Laboratório de Modelagem, Análise e Controle de Sistemas Não-Lineares
Departamento de Engenharia Eletrônica
Universidade Federal de Minas Gerais
Av. Antônio Carlos 6627, 31270-901 Belo Horizonte, MG Brasil

Fone: +55 3499-4866 - Fax: +55 3499-4850

Non linear Data-Driven Contol: Structure Selection of Controllers and Use of Auxiliary Information

João Carlos Vilela de Castro

Texto de qualificação submetido à banca examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Luis Antonio Aguirre, Ph.D.

Belo Horizonte, março de 2021

Contents

Resumo

Em procedimentos convencionais o projeto de controladores é feito a partir de um modelo matemático que representa o processo a se controlar. Outra abordagem, que difere em essência da convencional, com estudos crescentes nas últimas décadas, é a de projeto de controladores baseado em dados (DDC do inglês Data-Driven Control). No DDC, o projeto do controlador não faz uso direta ou indiretamente do modelo do processo e todo o projeto é feito a partir de dados amostrados diretamente do processo. Grande parte das técnicas DDC são métodos iterativos baseados principalmente no método do gradiente para minimizar algum índice de custo. Contudo algumas técnicas, em especial a VRFT, do inglês Virtual Reference Feedback Tuning, permitem, por um procedimento em batelada, realizar a minimização deste índice a partir de técnicas usuais no âmbito da identificação de sistemas. Em geral estes procedimentos são feitos fixando-se uma estrutura para o controlador, e a partir de métodos de identificação e otimização procura-se pelo melhor conjunto de parâmetros para o controlador que resulte em um comportamento próximo ao definido por um modelo de referência. Nesta pesquisa procura-se por uma abordagem alternativa, em que a melhor estrutura do controlador seja selecinada no processo de sintonia do controlador. Neste sentido tem-se buscado um método para auxilio na seleção da melhor estrutura do controlador a partir de uma estratégia de controle VRFT para sistemas não lineares. Esta seleção é feita por uma abordagem aleatorizada de seleção de estruturas de modelos recente no âmbito de identificação de sistemas e que é, neste trabalho, adaptada para lidar com identificação de controladores. Como um texto de qualificação, alguns resultados prévios alcançados são estudados, e propostas para continuidade do trabalho são apresentadas.

Palavras-chave: Controle baseado em dados; Controle livre de modelo; Aprendizado por reforço; Seleção de estruturas; Sistemas não lineares.

Abstract

In conventional procedures, the design of controllers is based on a mathematical model that represents the process to be controlled. Another approach, which differs in essence from the conventional one, with increasing studies in the last decades, is the design of data-driven controllers (DDC). In the DDC approach, the controller project does not make direct or indirect use of the process model and the entire project is made from data sampled directly from the process. Most of the DDC techniques are iterative methods based mainly on the gradient method to minimize some cost index. However, some techniques, especially the Virtual Reference Feedback Tunning (VRFT), allows, through a batch procedure, to minimize this index using the usual techniques in the scope of systems identification. In general, these procedures are done by setting a structure for the controller, and based on identification methods, the best set of parameters for the controller is identified, aiming to results in a behavior close to that defined by a reference model. In this research, an alternative approach is proposed, in which the objective is to select the best structure of the controller, before the process of tuning or identification of the controller's parameters. In this sense, a method has been developed to assist in the selection of the best controller structure based on a VRFT control strategy for non-linear systems. This selection is made with a modified version of a recent published randomized model structure selection approach used in model identification. An analise of the structure selection problem is made in sense of controller structure selection and parameter identification. As a qualification text, some preliminary results achieved are studied, and proposals for further work are presented.

proposta

Keywords: Data-Driven Control; Model-Free Control; Reinforcement Learning; Structure Selection; Nonlinear Systems.

List of Figures

List of Symbols

Chapter 1

y_k Output signal $\in \mathbb{R}^p$, at time k;

Chapter 2

N number of sampled data;
N_p number of sampled models;
$\mathcal{I}$
$\mathcal{I}_j^+$
$\mathcal{I}_j^-$
$\mathcal{J}$ performance index
$\mathcal{M}$ universe set of all considered possible models
$\tilde{\mathcal{M}}$
$\mathcal{R}$ set of all model candidate regressors
f^*
$\tilde{f}$
$\mathbb{E}[\cdot]$ expected value operator

Chapter 3

Chapter 4

u_k vector input signal $\in \mathbb{R}^m$, at time k;
u_k vector input signal $\in \mathbb{R}^m$, at time k;

Chapter 5

List of abbreviations and acronyms

AIC	Akaike's Information Criterion;
AICc	Corrected Akaike's Information Criterion;
ARMAX	AutoRegressive Moving Average model with eXogenous inputs;
ARX	AutoRegressive model with eXogenous inputs;
BIC	Bayesian Information Criterion;
CLS	Constrained Least Squares
EP	Error Prediction;
ERR	Error Reduction Ratio;
FPE	Final Prediction Error;
ILS	Iterative Least Squares;
IV	Instrumental Variables;
LTI	Linear Time Invariant;
LTV	Linear Time Variant;
MA	Moving Average;
MACSIN	Modeling, Analysis, Control and Nonlinear Systems Group;
MS1PE	Mean Square Prediction Error for one step ahead;
MSPE	Mean Square Prediction Error;
MSTE	Mean Square Tracking Error;
NARMAX	Nonlinear AutoRegressive Moving Average model with eXogenous inputs;
NARX	Nonlinear AutoRegressive model with eXogenous inputs;
OLS	Ordinary Least Squares;
PPGEE	Graduate Program in Electrical Engineering;
PRBS	Pseudo Random Binary Signal;
RIP	Regressor Inclusion Probabilitty;
RLS	Recursive Least Squares;
RMSE	Root Mean Error Square
RaMSS	Randomized Model Structure Selection;
RaCSS	Randomized Controller Structure Selection;
SISO	Single-Input Single-Output;
SNR	Signal Noise Ratio;
UFMG	Universidade Federal de Minas Gerais;
p.u.	Per-unit

Chapter 1

Introduction

The use of feedback control in mechanisms developed by humans is marked by the 1769 James Watt's invention, known as the Watt regulator and developed to regulate steam-machines spin velocities. From this time until the beginning of the 20th century, control designs were based on trial and error methods. With the emergence of theoretical publications on the subject, such as that of (Tolle, 1921), mathematical models were increasingly used in the design of controllers, mainly in the form of differential equations (Takahashi et al., 1972).

In the 30s and 50s the so-called Classical Control Theory originates, expressing itself basically in the frequency domain and in the s-plane, with models given by transfer functions, based on methods developed mainly by Nyquist, Bode, Nichols and Evans.

In the 1960s, a new control theory approach arises, using parametric models and state space representation. this approaches gives rise to the so-called Modern Control Theory and its main branches, such as systems identification, adaptive control, robust control, optimal control and stochastic control, which have been widely studied and developed until today, but still with many challenging topics, both in theoretical and practical aspects.

In both approaches, the classical control theory, mainly based on the use of transfer functions and linear systems, as in the modern control theory, mainly based on state space representations of linear and non-linear systems, a mathematical model of the process to be controlled is required*. Such model can be obtained via phenomenological modeling, or via systems identification methods. In the former case, the model is obtained using known laws from specific fields of science resulting in equations that represent it. In the latter case, using input-output data collected from the process and using systems identification techniques, models that represent the process are obtained, with a certain degree of reliability.

Several methodologies for identifying linear and non-linear models are available in the literature (Aguirre, 2015; Ljung, 1999).

Models obtained using first principles or even by systems identification can result in high order models, with a high degree of non-linearity, which makes difficult or even impractical their use for control purposes. Furthermore, modelling processes can be an arduous task and sometimes even impracticable, requiring steps to validate and

*with some exceptions like the cases where the controller is designed directly from the frequency response obtained experimentally.

determine the structure of the model.

For this reason, traditional model-based control methods (MBC) are unpractical in some cases. Besides, several processes generate and store large amounts of data and the use of this data for controller design would be very convenient (Hou and Wang, 2013).

Since the input and output data of a plant contains information about its dynamics, as long as it is properly excited, it may seem unnecessary to apply the identification theory to obtain a mathematical model of the plant for controller design (Ikeda et al., 2001). Furthermore, in an attempt to obtain a model that is faithful to the behaviour of the process, a very complex model can be arrived at, and a process of order reduction may be necessary during the controller design. In this case, additional effort in identifying the model may be unnecessary when designing the controller.

In this sense, in several practical control cases in which a mathematical model describing the plant is not available, or is too complex or the uncertainty in the model is too great for the use of MBC strategies, it is very convenient to obtain the controller from measurements obtained directly from the plant.

According to Campi et al. (2002), this problem has attracted the attention of control engineers since the work published by Ziegler and Nichols (1942), and several extensions have been proposed since then. Such procedures, despite being similar to trial and error procedures, were widely used in the industry, perhaps due to their simplicity of design, even if at the expense of final performance losses.

Around the 1990s, new approaches to controller design without the use of models for plants began to appear in the literature, which later came to be called control based on data (DDC). Hou and Wang (2013) claim that the term *data-driven* was first proposed in computer science and only recently entered the vocabulary of the control community and, to date, there are some DDC methods knouwn by different names, such as " *data-driven control* ", " *data-based control* ", " *modeless control* ", among others. Hou and Wang (2013) propose the following definition for DDC, based on 3 other definitions:

Definition 1.1 (Data-Driven Control)**.** (Hou and Wang, 2013) Data-driven control includes all control theories and methods in which the controller is designed by directly using on-line or off-line I/O data of the controlled system or knowledge from the data processing but not any explicit information from mathematical model of the controlled process, and whose stability, convergence, and robustness can be guaranteed by rigorous mathematical analysis under certain reasonable assumptions.

Therefore, the DDC is different from the MBC in essence, since the controller design does not make direct or indirect use of the process model. Although at first, they look like adaptive control methods, DDC methods differ from these in that, at first, they do not need any model information, and parameter settings depend on large batches of data, instead of only a few samples of the input-output signals.

Another topic that has been increasing attention in the systems identification and control community is the use of methodologies based on Monte Carlo-style approaches

for choosing the appropriate structure or even for identifying process model parameters. The following section talks a little about the study of this problem in the academic community in recent years and the chapter 4 presents a method of choosing parameters that is of great interest in this research.

In order to unite what has been developed in the sense of systems identification to design procedures of DDC controllers, where it is often intended to identify models for controllers, this research has been moving towards the development of procedures that assist in the selection of structure for the controller models during a DDC-style control project.

The next section provides an overview of the state of the art with regard to DDC procedures, as well as model structure selection procedures in recent years.

1.1 Objectives

As mentioned at the end of the last section, the main objective of this research is to study the controller structure selection in a procedure of the DDC type. To achieve this objective, two recent approaches in control and systems identification area are used. The first concerns the design of controllers by the VRFT procedure, developed by Campi et al. (2002) for linear cases and extended to non-linear cases in Campi and Savaresi (2006). In this approach, the controller is identified by common procedures systems identification community, based on batches of data collected of the process to be controlled. The other is a model structure selection (MSS) approach for model identification, proposed by Falsone et al. (2014), where a randomized procedure is used in order to try to select the best structue to represent a model on which the parameters are to be identified. the intention is to study In this research, the intention is to study the use of this randomized approach for choosing models within the scope of the DDC controller project, more specifically, in the VRFT style.

Other objective, although a secondary one are to investigate the use of auxiliary information, or its analogue, in order to guarantee aspects relevant to control, such as gain limitations due to actuator saturation, or imposition of an integrative effect on the controller to reduce errors on a permanent basis, or even aspects related to robustness.

Another secondary objective, that may be investigated, is the use of reinforcement learning techniques, an approach that has had increasing advances in the control community in order to make the framework proposed in this research applicable to real-time control systems.

1.2 Motivation

The VRFT method, discussed in more detail in Chapter 3, has great appeal in the academic community because it is a DDC approach, and as such, it does not need

a plant, or process, model for controller design. The basic philosophy is: instead of spending effort identifying an accurate model for the plant, that often needs to be simplified for proper use in the controller design, why not try to directly identify the controller model from data collected from the process input and output that, later processed, allow such identification?

The VRFT procedure shows that this is possible, mainly in cases where the structure of the controller to be tunned, is a specific structure, which solves the matched control problem (more detail in Section 3.2). In cases where this is not possible, which occurs more frequently in practice, since the plant model is not known, the method proposed by Campi et al. (2002); Campi and Savaresi (2006) proposes the use of filters that approximate the expected results, as long as the structure chosen for the controller is not too under parametrized. This solution is attractive for tuning procedure of previously available controllers. An example are controllers from the PID family, widely used in industries, where it is desired to tune parameters for a previously existing controller in a CLP bock, for instance.

However, there are cases in which such controllers are not sufficient to guarantee the behavior with the desired level of accuracy, or even to guarantee stability. Thus, more complex structures for the controller should be studied, such as, for example, non-linear models. However, trying to find a adequate structure for the controller that guarantees a behavior close enough to a desired reference model can be an arduous task.

In this sense, the present work aims at the development of a methodology that helps in the choice of the structure of the controller, using a randomized approach. In this way, instead of fixing the controller structure and adjusting its parameters, a more suitable structure is selected first so that the controller can be tuned, thus guaranteeing a greater degree of freedom and a closed-loop system behavior closer to the wanted.

It is intended to develop a framework capable of assisting in the controller structure selection based on recent techniques for MSS used in the scope of systems identification. So far, the author has not identified any work in the literature that deals directly with this problem. Only works for the selection of process structure (as presented in the section 1.3) are founded. Thus, it is believed that the final results can be of great value for the design of controllers in a DDC approach.

1.3 State of the art

Some conceptually distinct approaches using DDC appear in the literature in the last years, among them[†]: *Virtual Reference Feedback Tuning* (VRFT), Iterative Feedback Tuning (IFT), *Optimal Controller Identification* (OCI) (Campestrini et al., 2017), *Correlation*

[†]it was chosen here to mention some techniques that the author found most relevant to this proposal, however others can be found in the literature (Spall, 1992; Safonov and Tsao, 1995; Karimi et al., 2007; Huang and Kadali, 2008; Schaal and Atkeson, 1994; Shi and Skelton, 2000)

Based Tuning (CbT), originally presented by Campi et al. (2002), Hjalmarsson et al. (1994), Kammer et al. (2000) and Karimi et al. (2002), respectively.

Most of these methodologies use the concept of optimization from the minimization of a cost function, in general, measured in terms of the H_2 norm of a signal. Several DDC methods available in the literature do this optimization in an iterative way, among them, the IFT, CbT, ILC, ADP. Others do so in batches, such as the VRFT, OCI and *Noniterative data-driven model reference control* methods.

In iterative cases, the minimization of the cost function is done, typically, by gradient descent methodologies, from input-output data collected in a batch way (Bazanella et al., 2008). One drawback of these methodologies is the lack of conditions that guarantee convergence to a global minimum for the cost function in many cases. Extensions to improve the convergence properties and even reduce the number of required in-process experiments have been the subject of studies in last years (Huusom et al., 2009).

In non-iterative cases, convergence to a global minimum is generally not an issue.

The VRTF method, presented by Guardabassi and Savaresi (2000); Campi et al. (2002) and the OCI, presented by Campestrini et al. (2012) and anhanced by Campestrini et al. (2017), to deal with the design of linear SISO systems and than extended to deal with multivariable (Da Silva et al., 2018; Huff et al., 2019) and nonlinear systems (Campi and Savaresi, 2006), are examples of this non-iterative cases.

In order to solve the problem of convergence to a global minimum of a H_2 performance criterion, the VRFT focus on making the cost function be optimized sufficiently "well behaved" making optimization converge properly.

At first, given ideal conditions, convergence to the global minimum is not a problem when using the VRFT method, as it is a batch method. In addition, VRFT has no initialization problems and does not access the plant several times for experimentation, in contrast to iterative methods, allowing to maintaining the normal process operation. Extensions for non-linear controllers designs have been proposed since then (Campi and Savaresi, 2006; Jeng and Lin, 2014, 2018).

Due to its attractive characteristics, the VRFT has been used as one of the main methodologies in this work. This approach formulates the controller tuning problem as an identification problem via the introduction of a virtual reference signal. The control objective is to minimize a cost function given by the H_2 norm of the difference between a closed loop transfer function output and a reference model output, to the same reference signal r. The problem of finding the minimum is that there is no model available. In order to get around this problem, the concept of virtual signals is used. These signals, given by e^{vir} (virtual error) and u^{vir} (virtual control signal), are created from the output signal of the plant and the inverse reference model, enabling the use of a new cost function given by $J_{vir} = \|C(\theta, z^{-1})e^{vir} - u^{vir}\|$, in that $C(\theta, z^{-1})$ represents the controller model whose parameters θ must be identified by optimization. Campi et al. (2002) show that by minimizing J_{vir}, the first criterion is minimized under certain conditions. The minimization of the new functional can be done by techniques like

least squares estimators (MQ), instrumental variables (VI), among others (Aguirre, 2015). Bazanella et al. (2012) show examples of the use of instrumental variables to solve the problem of polarization of the parameters identified for cases of noisy signals.

So far it is possible to find extensions of the VRFT method for application to non-linear systems in the literature (Previdi et al., 2004; Campi and Savaresi, 2006; Wang, 2011; Bazanella and Neuhaus, 2014; Yan et al., 2016; Radac et al., 2018). But unlike VRFT linear, these extended versions for non-linear systems are either not batched, or their solutions cannot be determined by LS methods, losing a considerable advantage of VRFT (Jeng and Lin, 2018). However, Jeng et al. (2015) show that VRFT can be extended to control non-linear systems of the Hammerstein and Wiener type so that the non-iterative characteristic of VRFT is maintained and presents a method where only static nonlinearity (or its inverse), represented by *B-spline* series, is estimated simultaneously with the controller without the need for nonlinear optimization or procedures iterative. A drawback is that this approach is applied only to systems that can be represented by Hammerstein and Wiener models.

In general, the available methodologies in the literature are approaches in which a structure for the controller is previously defined and later the VRFT approach is applied. A question that arises is: wouldn't it be advantageous to have a methodology that helps in choosing the best structure for the controller in order to achieve the objective imposed by the reference model? This question refers to a problem that is not very recent in the systems identification area, which has been studied since the last century: the model structure selection (MSS) problem, where aim is to select the best structure to the model to be identified. The first studies in this sense are based on information indices that try to quantify how much information the inclusion of new terms (regressors) brings to the model in relation to the increase in complexity (number of terms) of the model. The most known indexes are the Akaike's Information Criterion (AIC) Akaike (1974), the Bayesian Information Criterion (BIC) (Schwarz, 1978) and the Final Prediction Error (Kashyap, 1977).

Other approaches for selecting structures based on different principles from the previous ones arise from the use of information such as the Error Reduction Ratio (ERR) (Billings et al., 1989). Methodologies similar to ERR have emerged, such as the case of the Simulation Error Reduction Ratio (SRR) (Piroddi and Spinelli, 2003) and ERR_2 (Alves et al., 2012) method.

More recently, methods based on Monte Carlo-style approaches have been proposed. One of them is the RJMCMC procedure proposed by Baldacchino et al. (2013) which uses a Bayesian approach that, based on sampled data, derives distributions for the model structure as well as for its parameters. Another is the method proposed by (Falsone et al., 2014, 2015), in which the MSS is made by a randomized approach, in which candidate regressors are ranked by probabilities of being the best candidates, at each iteration of the method. The process is done in such a way that a much smaller number of models need to be analyzed when comparing with a purely "Monte Carlo"

strategy.

In this sense, this work intends to extend this problem of MSS, already common in the means of system identification, for the controllers structure selection, a task which seems not to have been much studied until then.

Outra pergunta que surge é: seria possível o uso de técnicas de estimação do tipo MQ com restrições para modelos não lineares NARX (do inglês *Nonlinear model with eXogenous inputs*) ou MQ estendido para modelos NARMAX (do inglês *Nonlinear AutoRegressive Moving Average model with eXogenous inputs*) neste tipo de abordagem? Apesar de já terem sido desenvolvidas técnicas para incorporar informação auxiliar no processo de identificação, por exemplo via restrições e otimização multiobjetivo (Barroso, 2006), todas estas restrições dizem respeito à planta. Neste sentido surgem questões como: de que forma estas técnicas podem ser usadas na abordagem DDC?

Chapter 2

Basic concepts

2.1 Basic concepts in system identification

In order to describe natural phenomena or even mechanisms and processes created by the humans, over the centuries, different ways of representing such phenomena through mathematical expressions, known as mathematical models, or simply models, have been developed.

To obtain such models, in general, two approaches can be used: modeling by the first principles, or modeling by systems identification techniques. In the first case, the models are obtained from applications of laws, in general of physics, developed and documented over the years of observations of phenomena, natural or not, by scientists from the most diverse areas. In the second case, mathematical models are obtained from analyzes made on signal data collected from the system to be modeled using identification techniques developed for this purpose. In both cases, the models obtained are mathematical expressions that describe the approximate behavior of the modeled process.

In the present work, it is not of interest to obtain a model for the process, but a model for the controller. However, the methodologies used to identify systems can be used directly, or in some cases, with specific adaptations for the purpose of identifying controllers. In this sense, identification techniques used in the scope of model identification have been used in the design of data-driven controllers, targeting both linear (Campi et al., 2002) and non-linear (Campi and Savaresi, 2006) controller models.

Over the years, with the increase in the computer's processing power as well as the ease in the acquisition and storage of data, the use of techniques for identifying nonlinear models has increasingly shown to be interesting for predictions or even for better understanding of phenomena. Likewise, it is expected that non-linear controllers designed by data-oriented control techniques will often result in controllers with better performance or even greater robustness.

From a few decades ago, the task of modeling nonlinear systems has been studied, where some great works stand out (Billings, 1980; Leontaritis and Billings, 1985a,b; Korenberg et al., 1988; Billings et al., 1989; Chen et al., 1990; Chen and Billings, 1992; Aguirre and Billings, 1995; Aguirre et al., 2000; Zhu, 2005). The system identification process basically consists of the following steps: (1) Data collection; (2) Choice of model

type; (3) Structure selection; (4) Estimation of parameters; (5) model validation. The following sections briefly address each of these steps.

2.1.1 Data Collection and Pre-processing

The first step in the identification of systems models is the data collection. In this process, some care must be taken regarding the sampling interval considered when collecting signals, from the system to be identified. Care should also be taken when choosing the signal used to excite the process, when possible. The input signals must be designed in such a way as to excite the dynamics of the system in the frequency range of interest, through the choice of signals with adequate spectral powers. In this case it is said that the signal must be *persistently exciting*. Signals such as filtered white noise or binary pseudo-random signals (PRBS) are commonly used in practice.

The proper choice of the amplitude of the input signal is also a factor that deserves caution. For example, the signal should not be such as to cause the process response to exceed certain thresholds, close to an operating point, which guarantee a behavior close to linear in the case of the identification of linear models. Likewise, when the identification is of a non-linear model, the amplitude must be such that it explores the non-linear characteristics of the process. Problems such as oversampling, outliers, or cases in which the excitation signals cannot be previously chosen, can be resolved or mitigated by prior treatment of the data, a process known as pre-processing.

2.1.2 Choice of Model Class

There are several model classes that can be used to describe the input-output relationship of a process. These classes have different structures that are more or less suitable for a particular application. For example, for non-linear systems, the structure of the model must be complex enough to represent the non-linearities of interest. Among the several usual classes in the representation of models, the following stand out: base radial functions (Broomhead and Lowe, 1988), neural networks (Haykin, 1994), wavelets (Strang, 1989), Volterra series (Billings, 1980), polynomials and rationals functions (Billings et al., 1989).

2.1.3 Structure Selection

Once the class of a model is defined, choosing its structure (i.e., number, degree, time lags, of terms in a NARX polynomial model) becomes the next task, before identifying its parameters. This can be a difficult task, since we want to find a model with the lowest possible variance and polarization (bias), and these two quantities are contradictory. In short, the model must be rich enough to capture the dynamics and repeat the behavior of the modeled system, but not so much as to model the noise present in the sampled

signals. If the model is very simple, it may not fit well with the training data, and if too complex (with many terms), it can affect the behavior prediction for data other than those used in training.

In an attempt to find a solution to this problem, approaches have been presented in recent decades. Among them, the Akaike's Information Criterion (AIC) gained attention, as well as its corrected version, (AICc). The AIC, was introduced by Akaike (1974), and is defined as

$$AIC(n_\theta) = N \ln[\sigma^2_{\text{erro}}(n_\theta)] + 2n_\theta, \tag{2.1}$$

where N is the number of sampled data, $\sigma^2_{\text{erro}}(n_\theta)$ the variance of the residue and $n_\theta = \dim[\hat{\theta}]$ the number of parameters of the model. According to Aguirre (2015), (2.1) can be analyzed by the following: "as terms are included in the model, the number of degrees of freedom n_d increases, allowing a more accurate adjustment to the data. Thus, $\sigma^2_{\text{erro}}(n_\theta)$ decreases as n_d increases . . . ", but ". . . after a certain moment, the decrease in the variance of the residuals resulting from the inclusion of a new term is insignificant and would not justify the inclusion of the respective term ". In short, the first term of (2.1) quantifies the decrease in the variance of the residuals due to the inclusion of a term, whereas the second term penalizes the inclusion of each term.

In order to correct a problem presented by the AIC that increases the probability of the AIC to select models with a high number of parameters when the sample size is small, which leads to an over-adjustment, the AICc appears (Cavanaugh, 1997), given by the equation

$$AICc = AIC + \frac{2k^2 + 2k}{n - k - 1}.$$

Other criteria similar to AIC can also be found in the literature, among them, the Bayesian Information Criterion (BIC), (Schwarz, 1978); and the Final Prediction Error (FPE) (Kashyap, 1977).

Other approaches, already considered classic, that differ from the previously mentioned criteria can be found in the literature, with emphasis on the Error Reduction Ratio (ERR) (Billings et al., 1989). In this approach, the reduction in noise variance that occurs when a new term is included in the model is quantified and normalized with respect to the output variance. The ERR resulting from the inclusion of the i-th regressor is given by

$$[\text{ERR}]_i = \frac{MS1PE(v_{i-1}) - MS1PE(v_i)}{\langle \mathbf{y}, \mathbf{y} \rangle},$$

where $i = 1, 2, \ldots, m$ and m is the number of tested candidate terms; MS1PEv_i the one-step ahead error of the model with i terms, or regressors; and v represents a family of models with nested structures such that $v_{i-1} \subset v_i$.

Extensions to the ERR criterion are possible, such as ERR_2 (Alves et al., 2012), which uses 2-steps ahead prediction instead of 1. Another criterion similar to the ERR is the

Simulation Error Reduction Ratio (SRR), introduced by (Piroddi and Spinelli, 2003), and which is advantageous in non-ideal conditions, often resulting in more compact models, but with higher computational costs. It is given by

$$[\text{SRR}]_i = \frac{MSSE(v_{i-1}) - MSSE(v_i)}{\langle \mathbf{y}, \mathbf{y} \rangle},$$

where MSSEv_i now represents the mean square simulation error for the model with i regressors, which implies the use of free-run simulation.

More recent techniques, many of which are based on Monte Carlo approaches, have been introduced to the academic community in recent years. For the purposes of this research, the Randomized Model Structure Selection method, or simply, RaMSS (Falsone et al., 2014, 2015), stands out.

2.1.4 Parameter Estimation

In the process of identifying dynamic systems models, more specifically using parametric estimation, once the data is collected, pre-processed and the class of the model and its structure are chosen, the problem is to determine the best parameters for this model . This process is called parameter estimation. The objective is to find a parametric function $\hat{f}(\varphi_k, \hat{\boldsymbol{\theta}}) : \mathbb{R}^{n_\theta} \mapsto \mathbb{R}$ which is close to the ideal and generally unknown function, $f(\varphi_k) : \mathbb{R}^{n_\theta} \mapsto \mathbb{R}$ using sampled training data. Thus

$$y_k = f(\varphi_k) \approx \hat{f}(\varphi_k, \hat{\boldsymbol{\theta}}),$$

where $\varphi_{k-1} \in \mathbb{R}^{n_{\hat{\theta}}}$ is the vector of regressors, formed by linear or nonlinear combinations of the output $y_{k-1}, \ldots, y_{k-n_y}$ (e.g. FIR models) and/or input $u_{k-1}, \ldots, u_{k-n_u}$ (e.g., ARX models, or ARMAX) and even of the residues $\xi_k, \ldots, \xi_{k-n_\xi-1}$(e.g., ARMA and NARMAX models); $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{n_\theta}$ is a vector of estimated parameters; $y_k \in \mathbb{R}$, the signal sampled at time k; and $n_{\hat{\theta}}$, n_y, n_u represent respectively: the number of parameters and the maximum delays at the output and at the input.

Considering that the ideal function $f(\varphi_k)$ can be written in parametric form

$$y_k = f(\varphi_k, \boldsymbol{\theta}), \tag{2.2}$$

where $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ is the ideal parameter vector, and that both f and $\boldsymbol{\theta}$ do not change in relation to time k, you can write (2.2) in the following matrix form

$$\mathbf{y} = \Psi\boldsymbol{\theta}, \tag{2.3}$$

where $\Psi = \begin{bmatrix} \varphi_k & \varphi_{k-1} & \ldots & \varphi_{k-N+1} \end{bmatrix}^T$ and $\mathbf{y} = \begin{bmatrix} y_k & y_{k-1} & \ldots & y_{k-N+1} \end{bmatrix}^T$.

If there is $N = n_\theta$ restrictions, the vector of $\boldsymbol{\theta}$ parameters can be determined by

$$\boldsymbol{\theta} = \Psi^{-1}\mathbf{y}.$$

However if $N > n_\theta$ restrictions are taken, the system is said to be over-determined and the X matrix becomes non-square and non-invertible. One solution is to rewrite eqref eq: yMatrix so that the solution is not exact, by introducing an error term $\boldsymbol{\xi} \in \mathbb{R}^N$, known as the *residue* vector, resulting in

$$\mathbf{y} = \Psi\hat{\boldsymbol{\theta}} + \boldsymbol{\xi}.$$

To capture the behavior of the modeled process in the model, it is intuitive that the vector of estimated parameters $\hat{\boldsymbol{\theta}}$ is chosen so that $\boldsymbol{\xi}$ is reduced in some direction. In the strategy known as Ordinary Least Squares, OLS or, simply, least squares or LS, a cost function related to the residues vector is defined as

$$J_{LS} = \sum_{k=1}^{N} \boldsymbol{\xi}(k)^2 = \boldsymbol{\xi}^T\boldsymbol{\xi} = \|\boldsymbol{\xi}\|^2.$$

It is proved that the vector $\hat{\boldsymbol{\theta}}_{LS}$, defined as the set of parameters that minimizes J_{LS}, can be calculated using the pseudo-inverse of Ψ, so that (Aguirre, 2015):

$$\hat{\boldsymbol{\theta}}_{LS} = [\Psi^T\Psi]^{-1}\Psi^T\mathbf{y}. \tag{2.4}$$

The equation 2.4 represents the least squares estimator, where the parameters are determined by minimizing the cost function referring to the sum of the square of the modeling errors. Numerical alternatives more interesting than the classic algorithm of (2.4) can be found (Aguirre, 2015; Ljung, 1999), but the basic concepts remains the same.

2.1.5 Model Validation

Having estimated the parameters that minimize the residue in some sense, such as as a function of the sum of the square of the modeling errors, in the case of the least squares presented in (2.4), the performance of the model should be evaluated when it is subject to excitations different from those submitted during the identification process (training). For this, it is common to use a set of sampled data different from that used in the identification process. This data set is called a validation set. It is usual to collect a certain amount of data from the process, subject to an appropriate excitation signal, and then divide the resulting data set into a training set and a validation set.

The next step is to use some performance index in order to quantify the quality of the prediction in a test known as *free-run simulation*. In this simulation, the model

obtained in the identification process is submitted to the same excitation signal under which the validation set was submitted. The simulated and sampled results previously stored are then compared using some metric. Among the most common metrics, *Mean Square Error* (MSE) and *Mean Absolute Percentage Error* (MAPE) stand out.

The MSE is given by

$$\text{MSE} = \frac{1}{N}\sum_{k=1}^{N}(y_k - \hat{y}_k)^2,$$

being M the number of samples, y_k the data sampled in the time k and $\hat{y}_k$ the model prediction. The MSE can be calculated on both, training and validation data. However, for validation purposes, the validation data must be used, since this way it is possible to obtain a measure of the performance for the model behaving outside the training environment. A good model, in general, should present the MSE on the validation data next to the MSE on the training data.

Based on the MSE, two other performance indexes are defined: *Mean Square Prediction Error*, or MSPE, and *Mean Square Simulation Error*, or MSSE. The MSPE is defined as the MSE using one step ahead prediction as the prediction data $\hat{y}_k$ in MSE computing. MSSE, on the other hand, uses free simulation data as the $\hat{y}_k$ in the calculations. Modified versions of MSPE can also be used, such as MS2PE, which uses 2-step prediction ahead, among others.

The MAPE index calculates the absolute deviation of the prediction in relation to the observed data and is, in general, calculated as a percentage, in the form

$$\text{MAPE} = \frac{100}{N\sigma(\boldsymbol{y})}\sum_{k=1}^{N}|y_k - \hat{y}_k|\ \%, \tag{2.5}$$

where $\boldsymbol{y} = [y_1,\ y_2,\ \ldots,\ y_N]^T$ and $\sigma(\boldsymbol{y})$ represents the standard deviation of $\boldsymbol{y}$.

Different ways of choosing training and validation data in relation to the usual one presented in (2.5) can be used. One of these forms is the "leave-one-out cross-validation", in which only one sample of data is used at a time (Allen, 1974).

In addition to the quantitative validation, in general, a qualitative evaluation is also used, in which a graphical comparison is made, regarding the dynamic behavior, between the curve of the sampled signal and the signal curve estimated by the resulting model.

2.2 Randomized Model Structure Selection

The RaMMS method presents a randomized approach for choosing an appropriate model structure by sampling models from a set of models $\mathscr{M}$. Introduced by Falsone et al. (2014) and improved in Falsone et al. (2015), the method performs the task of structure selection in a probabilistic way that, despite the random behavior, avoids the

exhaustive search necessary to analyze all the possible models in the set of candidate regressors, defined by $\mathscr{R}$, that would be needed when using brute force in a purely Monte Carlo strategy.

Considering a set of all possible models formed by $\mathscr{R}$, named universe set and represented by $\mathscr{U}$, the method iteratively searches for the best subset of regressors in the $\mathscr{R}$ set in order to maximize the accuracy of the model's prediction according to a defined index. This is done by a set of candidate models, defined as $\mathscr{M} \in \mathscr{U}$, built from regressors sampled from the $\mathscr{R}$ set with a probability of choice given by an estimated probability function called RIP (*Regressor Inclusion Probability*).

Once a candidate model is chosen, it is evaluated according to some performance criteria and the RIPs are updated. At each iteration, N_p models are taken from $\mathscr{U}$ set and performance indexes are calculated for updating the RIPs.

These indexes are, in general, based on the MSPE, MSSE or a combination of the two, and are used to calculate an average performance index $\mathcal{I}$ of the evaluated regressor. Assuming a set $\mathscr{M}$ of $N_p = \dim(\mathscr{M})$ models, where some contain the j-th regressor (and others possibly not), an index that measures the *average performance of the regressor*, which can be used in calculating the RIP is defined as

$$\begin{aligned} \mathcal{I}_j &= \mathcal{I}_j^+ - \mathcal{I}_j^- \\ &= \mathbb{E}[\mathcal{J}(\mathscr{M})|\phi_j \in \mathscr{M}] - \mathbb{E}[\mathcal{J}(\mathscr{M})|\phi_j \notin \mathscr{M}], \end{aligned} \tag{2.6}$$

being $j = 1, \ldots, m$, where $m = \dim(\mathscr{R})$ is the number of regressors considered, $\mathcal{J}(\mathscr{M})$ represents the vector containing the performance indexes for the chosen models, $\mathscr{M}$ is the set of candidate regressors, ϕ_j the j-th regressor and $\mathbb{E}[\cdot]$ is the operator of mathematical expectation*. Thus, the performance index $\mathcal{I}_j$ compares the average performance of models containing the j-th regressor ($\mathcal{I}_j^+$) with the performance of models that do not contain this same regressor ($\mathcal{I}_j^+$). As, in general, the mathematical expectation of the equation (2.6) cannot be calculated analytically, this is done, in practice, by estimation, taking models from the universe model $\mathscr{U}$ set and calculating the sample mean, resulting in

$$\mathcal{I}_j^+ = \frac{1}{n_j^+} \sum_{i \in \mathscr{M}_j^+} \mathcal{J}_i^+ \qquad \text{e} \qquad \mathcal{I}_j^- = \frac{1}{n_j^-} \sum_{i \in \mathscr{M}_j^-} \mathcal{J}_i^-,$$

where $\mathscr{M}_j^+ \subset \mathscr{M}$ and $\mathscr{M}_j^- \subset \mathscr{M}$, represent the sets of all models that, respectively, contain and do not contain the j-th regressor ϕ_j, with $n_j^+ = \dim \mathscr{M}_j^+$ and $n_j^- = \dim \mathscr{M}_j^-$. The terms $\mathcal{J}_i^+$ and $\mathcal{J}_i^-$ represent the respective performance indexes for cases that contain and do not contain ϕ_j.

Assuming that the actual model, defined by f^*, belongs to the set of all possible structures (universe set) $\mathscr{U}$, it must be possible to find this model by exploring the set

*If the conditional event has zero probability of occurring, expectation is assumed to be null.

of models $\mathscr{M}$ and taking the models with the best performance. The problem of finding the real model as a function of an index can be represented by

$$f^* = \underset{\tilde{f} \in \tilde{\mathscr{U}}}{\operatorname{argmax}} \mathcal{J}(\tilde{f}), \tag{2.7}$$

where $\mathcal{J}$ is some[†] performance index calculated for candidate models $\tilde{f}$ taken from the non-redundant universe set $\tilde{\mathscr{U}} \subset \mathscr{U}$[‡], or even by $\mathscr{U}$.

In order to find the correct model structure, i.e. f^*, the RaMSS algorithm samples N_p models from $\mathscr{U}$, calculates the performance index $\mathcal{J}(\tilde{f})$ for each candidate model $\hat{f}_i$, with $i = 1, \ldots, N_p$ and estimates the average over all models involving the ϕ_j-th regressor. This is done for each of the $m = \dim(\mathscr{R})$ regressors of the $\mathscr{R}$ set of regressors.

Considering that the performance index $\mathcal{J}$ in (2.7) has values in the $[0, 1]$ range, i.e. $\mathcal{J}(\tilde{f}) \in \mathbb{R} : \mathcal{J}(\tilde{f}) \in [0, 1]$, its expected value, considering the probability distribution $\mathcal{P}$ and a random variable $\Phi \equiv \tilde{\mathscr{M}}$ that corresponds to a realization a model sample in $\tilde{\mathscr{U}}$, will be given by

$$\mathbb{E}[\mathcal{J}] = \sum_{\tilde{f} \in \tilde{\mathscr{U}}} \mathcal{J}(\tilde{f}) \mathcal{P}(\mathscr{U} = \tilde{f}) \tag{2.8}$$

If $\mathcal{P}$ is varied over all possible distributions in $\tilde{\mathscr{U}}$, the maximum of (2.8) is obtained by concentrating the entire probability mass on the most appropriate model. With that, the solution to the optimization problem can be obtained by

$$\mathcal{P}^* = \underset{\Phi \in \tilde{\mathscr{U}}}{\operatorname{argmax}} \mathbb{E}\left[\mathcal{J}(\Phi)\right] \tag{2.9}$$

and it's such that $\mathcal{P} = 1$. Thus, by solving (2.9), the most appropriate model, or real model f^*, is selected with the same solution as (2.7).

The RaMSS method estimates $\mathcal{P}$ and, consequently, the best model as follows. At each iteration, N_p candidate models are assembled from m candidate regressors. The choice of each ϕ_j regressor component of a candidate model is made based on a Bernoulli process, by a random variable associated with each ϕ_j regressor such that

$$\rho \approx \mathrm{Be}(\mu_j),$$

where possible results are: $\rho = 1$, with μ_j probability (of success) to occur; and $\rho = 0$, with probability μ_j; where $\mu_j \in [0, 1]$, with $j = 1, \ldots, m$ and m the number of candidate regressors. If $\rho = 1$, the ϕ_j regressor will be present in the candidate model built, otherwise, no.

[†]The definition of $\mathcal{J}$ adopted in the original proposal of RaMSS is presented later in the text (see eqs. 2.11 a 2.12).

[‡]Note that in the universe set it is common for redundant models to appear, ie, similar models that do not include statistically relevant terms. In practice, it is common for these to be removed and sampling to be performed on a reduced set of the universe model, defined by $\tilde{\mathscr{U}} \subset \mathscr{U}$.

It is assumed that ρ_j random variables are independent [§] and a $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_m]^T$ vector is defined as the vector of *Probability of Inclusion of Regressor* (RIP). Note that the RIP, or $\boldsymbol{\mu}$, is the one that dictates the $\mathcal{P}$ probability distribution on the models in $\tilde{\mathscr{U}}$ (or $\mathscr{U}$), that is, given a known $\boldsymbol{\mu}$, the probability of obtaining a $\tilde{f}$ structure model in any subset of n_θ regressors is

$$\mathcal{P}(\tilde{f}) = \prod_{j:\phi_j \in \tilde{f}}^{n_\theta} \mu_j \prod_{j:\phi_j \notin \tilde{f}}^{m-n_\theta} (1 - \mu_j),$$

to any $\tilde{f} \in \tilde{\mathscr{U}}$.

The procedure is performed so that at each iteration the regressors are chosen according to the Bernoulli process, where the probability of choosing μ_j for each ϕ_j regressor is represented by the RIP vector. The RIP vector, $\boldsymbol{\mu}$ is refined by successive iterations of an update rule defined as

$$\mu_j(i+1) = \mu_j(i) + \gamma \mathcal{I}_j, \tag{2.10}$$

where γ is a project parameter, ϕ_j the performance index defined in (2.6) and i the index of the algorithm's iteration.

After a few iterations, it is expected that the average performances of the models containing the right regressors will be significantly higher than those that do not, making the correct regressors more likely to be included in the final model. The method does not guarantee that $\boldsymbol{\mu}$ will be limited. To prevent the RIP values from increasing, or decreasing indefinitely, a saturation of the elements in a range with a minimum value greater than or equal to 0, and a maximum value less than or equal to 1, is applied by the algorithm.

The project parameter γ, from (2.10) is chosen in order to control the convergence speed. Larger values tend to lead to faster convergence but it can also lead to non-convergence of the method. To deal with this convergence problem, Falsone et al. (2015) proposes an adaptive parameter, given by

$$\gamma = \frac{1}{10(\mathcal{J}_{\max} - \bar{\mathcal{J}}) + 0.1},$$

where $\mathcal{J}_{\max}$ represents the performance index of the best model and $\bar{\mathcal{J}}$ represents the average performance index in the current iteration. The purpose of the adaptive step is that, if $\bar{\mathcal{J}}$ is far from the best performance index, γ is made small in order to compensate for the large probable variance of that model population. However, if $\bar{\mathcal{J}}$ is close to $\mathcal{J}_{\max}$, the sampled models have low performance variance, which indicates that the RIP should be updated to a greater value.

[§] Although there are results in the literature (Bianchi et al., 2016) suggesting that an approach using conditioned and multivariate Bernoulli distribution may result in improvements in the accuracy of the model selection process, an initial procedure adopted in this work considers independence.

The RaMSS method, as originally presented by the authors, uses as basis for calculating the performance index $\mathcal{J}$, quantities related to MSPE and MSSE, defined in the Section 2.1.5. The average performance index is calculated as an exponential value of these indices weighted by a K gain, used as a design parameter. This way,

$$\begin{aligned} \mathcal{J}_p &= e^{-K \cdot MSPE}, \\ \mathcal{J}_s &= e^{-K \cdot MSSE}. \end{aligned} \tag{2.11}$$

Note that $\mathcal{J}_p$ and $\mathcal{J}_s$ will have values in the [0 1] range, where values close to 1 correspond to better performance, and close to 0, worse performance. The K parameter defines the sensitivity of the performance index, where the difference between models is amplified to values greater than K.

The final value of $\mathcal{J}$ is calculated as

$$\mathcal{J} = \alpha \mathcal{J}_s + (1 - \alpha)\mathcal{J}_p \tag{2.12}$$

where $\alpha \in [0\ 1]$ is a parameter defined by the designer.

After a few iterations, the RIP vector, $\boldsymbol{\mu}$, is expected to converge to an equilibrium distribution. The final model will be the correct model expected from the system and will consist of regressors associated with RIPs with values greater than a certain threshold defined by the designer. Assuming that the real model $f^* \in \tilde{\mathscr{U}}$, this threshold is generally set to a value close to 1.

The pseudo-code for the described RaMSS method is presented in Algorithm 1. Following is an example for system identification using the method.

Example 2.2.1 (Idendificação de um modelo não-linear utilizando o RaMSS)**.** As an illustrative example of the use of the RaMSS algorithm in the structure selection, consider a nonlinear model provided in Baldacchino et al. (2013), given by:

$$y(k) = 0.7y(k-1)u(k-2) - 0.5y(k-2) + 0.6u^2(k-2) - 0.7y(k-2)u^2(k-2) + e(k)$$

where $u(k)$ is the excitation signal, adopted as a signal with uniform distribution between -1 and 1, i.e. $u(k) \sim \mathcal{U}(-1, 1)$ and $e(k)$ is a white Gaussian noise with a variance of 0.02 and zero mean, i.e. $e(k) \sim \mathcal{N}(0, 0.02)$.

The Algorithm 1 algorithm is applied for a maximum number of 100 iterations. At each iteration, a total of $N_p = 100$ models are chosen to calculate the indexes used in the RIPs update. Figure 2.1 shows a typical behavior of RIPs during updates. Note that the $u^2(k-2)$ regressor converges quickly, in approximately 14 iterations, followed by the $y(k-2)$, $y(k-1)u(k-1)$ and $y(k-2)u^2(k-2)$ regressors.

Algorithm 1: RaMSS algorithm

$\boldsymbol{y}, N_p, m, \mu_{\min}, \mu_{\max}, K, \mathscr{R} = \{\phi_1, \ldots, \phi_m\}$
while $iter < iter_{\max}$ **do**
  $\boldsymbol{\mu} \leftarrow \boldsymbol{\mu}_0$
  for $i = 1 : N_p$ **do** // Model Sampling
    $\tau \leftarrow 0$
    $\psi(k) \leftarrow [\,]$ // Initialize model
    for $j = 1 : m$ **do**
      $r_j \leftarrow \text{Be}(\mu_j)$ // Sample from a Bernoulli distribution
      if $r_j = 1$ **then**
        $\psi(k) \leftarrow \begin{bmatrix} \psi^T(k) & \phi_j \end{bmatrix}^T$
        $\tau \leftarrow \tau + 1$
    for $h = 1 : \tau$ **do**
      $\tilde{\psi}(k) \leftarrow \text{non-redundant}(\psi(\boldsymbol{k}))$ // Remove redundant terms
    $\hat{\boldsymbol{y}} \leftarrow \text{Predict}(\tilde{\psi}(k))$
    $\mathcal{J}_i \leftarrow e^{-K \cdot MSPE(\boldsymbol{y}, \hat{\boldsymbol{y}})}$
  for $j = 1 : m$ **do** // RIP Update
    $\mathcal{J}^+ \leftarrow 0; \mathcal{J}^- \leftarrow 0; n^+ \leftarrow 0; n^- \leftarrow 0;$
    for $i = 1 : N_p$ **do**
      if $\phi_j(k) \in \tilde{\psi}(k)$ **then**
        $\mathcal{J}^+ \leftarrow \mathcal{J}^+ + \mathcal{J}_i$
        $n^+ \leftarrow n^+ + 1$
      else
        $\mathcal{J}^- \leftarrow \mathcal{J}^- + \mathcal{J}_i$
        $n^- \leftarrow n^- + 1$
    $\mathcal{I}_j \leftarrow \left(\frac{\mathcal{J}^+}{\max(n^+,1)} - \frac{\mathcal{J}^-}{\max(n^-,1)} \right)$
    $\mu_j \leftarrow \mu_j + \gamma \mathcal{I}_j$
    $\mu_j \leftarrow \max\left(\min(\mu_j, \mu_{\max}), \mu_{\min}\right)$

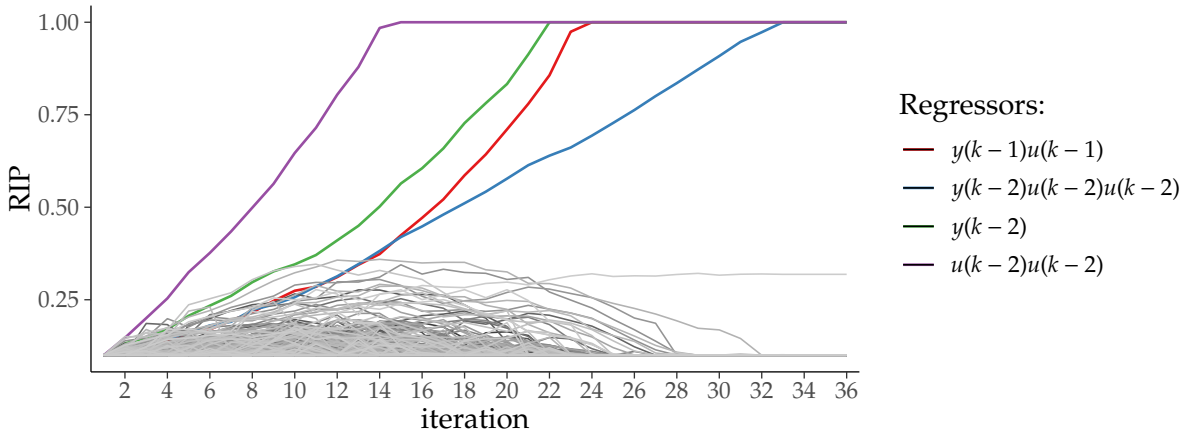

Figure 2.1: Convergence of RIPs considering a particular realization.

A more general analysis, which takes into account the behavior in 100 realizations, is shown in Figure 2.2. In this case it is observed that the previous behavior remains in the average.

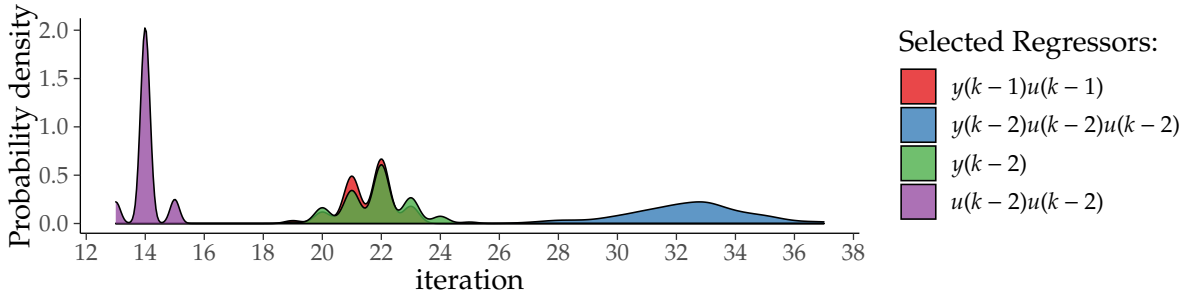

Figure 2.2: Probability density for convergence of RIPs, considering 100 realizations.

Note that the graphs only consider up to the 36th iteration. This is due to the adopted stopping criteria, which interrupts the procedure as soon as the variations in the RIPs become less than a certain predefined value. In all simulations the procedure stopped before the 100th iteration, selecting exactly the original regressors, indicating that there is always a convergence for these values in this exemple.

2.3 Control System Setup and Notations

In this section, we present the closed-loop system considered, defining the equations for its components and introducing the notation used throughout the text. As it is intended to deal with non-linear systems, the notation presented in Campi and Savaresi (2006) is adopted. When dealing with linear systems, however, this notation may appear to be heavier than necessary and a more usual notation may be adopted. These

notations are presented within the scope of the control system considered and are detailed in the next subsections.

2.3.1 The control system

It is considered a classic control system of the SISO type with a degree of freedom, which may or may not be under the effect of noise in the output signal. Figure **??** shows the block diagram of this system, where $C(q, \boldsymbol{\theta}, e)$ represents the controller, a function of the vector of parameters $\boldsymbol{\theta}$ and $P(q, u)$, the process model, or plant, which may or may not be known. $r(k)$ $u(k)$, $v(t)$, $e(k)$, $y(k)$, and $y_v(k)$ are, respectively, the reference, control, noise, tracking error, process output and, finally, noise output signals.

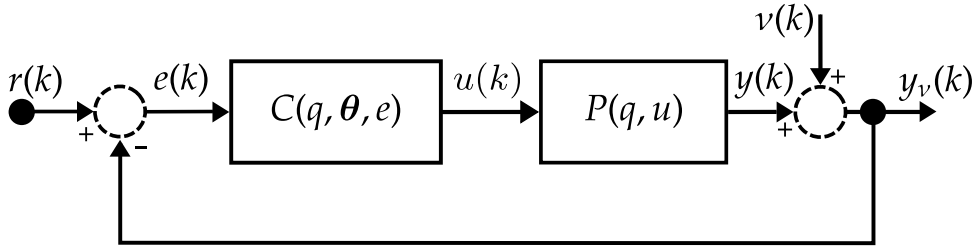

Figure 2.3: Sistema de controle.

2.3.2 The process

The process, or plant, P is a discrete non-linear time process of the SISO type, with nonlinear dynamics described as

$$y(k) = p\left(y(k-1), \ldots, y(k-n_{Py}), u(k-1), \ldots, u(k-n_{Pu})\right),$$

where p is a function representing the process, n_{Py} is the maximum delay in the output, n_{Pu} is the maximum delay in the input and k the temporal index.

For simplicity, the pure time delay from u to y in P is considered to be 1, but the procedure can be extended without problems for longer delays.

P is considered a non-linear map that operates from $\mathbb{R}^N$ to $\mathbb{R}^N$. When subject to the initial conditions *i.c.* $= (y(0), \ldots, y(1-n_{Py}), u(-1), \ldots, u(1-n_{Pu}))$ and an input signal in the range $[0, N-1]$, defined as $u(0{:}N-1) \triangleq [u(0) \cdots u(N-1)]^T$, P generates an output given by

$$y(1{:}N) = P[u(0{:}N-1), \text{ i.c. }]. \tag{2.13}$$

The following hypotheses are adopted:

Assumption 2.1 (Process smoothness). The function p that represents the process is smooth.

Assumption 2.2 (Uniqueness of the solution). For any initial conditions given, if $u_1(0{:}N-1) \neq u_2(0{:}N-1)$, so $P\left[u_1(0{:}N-1), i.c.\right] \neq P\left[u_2(0{:}N-1), i.c\right]$.

The Assumption 2.1 guarantees the invertibility of the P map. The invertibility of P for an input signal defined in the range $[0{:}N-1]$ implies in the invertibility of the map in a range $[0{:}T]$, with $T \leq N-1$ (Campi and Savaresi, 2004).

2.3.3 The controller

The controller considered, as well as the process, is represented by a non-linear model (which can also be linear). For the purpose of identifying the parameters by the VRFT method, subject addressed in the chapter 3, it is assumed that a structure (or class) is previously chosen. This controller can be described as

$$u(k) = c\left(u(k-1), \ldots, u(k-n_{Cu}), e(k), \ldots, e(k-n_{Ce})\right), \tag{2.14}$$

where n_{Cu} and n_{Ce} are the maximum delays in the control signal (controller output and tracking error signal (controller input).

As for the process, the controller is subject to the initial conditions

$$i.c. = u(-1), \ldots, u(-n_{Cu}), e(-1), \ldots, e(-n_{Ce})$$

and when fed with the error signal $e(0{:}N-1)$, it generates the control signal $u(0{:}N-1)$, which is represented by

$$u(0{:}N-1) = C[e(0{:}N-1), i.c.]. \tag{2.15}$$

In this work, one of the objectives is to select a suitable controller given a fixed class of controllers by the VRFT method, as discussed in the chapter 3. This class is formed by all controllers parameterized by

$$u(k) = c\left(u(k-1), \ldots, u(k-n_{Cu}), e(k), \ldots, e(k-n_{Ce}); \boldsymbol{\theta}\right), \tag{2.16}$$

being $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ a vector of n_θ parameters. The controller parameterized by $\boldsymbol{\theta}$ will be indicated here by C_θ, and the control signal (2.14) is in the form

$$u_\theta(0{:}N-1) = C_\theta[e(0{:}N-1), i.c.], \tag{2.17}$$

with

$$e(0{:}N-1) \triangleq r(0{:}N-1) - y(0{:}N-1),$$

representing the tracking error.

The following assumption is assumed for the controller:

Assumption 2.3. The c controller is represented by a scalar function parameterized as in (2.16), and assumed to be smooth, or in a simplified way

$$c : \mathbb{R}^{n_{Cu}+n_{Ce}+1+n_\theta} \mapsto \mathbb{R} \text{ is smooth.}$$

2.3.4 The feedback system

The interconnections of the closed-loop system shown in Figure 2.3, together with the equations of the process (2.13) and the controller (2.15) result in the relationship

$$\begin{aligned} y(1:N) &= P[u(0{:}N-1), i.c.] \\ &= P[C[e(0{:}N-1), i.c.], i.c.] \end{aligned} \tag{2.18}$$

2.3.5 Notation simplification

For simplifications in the notations, the initial conditions of the plant and the controller are considered null. Such a requirement is not necessary, but it only leads to unnecessary complications, especially in the notation. Furthermore, if the N is large enough, the initial conditions have little influence on stable cases. In addition, time arguments are omitted, resulting in the use of the symbols: u to represent $u(0{:}N-1)$, r for $r(0{:}N-1)$, e for $e(0{:}N-1)$ and y for $r(1{:}N)$. Note the time advance of y in relation to the other variables. In this way, $y(0{:}N)$ is represented by Dy, with D being a nilpotent delay displacement matrix defined as

$$D \triangleq \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

With that, (2.18) can be written as

$$y = P[C[r - Dy]] \tag{2.19}$$

Or, using the parametrized controller (2.17),

$$y_\theta = P[C_\theta[r - Dy_\theta]] \tag{2.20}$$

2.3.6 The control objective

A fundamental element for controller design and analysis in optimization-based control theory, as is the case with VRFT, is the concept of *control performance*. In general,

this concept can be expressed as the solution to a problem defined as

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}),$$

where $\boldsymbol{\theta} \in \mathbb{R}$ is a vector of n_θ parameters adopted as a decision variable and $J(\boldsymbol{\theta})$ is a *cost function*[¶]. The lower the value of $J(\boldsymbol{\theta})$, the better the performance according to some adopted criterion. Depending on the control objective, or even for reasons of analytical stability guarantees, different cost functions can be adopted. A very common approach is to choose a cost function based on the mean quadratic norm-2 [‖] in the form

$$\|x(k)\|^2 = \frac{1}{N}\sum_{k=1}^{N}[x(k)]^2 \triangleq \bar{\mathbb{E}}\,[x(k)]^2 \tag{2.21}$$

where $x(k)$ represents a generic function, N, the number of samples and $\bar{\mathbb{E}}\,[\cdot]$, an operator defined as

$$\bar{\mathbb{E}}[x(k)] \triangleq \frac{1}{N}\sum_{k=1}^{N}[x(k)],$$

which represents the calculation of the sample mean, and will be used throughout the text to replace the summation, for convenience.

The criterion presented in (2.21) is known as *performance criterion* H_2.

The H_2 criterion, for control purposes, is defined according to the control objective. These objectives, in general, are adopted as: (1) reference tracking, (2) noise rejection and (3) control effort saving. Mixed criteria adopting more than one of these objectives, can also be defined, as is the case of (4) composite performance criteria. The following sections deal with these criteria in a little more detail.

The Reference Tracking Objective

One of the main objectives of closed loop control is to make the controlled signal follow a desired reference signal, so that they are as close as possible. Disregarding the effect of noise on the output, and considering a parameterized controller, the response of the closed-loop system is given by (2.20).

This objective can be translated as the minimization of a cost function given by 2-norm of the tracking error, in the form

$$J_r^N(\boldsymbol{\theta}) \triangleq \|r - y_\theta\|^2. \tag{2.22}$$

[¶]Other terms are also known in the literature, such as *performance index (or function)*, or else, *objective function*.

[‖]The norm-2 of a discrete signal, or vector, $x(t)$ is defined as $\|x(t)\| \triangleq \sqrt{\sum_{t=1}^{N}[x(t)]^2}$. Raising this value squared we have the quadratic norm-2, which represents the energy of the signal which, divided by the number of samples, as in (2.21), results in its average energy.

The Reference Model Objective

A well-known fact in the control community is that in practice it is impossible to make the output follow the reference signal perfectly at all times, that is, (4.1) will never be zero. To get around this fact, one way out is to relax the tracking objective by another that satisfies the desired prerequisites (such as accommodation time, peak time, maximum startle, among others), but which is less demanding than the original reference.

This new objective is translated into a model known as *reference model* represented here by a map from $\mathbb{R}^N$ to $\mathbb{R}^N$, which maps a reference signal, say, $\tilde{r}$ to an exit signal $\tilde{y}$ with a desired behavior **.

$$M : \tilde{r} \in \mathbb{R}^N \mapsto \tilde{y} \in \mathbb{R}^N \tag{2.23}$$

This map, in principle, can be a non-linear model, as long as it is smooth and invertible. However, in general, for convenience, it is chosen as a linear model, but it must still be invertible. The following Assumption was defined:

Assumption 2.4 (Invertibility of the reference model)**.** M is lower triangular and invertible.

The fact that M is adopted as a lower triangular ensures that it will be causal, with a pure delay greater than or equal to 1. An example of a typical choice for M, for the linear case, is the filter

$$M(q) = \frac{b_1 q^{-1} + \cdots + b_{n_{Mr}} q^{-n_{Mr}}}{1 + a_1 q^{-1} + \cdots + a_{n_{My}} q^{-n_{My}}} \tag{2.24}$$

where n_{Mr} and n_{My} are the maximum delays for the reference and for the output, respectively, and q is a delay operator. In the time domain, (2.24) corresponds to

$$\tilde{y}(k) = -a_1 \tilde{y}(k-1) - \cdots - a_{n_{My}} \tilde{y}\left(k - n_{My}\right) + b_1 \tilde{r}(k-1) + \cdots + b_{n_{Mr}} \tilde{r}\left(k - n_{Mr}\right)$$

Once the reference model that reflects the desired behavior in closed loop has been defined, the new control objective is translated as

$$J_{RM}^N(\boldsymbol{\theta}) \triangleq \left\| y_\theta - \tilde{y} \right\|^2 = \left\| P[C_\theta[\tilde{r} - D y_\theta]] - M[\tilde{r}] \right\|^2 \tag{2.25}$$

where $\tilde{y} = M[\tilde{r}]$ represents the output signal of the desired reference model, when under the effect of a reference signal r. This criterion is known as *reference model criterion*.

The Noise Rejection Objective

Another common goal in control systems is to minimize the effect of noise on the output signal. The output signal due only to the action of the noise signal in the process

**the symbol ˜ emphasizes that these signals are for the reference model.

can be written as

$$y_{\theta v}(\boldsymbol{\theta}) \triangleq S_\theta[v] = v - P[C_\theta[Dy_{\theta v}],$$

where $v \triangleq v(0{:}N)$ represents the noise signal and S_θ a noise map for the process output, i.e. $M : v \mapsto y_{\theta v}$, which, for the linear case, is also known as the sensitivity function. When analyzing the magnitude of this signal as a function of some norm, results in the performance criterion of noise rejection $J_v^N(\boldsymbol{\theta})$ which, using the 2-norm, is given by

$$J_v^N(\boldsymbol{\theta}) \triangleq \left\| y_{\theta v} \right\|^2 = \left\| S_\theta[v] \right\|^2 .$$

As in the case of tracking, it is not possible to completely eliminate the effect of noise and relaxation in this criterion is desirable, for the same reasons. In this case, a desired sensitivity function is defined S_M [††], and the following criterion is adopted

$$J_{RMv}^N(\boldsymbol{\theta}) \triangleq \left\| S_\theta[v] - S_M[v] \right\|^2 .$$

The Economy of Control Effort Objective

A common objective in the design of controllers, especially in the area of optimal control, is to minimize the control effort by means of control signal. To minimize the control signal, the following index can be defined:

$$J_u^N(\boldsymbol{\theta}) \triangleq \left\| u(k) \right\|^2 , \tag{2.26}$$

where $u(k)$ is the control signal. However, this index should not be used alone, since $J_u^N(\boldsymbol{\theta}) = 0$ would imply $u \equiv 0$. So what is done is to use a combination of this index, with others, for example, with the reference tracking, resulting in

$$J_\lambda^N = J_{RM}^N(\boldsymbol{\theta}) + \lambda J_u^N(\boldsymbol{\theta}),$$

where $\lambda \in \mathbb{R}$ represents an adjustment parameter. When using controllers based on a reference model, such as the case of VRFT (to be seen in 3), a control effort saving effect can be obtained by choosing an appropriate reference model. Therefore, in this work, the inclusion of the (2.26) index in the cost function is disregarded.

The Composite Performance Objective

Another objective, more realistic than that of reference tracking, is *composite performance*, which aims to both, reduce tracking errors, and reject noise effects on the output. This criterion is given by

$$J_T^N(\boldsymbol{\theta}) \triangleq \left\| y_{\theta v} - \tilde{y} \right\|^2 ,$$

[††]The M index is used to maintain the analogy with the index used by the reference model for the tracking case.

where, unlike $\tilde{y}$ in (3.1), the signal $y_{\theta\nu} = P[C_\theta[r - y_\theta]] + S[\nu]$ depends on the noise. The final criterion will be the sum of the two previous criteria, since the reference and the noise are statistically independent signals and can be written as

$$J_T^N(\boldsymbol{\theta}) = \left\| P[C_\theta[r - Dy_\theta]] - M[r] \right\|^2 + \|S[\nu]\|^2 .$$

2.4 The Ideal Controller

Considering the feedback system presented in (2.19), and assuming the invertibility of P and that P is lower triangular, solving (2.19) for C, it is possible to calculate the ideal controller, which results in the optimal parameters for C_θ, as long as C_θ has the same structure (or belongs to the same class) as the ideal controller. This ideal controller is defined as:

$$C_0 \triangleq P^{-1}(I - MD)^{-1}M, \tag{2.27}$$

where I is the identity matrix and C^i is the ideal controller.

When C_0 is used in the closed loop, the map from r to y is given by M, that is, $y = M[r]$

For the linear SISO case, if the process model is known and considering that it is of minimum phase, the ideal controller (2.27) can be written in the form of a transfer function as

$$C_0(z) = \frac{M(z)}{P(z)\,(1 - M(z))}, \tag{2.28}$$

where $P(z)$ and $M(z)$, respectively, represent the transfer functions of the process and the reference model.

Note that, if the process is non-minimum phase, due to zeros outside the unit circle in the z-plane, the inverse of $P(z)$ in (2.28) results in system instability. Solutions to this problem have been presented for decades for MBC projects, such as rules for PID tuning, such as the one developed by Skogestad (2003). An most recent approach, within the DDC scope, is presented by Campestrini et al. (2011).

Chapter 3

Virtual Reference Feedback Tuning

The *Virtual Reference Feedback Tunning* method, or simply VRFT, proposed by Campi et al. (2002), is a procedure that aims to design closed loop controllers based only on data sampled from the process, without the need for a model that describes that process itself. Thus, it is classified as a data-based control method, or DDC.

The main objective of this method is to adjust the parameters of a controller, defined by a parametric function like (2.16), using the process sampled data only, so that the output signal of the controlled process, $y_\theta(k)$ behaves as close as possible to the output signal $\tilde{\boldsymbol{y}}$ of a previously defined reference model as defined in (2.23). To reach this objective, VRFT aims to optimize the tracking error by minimizing a performance index $J_{RM}^N(\boldsymbol{\theta})$ as stated in (3.1), rewritten here for convenience:

$$J_{RM}^N(\boldsymbol{\theta}) \triangleq \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} [y(k,\boldsymbol{\theta}) - \tilde{y}(k)]^2 = \left\| y_\theta - \tilde{\boldsymbol{y}} \right\|^2, \tag{3.1}$$

where N represents the number of data samples, $\boldsymbol{\theta} = \begin{bmatrix} \theta_1 & \theta_2 & \cdots & \theta_N \end{bmatrix}^T \in \mathbb{R}^N$ a vector of parameters and k a temporal index. The signal $y_\theta(k) \in \mathbb{R}$ represents the output of the system controlled with the parametrized controller C_θ, and $\tilde{y}(k) \in \mathbb{R}$ are the output of a reference model M, both subject to the same reference signal.

To achieve the objective of minimizing (3.1), Campi et al. (2002), for the linear case, and Campi and Savaresi (2006), for the non-linear case, show that, under certain conditions, presented in sequence, when minimizing a cost index defined as

$$J_{VR}^N(\boldsymbol{\theta}) \triangleq \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} [u(k) - C_\theta(q,\boldsymbol{\theta})\bar{e}(k)]^2, \tag{3.2}$$

the index $J_{RM}(\boldsymbol{\theta})$ defined in (3.1) is also minimized.

Notation: The symbol "tilde" will be adopted on the variables to indicate that they represent data obtained, or calculated, from the data collection experiment, used for the purpose of identifying the controller parameters.

According to the notation adopted, $\tilde{u}(k)$ em (3.2) represents the input signal applied to the process during data collection, $C(q,\boldsymbol{\theta})$ the controller model to be adjusted and

$\tilde{e}(k)$ is the so-called *virtual error*, defined as

$$\tilde{e}(k) = \tilde{r}(k) - \tilde{y}(k), \tag{3.3}$$

where $\tilde{r}(k)$ is a signal called *virtual reference*, obtained by filtering the output $\tilde{y}(k)$ by the inverse of the reference model M defined in (2.23) and in (2.24) for a linear case. So, for a linear reference model, the virtual reference can be defined as

$$\tilde{r}(k) = M^{-1}(q)\tilde{y}(k). \tag{3.4}$$

The term, $\tilde{y}(k)$ in (3.3) and (3.4), represents the sampled output signal used in the experiment.

Figure 3.1 shows a block diagram with the steps used in the experiment to obtain the controller parameters via the VRFT procedure.

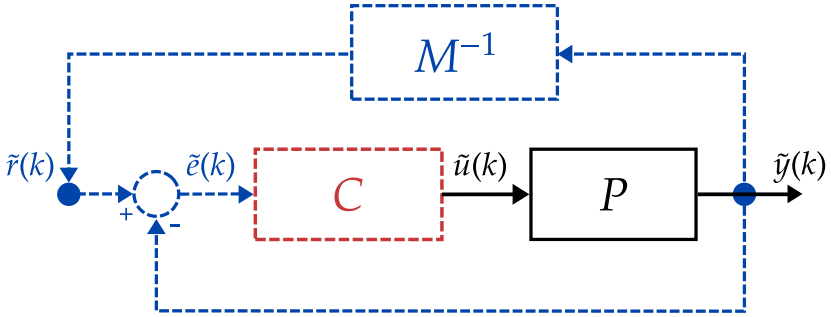

Figure 3.1: Experiment to obtain the data used to identify the controller parameters by the VRFT method: real data in *black solid lines* (——), virtual data in *blue dashed lines* (- - - -) and the cotroller to tune, in *red dashed block* (- - - -).

The term "virtual" is adopted in the reference $\tilde{r}$ and in the tracking error $\tilde{e}$ signals to emphasize that none of these signals are physically available, but only calculated for controller design purposes. The dashed lines (in blue) in Figure 3.1 represent these virtual data, and the red block represents the controller under which the vector of parameters $\boldsymbol{\theta}$ need to be adjusted.

The condition for $J_{RM}(\boldsymbol{\theta})$ and $J_{VR}(\boldsymbol{\theta})$ reach their minimum values for the same parameter solution $\boldsymbol{\theta}$ are presented in sequence, right after some definitions and Assumptions that are important for the rest of the chapter.

3.1 Initial Considerations

This Section presents some definitions and Assumptions used in the rest of the chapter.

Assumption 3.1 (The Process)**.** It is assumed that the process is the one presented in Section 2.3.2 and therefore, it is a nonlinear process, assumed to be smooth (Assumption 2.1) and invertible (Assumption 2.2).

Definition 3.1 (Ideal Controller)**.** The ideal controller is defined as in the Section 2.4 and is represented by the map C_0, which maps $\tilde{e} \mapsto \tilde{u}$. For the linear case, the notation $C_0(q, \theta)$ is used for temporal representations or $C_0(z, \theta)$ for transfer functions.

Considering that a structure for the controller is previously chosen (i.e. if it is linear or not, how many parameters are considered, or what regressors are considered in construction), it is said that this structure is defined by a $\mathscr{C}$ class. That is, the $\mathscr{C}$ class represents all possible controllers using the same structure.

Assumption 3.2 (Matched control)**.** The ideal controller belongs to control model class considered, i.e. $C_0(q) \in \mathscr{C}$, or, equivalently

$$\exists \theta_0 : C(q, \theta_0) = C_0(q).$$

Assumption 3.3 (Noise free)**.** The system is considered to be not affected by noise.

Unless otherwise noted, this chapter assumes that there are no measurement noise, that is, $v(k)$ is null and $y_v(k) = y(k)$ (see Figure 2.3).

Assumption 3.4 (The asymptotic counterpart)**.** The measured signals $\tilde{u}(k)$ and $\tilde{y}(k)$ are considered realizations of stationary and ergodic stochastic processes so, when the number of available data grows ($N \to \infty$), the following holds*:

$$J_{\mathrm{VR}}^{N}(\theta) \to J_{\mathrm{VR}}(\theta) = \mathbb{E}\left[(\tilde{u} - C_\theta[\tilde{e}])\right]^2,$$

so, J_{VR} is the *asymptotic counterpart* of J_{VR}^N and for the rest of text J_{VR} will be used in place of J_{VR}^N for analysis purposes.

Assumption 3.5 (Linear parametrized controller)**.** The controller is assumed to be linear in the parameters, as (2.16), and will be represented by the map C_θ, or by $C_\theta(q, \theta, \tilde{e})$, depending on the context. If it is linear, $C_\theta(q, \theta)$ is used as a function of time, or $C_\theta(z, \theta)$ when in the form of a transfer function.

Note that, from Assumption 3.5, we can write the controller $C_\theta(q, \theta, \tilde{e}) = \varphi^T \theta$, with φ being a vector of regressors as discussed in Section 2.1.4. Thus, considering the Assumption 3.4, we have

$$J_{VR}(\theta) = \frac{1}{N} \sum_{k=1}^{N} \left[\tilde{u}(k) - \varphi^T(k)\theta\right]^2 = \left\|\tilde{u} - \Psi^T \theta\right\|^2 \tag{3.5}$$

where Ψ the regressors matrix, as defined in the Section 2.1.4, equation 2.3.

*The operator $\mathbb{E}[\cdot]^2$ represents the expected value operator of a quadratic random variable: $\mathbb{E}[\cdot]^2 = \lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} (\cdot)^2$.

Definition 3.2 (SRq process (Bazanella et al., 2012))**.** A quasi-stationary process is said to be sufficiently rich of order q – or simply SRq – if its spectrum has at least q nonzero components.

Definition 3.3 (Persistence of excitation (Bazanella et al., 2012))**.** A quasi-stationary vector φ is said to be persistently exciting if $\bar{\mathbb{E}}[\varphi(k)\varphi(k)^T] > 0$

Assumption 3.6. The signal $\tilde{u}$ used in the excitation of process P is persistently exciting (definition), which implies being SRq (definition).

3.2 The Ideal Control Design Problem – The Matched Control

As already mentioned, the proposal of the VRFT procedure is to show that when solving the problem of minimizing $J_{VR}(\boldsymbol{\theta})$, that is, finding the vector of parameters $\boldsymbol{\theta}$ that leads to this solution, the cost function related to the reference model output tracking error, $J_{RM}(\boldsymbol{\theta})$ is also minimized with the same parameters, i.e.

$$\boldsymbol{\theta}_0 = \underset{\forall \boldsymbol{\theta} \in \mathbb{R}^{n_\theta}}{\operatorname{argmin}} J_{VR}(\boldsymbol{\theta}) = \underset{\forall \boldsymbol{\theta} \in \mathbb{R}^{n_\theta}}{\operatorname{argmin}} J_{RM}(\boldsymbol{\theta}).$$

Considering that *all* the Assumptions in the previous section are satisfied, Campi et al. (2002) shows that this is possible for the linear case, as well as for the nonlinear case (Campi and Savaresi, 2006). They define the following theorem:

Theorem 3.2.1 (Campi and Savaresi (2006))**.** *If $\boldsymbol{\theta}_0$ gives the perfect tracking, i.e. $\left\|\boldsymbol{y}_{\theta_0} - M[\tilde{\boldsymbol{r}}]\right\|^2 = 0$, then, $\boldsymbol{\theta}_0$ is a minimizer of $\|C_\theta[\tilde{\boldsymbol{e}}] - \tilde{\boldsymbol{u}}\|^2$.*

Proof:
As $\left\|\boldsymbol{y}_{\theta_0} - M[\tilde{\boldsymbol{r}}]\right\|^2 = 0$, it is possible to deduce that

$$\boldsymbol{y}_{\theta_0} = \tilde{\boldsymbol{y}}. \tag{3.6}$$

As $\boldsymbol{y}_{\theta_0}$ is the closed loop response of the plant with the controller C_{θ_0}, and $\tilde{\boldsymbol{y}}$ the plant P response from an input $\tilde{\boldsymbol{u}}$,

$$\boldsymbol{y}_{\theta_0} = P\left[C_{\theta_0}\left[\tilde{\boldsymbol{r}} - D\boldsymbol{y}_{\theta_0}\right]\right] \quad \text{and} \quad \tilde{\boldsymbol{y}} = P[\tilde{\boldsymbol{u}}].$$

Considering the Assumption 2.2, it is possible to conclude that

$$C_{\theta_0}\left[\tilde{\boldsymbol{r}} - D\boldsymbol{y}_{\theta_0}\right] = \tilde{\boldsymbol{u}}. \tag{3.7}$$

From (3.6),

$$\tilde{\boldsymbol{r}} - D\boldsymbol{y}_{\theta_0} = \tilde{\boldsymbol{r}} - D\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{e}}$$

which, replacing in (3.7), results in

$$C_{\theta_0}[\tilde{e}] = \tilde{u},$$

which implies in $\|C_\theta[\tilde{e}] - \tilde{u}\|^2 = \|\tilde{u} - \tilde{u}\|^2 = 0$.

Using the $J_{VR}(\theta)$ index in place of $J_{RM}(\theta)$ has the great advantage of not requiring the knowledge of a model for the process, which classifies VRFT as a DDC strategy. Another great advantage is that, when considering controllers linear in the parameters, the $J_{VR}(\theta)$ cost function becomes quadratic, which facilitates its minimization and allows to find its global minimum. The Figure 3.2 exemplifies this case. While $J_{RM}(\theta)$ may have local minimums that make it difficult to locate the global minimum, $J_{VR}(\theta)$, as long as linear in the parameters, results in a function with only a well-defined global minimum.

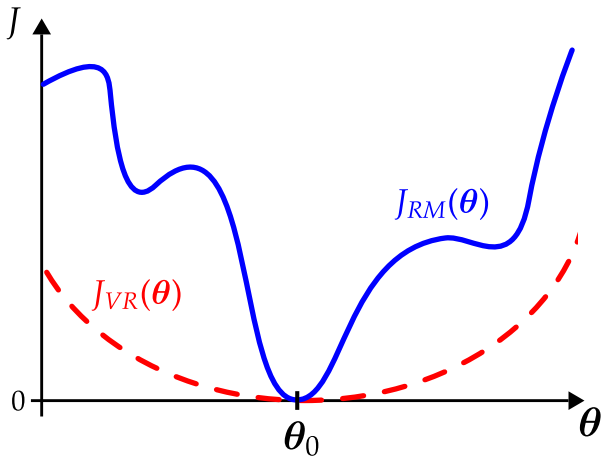

Figure 3.2: Typical cost functions for $J_{VR}(\theta)$ (- - - -) and $J_{RM}(\theta)$ (——).

Note that the Theorem 3.2.1 is valid if the Assumption 3.2 is satisfied. In a general case, the structure, or class of C_0 is unknown, which ends up leading to the choice of a structure for C_θ belonging to a class $\mathscr{C}$ such that Assumption 3.2 is not satisfied, i.e. $C_0 \notin \mathscr{C}$. A great contribution of VRFT is the choice of filters that, applied to the sampled signal $\tilde{u}$ and the predicted signal $C_\theta[\tilde{e}]$, bring the estimated value of θ closer to the optimum value θ_0. This subject is discussed in more detail in the Section 3.4.

3.3 Controller Parameter Identification

It is considered again that *all* the hypotheses of the Section 3.1 are satisfied, and that the data acquisition and calculation of virtual values experiment, as described at the beginning of the chapter and illustrated by Figure 3.1, is applied in order to obtain the necessary data. Thus, the vector of parameters that minimizes $J_{VR}(\theta)$, as (3.5), can be

estimated by

$$\boldsymbol{\theta} = \left[\sum_{k=1}^{N} \boldsymbol{\varphi}(k)\boldsymbol{\varphi}(k)^T\right]^{-1} \sum_{k=1}^{N} \boldsymbol{\varphi}(k)\tilde{u}(k). \tag{3.8}$$

Note that (3.8) is the same equation presented in (2.4), however the latter is presented in a matrix form, and the former in the form of a summation. Therefore, the solution can be written in terms of the ordinary least squares algorithm (OLS) or its derivatives. Attention is paid to the fact that, in this chapter, the estimated parameter vector $\hat{\boldsymbol{\theta}}$ of (2.4) is considered only as $\boldsymbol{\theta}$ for reasons of simplification of notation.

When the Assumption 3.3 is not satisfied, that is, the sampled is corrupted by noise, the OLS method may not be adequate due to the possible polarization of the estimator parameters caused by these noises (Aguirre, 2015). A common approach used to reduce the effects of this polarization, within the scope of VRFT, is to make use of the instrumental variable estimator, or VI (Young, 1970). Although, for the purposes of this qualification, noise effects on measurements have not yet been analyzed, and it is assumed that the estimators used are given by OLS estimators.

Another possibility to deal with polarization is to use the extended least squares estimator, or ELS (Ljung, 1999; Aguirre, 2015) in place of OLS, which, as the VI estimator, has the property of reducing polarization (at the cost of an increase in variance). This approach was adopted by Retes and Aguirre (2019) to structure selection in process model identification with good results. We intend to adapt the metodolgy for controller structure selection and parameter identification in this work. However, it is still in the study phase and will be presented as a future proposal in Chapter 5.

3.4 The Unmatched Control and the Filter Selection

As discussed at the end of the Section 3.2, if the Assumption 3.2 is not satisfied, i.e. $C_0 \notin \mathscr{C}$, the Theorem 3.2.1 no longer applies. Note that this is a recurring situation in practice, since there is, frequently, few information about the process.

However, Campi and Savaresi (2006) presents a filter that, when applied to the data used to find the minimizer of $J_{VR}(\boldsymbol{\theta})$, produces well-tuned controllers, even when the controller belongs to a class $\mathscr{C}$ other than the C_0 class, as long as this class is not "too far", that is, the controller should not be too under-parameterized with respect to the control objective.

To achieve the mentioned objective, an increased and filtered cost function is assumed, given by

$$J_{VR}(\boldsymbol{\theta}^+) \triangleq \|F[C_{\boldsymbol{\theta}^+}[\tilde{e}]] - F[\tilde{u}]\|^2$$

where F represents the mentioned filter and:

$$\boldsymbol{\theta}^{+} \triangleq [\boldsymbol{\theta}^{T}\ \tilde{\theta}]^{T}, \quad \tilde{\theta} \in \mathbb{R};$$
$$C_{\boldsymbol{\theta}^{+}} \triangleq C_{\theta} + \tilde{\theta}\,(C_0 - C_{\Delta}),$$

where $C_{\Delta} = C_{\boldsymbol{\theta}}$ is computed for $\boldsymbol{\theta} = \mathbf{0}$. It can be verified that:

- C_0 é obtido for $\boldsymbol{\theta}_0^{+} \triangleq [\mathbf{0}^{T}\ 1]^{T}$;
- $\{C_{\theta}\}$ it is obtained by imposing $\tilde{\theta} = 0$.
- C_{Δ} represents the part of C_0 that cannot be explained by the chosen controller family.

A new expanded cost function for the reference model is defined as

$$J\,(\boldsymbol{\theta}^{+}) \triangleq \left\| y_{\boldsymbol{\theta}^{+}} - M[\tilde{\boldsymbol{r}}] \right\|^{2}$$
$$\text{with } y_{\boldsymbol{\theta}^{+}} = P\,[C_{\theta} + [\tilde{\boldsymbol{r}} - D y_{\boldsymbol{\theta}^{+}}]]\,.$$

The ideal control objective is to minimize $J_{MR}(\boldsymbol{\theta})$, or

$$\boldsymbol{\theta}_0^{+} = \underset{\theta^{+} \in \mathbb{R}^{n_{\theta^{+}}}}{\operatorname{argmin}}\, J_{MR}(\boldsymbol{\theta}^{+}).$$

Campi and Savaresi (2006) prove that the objective of minimizing $J_{MR}(\boldsymbol{\theta}^{+}$ can be achieved with good approximation by selecting a filter that guarantees equality between expansions in Taylor series up to the term of degree 2 for $J_{MR}(\boldsymbol{\theta}^{+})$ and $J_{VR}(\boldsymbol{\theta}^{+})$. By doing this expansion (see Section A.2) it is possible to show that one must have

$$\left.\frac{\partial^{2} J_{\mathrm{VR}}\,(\boldsymbol{\theta}^{+})}{\partial \boldsymbol{\theta}^{+2}}\right|_{\boldsymbol{\theta}_0^{+}} = \left.\frac{\partial^{2} J\,(\boldsymbol{\theta}^{+})}{\partial \boldsymbol{\theta}^{+2}}\right|_{\boldsymbol{\theta}_0^{+}}. \tag{3.9}$$

If this equality is guaranteed, $J_{MR}(\boldsymbol{\theta}^{+})$ and $J_{VR}(\boldsymbol{\theta}^{+})$ will be close each other when they are smaller than one, and consequently the parameters that minimize $J_{VR}(\boldsymbol{\theta}^{+}$, minimize $J_{MR}(\boldsymbol{\theta}^{+}$. For this to happen, the following theorem is defined:

Theorem 3.4.1 (Campi and Savaresi (2006)). *If*

$$F = (I - MD)\left(\left.\frac{\partial P[\boldsymbol{u}]}{\partial \boldsymbol{u}}\right|_{\tilde{\boldsymbol{u}}}\right) \tag{3.10}$$

then (3.9) *is satisfied.*

The proof of the Theorem 3.4.1 is presented in the Section A.2 of the Appendix A.

A problem presented by the filter (3.10) is the dependence on the knowledge of a model for the plant. Possible solutions can be obtained using an approximate identified

model, which causes the method to lose the characteristic that qualifies it as a DDC system. However, the expression $(\partial P[\boldsymbol{u}]/\partial \boldsymbol{u})|_{\tilde{\boldsymbol{u}}}$ is not used for design, only as a filter for the data used in the parametric estimation and an accurate identification is not required. Furthermore, according to Campi and Savaresi (2006), in many cases, disregarding the derivative of the process and rewriting the filter equation simply as $F = (I - MD)$ results in good results, although there is no guarantee of good approximation.

To use the filter, it is sufficient that the sampled signal, $\tilde{\boldsymbol{u}}$ and the signal $C_{\boldsymbol{\theta}}[\tilde{e}]$ are filtered as follows:

$$\begin{aligned} \tilde{\boldsymbol{u}}_f &= F[\tilde{\boldsymbol{u}}], \\ \tilde{C}_f &= F[C_{\boldsymbol{\theta}}[\tilde{e}]]. \end{aligned}$$

When the controller is linear in the parameters, we can write

$$F[C_{\boldsymbol{\theta}}[\tilde{e}]] = F[\Psi_C \boldsymbol{\theta}] = F[\Psi_C]\boldsymbol{\theta} \implies \tilde{C}_f = F[\Psi_C]\boldsymbol{\theta}.$$

From the filtered data ($F[\tilde{\boldsymbol{u}}_f]$ and $F[\tilde{C}_f)]$, the VRFT procedure is applied in order to minimize the filtered cost function

$$J_{VR_f}(\boldsymbol{\theta}) = \|F[\Psi_C]\boldsymbol{\theta} - F[\tilde{\boldsymbol{u}}]\|^2,$$

and the vector parameter that minimizes this cost function can be found by solving

$$F[\Psi_C]^T (F[\Psi_C]\boldsymbol{\theta} - F[\tilde{\boldsymbol{u}}]) = 0,$$

where Ψ_C represents the regressors matrix, a function, in general, of the virtual error $\tilde{e}$, the input signal of the process $\tilde{\boldsymbol{u}}$ and possibly a vector of residues. The equation 3.4 can be solved, for example, by the OLS estimator or by non-polarized estimators for noisy cases. The solution will be the vector of estimated parameters $\hat{\boldsymbol{\theta}}$, that makes the matrix of regressors not correlated with the residual error of the identified controller model $(F[\Psi_C] - F[\tilde{\boldsymbol{u}}])$.

3.5 Aplication Examples

Example 3.5.1. (Controller Parameter identification for a Linear System usinv VRFT procedure)

To exemplify the VRFT procedure, a simple case is presented, where the intention is to identify a controller for a second order linear system, according to the VRFT procedure. The adopted model is given by a "artificial" system, taken from Wei and Billings (2008), and described in the following difference equation, presented as an ARX model:

$$y_k = -1.7y_{k-1} - 0.8y_{k-2} + u_{k-1} + 0.81u_{k-2} + \eta_k, \tag{3.11}$$

where, $u_{k-i} \in \mathbb{R}$ e $y_{k-i} \in \mathbb{R}$, represent, respectively, the input and output signals for a delay i in respect to k; k is the time index; and η_k represents the noise in the output signal, which for the purposes of this example, is considered null, i.e. $\eta_k = 0$. It is considered that the controller must be such that the output of the controlled system follows a reference trajectory imposed by a first order linear system reference model, with the smallest possible error. So, the reference model is choosed So, the reference model is chosen as

$$M(z) = \frac{1 - A}{(z - A)z^{d-1}} \tag{3.12}$$

where d is the desired pure time delay, $A = e^{-T_s/\tau_M}$ a quantity related to the desired time constant for the closed loop system, given by τ_M, and T_s the sampling time used in the sampling (or decimation) process.

From the application of an input signal $\tilde{\boldsymbol{u}} = [\tilde{u}_1, \ldots, \tilde{u}_N]^T$, as a step shape to the process, where $\tilde{\boldsymbol{u}} = \boldsymbol{u}$ (see (3.11)), the response $\tilde{\boldsymbol{y}} = [y_1, \ldots, y_N]^T$ is observed. The figure 3.3 shows the temporal evolution of these signals (first two top graphs). The

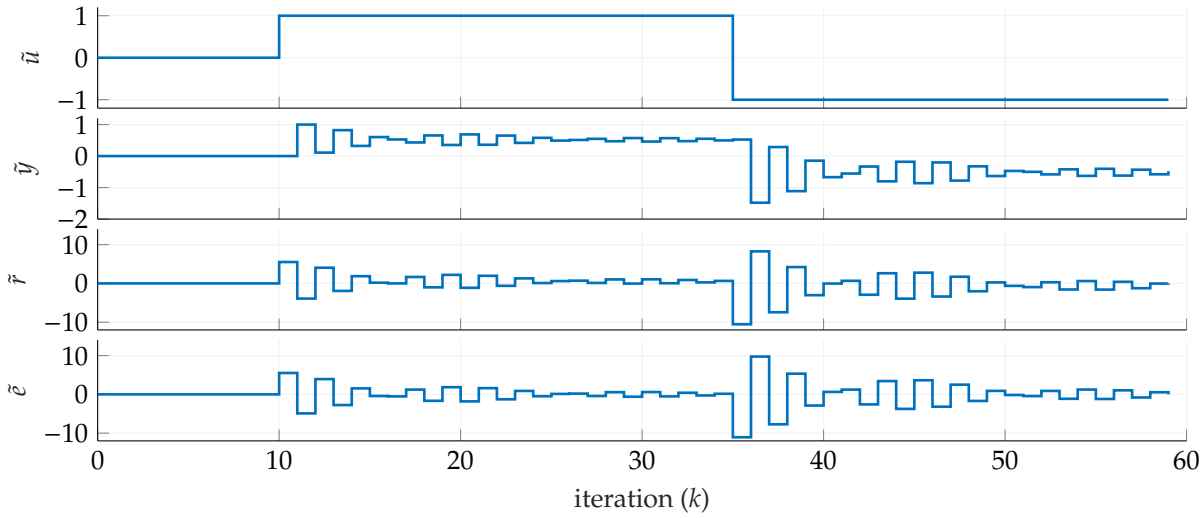

Figure 3.3: Temporal evolution of the signals: input of process $\tilde{\boldsymbol{u}}$; output of process $\tilde{\boldsymbol{y}}$; the calculated virtual reference $\tilde{\boldsymbol{r}}$; and the calculated virtual error $\tilde{\boldsymbol{e}}$; considering the top to bottom graphs sequence.

process (3.11) can be written in the form of a transfer function as

$$P(z) = \frac{b_{p1}z + b_{p2}}{z^2 + a_{p1}z + a_{p2}}, \tag{3.13}$$

where $a_{p1} = 1.7$, $a_{p2} = 0.8$, $b_{p1} = 1$ and $b_{p2} = 0.81$ are the process parameters, given in (3.11).

Form $\tilde{\boldsymbol{y}}$ collected, and the reference model (3.12), considering $d = 1$, the virtual

reference could be expressed as (3.4), in the transfer function forma

$$\tilde{\boldsymbol{r}}_k = M^{-1}(q)\tilde{\boldsymbol{y}} = \frac{q-A}{1-A}\tilde{\boldsymbol{r}}_k. \tag{3.14}$$

In (3.14), q (time shift operator) is adopted instead of z (z-transform variable) to emphasize that we are dealing with time responses. This procedure will be adopted throughout this text when needed.

Note that (3.14) is a non causal system. However, since $\tilde{\boldsymbol{y}}$ is available in advance, in batch, the calculation can be performed. With $\tilde{\boldsymbol{r}}$, the virtual error $\tilde{\boldsymbol{e}}$ can be easily calculated from (3.3), as

$$\tilde{\boldsymbol{e}}_k = \tilde{\boldsymbol{r}}_k - \tilde{\boldsymbol{y}}_k$$

Figure 3.3 shows this *virtual* values (2 bottom ones) calculated over the sampled data $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{y}}$ (2 top ones).

As the process in this example is considered known, the ideal controller $C_0(z)$ can be calculated according to (2.28), replacing (3.12) and (3.13), i.e.

$$\begin{aligned} C_0(z) &= \frac{M(z)}{P(z)\,(1-M(z))} \\ &= \frac{(1-A)z^2 + (Aa_{p1} - a_{p1})z - a_{p2} + Aa_{p2}}{b_{p1}z^2 + (b_{p2} - b_{p1})z - b_{p2}} \\ &= \frac{0.1813z^2 + 0.3082z + 0.145}{z^2 - 0.19z - 0.81} \end{aligned}$$

which can be rewritten in the time domain as:

$$\begin{aligned} u_k &= \frac{1-A}{b_{p1}}e_k + \frac{Aa_{p1} - a_{p1}}{b_{p1}}e_{k-1} + \frac{Aa_{p2} - a_{p2}}{b_{p1}}e_{k-2} - (b_{p2} - b_{p1})u_{k-1} + b_{p2}u_{k-2} \\ &= \begin{bmatrix} e_k & e_{k-1} & e_{k-2} & u_{k-1} & u_{k-2} \end{bmatrix} \boldsymbol{\theta}^\star \end{aligned} \tag{3.15}$$

with ideal parameters given by $\boldsymbol{\theta}^\star = [\theta_1^\star\ \theta_2^\star\ \theta_3^\star\ \theta_4^\star\ \theta_5^\star]^T$. Using the parameters of the process (see (3.12) and (3.11)), a assuming a constant time of $\tau = 5$ and a sample time of $T_s = 1$, we have $A = 0.8187$ and

$$\boldsymbol{\theta}^\star = \begin{bmatrix} 0.181 & 0.308 & 0.145 & 0.19 & 0.81 \end{bmatrix}^T. \tag{3.16}$$

If the ideal parameters are not known, they can be estimated using the sampled data, $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{y}}$ instead of $\boldsymbol{u}$ and $\boldsymbol{y}$ in (3.15) and, by constructing a matrix of regressors as, for example (it's not the only chice), given by

$$\Psi^T = \begin{bmatrix} \boldsymbol{\varphi_1} & \boldsymbol{\varphi_1} & \cdots & \boldsymbol{\varphi_k} & \cdots & \boldsymbol{\varphi_N} \end{bmatrix}$$

where $\boldsymbol{\varphi}_k^T = \begin{bmatrix} \tilde{\boldsymbol{e}}_k & \tilde{\boldsymbol{e}}_{k-1} & \tilde{\boldsymbol{e}}_{k-2} & \tilde{\boldsymbol{u}}_{k-1} & \tilde{\boldsymbol{u}}_{k-2} \end{bmatrix}$ is the vector of regressors. Using parameter

estimation techniques such as the OLS method, it is possible to estimate $\boldsymbol{\theta}$ with good approximation. If the excitation signal is persistently exciting, and there is no correlated noise affecting the output, when the number of samples N tends to infinity, i.e. $N \to \infty$, $\boldsymbol{\theta}$ tends to $\boldsymbol{\theta}^\star$. In this example, noise is disregarded and $\boldsymbol{\theta}$ is estimated by

$$\hat{\boldsymbol{\theta}} = (\Psi^T \Psi)^{-1} \Psi \tilde{\boldsymbol{u}}$$

Performing the procedure described for the data presented, the following cases are considered for controller design:

case A (Matched Case) A controller in which the vector of regressors $\boldsymbol{\varphi}$ has the same structure as the ideal controller is considerr. That is, $\mathscr{C}^\star \in \mathscr{C}_A$, where $\mathscr{C}^\star$ is the class of the ideal controller and $\mathscr{C}_A$, the controller class of the case A;

case B (Unmatched Case) It is considered a sub-parameterized controller, with vector of regressors given by $\boldsymbol{\varphi}_B^T = \begin{bmatrix} \tilde{e}_k & \tilde{e}_{k-1} & \tilde{e}_{k-2} & \tilde{u}_{k-1} \end{bmatrix}$, i.e. $\mathscr{C}^\star \notin \mathscr{C}_B$;

case C (PID Unmatched Case) It is considered a sub-parameterized controller with a vector of regressors built according to the structure of a PID controller, defined as

$$\begin{aligned}
C_C(q) &= \left(K_p + K_i \frac{q}{q-1} + K_d \frac{q-1}{q} \right) \tilde{\boldsymbol{e}}_k \\
&= \left(K_p + K_i \frac{q}{q-1} + K_d \frac{q-1}{q} \right) (\tilde{\boldsymbol{r}}_k - \tilde{\boldsymbol{y}}_k) \\
&= \left(K_p + K_i \frac{q}{q-1} + K_d \frac{q-1}{q} \right) \left(\frac{q-A}{1-A} - 1 \right) \tilde{\boldsymbol{y}}_k \\
&= \left(K_p + K_i \frac{q}{q-1} + K_d \frac{q-1}{q} \right) \frac{q-1}{1-A} \tilde{\boldsymbol{y}} \\
&= \begin{bmatrix} \dfrac{\tilde{\boldsymbol{y}}_{k+1} - \tilde{\boldsymbol{y}}_k}{1-a} & \dfrac{\tilde{\boldsymbol{y}}_{k+1}}{1-a} & \dfrac{\tilde{\boldsymbol{y}}_{k+1} - 2 * \tilde{\boldsymbol{y}}_k + \tilde{\boldsymbol{y}}_{k-1}}{1-a} \end{bmatrix} \begin{bmatrix} K_p & K_i & K_d \end{bmatrix}^T
\end{aligned}$$

The vector of regressors, in this case, is described by

$$\boldsymbol{\varphi}_C^T = \begin{bmatrix} (\tilde{\boldsymbol{y}}_{k+1} - \tilde{\boldsymbol{y}}_k)/(1-a) & (\tilde{\boldsymbol{y}}_{k+1})/(1-a) & (\tilde{\boldsymbol{y}}_{k+1} - 2 * \tilde{\boldsymbol{y}}_k + \tilde{\boldsymbol{y}}_{k-1})/(1-a) \end{bmatrix}$$

Note that, although the regressor in case B has fewer terms than the regressor in case A, the controller belongs to the class of controller A, since it can be written by

$$C_C(z) = \frac{(Kd + Ki + Kp)z^2 + (-2Kd - Kp)z + Kd}{z^2 - z}$$

however, the controller class in case A is broader (class B is contained in class A). By the same procedure of identification described earlier, the following parameters are

obtained for the 3 cases:

$$\boldsymbol{\theta}_A^T = \begin{bmatrix} 0.181 & 0.308 & 0.145 & 0.19 & 0.81 \end{bmatrix} = \boldsymbol{\theta}^\star$$
$$\boldsymbol{\theta}_B^T = \begin{bmatrix} 0.168 & 0.166 & 0.039 & 0.988 \end{bmatrix}$$
$$\boldsymbol{\theta}_C^T = \begin{bmatrix} -0.202 & 0.351 & 0.0229 \end{bmatrix}$$

Figure 3.4 shows the temporal response of the closed-loop system with the controllers obtained for the 3 cases, together with the signal from the reference model. It is considered a square reference signal and that a disturbance of magnitude 0.5 is applied at the output, at time $k = 75$. The figure also shows the answer for a fourth case to be discussed ahead.

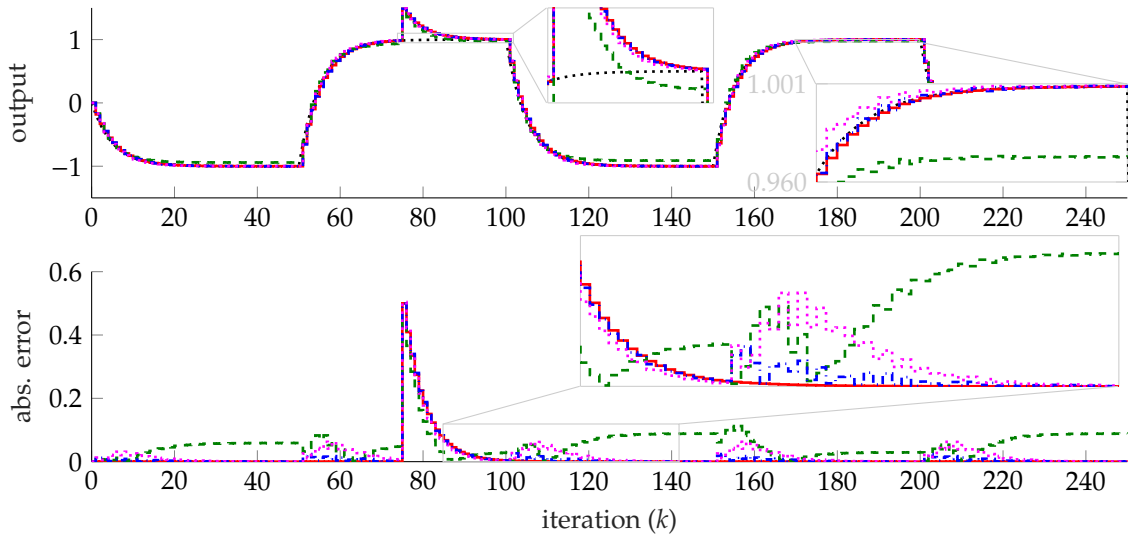

Figure 3.4: Closed-loop system response to a square reference signal (upper graph) and absolute error (lower graph) for: case A (——); case B (- - - -); case C (-·-·-); and case D (·····). The response of the reference model is represented by (·····) in the upper graph.

Note that in Case A, because it has the same structure and parameters as the ideal controller, it has an identical response to the reference model (except, obviously, at the transient period after the application of the disturbance). Differently, Case B presents some tracking error, even on steady state. This can be explained by the fact that the controller model was not able to reproduce an integration effect. This effect can be observed when the sum of the coefficients of the terms in $\tilde{\boldsymbol{u}}$ that make up the regressor, results in 1. In this case, the coefficient of $\tilde{\boldsymbol{u}}_{k-1}$ (only regressor in $\tilde{\boldsymbol{u}}$) is $0.988 \approx 1$, but not exactly 1. Case 3, due to the way in which the regressor is built, a restriction is imposed on the controller structure in such a way that an integrator effect is forced, which allows the controller to be expressed with only 3 regressors. The result, as can

be seen in Figure 3.4, is the elimination of tracking error in steady state, even under the effect of external disturbance. It is emphasized here that, although the integration effect is not present in case B, this can be imposed, for example, with the use of methods such as *Restricted Least Squares*, or CLS (Draper and Smith, 1998). Case D presented below considers this modification applied in case B, restricting the coefficient of $\tilde{\boldsymbol{u}}_{k-1}$ to be equals one and imposing an integrator in controller. The time response is also shown in Figure 3.4.

case D In this case, it is considered that the CLS procedure is adopted after the parameter estimation in case B. The $c = S\boldsymbol{\theta}_B$ restriction is assumed, with $c = 1$ and $S = [0\ 0\ 0\ 1]^T$. The new parameter vector $\boldsymbol{\theta}_D$ is calculated by (Draper and Smith, 1998):

$$\boldsymbol{\theta}_D = (\Psi^T\Psi)^{-1}\tilde{\boldsymbol{y}} - (\Psi^T\Psi)^{-1}S^T\left[(\Psi^T\Psi)^{-1}S^T\right]^{-1}(S\boldsymbol{\theta}_B - c)$$

where $\Psi^T = \left[\boldsymbol{\varphi}_B(1) \quad \boldsymbol{\varphi}_B(2) \quad \dots \quad \boldsymbol{\varphi}_B(N)\right]$ represents the regressor matrix, and $\boldsymbol{\varphi}_B(k) = \left[\tilde{e}_k \quad \tilde{e}_{k-1} \quad \tilde{e}_{k-2} \quad \tilde{u}_{k-1}\right]^T$, with $k = 1, \dots, N$, is the vector of regressors at instant k.

As can be seen in Figure 3.4, the error in steady state is eliminated, as expected.

3.5.1 Final Considerations of the Chapter

A benefit of specifying the controller as in case C of the Example 3.5.1, is that it starts from a priori knowledge (i.e., it is known that the controller must have an integrator). For many cases, where is desired to tune a pre-existing controller, such as controllers in the PID family, this can be very useful. This fact has great appeal in industrial applications.

However, there are cases where a precise control is needed. Especially for strongly non-linear systems, where a controller with an adequate pre-established structure (which may be non-linear) can be difficult to obtain. In this sense, the main focus of this work is on the selection of this appropriate structure with no (or some) prior knowledge. Only from sampled data. As will be seen later, in Chapter 4, a method for searching the appropriate controller structure by a randomized approach is proposed. possa In this case, it will be necessary to easily generate a sufficiently large universe of models. For this purpose, representing these controllers using polinomial structures like NBRX (or NBRMBX) models, generically, as done in case B of Example 3.5.1, instead of predefined structures as done in case C, is an advantage. In order to take advantage of known or desired information about the controller, a proposal of this work is to include this information in the identification process in a posteriori way, using, for example, the CLS estimator, as shown in case D of the example.

As will be seen, the VRFT method will be one of the main tools used for the framework proposed in chapter 4.

Chapter 4

Randomized Controller Structure Selection

In this chapter, an approach for the design of DDC controllers is presented, based on the RamSS methodology for the choice of structure and on the VRFT method for tuning the controller. This new approach is here called Randomized Controller Structure Selection, or RaCSS, for short.

In this research, focus has been given to the study and development of structure selection techniques for the model of feedback controllers in a DDC design fashion. The intention is, at first, in the scope of non-linear control systems, to adapt known techniques of structure selection in the area of systems identification to deal with the control case, where the system to be identified becomes the controller. To achieve these objectives, it was decided, to use polynomial models to represent the controllers, initially with NARX structures. This choice is convenient due to the structural flexibility characteristic and the ability of these models to describe non-linear systems (Pearson, 1999; Martins et al., 2013).

As mentioned in Chapter 2, one of the main steps in the system identification procedure from data is the step of choosing the model structure. Works that aim to help on this task have been proposed, as is the case of the papers published by Baldacchino et al. (2012, 2013) and Falsone et al. (2014, 2015), in which a random approach is used in order to select the best candidates among the possible regressors in the formation of the model to be identified.

More recently, Retes and Aguirre (2019) extended the RaMSS strategy for choosing NARMAX model structures, which take into account the effect of noise on signals in order to reduce the polarization in parametric estimates.

In the scope of controller design based on data, more specifically by the VRFT method, in which the controller parameters are adjusted by conventional identification techniques such as OLS, ILS, IV, among others, the same problem of choosing the model structure occurs, with the difference that the model now represents the controller, and no longer the process. Therefore, analogies must be able to be made intended to use the RaMSS method in order to extend it for the purpose of identifying the most suitable structure for the controller.

In this sense, the present work has been directed to the study of the possibilities and consequences of using these technologies for control purposes, aiming to choose

the most suitable controller structure.

The next section presents the basics of the methodology adopted for the RaCSS procedure, and then, some preliminary results obtained in the first stages of this research are studied.

4.1 Methodology

In order to achieve the objective of using a strategy for structural and parametric identification of controllers from a DDC approach, this work intends to make use of the two methodologies covered in previous chapters: VRFT, for tuning controller parameters and RaMSS, to select the best controller structure. From the study and adaptations of previous works (Retes and Aguirre, 2019), an algorithm has been developed, capable of uniting the two strategies in order to assist in the selection of structures, as in the tuning of non-linear (or linear) controllers. The algorithm has served as a framework for the study and validation of the proposed approach, based on numerical simulations. In this sense, part of the research efforts has turned to the numerical implementation of this framework. In a first step, the behavior of the algorithm in the identification of process models was analyzed, using the RaMSS algorithm, as originally proposed. Some of the validation results for this step was presented in Chapter 2, in the form of Example 2.2.1.

Once the behavior of the algorithm has been validated, the VRFT method, discussed in Chapter 3 is added to the process, so that the performance of RaMSS in selecting structures for a controller can be evaluated in practice. In this step, the RaMSS algorithm, presented in Chapter 2 is slightly modified to deal with parametric estimation via VRFT, for the first tests of controller design. In this scenario, some first results are expected, without major commitments with good performances, but they are of great value for analysis and comparison of the proposed modifications. Some of these results are presented by the examples in the Section 4.2 below, more specifically in the Example 4.2.1.

As discussed in Chapter 3, under certain conditions, the controller can be identified by traditional systems identification procedures using the VRFT method. If the necessary conditions are not met (see Assumption 3.2 - matched control) a filter can be designed so that the minimization of the cost index related to the input and output signals of the controller, given by $J_{VR}(\boldsymbol{\theta})$, also results in the minimization of the $J_{MR}(\boldsymbol{\theta})$ cost index related to the tracking error of a reference model. This relationship is guaranteed by the filter as long as the model structure is not too "distant" from the ideal structure, i.e. as long as it is not too under-parameterized.

A proposal of the present research is to use the RaMSS procedure to try to identify the ideal structure for the controller, or at least a close structure that generates good results when using the VRFT procedure. The identification of this structure is made from data collected under the VRFT approach. In an attempt to validate the proposal, simulations

are carried out in different scenarios. In a first scenario, the RaMSS procedure is applied to a linear process where their model and the model of the ideal controller (matched control) are previously known. The aim here is to analyze the behavior of the procedure in the selection of candidate regressors for a possible controller. A randomized analysis is applied to the process in 2 cases: in the first, the procedure is performed without the influence of noise on the data used; in the second, noise is added to the data in order to simulate a more realistic environment. Example ref ex: sis2aord analyzes these cases.

The RaMSS procedure was originally proposed to deal with process identification, not controllers. The update of the vector of the Regressor Inclusion Probabilities (RIPs) is done based on performance indexes that evaluate the model's prediction quality. This prediction is made on training data, using an exponential version of MSPE, and possibly, in order to increase robustness on validation data (free run simulation), through an exponential version of MSSE. However, for control purposes, these indexes may not be the best. Mainly the index based on MSSE, which requires higher computational cost under the proposal to promote improvements in the robustness of the identified model. The effect is a better prediction of the output signal of the identified model, even when excited with signals different from those used in the identification. Despite the lack of further studies, at first, this index does not seems to bring many benefits when used for the controller's MSS (Model Structure Selection) purposes, since the main objective, in the design of a controller, is to make the process output (and not the controller's output), behave as desired.

In this sense, it is proposed adaptations to the original RaMSS method aiming at objectives more suitable to MSS of controllers. A studied modification at the moment is the replacement of the exponential version of the MSSE, given by $J_s = e^{-K \cdot MSSE}$, by one that takes into account the closed-loop system tracking, one of the main objectives of the controller. This new index is defined as

$$J_r = e^{-K \cdot MSTE}, \tag{4.1}$$

where $MSTE$, is defined here as the mean squared tracking error, given by

$$MSTE = \frac{1}{N} \sum_{k=1}^{N} (y_{rm}(k) - y(k))^2, \tag{4.2}$$

where y_{rm} is the reference model output and $y(k)$ is the closed loop output, to same the reference signal.

The final performance index $\mathcal{J}$, used in $\mathcal{I}$ (see Equation 2.6) to compute the RIPs, is now calculated by:

$$\mathcal{J} = \alpha \mathcal{J}_r + (1 - \alpha) \mathcal{J}_p \tag{4.3}$$

where J_p remains the same given by (2.11), an exponential version of MSPE. This new index takes into account the performance of the closed loop controller in the process of

updating the RIPs.

Since the choice of structure is made by the modified RaMSS method, incorporating elements of the VRFT strategy, and the aim is the selection of structure and parameters for controllers, this strategy is named here *Randomized Controler Structure Selection* , or RaCSS.

Preliminary results from the use of this index are presented in next sections. Examples 4.2.2 and 4.2.3 show some of these results and a general analysis is done in Chapter 5.

In Chapter 5, proposals for continuing the work, such as the inclusion of auxiliary information to the procedure, aiming to incorporate previously desired or known characteristics for the controller, as well as possibly the use of reinforcement learning strategies, in order to reduce the computational cost due to the use of closed loop information from the controller are presented.

Finally, discussions about stability, convergence, polarization and robustness are presented, not yet in a context of guarantees, but as conjectures of steps that we intend to deepen in the sequence of this research.

4.2 Preliminary Results

Example 4.2.1 (Second order Linear System – continued.)**.** Consider the 2nd-order linear system describe in Example 3.5.1:

$$y_k = a_1 y_{k-1} + a_2 y_{k-2} + b_1 u_{k-1} + b_2 u_{k-2}.$$

According to the VRFT strategy (Chapter 3), one must first define a class of allowable controllers $\mathscr{C}$, containing the desired structure, and define a reference model that expresses the desired close loop system's behavior.

In this example, we want to analyze the use of the RAmSS algorithm directly, without modifications, in order to find the best structure for a controller designed from the VRFT strategy. In order to examine the behavior of the RaMSS Algorithm in finding the best set of regressors, we proceed as follows.

A feasible reference model is defined, that is, with a relative degree equal to or greater than that of the process. For simplicity, a 1st order model with the same relative degree of the process is chosen, as Example 3.5.1, given by the transfer function

$$M(z) = \frac{1 - A}{z - A},$$

The parameter adopted here is the same as Exemple 3.5.1, i.e. $A = -T_s/\tau_d = 0.8187$. The ideal controller, represented by an ARX model, will be given by the structure composed

of the regressors presented in (3.15), represented here by

$$u_k = \theta_0 e_k + \theta_1 e_{k-1} + \theta_2 e_{k-2} + \theta_3 u_{k-1} + \theta_4 u_{k-2},$$

and will have the parameters shown in (3.16), described here as

$$\boldsymbol{\theta}^\star = \begin{bmatrix} \theta_0^\star & \theta_1^\star & \theta_2^\star & \theta_3^\star & \theta_4^\star \end{bmatrix}^T = \begin{bmatrix} 0.181 & 0.308 & 0.145 & 0.19 & 0.81 \end{bmatrix}^T.$$

Using the RaMSS algorithm, as discussed in the section 2.2, the procedure is performed for 2 cases:

case A the universe set $\mathscr{M}$ is taken as all possible 3rd degree non linear models formed by the monomials up to 4 delays for input $\tilde{e}_k$ (virtual error) and output $\tilde{u}_k$ (virtual process input) collected data. No noise is considered in this case.

case B the same case as case A, but now with a noise in the output, given by a gaussian distribution, i.e. $\nu \sim \mathcal{N}(\mu, \sigma)$, where $\mu = 0$ is the mean, and $\sigma = 0.1$ is the standard deviation adopted. Note that the noise is added in the output, in a way that the model for the process is represented by an output error model (OEM).

The RaMSS procedure is applied to a data set of 700 samples, obtained via the VRFT procedure, in which the VRFT filter is considered to be unitary, i.e. the data is not filtered. The same data set is applied for the 2 cases. At each iteration, 100 candidate models are randomly chosen to be analyzed, using a Bernoulli process. The RIPs are updated in a maximum of 100 iterations, or until they all converge to a margin above 0.9 or below 0.1, i.e. the following stopping criterion is adopted:

$$\textbf{if } [iter < iter_{\max}] \textbf{ or } \left[\Delta_S < \frac{1}{2}\left(1 - \min_{\forall \mu \in \boldsymbol{\mu}_k} \left|2\mu - 1\right|\right)\right] \textbf{ then}, \text{ STOP the procedure.} \quad (4.4)$$

where $\Delta_S = 0.1$ is the convergence margin, $iter$ is the iteration number and $iter_{\max} = 100$ is the maximum allowed iterations.

Applying the procedure in both cases, the following controllers are obtained, for a specific realization:

$$\begin{aligned} u_1(k) &= 0.636u_1(k-1) + 0.513u_1(k-2) + -0.321u_1(k-3) + 0.17u_1(k-4) \\ &\quad + 0.181e_1(k) + 0.227e_1(k-1) + 0.045e_1(k-2) + 0.03e_1(k-4) \\ u_2(k) &= 0.737u_2(k-1) + 0.242u_2(k-2) + -0.329u_2(k-3) + 0.32u_2(k-4) \\ &\quad + 0.175e_2(k) + 0.197e_2(k-1) + 0.049e_2(k-2) + 0.049e_2(k-3) + 0.059e_2(k-4) \end{aligned} \quad (4.5)$$

where the index subscribed to the variables represent the respective cases.

The table 4.1 sumarizes the simulation parameters used in 1, for the 2 cases, where o is the maximum allowed degree for the regressors, $n_{\tilde{e}}$ is the maximum delay for the virtual error signal $\tilde{e}(k)$, $n_{\tilde{u}}$ is the maximum delay for the input signal of the sampled

plant, N_p is the number of models chosen at each update of the RIPs, $iter_{\max}$ is the maximum number of allowed iterations, Δ_S is the trashold for convergence of RIPs, K is the gain for the performance indexes presented in 2.2, γ_0 is the initial gain of 2.2, $\mu_{\min}$ and $\mu_{\max}$ are the minumum and maximum values allowed for the RIPs and ν is the noise added to the output.

Table 4.1: Parameters for simulating the RaMSS algorithm of the example 4.2.1

Case	o	$n_{\tilde{e}}$	$n_{\tilde{u}}$	N_p	$iter_{\max}$	K	γ_0	$\mu_{\min}$	$\mu_{\max}$	ν
A	3	4	4	100	100	1	2	0.05	1	0
B	3	4	4	100	100	1	2	0.05	1	$\mathcal{N}(\mu, \sigma)$

Note that the controllers presented in (4.5) are obtained for a specific realization of the procedure. All terms that end with the RIP above 0.95 are considered in the hundredth iteration. Figure 4.1 shows the evolution of the RIPs for this realization.

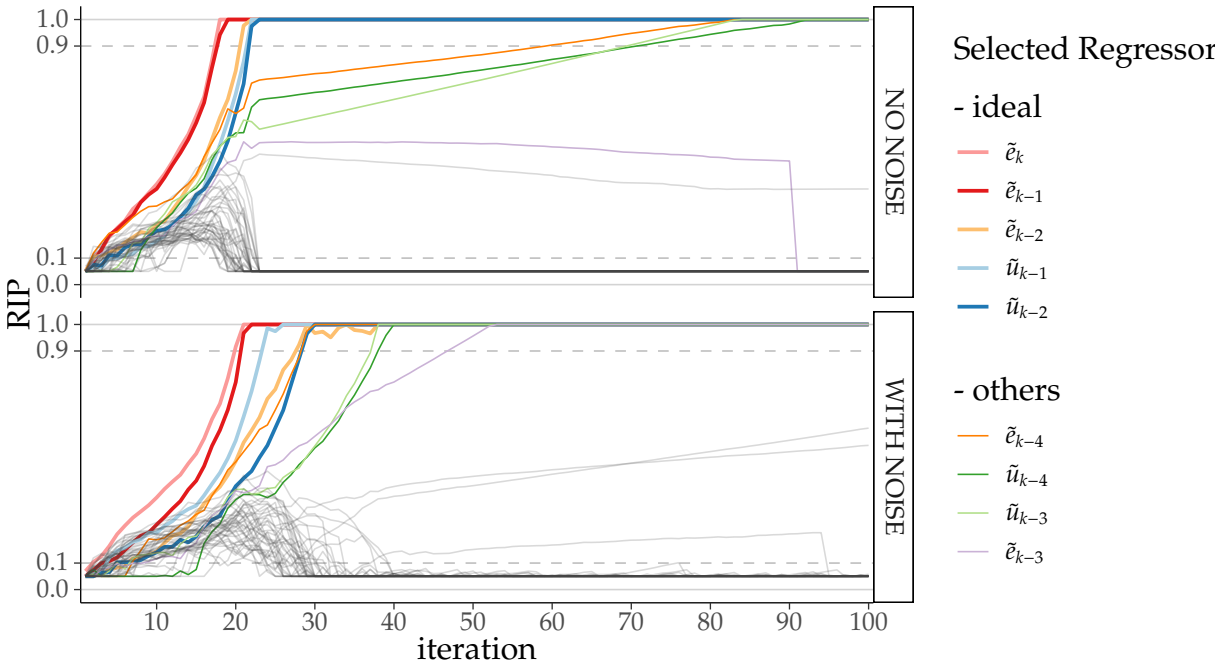

Figure 4.1: Typical evolution of RIPs for choosing regressors for case 1 and 2.

The ideal regressors are selected before the first 40 steps. But as the iterations continue, the $\tilde{u}(k-3)$, $\tilde{u}(k-4)$ and $\tilde{e}(k-4)$ regressors continue to increase monotonically in value and eventually end up being selected. The effect of including these terms, although in general they are small, deteriorates the desired behavior of the controller. When considering noise in the measurement (case 2), the selection of non-ideal regressors occurs even earlier, as can be seen in the lower graph of 4.1. Part of this result

is due to the worsening of parametric identification during the VRFT procedure, provoked by the polarization effect introduced by noise at the output. The result is that the regressor vectors are no longer orthogonal to the residuals and the OLS estimator becomes polarized. A strategy to mitigate this effect is to use non-polarized estimators, such as VI estimators, or even the ELS.

Figure 4.2 shows the temporal response for the controllers identified in (4.5), when the reference signal is taken as a square wave.

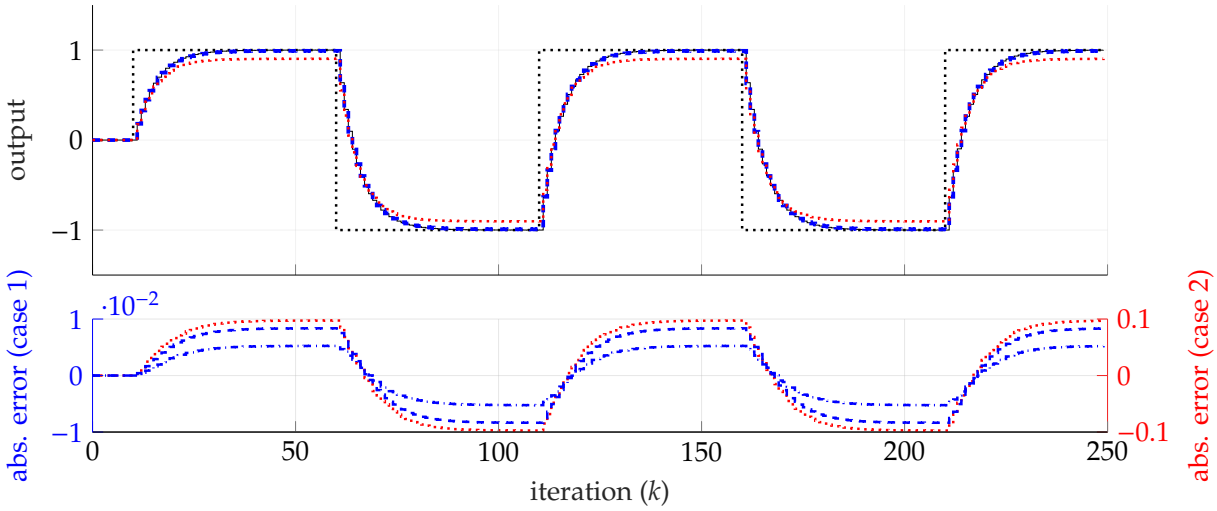

Figure 4.2: Closed-loop system response (upper graph) and respective absolute errors (lower graph) using the controllers identified in case A (- - - -) and case B (·····). The reference (·····) and the reference model response (——) signals are shown in the upper graph. The error for case considering only the ideal structure is represented by (-·-·-·), in the lower graph.

The behavior of the case with noise is deteriorated in relation to case 1 (without noise). The steady-state error for case 2 is about 10 times greater than case 1, as shown in the lower graph in Figure 4.2 (note the different scales in the error graph in the figure). This greater error is due to two factors: selection of an over-parameterized structure of the model by the RaMSS procedure, and worse parametric identification due to the presence of noise at the output, which can result in parameter polarization. Note that, as the noise is added to the process output (OEM), and by the procedure of filtering by the inverse of the plant when applying the VRFT, the noise, although white, can cause polarization in the identified parameters. The error graph in Figure 4.2 also shows the time response for the case without noise, but considering that only the ideal regressors are taken into account in the identification process. In this case, the error decreases, and and this decrease is attributed to the over parameterization caused by the extra regressors selected in case 1. Despite this, a small error remains on steady state. This fact is attributed to the effect discussed in Chapter 3, in which small errors in the identification of parameters make the sum of the coefficients in $y(k)$ not

exactly zero, resulting in a system with high gain at low frequencies, but not infinite. As discussed, in the chapter 3 and to be proposed in the chapter 5, it is hoped that this problem can be solved by imposing restrictions on the identification process (use of auxiliary information).

The results discussed in the example 4.2.1, take into account only one performance of one specific realization. A more realistic analysis would be to consider various realizations, and observe the behavior in the form of statistical distributions. These results are shown in the next example.

Example 4.2.2 (RaCSS applied to a 2nd order linear system)**.** In this example, the same cases contemplated in the previous example are considered, but with the following modifications:

1. The index used to calculate the RIPs (2.12), is formed according to (4.3), i.e., the RaCSS procedure is used, described in Section 4.1, now taking into account the tracking error information in the RIPs updating procedure.
2. Several simulations, or realizations, are analyzed, but maintaining the same training data, i.e. $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{y}}$ remain the same for each realization. What changes are the candidate models chosen during the procedure, since this choice is based on a Bernoulli process.

As a first step, the evolution of RIPs are analyzed for 5 different values of α, which are: $\alpha = 0$, $\alpha = 0.25$, $\alpha = 0.5$, $\alpha = 0.75$, $\alpha = 1$.

The 5 cases, considering that there is no measurement noise and with the same values as Example 4.2.1, are analyzed for the RaCSS procedure. The evolution of the RIPs for each case is presented by Figure 4.3. It is observed that when increasing the value of α, regressors that do not belong to the set of ideal regressors are less frequently selected, when compared to the case where no closed loop information is used (case with $\alpha = 0$) *.

Note that, for this realization, for $\alpha = 0.5$ and $\alpha = 1$ the algorithm was interrupted around iterations 60 and 90, respectively, due to the stop criterion (4.4) having been satisfied. It is emphasized that this occurs for this specific realization, the same may not happen for other realizations, due to the random character of the method. Despite this, in general, when performing several iterations, it is possible to notice that the general behavior of Figure 4.3 prevails, with greater choices of spurious RIPs for the case, in which $\alpha = 0$, when compared to the others ($\alpha > 0$).

To analyze the behavior more generally, 50 realizations are made (with the same training data) for each value of α. The density of probability of convergence of the RIPs for each value of α, as a function of the number of iterations, is shown in Figure 4.4. In the figure, it is clear that the method selects the ideal parameters well for all cases. In

*note that the case where $\alpha = 0$ in this example corresponds to case A of Example 4.2.1.

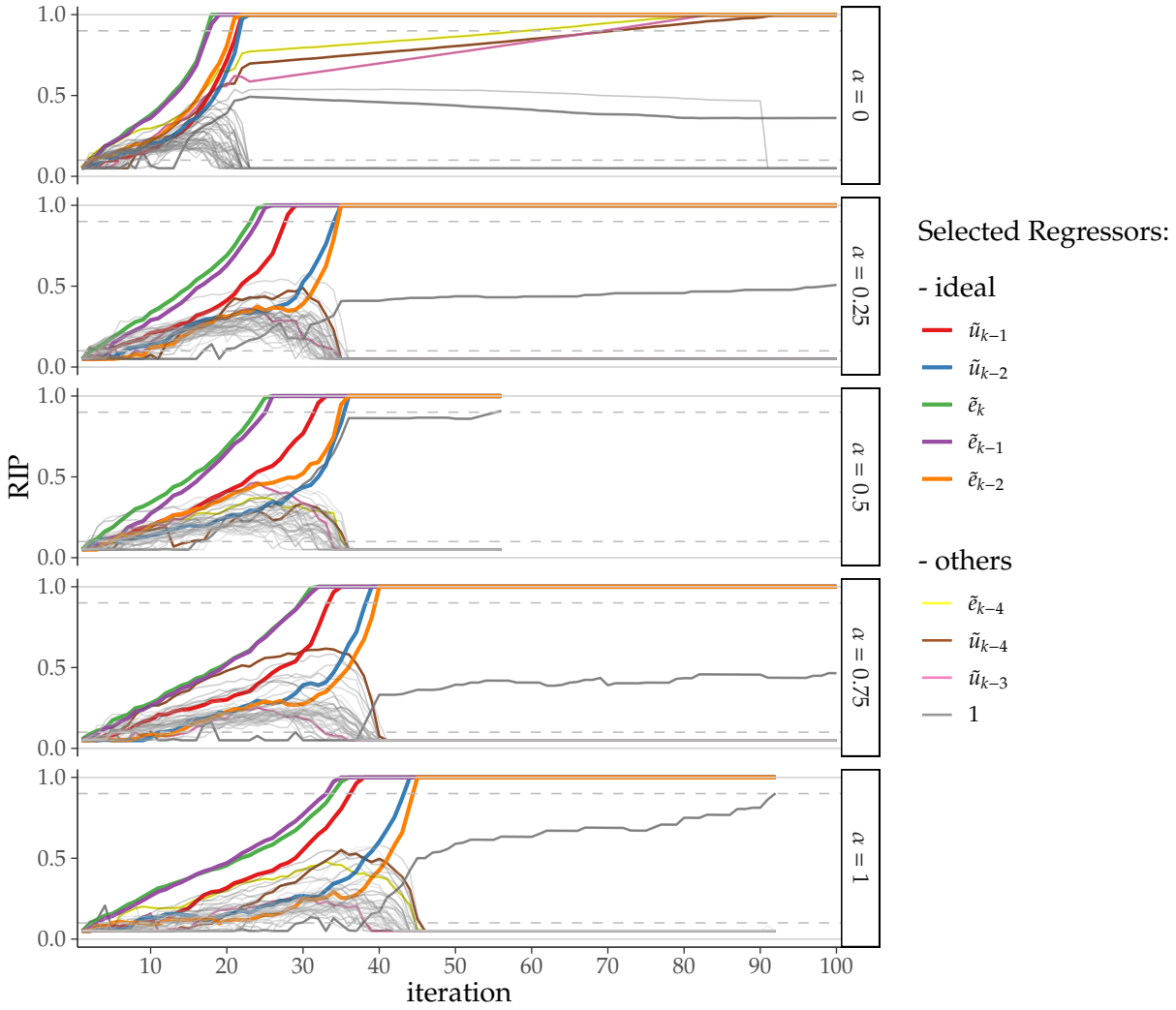

Figure 4.3: RIPs evolution to different values of parameter α considering data without noise.

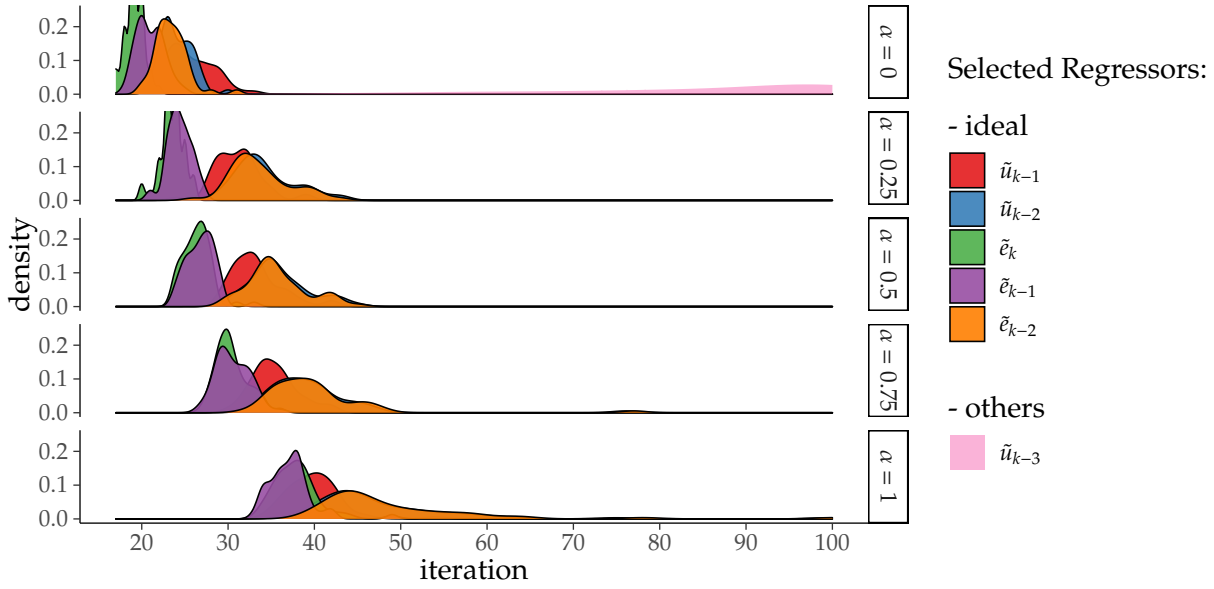

Figure 4.4: Probability densities of selection for the regressors selected within 100 iterations considering different values of α for the case without noise.

the case without using the tracking error information, $\alpha = 0$, the $\tilde{u}(k-3)$ regressor is chosen with low probability for few iterations but with increasing probability with the increase in iterations.

When tracking error information is taken into account, cases with $\alpha > 0$, only the ideal regressors are selected for all 50 realizations. Note that the overall average time to choose the regressors increases with the increase in the α parameter. One possible explanation is that the performance index related to the tracking error, J_r, changes little compared to the index related to the prediction error, J_p. A probable solution may be to adopt a larger K gain in the calculation of the MSTE (see equation 4.1).

The same previous simulations are done for the case in which there is measurement noise (noise at the output), as in case B of Example 4.2.1. Figure 4.5 shows the evolution of RIPs and Figure 4.6 shows the probability density of choice of regressors as a function of the number of iterations for each value of α, considering this new case.

Looking at Figure 4.5, it is observed the following facts: 1) in all cases the ideal regressors were selected. 2) With the increase of α, non-ideal regressors have a lower probability of being chosen. This is more evident when comparing several realizations, as shown in Figure 4.6. Comparing $\alpha = 0$ with $\alpha > 0$, it can be seen that the probability of selecting non-ideal regressors decreases for smaller iterations. However, for values greater than $\alpha = 0$, there is a longer convergence time for the regressors. For example, we notice a greater number of regressors that do not converge to the minimum value $\mu_{\min}$ for higher values of α. The polarization problem may be one of the factors responsible for this behavior.

Nos exemplos anteriores, o algoritmo RaCSS é analisado para um caso em que o

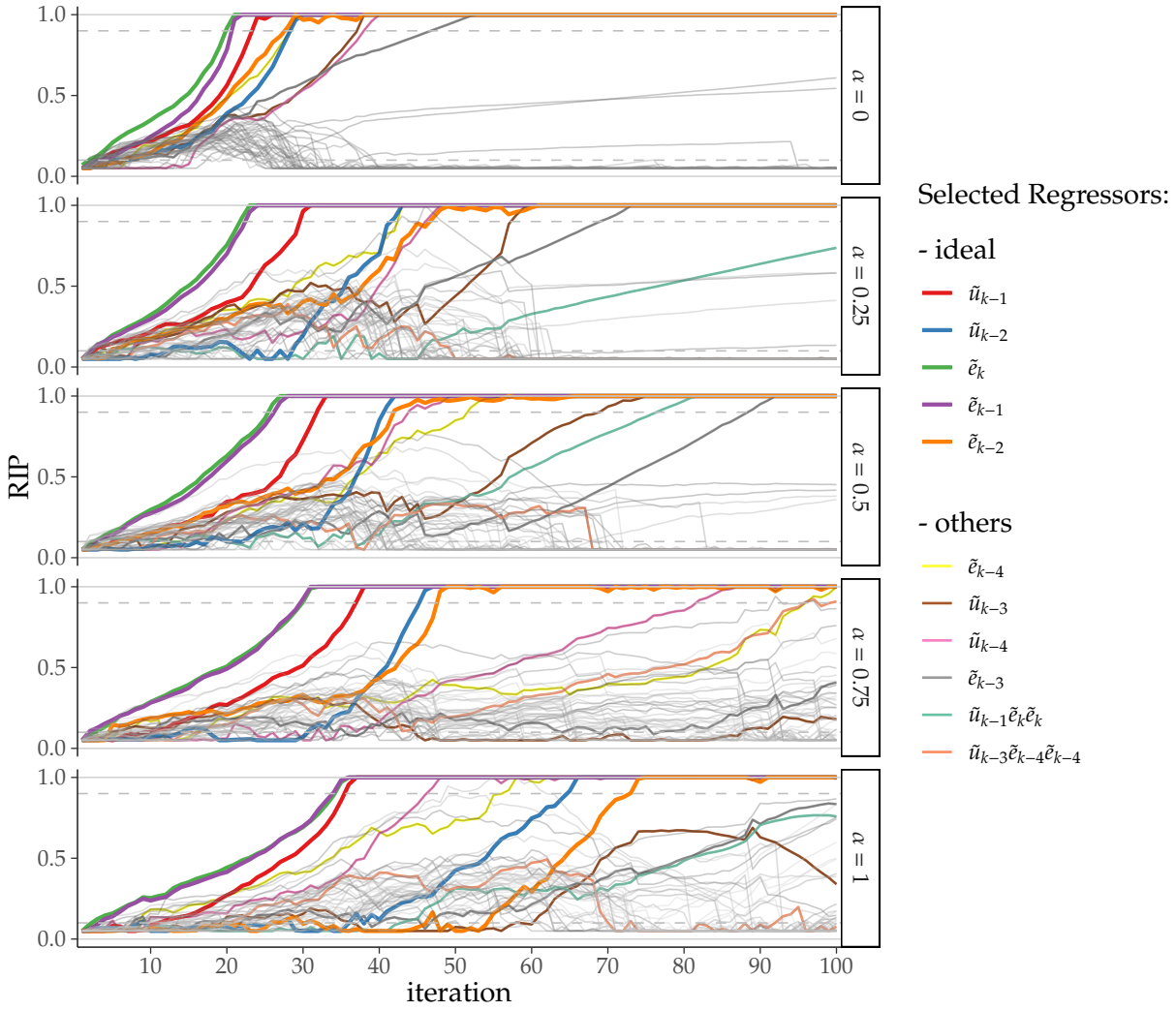

Figure 4.5: RIPs evolution to different values of parameter α considering data with noise.

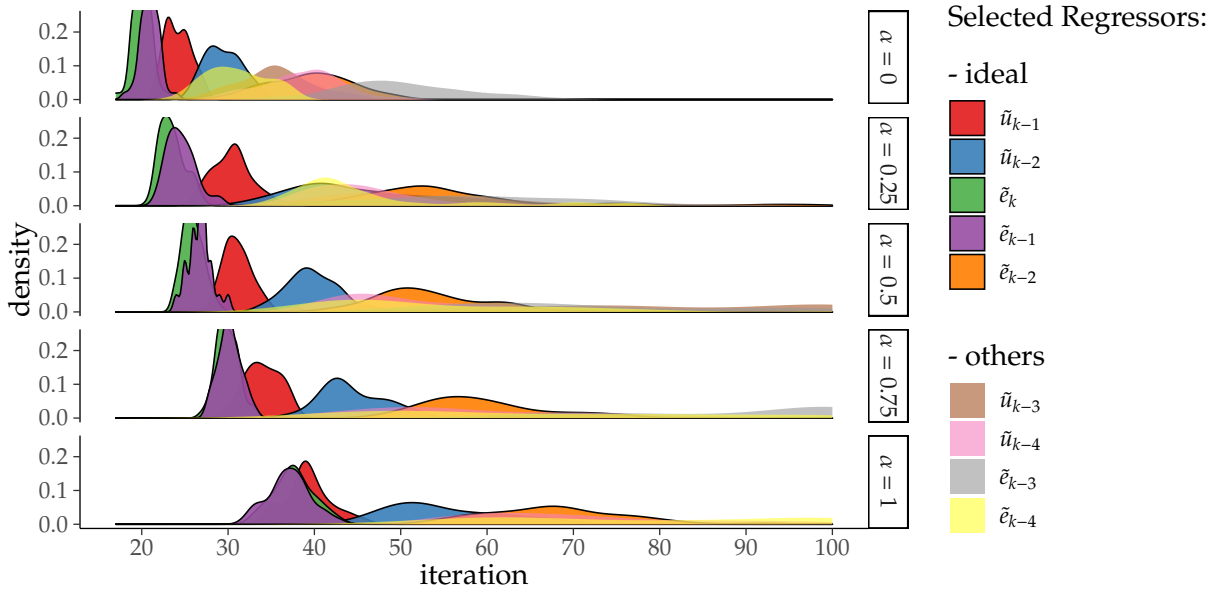

Figure 4.6: Probability densities of selection for the regressors selected within 100 iterations considering different values of α for the case without noise.

processo é linear e o controlador ideal é conhecido. Na sequência, apresenta-se um exemplo de aplicação para um processo não linear.

Example 4.2.3 (RaCSS aplicado a um sistema não-linear)**.**

In this example, the behavior of the RaCSS algorithm is analyzed in the structure selection for a non-linear process. The model of a small electric heater, with variable dissipation, is adopted. The variation in dissipation is the result of the activation of a fan. The input signal is the electrical voltage applied to the heater and the output is the amplified signal from a thermocouple. The model obtained by an identification process in a real system, as detailed in (Aguirre, 2015), section 16.6., Is given by

$$\begin{aligned} y(k) = 1.2929y(k-1) + 0.0101u(k-2)u(k-1) + 0.0407u^2(k-1) - 0.3779y(k-2) \\ - 0.1280u(k-2)y(k-1) + 0.0957u(k-2)y(k-2) + 0.0051u^2(k-2) \end{aligned}$$

The same signals sampled presented in Aguirre (2015) are used to calculate the virtual values used in the RaCSS procedure. These signals are shown in Figure 4.7. The original sampling period is 12 seconds, however, for simplicity, it will be considered here that this sampling period is one second, $T_s = 1$ s. This has no effect on the results to be obtained for the purposes of this numerical example. Making this simplification, it is considered a reference model given by a first order system, with a transport delay of 3 seconds and a time constant of 10 seconds, resulting in the following transfer function in continuous time

$$M(s) = \frac{e^{-\tau_d s}}{\tau s + 1},$$

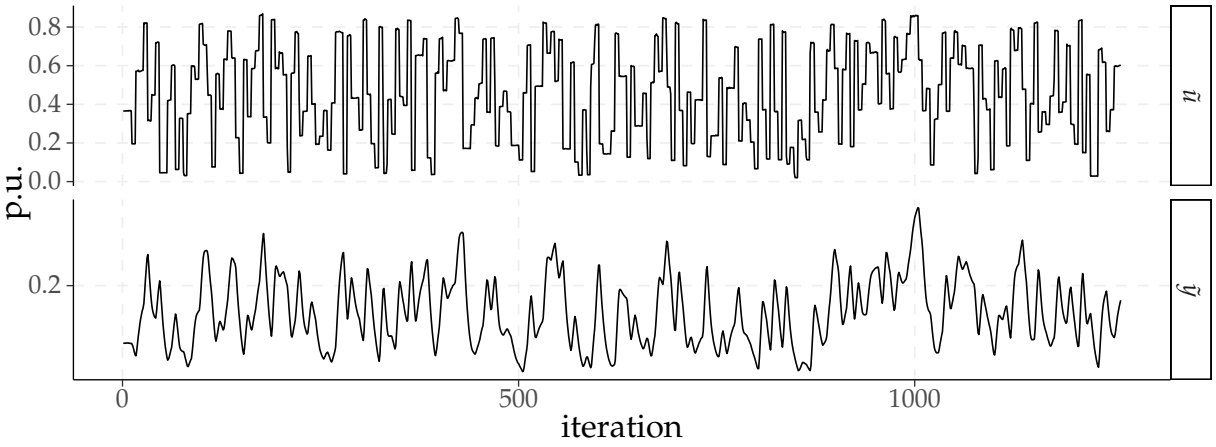

Figure 4.7: Input signal $\tilde{u}$ (upper graph) and heater output $\tilde{y}$ (lower graph), in p.u.

where $\tau = 10$ s is the desired time constant and $\tau_d = 3$ s is the transport delay. A discrete version of this model is given by

$$M(z) = \frac{1 - e^{-T_s/\tau}}{(z - e^{-Ts/\tau})z^{\tau_d/Ts-1}}.$$

Disregarding noise in the process, the RaCSS procedure is applied in order to obtain a controller that results in a closed-loop system that behaves according to a given reference model. It is considered all regressors formed by monomials up to 8 delays in $\tilde{u}$ (controller output signal) and $\tilde{e}$ (virtual error) as candidate regressors to RaCSS. The parameters adopted in the RaCSS algorithm are presented in Table 4.2, where α is the factor given in (4.3) and the meanings of the other symbols are the same as presented in Table 4.1 .

Table 4.2: Parameters used in the RaMSS algorithm of the example 4.2.3

o	$n_{\tilde{e}}$	$n_{\tilde{u}}$	N_p	$iter_{\max}$	K	γ_0	$\mu_{\min}$	$\mu_{\max}$	ν	a
2	8	8	100	200	1	2	0.01	1	0	0.5

A typical evolution for the RIPs, after 200 iterations is shown in Figure 4.8.

There is a much less regular behavior with those obtained in Example 4.2.2. At the end of the 200 iterations, 15 regressors are selected. Although there are still other regressors that have not yet converged to the minimum and maximum value, and a certain degree of classification can already be obtained from the results. It should be noted that there are a total of 171 possible regressors, which results in a total of approximately 2^{171} possible models.

The closed loop response considering the first 15, 19 and 27 regressors is shown in Figure 4.9. The MAPE, MSTE and J_r index values (see (4.2) and (4.3)) are presented in Table 4.3.

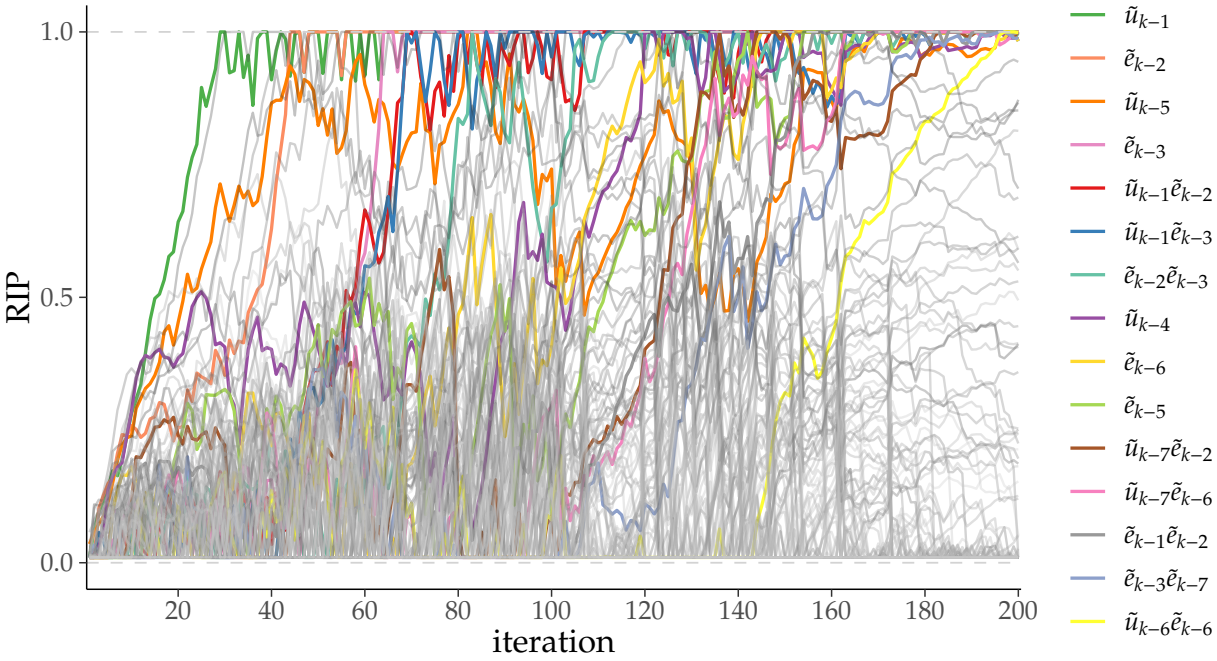

Figure 4.8: RIPs evolution for example 4.2.3.

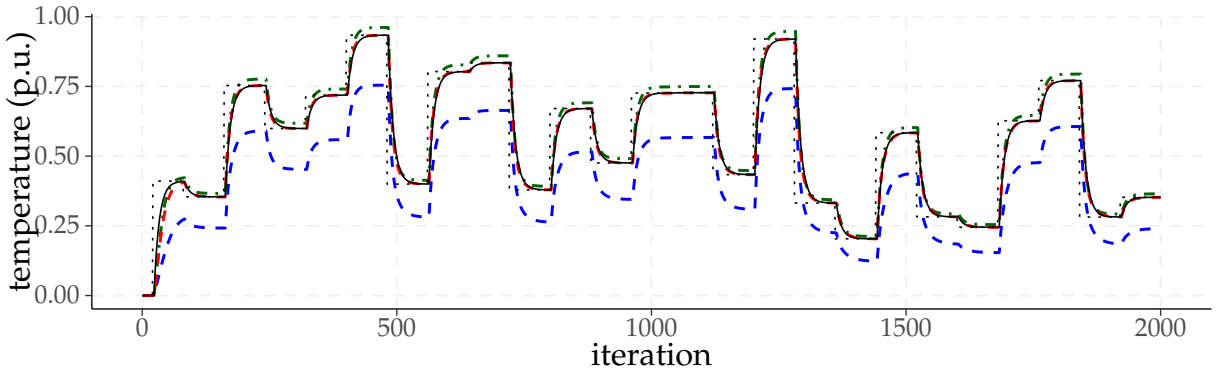

Figure 4.9: Closed loop response for controllers obtained using RaCSS choosing the first 15 (- - - -), 19 (-·-·-·-) and 27 (- - - -) regressors with greater RIPs , together with the reference signal (·····) and the reference model response (——).

It is observed that when considering 27 regressors the answer is practically identical to the desired one, with a MAPE value of only 2.14. No further analysis was done on the classified parameters. In a new analysis, we intend to separate the selected regressors and proceed again with the RaCSS procedure, but this time, only with these pre-selected regressors.

Table 4.3: Parameters used in the RaMSS algorithm of the example 4.2.3

	MAPE	*MSTE*	J_s
15	69.02	0.2142	0.8072
19	9.59	0.0050	0.9950
27	2.14	0.00116	0.9988

Chapter 5

Conclusions

5.1 Final Considerations

The current research is in the development phase, but with the first results it is already possible to infer that the RaCSS approach works, in the sense that it helps in the selection of structures and design of DDC controllers. Further analysis will be carried out in order to obtain information regarding the method's robustness, performance and limitations.

During the development, some proposals for investigation were emerging, which, for the most part, could not be investigated until now. As this work is still in the development phase, these proposals are presented as continuity proposals and are listed in more detail in the following section.

5.2 Continuity proposals

The main focus of this research has been the study and development of design methodologies for DDC controllers, with a special attention on the controller structures selection using the VRFT approach.

In particular, the RaMSS method, has shown to be promising for the desired application. This method appeals to exploratory techniques that uses Monte Carlo-style random searches, but with selection mechanisms that dramatically reduce computational cost, avoiding an exhaustive search, while trying to ensure an adequate selection of regressors. Supported by this technique, some adaptations have been studied in order to deal with the identification and structure selection of the controller, and not of the process, as proposed by RaMSS approach.

Following, specific items to be addressed are presented, which can be taken as proposals for continuity of the current work:

Proposal 1

A first proposal concerns the study of the use of performance indices that best represent the reality of the controller when updating the terms of RIPs. As already discussed (Section 2.2), the RaMSS originally uses measures such as MSPE and MSSE for the RIPs updates. For modeling of physical processes purposes, where the prediction of input-output behavior is often the target, such indices are adequate Falsone et al. (2015).

However, when dealing with the controller identification problem, the minimization of the prediction error, be it a step-ahead error, as in the case of MSPE or a free simulation error, as in the case of MSSE, there is no guarantee that the error of tracking is reduced. And in general, this tracking error is the main target in control systems. , A study on the use of this index is underway, according to results presented in Chapter 4. What has been observed is that minimizing the MSPE directly by the RaMSS strategy, despite often showing good results at prediction of the controller output view point, does not guarantee that the optimal tracking is achieved, i.e. the one in which the mean squared tracking error is minimal.

As an alternative, it is proposed to use an index that takes into account the result in closed loop when applying the intermediate controllers identified and selected by the procedure. Something similar has been used with regard to the identification of processes in which the mean quadratic simulation error (MSSE) is used to replace the MSPE (Aguirre et al., 2010). In this case, according to (Piroddi and Spinelli, 2003), the use of information from the free simulation can improve the robustness in the selection of the process structure when under partial identifiability conditions.

The MSSE depends on free-run simulation, which in short is the open loop response to a known (and new) excitation signal. Something similar could be used when evaluating the structure in the VRFT procedure. In this case, it would be desirable to use the system closed loop response with the controller obtained from the evaluated structure to calculate a new index. As discussed in Chapter 4, the MSE of the tracking error seems to be a good choice (see equation 4.2). The big problem in this case is that this simulation becomes dependent on a model, even if approximate, on the process, or on the process itself. However, as DDC strategies aim at exactly not identifying a model for the process, such a situation can be a practical obstacle. A proposal for a solution to this problem is presented later in this text.

Proposal 2

A second proposal, which has also been analyzed, consists in the use of auxiliary information during the process of identifying the parameters of the controller. Although techniques have already been developed to incorporate auxiliary information in the identification process, for example via restrictions and multiobjective optimization (Barroso, 2006), all these restrictions are related to the plant. In this sense, questions arise such as: how can these techniques be used in the DDC approach? Would it be possible to find an analogue of auxiliary information, used in traditional methods, for DDC strategies, in which there is no plant information? Could this auxiliar information be defined, for example, from restrictions that guarantee aspects relevant to control, such as gain limitations due to saturation of actuators, insertion of integrators in the controller, in addition to aspects related to robustness? In this sense, a procedure to guarantee the presence of integrators in the controller model has already been studied, but still with no conclusive results.

Proposal 3

Another proposal to be studied concerns in the use of filters in the process of identifying the controller, as is common in the VRFT strategy. In this strategy, when the ideal or compatible controller does not belong to the class of controllers considered, that is, the hypothesis 3.2 is not satisfied, a filter to be applied to the signals and regressors used in the identification process must be designed in order to approach an optimal solution, as presented by Campi et al. (2002); Campi and Savaresi (2006) and discussed in Chapter 3. So far, the results presented in Chapter 4 have not made use of this filter. This should be the next step to be taken in the procedure under development.

Proposal 4

A inherent problem to both, systems identification methodologies, and the VRFT method, is the polarization of parameters that appear when the process to be identified (be it the plant or controller) is subject to colored noise. It is common to use instrumental variable estimators (VI) to mitigate the effects of polarization in the VRFT approach. It is proposed to study the use of the extended least squares estimator (ELS) during the controller structure selection and parameter identification, aiming to reduce the polarization.

Proposal 5

A known problem in controller designs based on reference models is the choice of the appropriate reference model. Especially with regard to the choice of the pure time delay to be considered in the reference model. If a model is chosen with shorter delays than the plant, for example, the identification of the controller is severely impaired. In this sense, we intend to analyze the possibility of searching for a suitable time constant for the reference model in a randomized way, during the structure selection procedure. Another possibility is to analyze some pre-established structures for the reference model in order to allow better adjustments of the controller. For example, during the RaCSS procedure, there is also a search for reference models in different orders, but with, for example, the same accommodation time.

Proposal 6

One last proposal concerns the problem raised earlier, in first proposal, where the output signal, and consequently the tracking error, are required to calculate the update rate of the RIPs in the RaMSS (or RaCSS) procedure. As the simulation of the closed-loop (or even open-loop) response is not viable in DDC strategy, due to a lack of a model for the process, is intend to make use of Reinforcement Learning (RL) techniques, discussed in Chapter ref cap: RL, so that data collected from the real process, while in operation, can be used for real-time calculation of RIPs and consequently for choosing the best structure.

Some RL techniques are promising in this case, since they often lead to the optimization of performance indexes based on sampled data, without the need for the process model, while avoiding the high number of sample achievements as in Monte Carlo methodologies. Among them, the TD Learning strategy (Sutton and Barto, 2018),

has been considered, with attention to the method known as Q-learning Watkins and Dayan (1992), which allows a controller to be adjusted in an approach known as *off-policy*, that allows the learning of a control policy while the closed-loop system is under the influence of another. In the scope of this work, the feeling is that it should be possible to use a similar strategy together with the RAmSS concept, to be able to update the RIPs and, consequently, the structure, and perhaps even parameters, of the controller with less effort and, if possible, with evidence of convergence.

Bibliography

Aguirre, L. A. (2015). *Introdução à Identificação de Sistemas: Técnicas Lineares e Não Lineares: Teoria e Aplicação*. Editora UFMG, Belo Horizonte, fourth edition.

Aguirre, L. A., Barbosa, B. H. G., and Braga, A. P. (2010). Prediction and simulation errors in parameter estimation for nonlinear systems. *Mechanical Systems and Signal Processing*, 24(8):2855–2867.

Aguirre, L. A. and Billings, S. (1995). Dynamical effects of overparametrization in nonlinear models. *Physica D: Nonlinear Phenomena*, 80(1-2):26–40.

Aguirre, L. A., Donoso-Garcia, P. F., and Santos-Filho, R. (2000). Use of a priori information in the identification of global nonlinear models-a case study using a buck converter. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 47(7):1081–1085.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Allen, D. M. (1974). The Relationship between Variable Selection and Data Agumentation and a Method for Prediction. *Technometrics*, 16(1):125–127.

Alves, M. A., Corrêa, M. V., and Aguirre, L. A. (2012). Use of self-consistency in the structure selection of NARX polynomial models. *International Journal of Modelling, Identification and Control*, 15(1):1.

Baldacchino, T., Anderson, S. R., and Kadirkamanathan, V. (2012). Structure detection and parameter estimation for NARX models in a unified EM framework. *Automatica*, 48(5):857–865.

Baldacchino, T., Anderson, S. R., and Kadirkamanathan, V. (2013). Computational system identification for Bayesian NARMAX modelling. *Automatica*, 49(9):2641–2651.

Barroso, M. F. S. (2006). *Otimização Bi-Objetivo Apliada à Estimação de Parâmetros de Modelos Não-Lineares: Caraterização e Tomada de Decisão*. PhD thesis, UFMG.

Bazanella, A. S., Campestrini, L., and Eckhard, D. (2012). *Data-Driven Controller Design*. Communications and Control Engineering. Springer Netherlands, Dordrecht.

Bazanella, A. S., Gevers, M., Mišković, L., and Anderson, B. D. (2008). Iterative minimization of H_2 control performance criteria. *Automatica*, 44(10):2549–2559.

Bazanella, A. S. and Neuhaus, T. (2014). Tuning nonlinear controllers with the virtual reference approach. *IFAC Proceedings Volumes*, 47(3):10269–10274.

Bianchi, F., Falsone, A., Prandini, M., and Piroddi, L. (2016). A randomised approach for NARX model identification based on a multivariate Bernoulli distribution. *International Journal of Systems Science*, 48(6):1203–1216.

Billings, S. (1980). Identification of nonlinear systems–a survey. *IEE Proceedings D Control Theory and Applications*, 127(6):272.

Billings, S. A., Chen, S., and Korenberg, M. J. (1989). Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator. *International Journal of Control*, 49(6):2157–2189.

Broomhead, D. S. and Lowe, D. (1988). Radial basis functions, multi-variable functional interpolation and adaptive networks. Technical report, Royal Signals and Radar Establishment Malvern (United Kingdom).

Campestrini, L., Eckhard, D., Gevers, M., and Bazanella, A. (2011). Virtual Reference Feedback Tuning for non-minimum phase plants. *Automatica*, 47(8):1778–1784.

Campestrini, L., Eckhard, D., Sanfelice Bazanella, A., and Gevers, M. (2017). Data-driven model reference control design by prediction error identification. *Journal of the Franklin Institute*, 354(6):2628–2647.

Campestrini, L., Ola, Eckhard, D., Bazanella, A. S., and Gevers, M. (2012). *Model Reference Control Design by Prediction Error Identification*, volume 16. IFAC.

Campi, M. C., Lecchini, A., and Savaresi, S. (2002). Virtual reference feedback tuning: A direct method for the design of feedback controllers. *Automatica*, 38(8):1337–1346.

Campi, M. C. and Savaresi, S. M. (2004). Invertibility of nonlinear maps. Int. rep., Uinv. Brescia, Brescia, Italy.

Campi, M. C. and Savaresi, S. M. (2006). Direct nonlinear control design: The virtual reference feedback tuning (VRFT) approach. *IEEE Transactions on Automatic Control*, 51(1):14–27.

Cavanaugh, J. E. (1997). Unifying the derivations for the Akaike and corrected Akaike information criteria. *Statistics & Probability Letters*, 33(2):201–208.

Chen, S. and Billings, S. A. (1992). Neural networks for nonlinear dynamic system modelling and identification. *International Journal of Control*, 56(2):319–346.

Chen, S., Billings, S. A., Cowan, C. F. N., and Grant, P. M. (1990). Practical identification of NARMAX models using radial basis functions. *International Journal of Control*, 52(6):1327–1350.

Da Silva, G. R., Campestrini, L., and Bazanella, A. S. (2018). Multivariable virtual reference feedback tuning for non-minimum phase plants. *IEEE Control Systems Letters*, 2(1):121–126.

Draper, N. R. and Smith, H. (1998). *Applied Regression Analysis*. Wiley Series in Probability and Statistics. Wiley, New York, 3rd ed edition.

Falsone, A., Piroddi, L., and Prandini, M. (2014). A novel randomized approach to nonlinear system identification. In *53rd IEEE Conference on Decision and Control*, pages 6516–6521.

Falsone, A., Piroddi, L., and Prandini, M. (2015). A randomized algorithm for nonlinear model structure selection. *Automatica*, 60:227–238.

Guardabassi, G. O. and Savaresi, S. M. (2000). Virtual reference direct design method: An off-line approach to data-based control system design. *IEEE Transactions on Automatic Control*, 45(5):954–959.

Haykin, S. S. (1994). *Neural Networks: A Comprehensive Foundation*. Macmillan ; Maxwell Macmillan Canada ; Maxwell Macmillan International, New York : Toronto : New York.

Hjalmarsson, H., Gunnarsson, S., and Gevers, M. (1994). A convergent iterative restricted complexity control design scheme. *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, 2(December):1735–1740.

Hou, Z. S. and Wang, Z. (2013). From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, 235:3–35.

Huang, B. and Kadali, R. (2008). Dynamic modeling, predictive control and performance monitoring. In *Lecture Notes in Control and Information Sciences*, page 249.

Huff, D. D., Campestrini, L., Gonçalves da Silva, G. R., and Bazanella, A. S. (2019). Data-Driven Control Design by Prediction Error Identification for Multivariable Systems. *Journal of Control, Automation and Electrical Systems*, 30(4):465–478.

Huusom, J. K., Poulsen, N. K., and Jørgensen, S. B. (2009). Improving convergence of iterative feedback tuning. *Journal of Process Control*, 19(4):570–578.

Ikeda, M., Fujisaki, Y., and Hayashi, N. (2001). A model-less algorithm for tracking control based on input-output data. In *Nonlinear Analysis, Theory, Methods and Applications*, volume 47, pages 1953–1960.

Jeng, J.-C. and Lin, Y.-W. (2014). Controller design for nonlinear hammerstein and wiener systems based on VRFT method. In *Computer Aided Chemical Engineering*, volume 33, pages 547–552.

Jeng, J.-C. and Lin, Y.-W. (2018). Data-driven nonlinear control design using virtual-reference feedback tuning based on the block-oriented modeling of nonlinear systems. *Industrial & Engineering Chemistry Research*, 57(22):7583–7599.

Jeng, J.-C., Lin, Y.-W., and Lee, M.-W. (2015). Extended VRFT method for controller design of nonlinear systems based on block-oriented model structures. In R, G. K., JK, H., and Gani, editors, *Computer Aided Chemical Engineering*, volume 37 of *Computer Aided Process Engineering*, pages 1691–1696.

Kammer, L., Bitmead, R., and Bartlett, P. (2000). Direct iterative tuning via spectral analysis. In *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No.98CH36171)*, volume 3, pages 2874–2879. IEEE.

Karimi, A., Mišković, L., and Bonvin, D. (2002). Convergence analysis of an iterative correlation-based controller tuning method. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 35(1):413–418.

Karimi, A., van Heusden, K., and Bonvin, D. (2007). Non-iterative data-driven controller tuning using the correlation approach. In *2007 European Control Conference (ECC)*, volume 3536, pages 5189–5195. IEEE.

Kashyap, R. (1977). A Bayesian comparison of different classes of dynamic models using empirical data. *IEEE Transactions on Automatic Control*, 22(5):715–727.

Korenberg, M., Billings, S. A., Liu, Y. P., and McILROY, P. J. (1988). Orthogonal parameter estimation algorithm for non-linear stochastic systems. *International Journal of Control*, 48(1):193–210.

Leontaritis, I. J. and Billings, S. A. (1985a). Input-output parametric models for non-linear systems Part I: Deterministic non-linear systems. *International Journal of Control*, 41(2):303–328.

Leontaritis, I. J. and Billings, S. A. (1985b). Input-output parametric models for non-linear systems Part II: Stochastic non-linear systems. *International Journal of Control*, 41(2):329–344.

Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall, New Jersey, US.

Martins, S. A. M., Nepomuceno, E. G., and Barroso, M. F. S. (2013). Improved Structure Detection For Polynomial NARX Models Using a Multiobjective Error Reduction Ratio. *Journal of Control, Automation and Electrical Systems*, 24(6):764–772.

Pearson, R. K. (1999). *Discrete-Time Dynamic Models*. Topics in Chemical Engineering. Oxford University Press, New York.

Piroddi, L. and Spinelli, W. (2003). An identification algorithm for polynomial NARX models based on simulation error minimization. *International Journal of Control*, 76(17):1767–1781.

Previdi, F., Schauer, T., Savaresi, S. M., and Hunt, K. J. (2004). Data-driven control design for neuroprotheses: A virtual reference feedback tuning (VRFT) approach. *IEEE Transactions on Control Systems Technology*, 12(1):176–182.

Radac, M. B., Precup, R. E., and Roman, R. C. (2018). Data-driven model reference control of MIMO vertical tank systems with model-free VRFT and Q-Learning. *ISA Transactions*, 73:22–30.

Retes, P. F. L. and Aguirre, L. A. (2019). NARMAX model identification using a randomised approach. *International Journal of Modelling, Identification and Control*, 31(3):205.

Safonov, M. G. and Tsao, T.-C. (1995). The unfalsified control concept: A direct path from experiment to controller. *Feedback Control, Nonlinear Systems, and Complexity*, pages 196–214.

Schaal, S. and Atkeson, C. G. (1994). Robot juggling: Implementation of memory-based learning. *IEEE Control Systems*, 14(1):57–71.

Schwarz, G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, 6(2):461–464.

Shi, G. and Skelton, R. E. (2000). Markov data-based LQG control. *Journal of Dynamic Systems, Measurement, and Control*, 122(3):551.

Skogestad, S. (2003). Simple analytic rules for model reduction and PID controller tuning. *Journal of Process Control*, page 19.

Spall, J. (1992). Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341.

Strang, G. (1989). Wavelets and Dilation Equations: A Brief Introduction. *SIAM Review*, 31(4):614–627.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning Series. The MIT Press, Cambridge, Massachusetts, second edition edition.

Takahashi, Y., Rabins, M. J., and Auslander, D. M. (1972). *Control and Dynamic Systems*. Addison-Wesley, California.

Tolle, M. (1921). *Regelung Der Kraftmaschienen*. Springer, Berlin.

Wang, J. (2011). PID controller design of based on neural network and virtual reference feedback tuning. In *2011 CHINESE CONTROL AND DECISION CONFERENCE, VOLS 1-6*, Chinese Control and Decision Conference, pages 3078–3083.

Watkins, C. J. C. H. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3-4):279–292.

Wei, H. L. and Billings, S. A. (2008). Model structure selection using an integrated forward orthogonal search algorithm assisted by squared correlation and mutual information. *International Journal of Modelling, Identification and Control*, 3(4):341.

Yan, P., Liu, D., Wang, D., and Ma, H. (2016). Data-driven controller design for general MIMO nonlinear systems via virtual reference feedback tuning and neural networks. *Neurocomputing*, 171:815–825.

Young, P. (1970). An instrumental variable method for real-time identification of a noisy process. *Automatica*, 6(2):271–287.

Zhu, Q. (2005). An implicit least squares algorithm for nonlinear rational model parameter estimation. *Applied Mathematical Modelling*, 29(7):673–689.

Ziegler, J. G. and Nichols, N. B. (1942). Optimum settings for automatic controllers. *Transactions of ASME*, 64:759–768.

Appendix A

VRFT Complements

A.1 Proof. for Theorem 3.2.1 for linear case

Para o caso linear, uma forma de demostrar tal resultado é rescrevendo (3.5) como

$$\begin{aligned} J_{VR}(\boldsymbol{\theta}) &= \left\| (\boldsymbol{\theta}_0 - \boldsymbol{\theta})^T \boldsymbol{\varphi}(k) \right\|^2 \\ &= \left\| (\boldsymbol{\theta}_0 - \boldsymbol{\theta})^T C_\theta(q) \frac{1 - M(q)}{M(q)} P(q)\tilde{u}(k) \right\|^2 \\ &= \left\| (\boldsymbol{\theta}_0 - \boldsymbol{\theta})^T C_\theta(q) \frac{1}{C_0(q)} P(q)\tilde{u}(k) \right\|^2 \end{aligned} \tag{A.1}$$

onde foi usado o fato que $\tilde{u}(k) = C_0\tilde{e}(k) = \boldsymbol{\varphi}^T(k)\boldsymbol{\theta}_0 = \boldsymbol{\theta}_0^T\boldsymbol{\varphi}$ e que

$$\varphi(k) = C_\theta(q)\tilde{e}(k) = C_\theta(q)\frac{1 - M(q)}{M(q)}\tilde{y}(k),$$

sendo $\tilde{y} = P(q)\tilde{u}(k)$.

Aplicando o teorema de Parserval Ljung (1999), em (A.1) e rearranjando os termos, pode-se rescrevê-la como

$$J_{VR}(\boldsymbol{\theta}) = (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^T A_{VR} (\boldsymbol{\theta} - \boldsymbol{\theta}_0),$$

onde

$$A_{VR} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{|C_0(e^{j\omega})|^2} C_\theta(e^{j\omega}) C_\theta(e^{j\omega}) C_\theta^*(e^{j\omega}) \Phi_u(e^{j\omega})\, dw$$

Desde que $C_0(z)$ não tenha zeros no círculo unitário, nota-se que $J_{VR}(\boldsymbol{\theta}_0) = 0$ se $\boldsymbol{\theta} = \boldsymbol{\theta}_0$, e será a única solução que resulta em um mínimo desde que $A_{VR} \neq 0$. Considerando o espectro do sinal de entrada $\tilde{u}(k)$ como discreto, i.e.

$$\Phi_u(e^{j\omega}) = \sum_{k=1}^{q} \lambda_k \delta(\omega - \omega_k)$$

sendo λ_k um valor real positivo e $\delta(\cdot)$ a função delta de Dirac, de forma que

$$A_{VR} = \frac{1}{2\pi}\sum_{k=1}^{q}\frac{1}{|C_0(e^{j\omega})|^2}C_\theta(e^{j\omega})C_\theta(e^{j\omega})C_\theta^*(e^{j\omega})\lambda_k$$

Como $C_\theta(e^{j\omega})$ e sua conjugada transposta são matrizes definidas positivas, A_{VR} será uma soma ponderada de matrizes definidas positivas. O posto desta soma é dada por $\min(q,p)$ onde p é a ordem de A_{VR}. Portanto, caso o espectro do sinal de entrada $\Phi_u(e^{j\omega}) \geq p$, i.e., seja suficientemente rico de ordem $q \geq p$ (definition 3.6), ou o vetor de regressores seja persistentemente excitante (definition 3.6), A_{VR} é definida positiva e $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ é a única solução ótima (Bazanella et al., 2012).

A.2 Proof of the VRFT filter choice (Theorem 3.4.1)

Note que

$$\tilde{u} = C_{\theta_0^+}[\tilde{e}] \tag{A.2}$$

uma vez que

$$\begin{aligned}\tilde{u} &= P^{-1}[\tilde{y}] = P^{-1}\left[(I-MD)^{-1}(I-MD)\tilde{y}\right]\\ &= P^{-1}\left[(I-MD)^{-1}(M\tilde{r}-MD\tilde{y})\right] = P^{-1}\left[(I-MD)^{-1}M\tilde{e}\right] = C^0[\tilde{e}]\\ &= C_{\theta_0^+}[\tilde{e}].\end{aligned}$$

De forma semelhante

$$\tilde{y} = y_{\theta_0^+} \tag{A.3}$$

uma vez que, $\tilde{y} = P[\tilde{u}]$ a partir de (A.2), assumindo $\tilde{e} = \tilde{r} - D\tilde{y}$, tem-se que $\tilde{y} = P\left[C_{\theta_0^+}[\tilde{e}]\right] = P\left[C_{\theta_0^+}[\tilde{r}-D\tilde{y}]\right]$. Como $y_{\theta_0^+} = P\left[C_{\theta_0^+}\left[\tilde{r}-Dy_{\theta_0^+}\right]\right]$, $\tilde{y}$ e $y_{\theta_0^+}$ corresponde ao mesmo $\tilde{r}$ no mapa $r \mapsto y$ dado por $y = P\left[C_{\theta_0^+}[r-Dy]\right]$. Uma vez que tal mapa, dado um r há somente um y correspondente, conclui-se (A.3). De (A.3) é possível concluir que

$$\tilde{r} - Dy_{\theta_0^+} = \tilde{e}, \tag{A.4}$$

que, em (A.2), resulta em

$$\tilde{u} = C_{\theta_0^+}\left[\tilde{r} - Dy_{\theta_0^+}\right]. \tag{A.5}$$

Por simplicidade e maior clareza no desenvolvimento, adota-se a seguinte notação:

$$x_{\theta^+} \triangleq F\left[C_{\theta^+}[\tilde{e}]\right] - F[\tilde{u}] \tag{A.6}$$

$$w_{\theta^+} \triangleq y_{\theta^+} - \tilde{y} \tag{A.7}$$

$$\frac{\partial g}{\partial \theta^+} \triangleq \begin{bmatrix} \partial g_1/\partial\theta_1^+ & \dots & \partial g_1/\partial\theta_j^+ & \dots & \partial g_1/\partial\theta_{n_{\theta+}}^+ \\ \vdots & & \vdots & & \vdots \\ \partial g_i/\partial\theta_1^+ & \dots & \partial g_i/\partial\theta_j^+ & \dots & \partial g_i/\partial\theta_{n_{\theta+}}^+ \\ \vdots & & \vdots & & \vdots \\ \partial g_N/\partial\theta_1^+ & \dots & \partial g_N/\partial\theta_j^+ & \dots & \partial g_N/\partial\theta_{n_{\theta+}}^+ \end{bmatrix} \tag{A.8}$$

Sendo (A.8) (em que g é uma função genérica) definida tal que o (i, j)-ésimo elemento é $\partial g(i)/\partial\theta_j^+$, de modo que as colunas (j) correspondem às derivadas de g em relação a diferentes parâmetros e as linhas (i) correspondem à evolução temporal.

Usando (A.6) e (A.7), as função de custo (**??**) e (3.1) são rescritas como

$$J_{\text{VR}}(\theta^+) = \|x_{\theta^+}\|^2 \qquad J(\theta^+) = \|w_{\theta^+}\|^2$$

Calculando a primeira e segunda derivadas de $J_{\text{VR}}(\theta^+)$ com respeito ao vetor de parâmetros θ^+, para aproximação por séries de Taylor:

$$\frac{\partial J_{\text{VR}}(\theta^+)}{\partial\theta^+} = \frac{\partial x_{\theta^+}^T x_{\theta^+}}{\partial\theta^+} = 2x_{\theta^+}^T\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right),$$

$$\frac{\partial^2 J_{\text{VR}}(\theta^+)}{\partial\theta^{+2}} = 2x_{\theta^+}^T\left(\frac{\partial^2 x_{\theta^+}}{\partial\theta^{+2}}\right) + 2\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right)^T\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\right).$$

Utilizando (A.6) e (A.2) e calculando as derivadas para o ponto de equilíbrio θ_0^+, resulta em

$$J_{\text{VR}}(\theta^+)\Big|_{\theta_0^+} = \frac{\partial J_{\text{VR}}(\theta^+)}{\partial\theta^+}\Big|_{\theta_0^+} = 0 \tag{A.9}$$

$$\frac{\partial^2 J_{\text{VRFT}}(\theta^+)}{\partial\theta^{+2}}\Big|_{\theta_0^+} = 2\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\Big|_{\theta_0^+}\right)^T\left(\frac{\partial x_{\theta^+}}{\partial\theta^+}\Big|_{\theta_0^+}\right) \tag{A.10}$$

Fazendo o mesmo procedimento para $J(\theta^+) = \|w_{\theta^+}\|^2$:

$$\frac{\partial J(\theta^+)}{\partial\theta^+} = \frac{\partial w_{\theta^+}^T w_{\theta^+}}{\partial\theta^+} = 2w_{\theta^+}^T\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right),$$

$$\frac{\partial^2 J(\theta^+)}{\partial\theta^{+2}} = 2x_{\theta^+}^T\left(\frac{\partial^2 w_{\theta^+}}{\partial\theta^{+2}}\right) + 2\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right)^T\left(\frac{\partial w_{\theta^+}}{\partial\theta^+}\right).$$

Usando (A.7) e (A.3):

$$J(\theta^+)\Big|_{\theta_0^+} = \frac{\partial J(\theta^+)}{\partial \theta^+}\Big|_{\theta_0^+} = 0 \tag{A.11}$$

$$\frac{\partial^2 J(\theta^+)}{\partial \theta^{+2}}\Big|_{\theta_0^+} = 2\left(\frac{\partial w_{\theta^+}}{\partial \theta^+}\Big|_{\theta_0^+}\right)^T\left(\frac{\partial w_{\theta^+}}{\partial \theta^+}\Big|_{\theta_0^+}\right). \tag{A.12}$$

Somando os termos de (A.9) a (A.10), e os (A.11) e (A.12), tem-se, respectivamente uma aproximação de segunda ordem por séries de Taylor. E para que o objetivo do filtro (**??**) seja alcançado, ou seja $J_{\text{VR}}(\theta^+) \approx J(\theta^+)$, comparando (A.10) com (A.12), deve-se ter

$$\frac{\partial x_{\theta^+}}{\partial \theta^+}\Big|_{\theta_0^+} = \frac{\partial w_{\theta^+}}{\partial \theta^+}\Big|_{\theta_0^+} \tag{A.13}$$

Usando a notação

$$\frac{\partial P[u]}{\partial u} \triangleq \begin{bmatrix} \partial P[u]_1/\partial u_0 & \dots & \partial P[u]_1/\partial u_{(j-1)} & \dots & \partial P[u]_1/\partial u_{(N-1)} \\ \vdots & & \vdots & & \vdots \\ \partial P[u]_i/\partial u_0 & \dots & \partial P[u]_i/\partial u_{(j-1)} & \dots & \partial P[u]_i/\partial u_{(N-1)} \\ \vdots & & \vdots & & \vdots \\ \partial P[u]_N/\partial u_0 & \dots & \partial P[u]_N/\partial u_{(j-1)} & \dots & \partial P[u]_N/\partial u_{(N-1)} \end{bmatrix}$$

e resolvendo o lado esquerdo de (A.13), considerando que o filtro F é linear, chega-se a

$$\frac{\partial x_{\theta^+}}{\partial \theta^+}\Big|_{\theta_0^+} = \frac{\partial F[C_{\theta^+}[\tilde{e}]]}{\partial \theta^+}\Big|_{\theta_0^+} = F\left(\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial \theta^+}\Big|_{\theta_0^+}\right) \tag{A.14}$$

Como $C_{\theta_0^+} = \tilde{u}$, ver (A.2), o lado direito de (A.13) é calculado como

$$\begin{aligned} \frac{\partial P[u]}{\partial u}\Big|_{\tilde{u}} \frac{\partial C_{\theta_0^+}[e]}{\partial e}\Big|_{\tilde{e}} &= \frac{\partial P\left[C_{\theta_0^+}[e]\right]}{\partial e}\Big|_{\tilde{e}} \\ &= \frac{\partial (I - MD)^{-1} M[e]}{\partial e}\Big|_{\tilde{e}} \\ &= (I - MD)^{-1} M. \end{aligned} \tag{A.15}$$

Aplicando a regra da cadeia, em $(\partial y_{\theta_0^+}/\partial\theta^+)$:

$$\left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = \left.\frac{\partial P\left[C_{\theta^+}\left[\tilde{r} - Dy_{\theta^+}\right]\right]}{\partial\theta^+}\right|_{\theta_0^+}$$

$$= \left.\frac{\partial P[u]}{\partial u}\right|_{C_{\theta_0^+}\left[\tilde{r}-Dy_{\theta_0^+}\right]} \left\{ \left.\frac{\partial C_{\theta^+}\left[\tilde{r} - Dy_{\theta_0^+}\right]}{\partial\theta^+}\right|_{\theta_0^+} - \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{r}-Dy_{\theta_0^+}} \left.\frac{\partial Dy_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} \right\}$$

Usando (A.4) e (A.5),

$$\left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left\{ \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial\theta^+}\right|_{\theta_0^+} - \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{e}} \left.\frac{\partial Dy_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} \right\}$$

e então

$$\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial\theta^+}\right|_{\theta_0^+} = \left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} + \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta_0^+}[e]}{\partial e}\right|_{\tilde{e}} \left.\frac{\partial Dy_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+}.$$

Substituindo em (A.15) resulta em

$$\left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} + (I - MD)^{-1}M \left.\frac{\partial Dy_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial\theta^+}\right|_{\theta_0^+},$$

que, multiplicando por $(I - MD)$, nos dá

$$\left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} - MD \left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} + M \left.\frac{\partial Dy_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = (I - MD) \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial\theta^+}\right|_{\theta_0^+}$$

Por fim, o termo do lado direito de (A.13), pode ser escrito como

$$\left.\frac{\partial w_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = \left.\frac{\partial y_{\theta^+}}{\partial\theta^+}\right|_{\theta_0^+} = (I - MD) \left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}} \left.\frac{\partial C_{\theta^+}[\tilde{e}]}{\partial\theta^+}\right|_{\theta_0^+}$$

que, comparando com (A.14), conclui-se que

$$F = (I - MD)\left(\left.\frac{\partial P[u]}{\partial u}\right|_{\tilde{u}}\right).$$