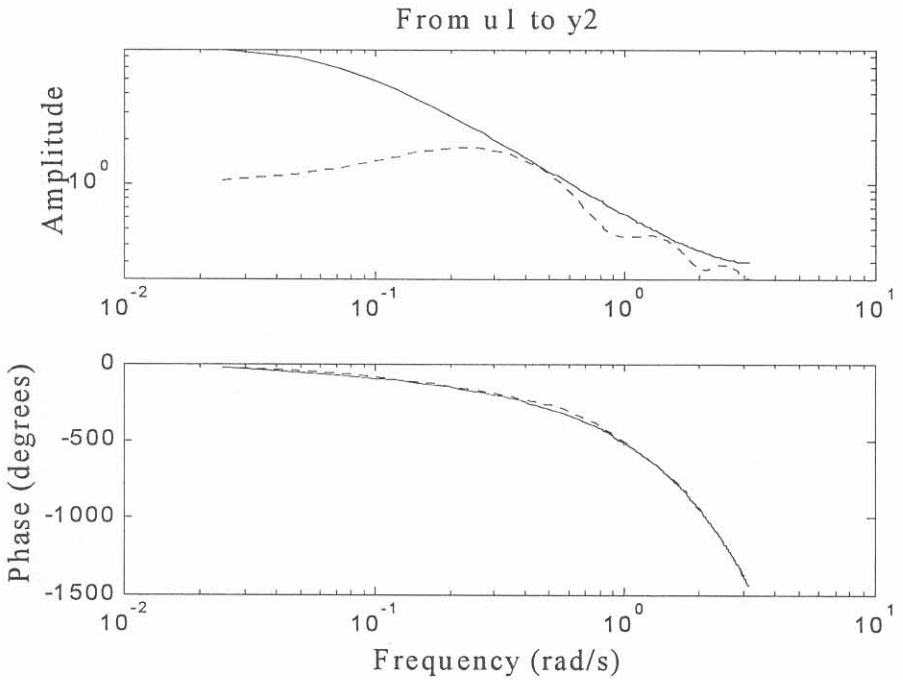

Figure 5.28: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 13 (dotted line).

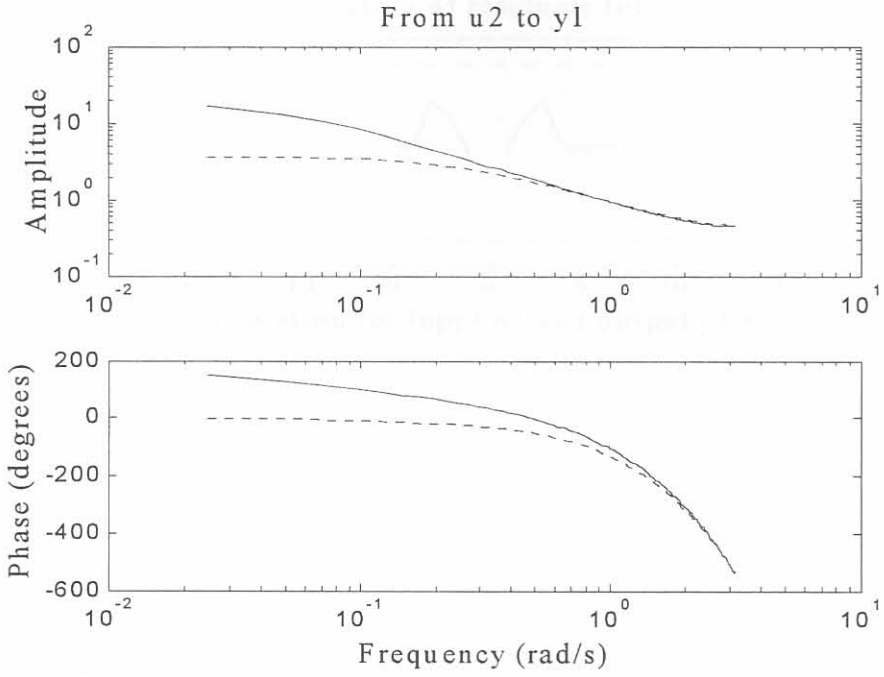

Figure 5.29: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 13.

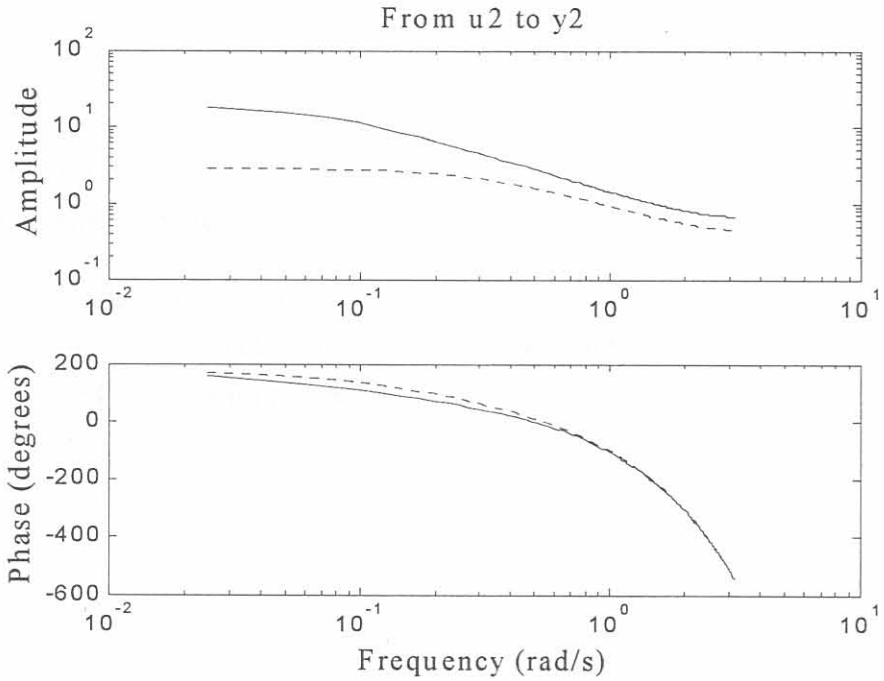

Figure 5.30: The bode plots for the model identified from open-loop data (solid line) and of the model identified in case 13 (dotted line).

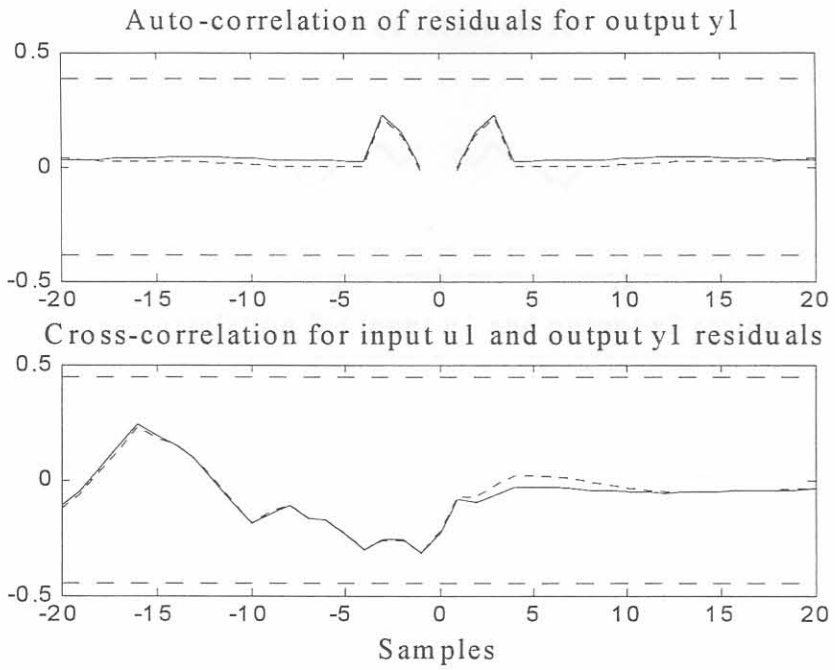

Figure 5.31: Comparison of the cross-correlation and auto-correlation of the residuals for y_1 and u_1 of the models identified from the open-loop data (solid line) and case 6 (dotted line).

5.4.2.5 Residual Comparison with the Open-Loop Identified Model

In Figs. 5.31, 5.32, 5.33 and 5.34 the auto-correlation and cross-correlation of the residuals for the model identified from the open-loop data and the models identified from the closed-loop data in case 6 are compared. The figures show that for the **class A** models and the open-loop identified model, these functions are comparable.

The auto-correlation and cross-correlation functions of the open-loop identified model and the functions obtained in the **class B** cases, shown in Figs. 5.16, 5.17, 5.18 and 5.19, can also be compared. This comparison shows that these functions for the **class B** models are not comparable to the open-loop identified model's functions.

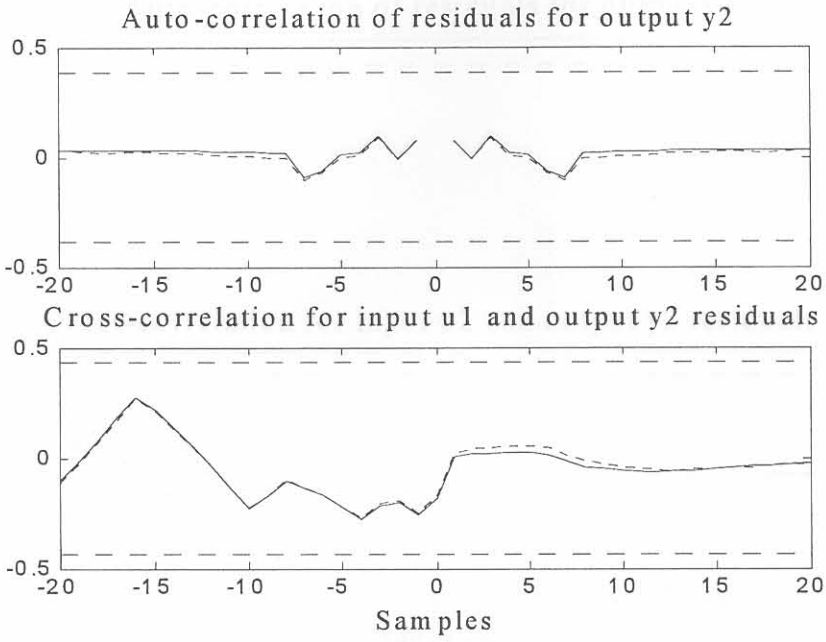

Figure 5.32: Comparison of the cross-correlation and auto-correlation of the residuals for y_2 and u_1 of the models identified from the open-loop data (solid line) and case 6 (dotted line).

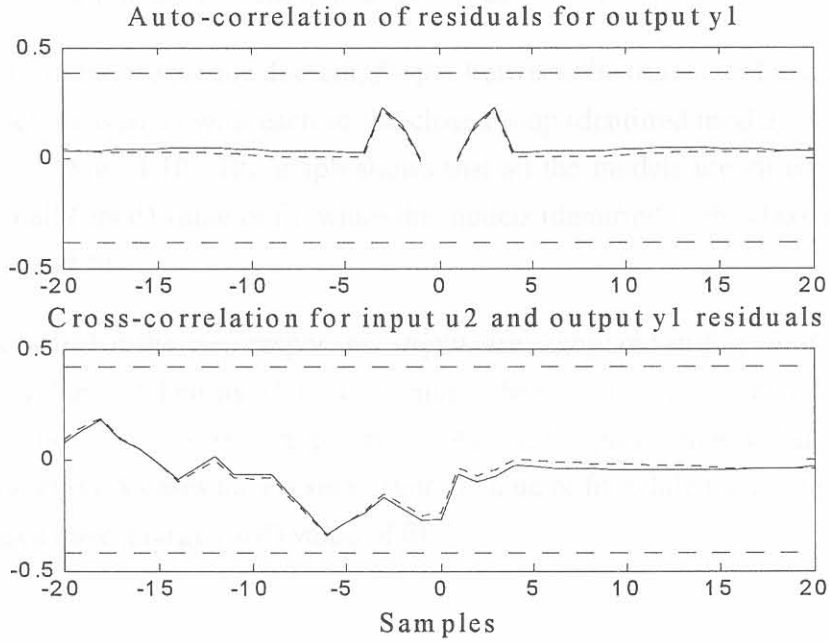

Figure 5.33: Comparison of the cross-correlation and auto-correlation of the residuals for y_1 and u_2 of the models identified from the open-loop data (solid line) and case 6 (dotted line).

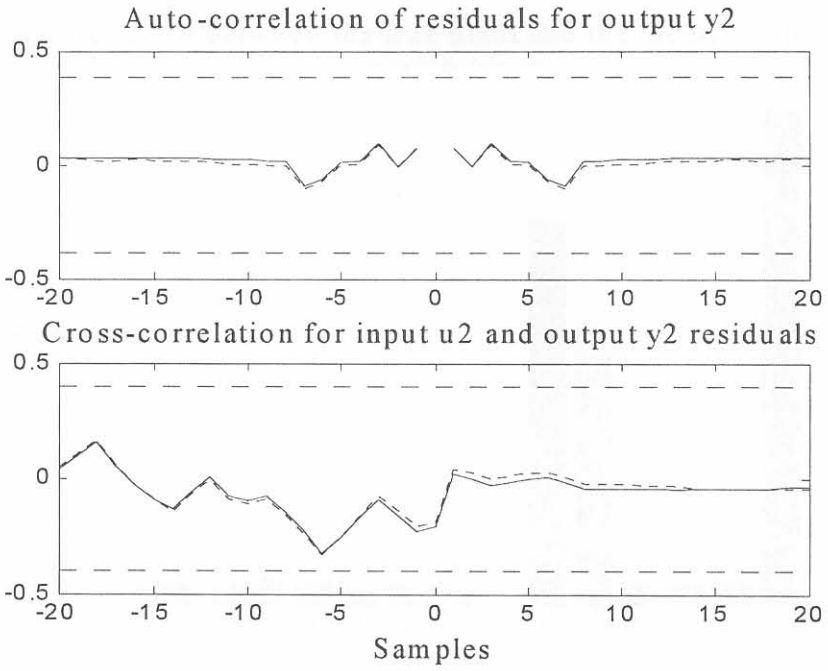

Figure 5.34: Comparison of the cross-correlation and auto-correlation of the residuals for y_2 and u_2 of the models identified from the open-loop data (solid line) and case 6 (dotted line).

5.4.2.6 Comparison of Both Models with the True Model

The values of fit in the frequency domain, *freqfit*, between the true model and the open-loop identified model, as well as with each of the closed-loop identified models, were computed and are plotted in Fig. 4.10. The graph shows that all the models identified in the **class A** cases have a small (good) value of fit, while the models identified in the **class B** cases have a large (bad) value of fit.

The values of fit for the step responses, *stepfit*, are compared in Fig. 5.36. In Fig. 5.36 the maximum value is taken as 150. The values obtained in cases 9 and 13 are actually 3.7944×10^{14} and 1.0918×10^{22} respectively. Again, the graph shows that all the models identified in the **class A** cases have a small (good) value of fit, while the models identified in the **class B** cases have a large (bad) value of fit.

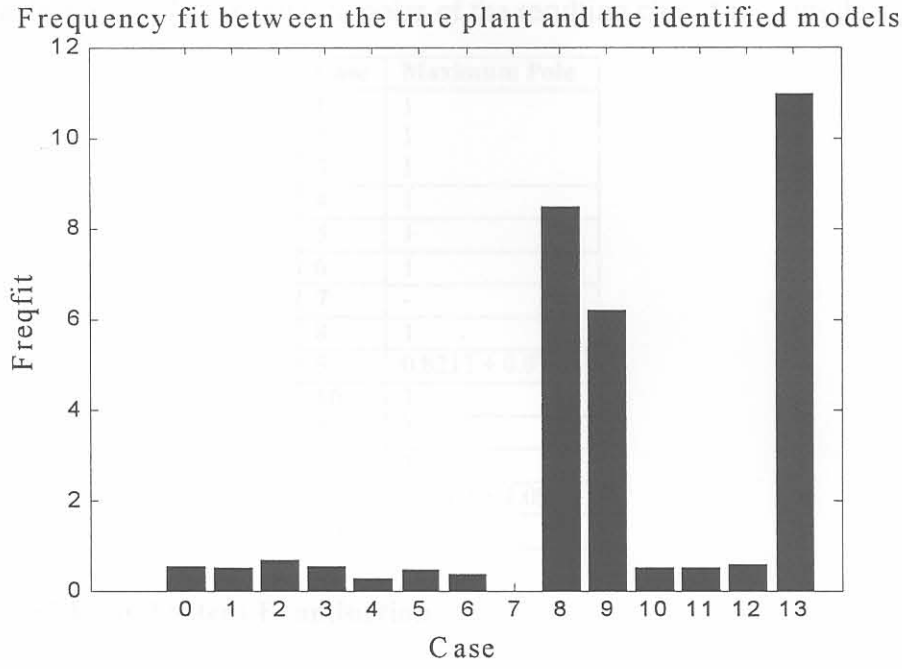

Figure 5.35: The fit in the frequency domain between the true plant and the identified models. Case 0 is the open-loop identified model.

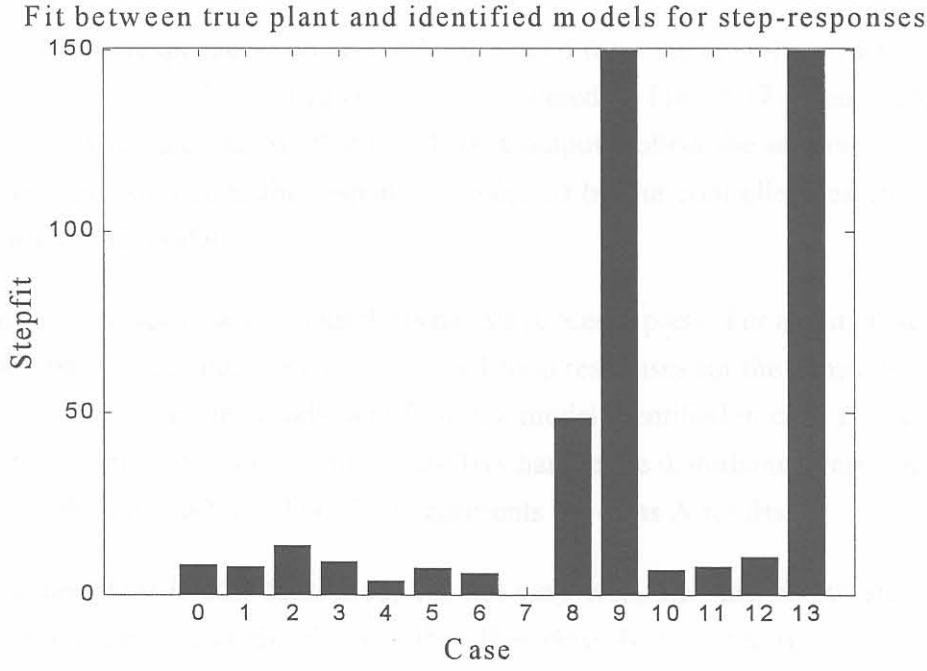

Figure 5.36: The fit in the time domain between the step-responses of the true plant and the identified models. Case 0 is the open-loop identified model.

Table 5.3: The maximum poles of the resulting closed-loop models

Case	Maximum Pole
1	1
2	1
3	1
4	1
5	1
6	1
7	-
8	1
9	0.8213 + 0.9758i
10	1
11	1
12	1
13	0.9462 + 1.0921i
open	1

5.4.2.7 Closed-Loop System Examination

The maximum poles of the resulting closed-loop function are given in Table 5.3. In all the **class A** cases the maximum poles are on the unit circle and the closed-loop systems are thus marginally stable. In two of the **class B** cases the maximum poles are outside the unit circle and the closed-loop systems are thus unstable.

The closed-loop responses for a controller designed from the open-loop identified model and for a controller identified in case 11, are compared in Fig. 5.37. This represents the **class A** results. The figure shows that the **class A** outputs follow the set-points and that the responses are very similar to the responses generated by the controller designed from the open-loop identified model.

The real test is to see how the controllers handle process upsets. For a unit pulse change at $t = 1$ in the disturbance adding to u_1, the closed-loop responses for the controllers designed from the open-loop identified model and from the model identified in case 11 are compared in Fig. 5.38. The plot shows that both controllers handle the disturbance very similarly and the set-point values are still reached. This represents the **class A** results.

In case 8 the plant is still controlled, but not very well, i.e. the output signals do not follow the reference signals closely. For the other **class B** cases the resulting closed-loop responses are also unsatisfactory, because the outputs do not follow the reference signals and the systems become unstable.

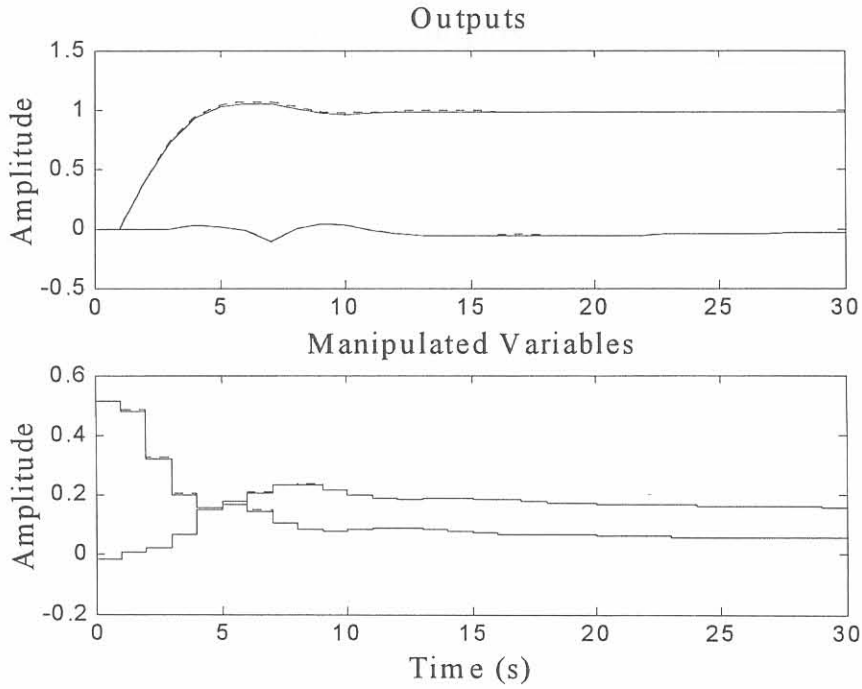

Figure 5.37: The closed-loop response for a controller designed from the open-loop identified model (solid line) and for a controller designed from the model identified in case 11 (dotted line) with $r_1(t) = 1$ and $r_2(t) = 0$.

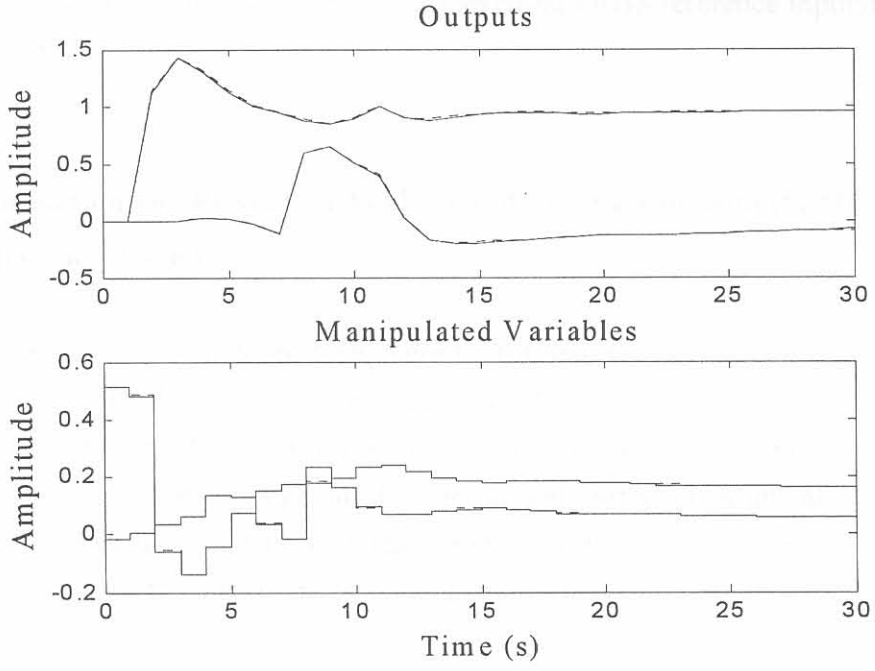

Figure 5.38: The closed-loop response for a controller designed from the open-loop identified model (solid line) and for a controller designed from the model identified in case 11 (dotted line) with $r_1(t) = 1$ and $r_2(t) = 0$. A pulse disturbance was added at $t = 1$ to u_1.

5.4.3 Discussion

As mentioned, two trials were run with different test signals. The PRBS signals gave slightly better results than the step inputs and therefore these results are shown. A possible cause for this slight difference can be that the step input is not PE enough. For a step to be PE, the step should be preceded by zeros for a length of time approximately equal to the impulse response of the true system [36]. In the first trial, $r_1(t)$ was stepped at time zero and therefore did not satisfy this condition.

The results obtained in trial 2 (persistently exciting PRBS reference inputs) are now further discussed.

5.4.3.1 Simulation and Prediction Analysis and Comparison with the Open-Loop Identified Model

In all the cases where structured tests were performed, the identified models are capable of reproducing the validation data satisfactorily. The percentages of fit obtained for the open-loop identified model are also comparable with these values. Thus, according to the simulation and prediction analysis, all these models are satisfactory and also comparable to the open-loop identified model. This result indicates that the proposed SID methodology works irrespective of the disturbances and constraints that were used.

In the case where a nonlinear controller was used, a good percentage of fit results only when there were no added disturbance as in case 12. This model is thus also satisfactory. Case 7 is similar to case 12, except that the controller is linear. In this case no model could be identified from the zero input and output signals that resulted. These results confirm the fact that nonlinear feedback ensures identifiability.

Poor results are obtain in the other cases where the reference inputs were zero (**class B**). The percentages of fit are much lower than the percentages of fit obtained for the open-loop identified model. In case 8 this can be contributed to the fact that the system is not identifiable. However, in case 9 where output inter-sampling was used and in case 13 where a nonlinear controller was used, the systems were identifiable. Here imprecise models still resulted because of the bad SNR.

5.4.3.2 Residual Analysis and Comparison with the Open-Loop Identified Model

The auto-correlation and cross-correlation of the errors with the outputs, indicate that the closed-loop identified models in all the structured test cases, as well as in the zero reference case with the nonlinear controller and good SNRs (case 12), describe the plant accurately. In these cases the auto-correlation functions are between the confidence bounds, which mean the errors are white and the models unbiased. A visual inspection of the errors confirms this result. The cross-correlation functions are also between the confidence bounds, which indicate that the error signals and u_i are independent. These results are similar to that of the model identified from the open-loop data. Again, this indicates that the proposed SID methodology works irrespective of the disturbances and constraints that were used and it confirms that nonlinear feedback ensures identifiability.

The residual analysis also indicates that the other models, identified in the cases when $r_i(t) = 0$, are deficient, since these cross-correlation functions and the auto-correlation functions go outside the confidence bounds. This confirms that the other methods that ensure identifiability do not necessarily deliver accurate models.

5.4.3.3 Visual Time and Frequency Domain Comparison with the Open-Loop Identified Model

Since, in the structured test cases, the step responses of the closed-loop identified models and the open-loop identified model agree very well, it can be concluded that these models are approximately equal in the time domain. Thus, the proposed SID methodology deliver satisfactory models. Also the model identified in case 12 and the open-loop identified model compare well in the time domain, which again confirms the fact that nonlinear feedback ensures identifiability.

The comparison in the frequency domain shows that all the closed-loop models identified from the structured tests and case 12 are equal in accuracy in the low, high and cross-over frequency regions to the open-loop identified model. Again, this indicates that the proposed SID methodology works irrespective of the disturbances and constraints that were used and it confirms that nonlinear feedback ensures identifiability.

The other models identified with no PE reference signal present, did not agree very well with the open-loop identified model in either the time or frequency domain. Therefore, these

models do not give the correct description of the plant. This confirms that the other methods that ensure identifiability do not necessarily deliver accurate models.

5.4.3.4 Comparison of Both Models with the True Model

The computed values of fit between the identified models and the true model, show that models estimated from the structured tests closed-loop data represent the true model just as well as the open-loop identified model. Thus, the proposed SID methodology delivers satisfactory models.

The computed values of fit, between the identified model in case 12 and the true model, also shows that this closed-loop estimated model represents the true model just as well as the open-loop identified model. Again, it confirms that nonlinear feedback ensures identifiability and that good SNRs are necessary.

These values of fit are much worse (larger) for the models identified in the other cases where structured tests were not performed. From this it is apparent that the other models, identified from experimental data with no reference signals present, are not comparable with the open-loop identified model and are also very different from the true plant. Again, this confirms that the other methods that ensure identifiability do not necessarily deliver accurate models.

5.4.3.5 Closed-Loop System Examination

The closed-loop responses show that the closed-loop models identified from the structured test data, as well as the model identified in case 12, are good enough for their purpose, namely closed-loop MPC control. In these cases the resulting closed-loop systems are stable. These responses also show that the open-loop and closed-loop identified models result in very similar controllers. Again, this indicates that the proposed SID methodology works irrespective of disturbances and constraints that were used and it confirms that nonlinear feedback ensures identifiability.

In case 8, where $r_i(t) = 0$, the plant is still controlled, but not very well. This can be due to the fact that the model may still be relatively accurate in the cross-over frequency region and therefore still results in a stable closed-loop system. For the other two cases with $r_i(t) = 0$, the resulting closed-loop responses are unsatisfactory, because, firstly, the outputs do not

follow the reference signals and, secondly, the closed-loop systems are unstable. Again, this confirms that the other methods that ensure identifiability do not necessarily deliver accurate models.

5.4.3.6 Synopsis

As expected, irrespective of the disturbances and constraints that were used, satisfactory models are obtained in all the cases where structured tests, which ensured PE reference signals and good SNRs, were performed. Thus, the proposed SID methodology works for the type of disturbances and constraints used.

There is also no significant bias in the models identified from the structured test data. The reason can be that the noise model is good and that the SNRs were also acceptable. Furthermore, there is also no significant bad fit in the low frequency regions, which may sometimes be expected from ARX models.

There is also no significant difference between the models identified from the open-loop data and the closed-loop structured test data and therefore the plant probably did not exhibit very different dynamics in closed-loop than in open-loop [6]. For this reason these closed-loop identified models produced equivalent controllers to the one produced by the open-loop identified model.

Then, in the cases where the reference signals were zero, unsatisfactory models were obtained. For cases 7 and 8 the reason is that the data were not informative enough. In case 9, where the inter-sampling method ensured identifiability, an imprecise model was identified, because it is shown in Section 4.4 that with $r_i(t) = 0$ a large variance will result. The only exception, as expected, is case 12 with the constrained controller and no disturbance, since in this case the nonlinearities in the controller ensured identifiability and the good SNR ensured a small variance in the model. The changes in the input and output signals were very small. Therefore, when the disturbance was added in case 13, the SNR became unacceptable and a large variance resulted in the identified model. With the reference signals zero, the SNRs are only good for *very* small disturbances.

5.5 Conclusion

From these simulation results, it can be concluded that the proposed closed-loop system identification methodology gives reliable results, i.e. accurate and precise models, for MIMO

plants controlled by MPC controllers, for the type of system disturbances and constraints that were used, as long as the reference signals are PE and the SNRs (ratios between noise and plant input signals) are good.

When PE reference signals are used, the proposed closed-loop SID methodology and open-loop SID method deliver comparable identification results. When the reference signals are not PE, or when they result in bad SNRs, the methodology should be reconsidered.

Other methods that ensure identifiability, e.g. inter-sampling and nonlinear controllers, also do not guarantee precise models if the SNR is not good, which can happen when no structured tests are performed. Structured tests should be conducted to ensure good SNRs, and the easiest way to ensure identifiability is to make the reference signals PE.

These validation results hold for the ideal case, at least. In the next chapter the implementation of the methodology on real process data is discussed.