System Identification

Integrated Master Degree in Mechanical Engineering

Scientific Area of Control, Automation, and Industrial Informatics

Rasteirinho

Author:
João Gil, 80838
Luís Miranda, 81089

Instructor:
Prof. Jorge Martins

October 27, 2020

Contents

1 Introduction

This study is integrated in the System Identification course, of the Systems Master Degree in Mechanical Engineering of Instituto Superior Técnico.

In this report several methods of identification are described and their results evaluated. There are identification methods in both time and frequency domain. The system to be identified is the *Rasteirinho*, which is a small mobile robot performing a couple of routines.

Firstly, a Deterministic approach is taken, where the inputs given to the system are deterministic. Then, the system is treated has Stochastic, receiving random white noise inputs. Finally, a set of Discrete Stochastic Models are trained to fit the experimental collected data. Throughout the implementation of these approaches, the *Rasteirinho* is described in both parametric and non-parametric representations.

2 System's Dynamics

The *Rasteirinho* is a system which consists of a laptop, with a connected webcam, placed in a metal frame with wheels. By means of a motor, controlled with an Arduino, it moves forward, following a black line on the ground. This mobile robot will do certain types of routines about the black line, center of the 160 pixel image, depending on the given input signal. It is of high importance to mention the non-linearities present in this system, both soft non-linearities:irregularities of the floor, camera's resolution and angle, light of the laboratory and direction of the third wheel, as well as hard non-linearities: Dead-Zone and Saturation. The hard non-linearities are prevented by not exceeding the amplitudes delimiting them, when feeding a certain input signal.

2.1 Sample Time

The default sample time of $T_s = 1/30s$ was the sample time chosen since it is adequate to our system. This is verified by the Nyquist Sample Theorem:

$$Ts \leq \frac{2\pi}{\omega_{max}} \tag{1}$$

The *Rasteirinho* is a slow system so the maximum frequency of $\omega_{max} = 190.4 rad/s$ won't be exceeded if this sample time is used.

3 Deterministic Identification

The goal is to estimate a mathematical model, transfer function, that best describes the dynamics of the system. By feeding deterministic inputs and analysing the output it is possible to understand the behaviour of the *Rasteirinho*, using different identification methods. The inputs chosen were the squared and sinusoidal functions allowing identification of the system in both time and frequency domain.

The output samples are visualized and compared with the reference, extracting the intervals of interest for the analysis. With this data the following identification methods are applied:

- Time Domain
  - Step Response - Deconvolution
- Frequency Domain
  - Levy Method
  - Lissajous Method

3.1 Step Response Method

In order to apply this method, the system was excited with several square functions of different amplitudes and the corresponding outputs saved, creating a pool of time samples with various step responses in each sample. This saved a lot of time, since it would take much longer if the system was fed with single step signals.

The objective was to get a good step response and study it's transient part, which gives the most useful information regarding the system's behaviour such as overshoot, peak time and settling time. With that in mind, each output sample was then compared with the reference and the best step responses extracted. These can be seen in following figure,

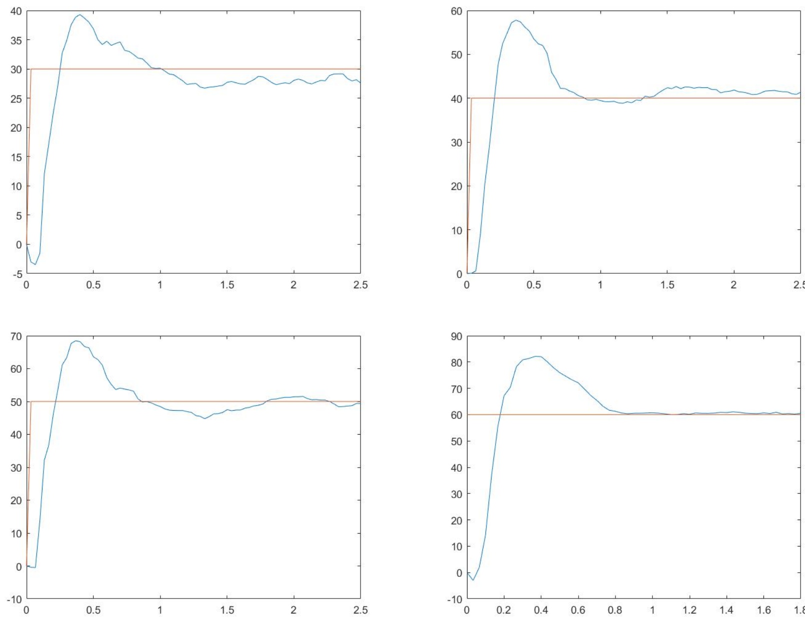

Figure 1: step responses for inputs with 15 and 20 25, and 30 of amplitude

By examining these plots its possible to draw some conclusions. As expected, the system is not a first order system since it has an overshoot, so it has two poles and possibly more. Also, the response observed in the first 0.1 seconds indicates that the system has a non-minimum phase zero.

The fact that the gain of the response varies according to the amplitude of the input signal attests that even though the *Rasteirinho* is a non-linear system, it follows the superposition principle. This is very important because it validates the identification being made, where the system is considered linear.

On the other hand the settling time isn't always the same, as the amplitude of the input changes. This proves the non-linearity of the system.

3.1.1 Deconvolution Method

The deconvolution integral enables a non-parametric estimation of the impulse response of the *Rasteirinho*, using the step responses and corresponding references mentioned previously. With the impulse response it is possible to estimate transfer function in the Z-domain(discrete) which is then translated to the S-domain(continuous).

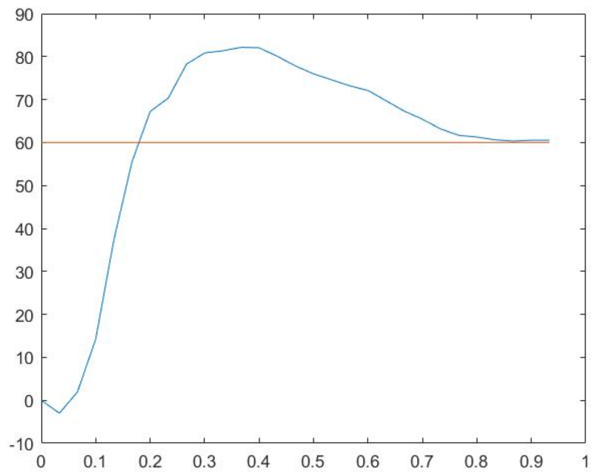

Figure 2: Transient region of the step response with 30 of amplitude

Firstly, the points of interest in the step response are selected by choosing an adequate Sample Time T_s. It will be an input of the Matlab function *Deconv.m* as well as the step signal and respective response. The function then performs the deconvolution integral, outputting discrete data which represents the impulse response.

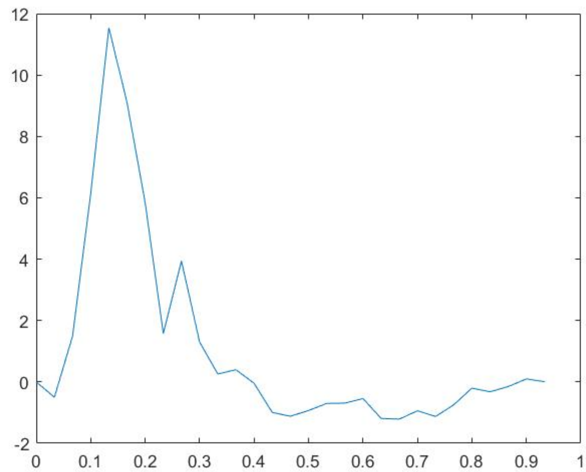

Figure 3: Deconvolution impulse response

With the obtained data a discrete transfer function is estimated with help of the Matlab Function $ir2dtf.m$. To use this function it is required to have a hint about the order of the system at hand to know many points the function will use: $nr.ofpoints = 2 * n + 1$. In the beginning of this Chapter, some conclusions were drawn from the observation of the step responses, telling that the system has at least two poles but most likely three due to it's oscillatory behaviour. The selected sample time was $T_s = 0.066s$.

In the final step, the discrete time system was re-sampled back to $T_s = 0.033s$ and then the function $d2c.m$ was implemented, to go from discrete to continuous time domain.

In order to make a comparison between the deconvolution transfer function and the Levy Method one (4), the estimated transfer function was truncated to a second order system. This simplification was performed with the Model Reducer App from MATLAB and the transfer function was the following,

$$G_{deconv}(s) = \frac{-1.674s + 142.7}{s^2 + 18.39s + 179.6} \tag{2}$$

3.1.2 Levy's Method

The Levy method is a numerical identification method that performs a curve fitting by minimizing the error between the data and the frequency response of the estimated model. The order of the denominator and numerator of the transfer function must be given as input in order for this method to be performed.

This method will be applied to the output of our deconvolution integral, after the Fourier Transform. The Fourier transform will be performed by using the Euler formula,

$$G(jw) \simeq T_0 \sum_{k=0}^{m} g(kT_0) \cos(kT_0) - jT_0 \sum_{k=0}^{m} g(kT_0) \sin(kwT_0) \tag{3}$$

The resulting transfer function along with the its respective bode diagram are shown below.

$$G(s) = \frac{-9.210s + 78.19}{s^2 + 4.471 + 109.4} \tag{4}$$

The transfer function has two poles and one zero because this structure was the one who had a better fitting between the fourier transformation points and the levy method bode diagram, as seen in the following figure,

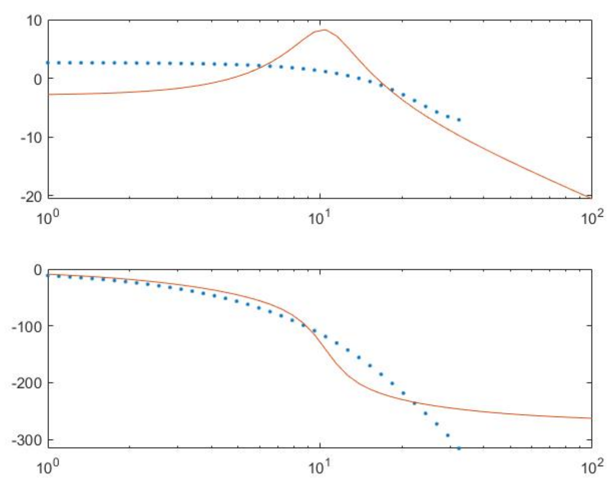

Figure 4: Levy Method Bode Diagram with Fourier Transformation points

After this point, the transfer functions obtained through the Deconvolution method and the Levy method can be compared. This comparison is visible by overlaying both bode diagrams.

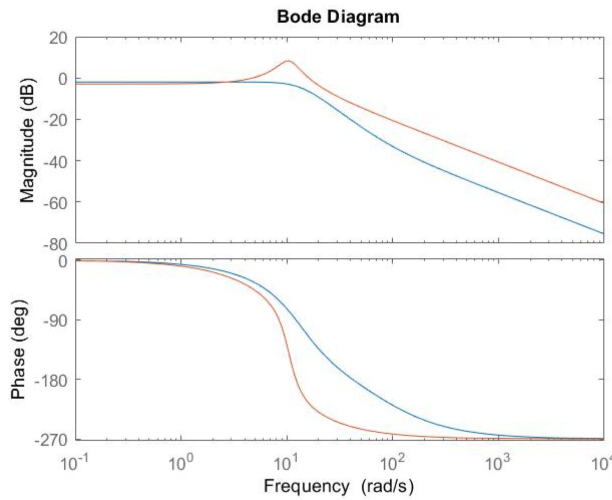

Figure 5: Deconvolution (BLUE) and Levy's Methods (ORANGE) bode diagrams comparison

Observing the above plots its possible to see that the TF estimated with Levy's Method has lower damping than the one estimated through the Deconvolution Method. A displacement on the magnitude above frequency values of $10 rad/s$ will result in a stationary error. A bigger undershoot can also be seen for the Levy Method, being promoted by a zero relatively close to the origin and on the right hand side of the complex plane. All this thoughts can be seen in the comparison made on the Fig.6 below,

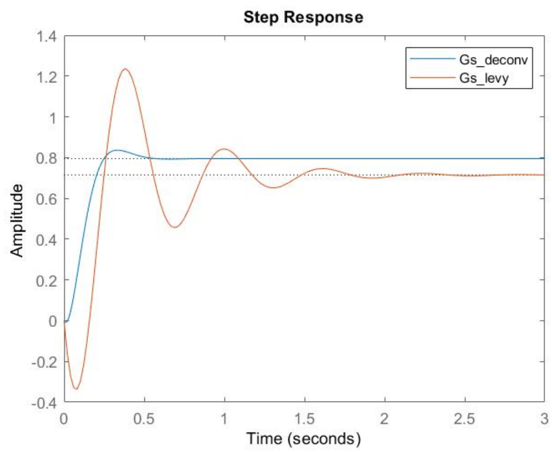

Figure 6: Deconvolution and Levy Methods step responses

3.2 Sinusoidal Input Data Validation

3.2.1 Lissajous Method

The Lissajous Method uses sinusoidal inputs and its corresponding output. Analyzing each sinusoidal response individually, parameters are obtained and processed into frequency, magnitude and phase values.

$$G = \frac{b}{a} \tag{5}$$

$$\phi = arcsin\left(\frac{c}{b}\right) \tag{6}$$

The "a" is the amplitude of the input sinusoid, "b" the amplitude of output and "c" the gap between both parameters. G will be converted in magnitude and ϕ to the phase.

After having all the values, the points obtained from the data are plotted over the bode diagram of Fig.5

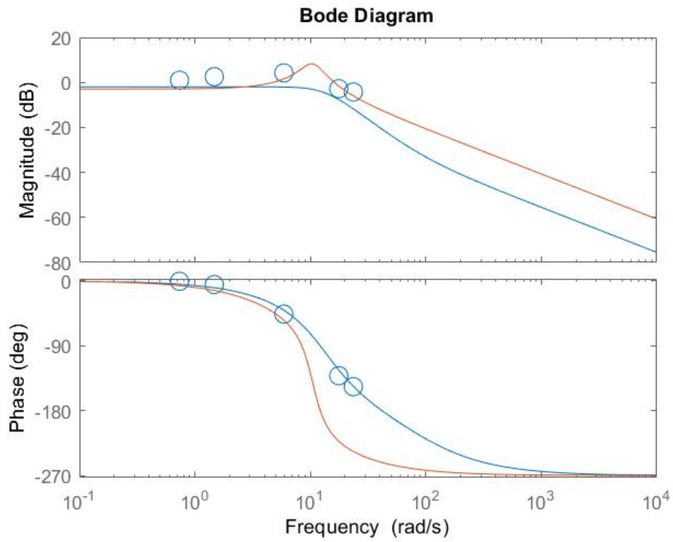

Figure 7: Deconvolution and Levy's Method bode diagrams with validation data points

From the previous figure, it's clear that the points fit better to the deconvolution transfer function bode diagram, even tough there is a constant displacement in gain values above the reference. This problem indicates that there will be an stationary error. From this point on, the deconvolution transfer function (2) will be accepted as the best one so far.

4 Stochastic Identification

In this chapter, the system at hand is analysed as stochastic. This approach is of high importance since it takes into account non-linearities and noise that are neglected in the Deterministic Identification, therefore, it adds more information which results in a description of the *Rasteirinho* that is closer to it's true model.

The inputs are random, so it's not possible to represent them in an exact mathematical function. Logically, the same goes for the outputs. To describe these random variables, as well as their relation, some statistical tools are used such as the Autocorrelation and Correlation Functions. The mean and variance values are also very relevant.

For this identification to be valid, the gathered data must obey certain rules, namely being Stationary (constant in time) and Ergodic (every sample has all statistical information needed).

4.1 Input Signal

Different inputs were given to the system, with noise power ranging from 2 to 12. By inspection, it was possible to conclude that signals with power higher than 8 would disturb the system too much making it lose the black line reference, thus the selected input signal was the one with $power = 4$.

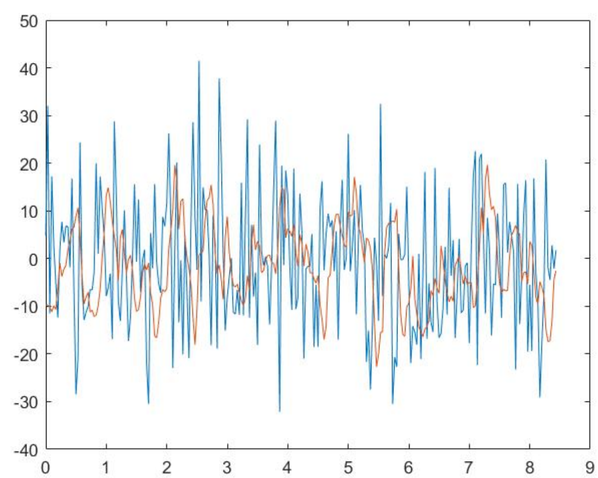

Figure 8: Random input signal (BLUE) and corresponding output signal(ORANGE)

4.2 Input-Output Correlation

In order to obtain the relation between the input and output of the system, the Matlab function *cra* was used. With the input and output of the system, as well as the number of lags, this function outputs the covariance of the input and output and, most importantly, the correlation between them which represents the impulse response of the system. The obtained graphs are shown below.

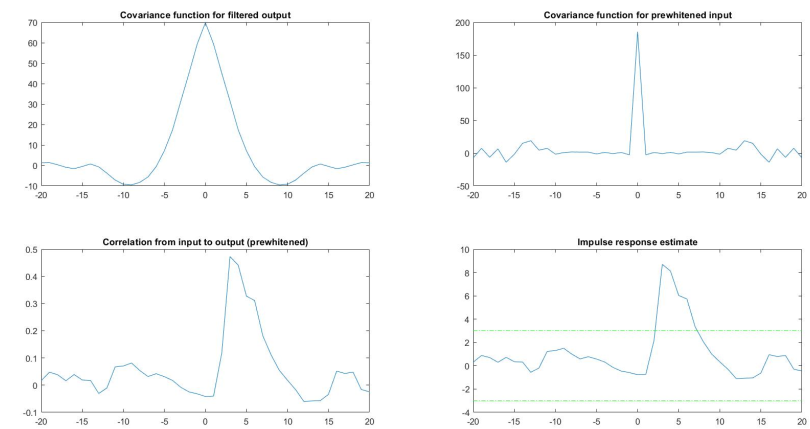

Figure 9: Covariance and impulse response after the cross-correlation between output and input signals

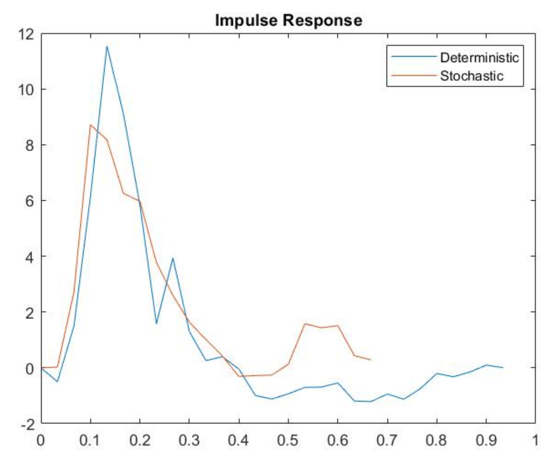

Figure 10: impulse response comparison

4.3 Time Domain Estimation

Once again, to achieve a representation of the model, transfer function, the Matlab functions $ir2dtf.m$ and $d2c.m$ were implemented, using the calculated impulse response. The value of n in $ir2dtf.m$ was 3 (third order system). The result was a fourth order transfer function which was then truncated for simplification purposes, enabling comparison with the other methods and calculation of useful parameters.

$$G(s) = \frac{-3.388s + 148.7}{s^2 + 22.03s + 204.3} \tag{7}$$

4.4 Frequency Domain Estimation

For the stochastic frequency domain estimation, the impulse response data was translated to frequency domain. This was done applying the Fourier Transform by means of a Matlab function that computes the equation presented in section 3.1.2. This function outputs the magnitude and phase of each frequency. As stated before, the *Rasteirinho* is a slow system, therefore only frequencies from 1 to 30 rad/s were considered.

Lastly the Matlab function *levy.m* estimates the transfer function.

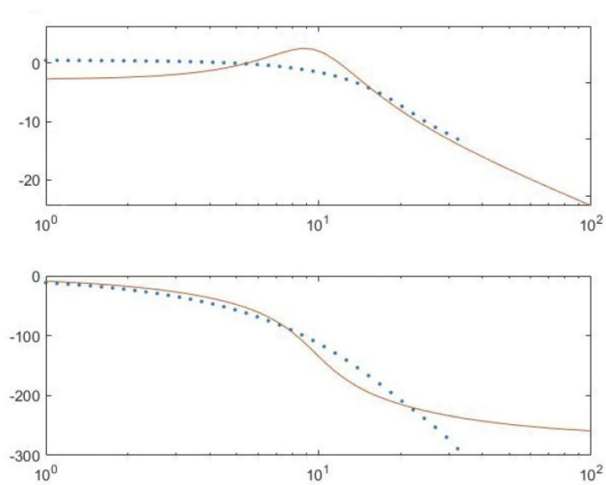

Figure 11: Levy Method bode diagram with Fourier transformation points

$$G(s) = \frac{-5.632s + 69.6348}{s^2 + 6.852s + 93.63} \tag{8}$$

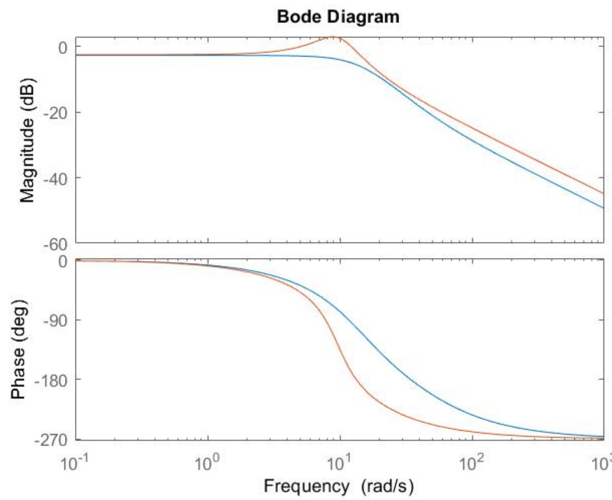

Figure 12: Impulse Response Method (BLUE) and Levy Method (ORANGE) bode diagrams comparison

Observing the above plots its possible to see that it's very similar with the Deterministic approach. The TF estimated with Levy's Method has lower damping than the one estimated through the impulse response but this time there is no significant stationary error. A bigger undershoot can also be seen for the Levy Method, being promoted by a zero relatively close to the origin and on the right hand side of the complex plane. All this thoughts can be seen in the comparison made on the Fig.13 below,

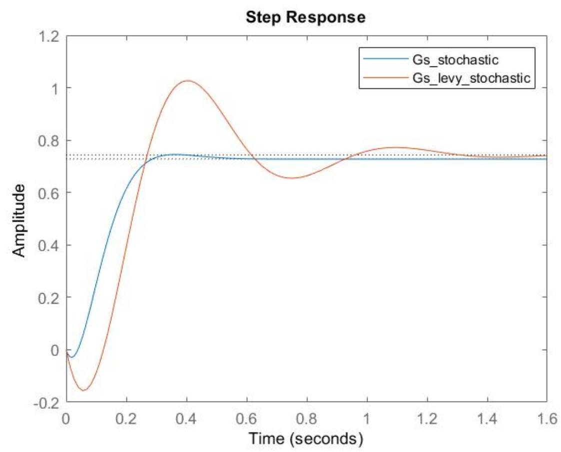

Figure 13: impulse response Method and Levy Method step responses

4.5 Sinusoidal Input Data Validation

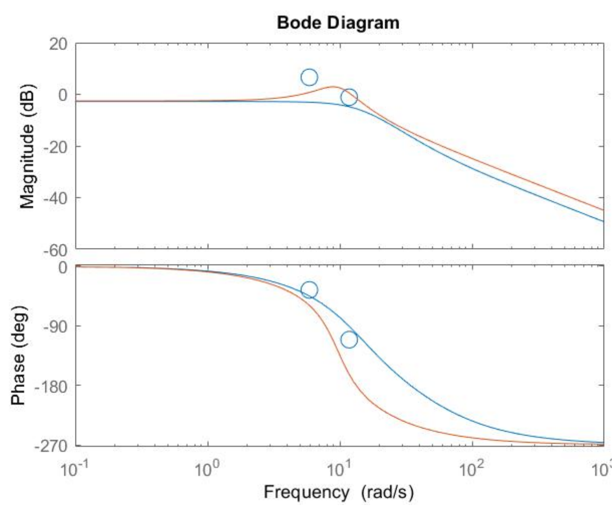

Figure 14: Impulse Response Method and Levy Method bode diagrams with validation data points

After analysing the Fig, 14 the validation points seem to follow the impulse response method bode diagram instead of the Levy Method one. This also occured on the Deterministic approach.

Due to the lack of values, no proper conclusions can be achieved after this analysis. So, the best transfer function will continue to be the one obtained from the Deconvolution method (2) on the previous chapter. Nevertheless, the impulse response method transfer function (7) will be selected as the best one from the Stochastic approach for further comparison.

5 Stochastic Identification - Discrete Models

The System Identification Toolbox, provided by MATLAB, will be used in order to train a model to fit a white noise input. The working data is a random signal with power 4 and the validation is also random signal but with a power of 6.

A model with order higher than five will not represent our system since no transfer functions, estimated previously, were of a fith order system. A threshold of five will be set for the parameter's order. This threshold was set to ensure the dynamic behaviour has real representation.

In section 5.1 the tested models are described and the results of each model output are shown.

In the following sections the validation criteria is used to attest the validity of the models and the best fitting model is chosen.

5.1 Types of Models

5.1.1 Autoregressive Model with Exogenous Input Model (ARX)

This model is the simplest model incorporating the system input signal and noise.

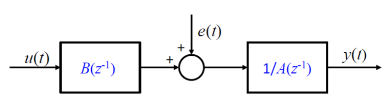

Figure 15: Block diagram of an ARX model

The dynamics of the noise and the dynamics of the system are not independent. For the test data where the input was white noise, the training plot was the following,

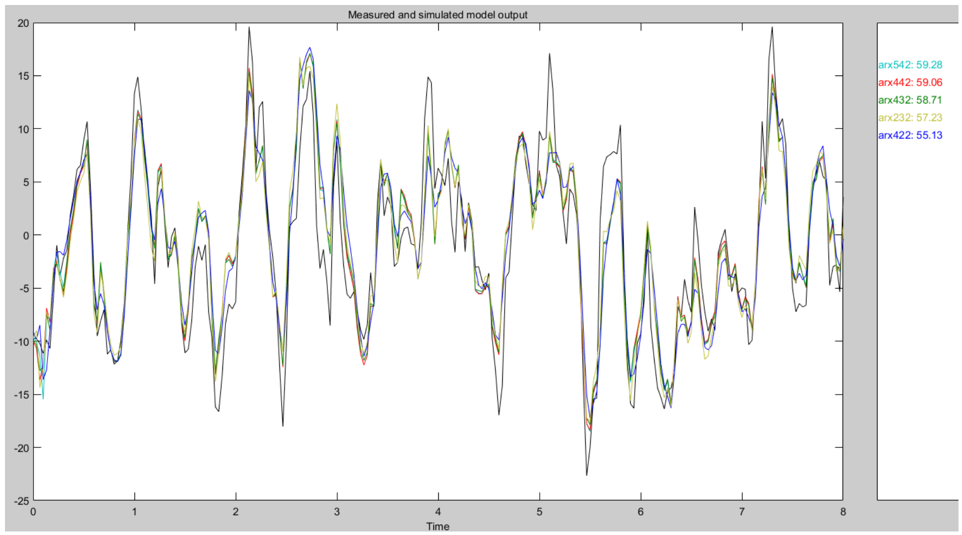

Figure 16: ARX Model Resideus

5.1.2 Autoregressive Moving Average Model with Exogenous Input Model (ARMAX)

This model considers that the noise dynamics contains a moving average component plus the coupling with the dynamics of the system.

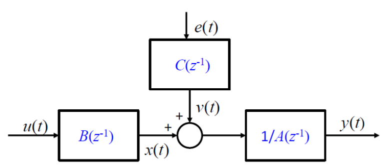

Figure 17: Block diagram of an ARMAX model

This model is specially useful when we have dominant disturbances that enter early in the process.

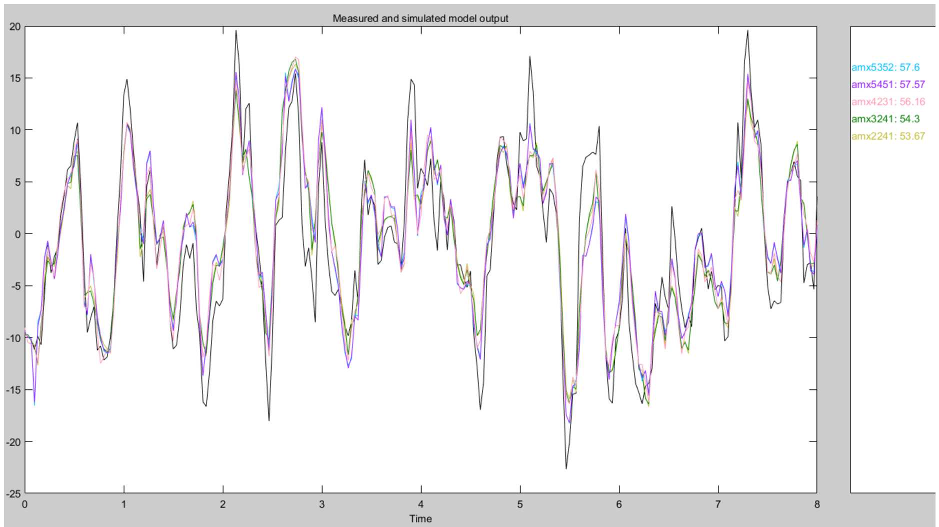

Figure 18: ARMAX Model Ouput

5.1.3 Output Error Model (OE)

OE describes the system's dynamics and disturbances' dynamics separately. No parameters are used for modelling the disturbance.

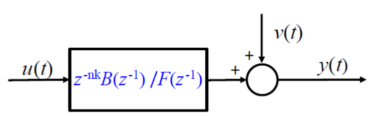

Figure 19: Block diagram of an OE model

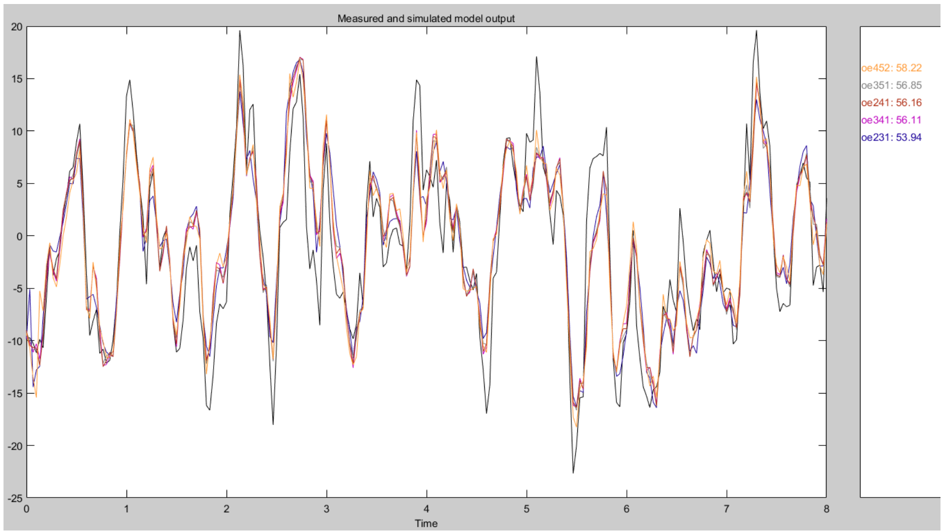

Figure 20: OE Model Ouput

5.1.4 Box-Jenkins Model (BJ)

This model provides a complete model with disturbances dynamics separated from system dynamics.

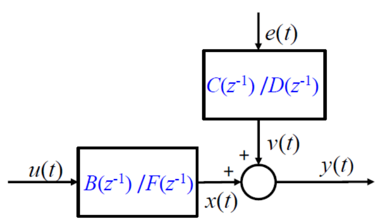

Figure 21: Block diagram of an BJ model

Useful when you have disturbances that enter later in the process.

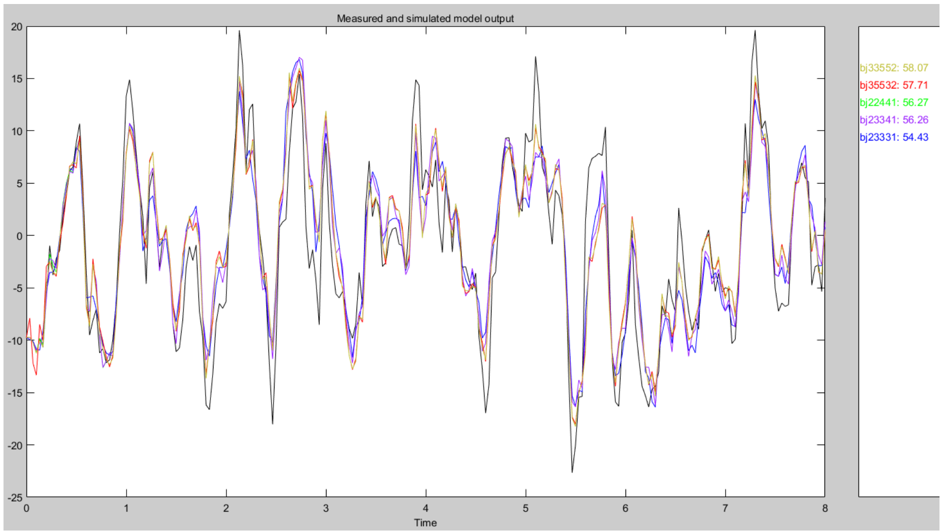

Figure 22: BJ Model Ouput

5.2 Validation Criteria

5.2.1 Akaike information criterion (AIC) and Bayesian information criterion (BIC)

Akaike information criterion (AIC) is based on the theory of information and uses the likelihood function.

$$AIC(m) = n \ln \hat{\sigma_\varepsilon^2} + 2m \tag{9}$$

Bayesian information criterion (BIC) is based on the AIC and the Bayes decision theory.

$$BIC(m) = n \ln \hat{\sigma_\varepsilon^2} + m(1 + \ln n) + m \ln \left[\frac{1}{m} \left(\frac{\sigma_x^2}{\sigma_\varepsilon^2} - 1 \right) \right] \tag{10}$$

The difference is that the between AIC and BIC is that BIC increases the weight associated to the lack of parsimony. Resulting, usually, in a lower order model than the AIC. Due to this, BIC tries to calculate a more representative model of the system.

Model	AIC	BIC	Fit(%)
ARX(4,2,2)	1203.6	1238.4	55.13
ARX(2,3,2)	1177.5	1205.3	57.23
ARX(4,3,2)	1160.9	1199.2	58.71
ARX(4,4,2)	1152.3	1194.1	59.06
ARX(5,4,2)	1149.0	1197.8	59.28

Table 1: ARX evaluation criterion's values

Comparing the best BIC and AIC values, the ARX(4,4,2) only loses 0.22% fitting facing ARX(5,4,2), while prioritizing the BIC criterion. This way, the model is simpler and with almost the same fitting percentage. The chosen model is ARX(4,4,2).

Model	AIC	BIC	Fit(%)
ARMAX(2,2,4,1)	1166.6	1194.5	53.67
ARMAX(3,2,4,1)	1130.3	1175.6	54.30
ARMAX(4,2,3,1)	1073.0	1118.3	56.16
ARMAX(5,4,5,1)	1090.9	1139.7	57.57
ARMAX(5,3,5,2)	1088.8	1134.2	57.60

Table 2: ARMAX evaluation criterion's values

On this case, the best values of AIC and BIC are obtained on the same model. So, the chosen model is ARMAX(4,2,3,1).

Model	AIC	BIC	Fit(%)
OE(2,3,1)	1209.8	1237.7	53.94
OE(3,4,1)	1171.1	1209.5	56.11
OE(2,4,1)	1170.0	1204.8	56.16
OE(3,5,1)	1172.5	1217.8	56.85
OE(4,5,2)	1150.1	1198.9	58.22

Table 3: OE evaluation criterion's values

On this case, the situation is similar to the ARMAX models. Both best values of AIC and BIC are obatined on the same model. The chosen model is OE(4,5,2).

Model	AIC	BIC	Fit(%)
BJ(2,3,3,3,1)	1137.4	1196.7	54.43
BJ(2,3,3,4,1)	1071.3	1137.6	56.26
BJ(2,2,4,4,1)	1069.8	1139.5	56.27
BJ(3,5,5,3,2)	1095.8	1151.6	57.71
BJ(3,3,5,5,2)	1091.7	1147.5	58.07

Table 4: BJ evaluation criterion's values

Comparing the best BIC and AIC values, the BJ(2,3,3,4,1) only loses 0.01% fitting facing BJ(2,2,4,4,1), while prioritizing the BIC criterion. This way, the model is simpler and with almost the same fitting percentage. The chosen model is BJ(2,3,3,4,1).

After selecting the best model of each type, the model with best fitting percentage is ARX(4,4,2) with a fit of 59.06%. This model is still causal even with four poles and 4 zeros. This is due to the fact that the system has two pure time-delays. It was expected that the most complex model would give the best fitting percentage, however, in this case the simplest model presented the best results. This might be explained by the fact that the samples were too short, consequently, the complex models would overfit.

5.2.2 Model Residues Analyzes

The model residues, in this case, will not be a meaningful validation criteria due to the space restrictions on the lab. The small space offered resulted in data acquired during a very small time space. The fact that the collected data is small, results in a non trustful larger confidence interval and in a non compact autocorrelation values, getting often out of that same interval. This reasoning can be observed on the following Fig.23 and Fig,24,

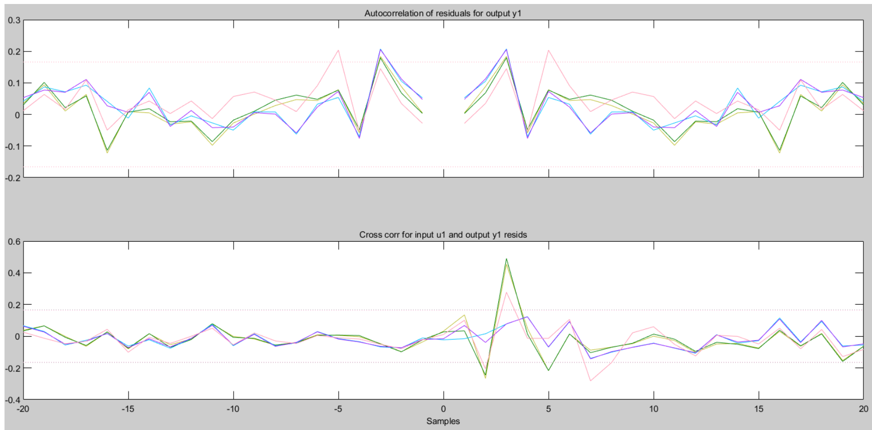

Figure 23: ARMAX Model Residues

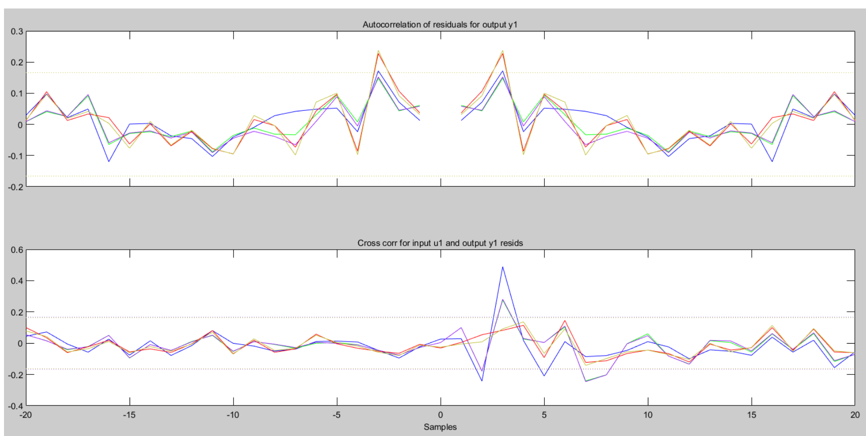

Figure 24: BJ Model Residues

6 Results Analysis and Result Comparison

The Deconvolution Method transfer function was previously selected, through validation, has the best representation of our system. In this chapter, that transfer function along with the second best estimated one will be compared with the best stochastic discrete model obtained through the Identification Toolbox.

First, the step responses of this three methods are shown in the Fig.25 below and, secondly, the detailed information of all curves can be seen,

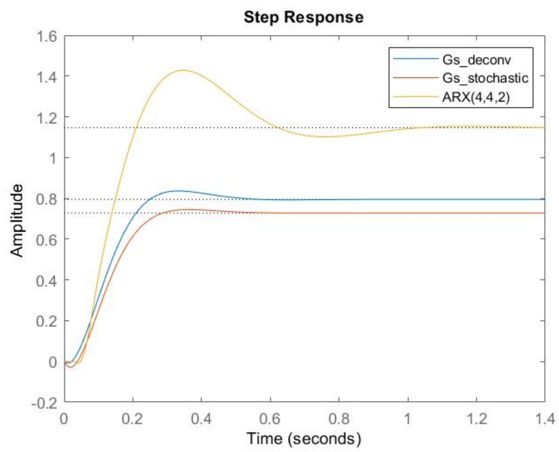

Figure 25: step responses of best models

Gs_deconv	Gs_stochastic	Arx(4,4,2)
RiseTime: 0.1539	RiseTime: 0.1600	RiseTime: 0.1186
SettlingTime: 0.4583	SettlingTime: 0.4003	SettlingTime: 0.9032
SettlingMin: 0.7186	SettlingMin: 0.6573	SettlingMin: 1.0349
SettlingMax: 0.8364	SettlingMax: 0.7450	SettlingMax: 1.4282
Overshoot: 5.2247	Overshoot: 2.3303	Overshoot: 24.5811
Undershoot: 1.0749	Undershoot: 3.9237	Undershoot: 0.6585
Peak: 0.8364	Peak: 0.7450	Peak: 1.4282
PeakTime: 0.3306	PeakTime: 0.3638	PeakTime: 0.3469

Figure 26: step responses information of the best models

Both *Gs_deconv* and *Gs_stochastic* transfer functions have a very similar behaviour. The main diferences are in the under and overshoot, with *Gs_stochastic* having 27.39% more undershoot but with a 55.40% less overshoot. Taking in consideration the restriction of a fifth order threshold for the stochastic discrete model parameters, the best model was an ARX(4,4,2) model. This model only has 59.06% fitting so it was expected for the dynamic behaviour to not be in sync with the selected transfer functions. Even though ARX(4,4,2) has a closer undershoot to the *Gs_deconv* than the *Gs_stochastic*, it has a 50.74% bigger settling time, and stationary error 30.87% of and a 4,7 times bigger overshoot.

7 Conclusion

The *Rasteirinho* is a slow system so frequencies higher than 30rad/s will not show a proper behaviour of the system dynamics.

The best validated transfer function obtained with a deterministic approach was the Deconvolution Method one (*Gs_deconv*).The best transfer function with a stochastic approach was the impulse response method one (*Gs_stochastic*). Both non validated Levy Methods had similar behaviours with high overshoot and and settling time. Lastly, the best stochastic discrete model obtained was a ARX(4,4,2).

The *Rasteirinho* is a system with a highly nonlinear dynamics behaviour. Even with a low frequency and small amplitude, the steps in the Fig.1 don't follow the reference properly. The fact that the camera used is a webcam poorly attached to the computer that also oscillates with the screen when the *Rasteirihno* starts moving forward and when turning don't allow proper data collection. The change in direction is the dynamics that we tried to identify so the collected data was affected by this non-linearities as well.

Lastly, even with all the thresholds and restrictions imposed during the method and the limiting factors of the system and the lab itself, the *Gs_deconv* accepted has a valid representation of the system and *Gs_stochastic* got somewhat similar/close dynamics to the system.

References

[1] Sá da Costa, J., *Notas de apoio à unidade curricular: Identificação de Sistemas*. IST, 2011.

[2] Ljung, L., *System Identification: theory for the user*. Prentice Hall, 2nd edition, 1999.

[3] *MATLAB* & *Simulink*. Student Version 2010a, 2010.

[4] *System Identification Toolbox Documentation*. URL: *https://www.mathworks.com/help/ident/*