

University of Alberta

GENERALIZED PREDICTIVE CONTROL FOR MULTIRATE SYSTEMS

by

Jie Sheng



A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Department of Electrical and Computer Engineering

Edmonton, Alberta
Fall 2002



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-81264-2

Canada

University of Alberta

Library Release Form

Name of Author: Jie Sheng

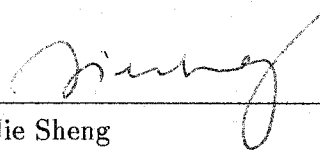
Title of Thesis: Generalized Predictive Control for Multirate Systems

Degree: Doctor of Philosophy

Year this Degree Granted: 2002

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.



Jie Sheng
Department of Electrical and
Computer Engineering
Edmonton, Alberta
Canada, T6G 2V4

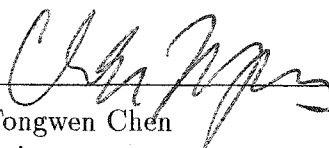
Date: July 18, 2002

The important thing is not to stop questioning.
- Albert Einstein


University of Alberta

Faculty of Graduate Studies and Research

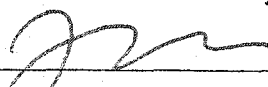
The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Generalized Predictive Control for Multirate Systems** submitted by Jie Sheng in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.



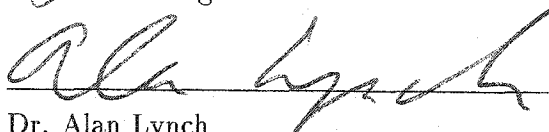
Dr. Tongwen Chen
Supervisor



Dr. Sirish L. Shah
Co-Supervisor



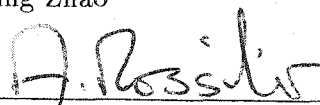
Dr. Biao Huang



Dr. Alan Lynch



Dr. Qing Zhao



Dr. J. Anthony Rossiter
External Examiner

Date: July 5, 2002

To my parents

Abstract

Model predictive control (MPC) is a paradigm that emerged in the late 1970's, and has developed considerably over the last two decades. The two versions which appear to have had the most acceptability are one derived by Clarke *et al.* [24] and is called GPC for generalized predictive control, and secondly that derived by Cutler and Ramaker [30] called DMC for dynamic matrix control.

Multirate systems are common in industry, especially in chemical process control. The synthesis and analysis of GPC for single-rate systems has drawn considerable attention since the 1980's. On the other hand multirate GPC, practically useful as it may be, has not received much attention. Most of the research in the multirate control area deals with the time-varying nature of multirate systems ([54, 99]) and hence makes the design and analysis more complex. The use of the lifting technique as introduced by Kranc [50] allows one to convert multirate systems into single-rate and linear time-invariant (LTI) equivalent systems. However, its use also presents a challenge in the synthesis problem: the lifted controllers should satisfy certain causality constraint. The main purposes of this thesis are:

1. To develop GPC controllers for multirate systems, taking into account the causality design condition.
2. To develop a sampled-data GPC scheme for multirate systems so that the continuous-time closed-loop performance is optimized, and the inter-sample behavior is improved.
3. To perform robust stability analysis of a general class of multirate MPC controllers in the presence of model-plant mismatch (MPM). Note that GPC belongs to the MPC class of controllers which are all model based algorithms.

The thesis focuses on the synthesis and analysis of GPC for multirate systems, as well as the more general and special case of multirate systems such as non-uniformly sampled systems and fast-control, slow-sampling, dual-rate systems. In addition to these, related topics such as the multirate digital redesign problem and the continuous time delay estimation

problem are also investigated in this study.

Acknowledgements

I would like to express my gratitude towards my supervisor, Prof. Tongwen Chen for his excellent supervision and guidance on all aspects of this thesis. I am grateful to him for his constant encouragement, patient advice, valuable directions and weekly research meeting with me. I feel fortunate to have learned so much on many aspects of control from him.

I would like to thank Prof. Sirish L. Shah, my co-supervisor, for his suggestions in my research and giving me the chance to be involved in the GPC project in cooperation with the Mitsubishi Chemical Co. of Japan. His emphasis on the practical aspects and efforts to bridge the gap between industry and academia led me to understand and appreciate the needs and problems of process control.

I would also like to express my gratitude towards my colleagues in the GPC research group: Prof. Biao Huang, for sharing his expertise in performance assessment and monitoring, Prof. Sachin Patwardhan, for sharing his ideas in handling disturbances in GPC problems during his visit in the Department of Chemical Engineering, Dr. Dongguang Li, for sharing his thoughts and experiences in practical problems, and Bhushan Gopaluni, for many interesting discussions and friendly help from him.

I thank the Computer Process Control group in the Department of Chemical Engineering, as an exciting place for one to learn about process control. I also thank all the members belonging to the Advanced Control Systems Lab in the Department of Electrical Engineering, for making my stay in Edmonton enjoyable and unforgettable. In particular, I would like to express my gratitude to Prof. Wen Tan, for his technical help on many matters, during his visit to the University of Alberta.

I gratefully acknowledge the F.S Chia PhD Scholarship and the Dissertation Fellowship Programs at the University of Alberta, and also the NSERC research funds, made available by Prof. Tongwen Chen, and the Department of Electrical Engineering for their financial support.

My parents have provided me with constant support and have encouraged me to work towards my own goals. I am unable to find enough words to express my thanks and gratitude

to them, hence I dedicated this thesis to my beloved parents.

Contents

1	Introduction	1
1.1	A historic perspective of MPC	1
1.2	Motivation	2
1.3	Scope of the thesis	4
1.4	Research objectives	5
1.5	About the thesis	7
1.5.1	Outline of the thesis	7
1.5.2	Notation	8
2	Review of Single-Rate GPC	10
2.1	Prediction model	11
2.2	Objective function	13
2.3	Obtaining the control law (constraint-free)	15
2.4	Some remarks	20
3	GPC for Non-Uniformly Sampled Systems	22
3.1	Introduction	22
3.2	Modeling of non-uniformly sampled systems	24
3.2.1	The sampling and updating scheme	24
3.2.2	Lifting, lifted models and the causality constraint issues	25
3.2.3	Controllability and observability	29
3.3	GPC algorithm for non-uniformly sampled systems	31
3.3.1	Conventional GPC design	32
3.3.2	Proposed GPC design	34
3.4	Example	38
3.5	Conclusions	41
4	Multirate GPC for Sampled-Data Systems	43
4.1	Introduction	43
4.2	Problem formulation	44
4.3	Dual-rate GPC design for sampled-data systems	47
4.3.1	Discretization at the fast rate	47
4.3.2	Lifting for conversion to the slow rate	49
4.3.3	Lifting the cost function	50
4.3.4	Optimal control law	51
4.4	A design example	52
4.4.1	Discrete-time dual-rate GPC scheme	52

4.4.2	Sampled-data single-rate GPC schemes	53
4.4.3	Simulation results	54
4.5	Conclusions	57
5	On the Robust Stability of Dual-Rate GPC Systems	59
5.1	Introduction	59
5.2	Dual-rate sampled-data systems with noise	60
5.3	Dual-rate GPC problem: a state-space solution	62
5.4	Stability robustness	67
5.4.1	CARIMA models	68
5.4.2	Dual-rate GPC solution in the polynomial domain	69
5.4.3	Stability robustness analysis	71
5.5	Illustrative examples	73
5.6	Conclusions	81
6	Issues on Multirate Systems	82
6.1	Introduction	82
6.2	Multirate discretization of analog controllers	85
6.2.1	Error system	85
6.2.2	Example	88
6.3	Time-delay estimation of multirate systems based on interactor matrices	94
6.3.1	Properties of time-delay matrices of lifted models	94
6.3.2	Interactor matrices	98
6.3.3	Estimation of continuous time delays	99
6.3.4	Examples	105
6.4	Conclusions	109
7	Conclusions and Future Work	111
7.1	Conclusions	111
7.2	Future work	113
	Bibliography	116

List of Figures

1.1	A sampled-data system	4
1.2	Different sampled-data systems	5
3.1	The non-uniform sampling and updating scheme	25
3.2	An illustration of the design	39
3.3	Tracking performance of the closed-loop with the multirate GPC controller	41
4.1	Dual-rate sampled-data open-loop system	44
4.2	Dual-rate discrete-time open-loop system	45
4.3	Lifted LTI and single-rate model	46
4.4	Tracking performance and control efforts with discrete-time dual-rate GPC	55
4.5	Comparison of the system tracking responses with the three SD schemes	56
4.6	Comparison of the control efforts with the three SD schemes	56
5.1	Dual-rate sampled-data open-loop system with noise	61
5.2	Dual-rate discretized system with noise	61
5.3	Lifted LTI closed-loop system	62
5.4	Dual-rate GPC closed-loop system	71
5.5	Dual-rate GPC closed-loop system with model uncertainty	72
5.6	Simplified system for robustness analysis	72
5.7	Schematic diagram of a stirred tank heater	73
5.8	Tracking performance over the whole time horizon	75
5.9	Tracking performance of the water level during time period $[0, 250]$ sec	76
5.10	Tracking performance of the temperature during time period $[0, 250]$ sec	76
5.11	Performance index comparison for three cases over time horizon $[0, 250]$ sec	77
5.12	Bode plots of W_2 and the true uncertainty	78
5.13	GPC tracking response with model \hat{P}_h	79
5.14	GPC tracking response with model P_h	80
5.15	GPC tracking response with model $(1 + W_2)\hat{P}_h$	80
6.1	A SISO multirate sampled-data system	82
6.2	Multirate digital K_d	85
6.3	Multirate sampled-data implementation of K	86
6.4	The error system	86
6.5	The equivalence of Figure 6.4	86
6.6	Bode plot of the elliptic filter	89
6.7	Error for single-rate discretization of Elliptic filter	90
6.8	Error for dual-rate discretization with different n	90
6.9	Error for bilinear approximation with different n	91

6.10 Error for step-invariant approximation with different n	91
6.11 Error signal $\hat{e}(j\omega)$ for different n	93
6.12 Error signal $\hat{e}(j\omega)$ for different m	93
6.13 Discrete-time equivalent of Figure 6.1	104
6.14 Lifted LTI system	104

List of Tables

4.1	Comparison of cost functions for the three SD designs	57
6.1	E_{max} for two approximations with different n	92

Chapter 1

Introduction

1.1 A historic perspective of MPC

Model predictive control, or MPC in short, emerged in the late seventies, when its early versions appeared in industry. For example, Richalet *et al.* suggested Model Predictive Heuristic Control (MPHC) (later known as Model Algorithmic Control (MAC)) in [87, 88]; and Culter and Ramaker proposed Dynamic Matrix Control (DMC) in [30]. Other well known variations of MPC were developed independently in the adaptive control area, such as Predictor-Based Self-Tuning Control by Peterka [77], Extended Horizon Adaptive Control (EHAC) by Ydstie [115], Extended Prediction Self Adaptive Control (EPSAC) by De Keyser *et al.* [31], and Generalized Predictive Control (GPC) by Clarke *et al.* [24, 25]. A comprehensive survey of all these predictive methods developed during the eighties can be found in the work by Garcia *et al.* [35].

Traditionally, MPC is derived in the transfer function framework. There have also been efforts to formulate MPC in the state-space framework because of the following advantages: (i) it permits the use of well-known state-space techniques, (ii) it makes possible the extension to more complex systems such as MIMO systems and those with disturbances and noise, although the state estimation arises as a new problem. Articles dealing with the state-space based MPC include: Navratil *et al.* [72] (1988), Li *et al.* [60] (1989), Ricker [90] (1990), Lee *et al.* [54] (1992), Ordys and Clarke [75] (1993), Lee *et al.* [55] (1994), Scattolini and Schiavoni [99] (1995), and Ling and Lim [61] (1996).

Multirate MPC is now one of the most popular and effective techniques in the process control area due to its capability to handle practical issues, such as future reference signals, noise and disturbances, constraints on manipulated variables which are very common in industry. All MPC strategies only differ in the following factors:

- The prediction model;

- The objective function and techniques for handling constraints;
- Methods to obtain the control law.

When these elements take various alternatives, different control algorithms arise; and the version which appears to have had the most acceptability in the academic domain is GPC as derived by Clarke *et al.* [24, 25] in 1987. Hence the MPC technique in the form of GPC will be the main subject of study in this thesis.

The main philosophy of MPC is to compute the future control sequence based on an explicit model of the process by minimizing certain performance index over a finite horizon. In the implementation phase, only the first element of the calculated sequence is applied to the real process at each step. For the next step, the whole procedure is repeated. This receding horizon strategy is difficult to theoretically analyze for properties such as stability and robustness. The reason is that the majority of stability results are limited to the infinite horizon case and there is a lack of a theory relating the receding horizon strategy to the closed-loop behavior and design parameters. However, promising results on these fundamental issues have been proposed beginning with the early nineties. Among them, methods such as CRHPC (Constrained Receding Horizon Predictive Control) by Clarke and Scattolini [28], SIORHC (Stabilizing I/O Receding Horizon Control) by Mosca *et al.* [70], and *stable* GPC by Kouvaritakis *et al.* [49] have been proposed to guarantee closed-loop stability. The results by Campo and Morari [15], Rawlings and Muske [86], Rossiter and Kouvaritakis [93], Allwright [3], and Zheng and Morari [117] have been obtained to tackle the problem of stability of constrained receding horizon controllers. Recently, using linear matrix inequality (LMI) (Boyd *et al.* [13]), robust constrained model predictive control was developed by Kothare *et al.* [48]; and stability and robustness analysis of finite receding horizon control was carried out by Primbs and Nevistić [78, 79].

1.2 Motivation

Multirate systems are abundant in industry, especially in the chemical process industry, mostly due to sensor and actuator speed and sampled time constraints; for example, in a distillation column, the composition variable is measured at a slower rate than the rate at which the manipulated or other variables can be adjusted or measured. This is typically some flow or temperature signal. So it is of great importance to study MPC for multirate sampled-data systems where the sampled outputs and the control inputs have different sampling periods. This thesis is focused on a detailed study of multirate GPC problems.

The main difficulty in studying multirate systems is the fact that they are in general *time-varying*, and hence complicating the synthesis and analysis problems for such systems, see, for example, the work on multirate MPC by Lee *et al.* [54] and Scattolini *et al.* [99]. To get around this difficulty or any difficulty arising from this, we adopt a powerful tool called the lifting technique. The idea of lifting was first introduced in the switch decomposition method proposed by Kranc [50] in 1957; this was later further developed into the widely used lifting technique for dealing with multirate systems by Khargonekar *et al.* [47]. Simultaneously treating two signals with different sampling rates, the lifting technique converts a multirate system into a single-rate system which has more inputs and outputs, but is time-invariant. Thus the usual linear techniques for time-invariant systems become formally applicable to the lifted systems. Methods similar to lifting have been introduced in early work on multirate MPC/GPC problems. For example, Scattolini presented a stochastic generalized minimum variance (GMV) self-tuner for MIMO multirate systems [98] in 1988; Carini *et al.* developed a multirate adaptive self-tuning controller for MIMO systems by extending the GPC method [16] in 1990, and Ling and Lim [61] presented a state-space GPC in a least squares framework for both state feedback control and state estimation.

Although lifting can result in an equivalent time-invariant model for the original time-varying multirate system, thus making design and analysis much easier, it introduces a hard design constraint owing to causality [19]. This is a new feature in studying multirate systems; and it remains an open or unsolved problem in multirate MPC/GPC design. As such, the main thrust of this research is to propose an optimal solution to tackle the causality problem in the GPC framework for multirate systems.

It is stressed here that techniques similar to lifting have been applied in [98], [16], and [61]; however, the causality constraint has never been explicitly handled there. References [98] and [16] only dealt with systems where the output sampling period is an integer multiple of the control input period; and it will be shown in later chapters that the causality constraint is automatically satisfied for such systems due to its special sampling-updating strategy. Reference [61] gave a solution in a periodic (lifted) form; but it did not consider the causality constraint in the design. Thus, how to find causal solutions to the general multirate systems is still an open problem in MPC/GPC design.

1.3 Scope of the thesis

In this thesis, we mainly study GPC problems for multirate sampled-data (SD) systems. Consider a sampled-data system in Figure 1.1, where P is the continuous-time process; S

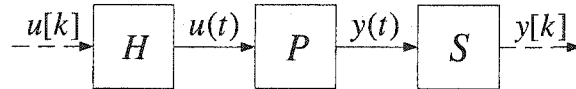


Figure 1.1: A sampled-data system

and H are the sampler and hold devices, representing A/D and D/A conversions, respectively. According to different settings for S and H , we have:

- *single-rate SD systems*, where both S and H have the same periods for sampling and updating.
- *multirate SD systems*, where S and H operate at different periods. More specifically, for a r -input s -output system P , the i -th channel of y , y_i is sampled with period $n_i h$ ($i = 1, \dots, s$) and the j -th channel of u , u_j is through the zero-order hold operation adjusted with period $m_j h$ ($j = 1, \dots, r$). Here n_i and m_j are all integers and h is the so-called *base period*.
- *dual-rate SD systems*, where only two different periods are involved in the sampled-data system shown in Figure 1.1. In our research, we are interested in a fast-control, slow-sampling dual-rate case, where the control updating rates are an integer multiple of the output sampling rates, i.e., $m_1 = m_2 = \dots = m_r = 1$, $n_1 = n_2 = \dots = n_s > 1$.
- *non-uniformly sampled systems*, where S and H adopt more general sampling and updating strategies, say, those instants are non-uniformly spaced within a larger interval $[0, T)$. This T can be a finite number and is known as the *frame period*; it can also be infinity. Later in this thesis, when we mention a non-uniformly sampled system, we refer to the former case, i.e., the whole sampling and updating scheme is repeated every period T .

All such sampled-data systems are related to each other, as shown in Figure 1.2; and they will be handled later in this thesis.

GPC belongs to class of discrete controllers where two common design approaches are used [20]. One is the indirect design by discretizing the original sampled-data systems first and then design in discrete time; another is the design for the sampled-data systems directly. Using the powerful tool of lifting for multirate systems, three types of problems are studied:

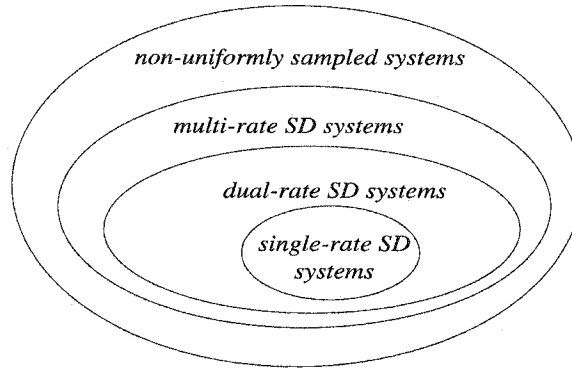


Figure 1.2: Different sampled-data systems

- Design optimal, causal indirect GPC controllers for multirate SD systems.
- Design optimal, direct GPC controllers for multirate SD systems.
- Study stability robustness of closed-loop systems with multirate GPC controllers when there is model-plant mismatch.

The indirect design problem includes: (i) studying modeling issues for the multirate SD systems from continuous time to the lifted discrete time, (ii) taking into account the causality constraint in the design of the lifted controllers and deriving a causal, optimal solution, and (iii) implementation of the designed controller.

The direct design problem can be split into three distinct tasks: (i) analysis of the associated GPC problem by considering the inter-sample behavior, (ii) formulating the multirate GPC problem and deriving an explicit solution, and (iii) application of the designed controller and the comparison of the closed-loop performance under SD controllers with that under pure discrete time controllers.

The robust stability problem involves: (i) setting up the stage for the analysis in the polynomial domain, (ii) using suitable tools to analyze the closed-loop stability robustness of multirate GPC controllers when there is model-plant mismatch, and (iii) concluding on how closed-loop robust stability is related with the multirate strategy.

1.4 Research objectives

More specifically, the research objectives of this thesis are:

Issues with respect to multirate sampled-data systems

1. *The issue of modeling multirate systems from continuous time to the lifted discrete time.*

Since non-uniformly sampled systems adopt the most general multirate strategy, all the results on non-uniformly sampled systems are also guaranteed to work on uniformly sampled, i.e., multirate systems. Using the lifting technique, we will derive a state-space model for the non-uniformly sampled system in discrete time, based on the continuous-time model. Our interest here are the conditions under which the lifted models are controllable and observable, if the original continuous-time systems are controllable and observable.

2. *The issue of time delay identification for a continuous process in the multirate SD setting by the knowledge of the interactor matrix of the lifted system.*

With the tool of lifting, an LTI lifted discrete model can be derived for a multirate sampled-data system based on the knowledge of the continuous process, the input control rate and the output sampling rate. However, this continuous model is usually unknown in practice. On the contrary, interactor matrices [113, 39] of lifted models could be obtained from multirate input-output data by using certain existing algorithms [92, 76]. Assuming the interactor matrix of a lifted model already exists, we will reveal the relationship between this interactor matrix and the unknown continuous time delay.

Development of causal and optimal GPC controllers for multirate sampled-data systems

1. *Use of the indirect design method.*

The causality constraint presents a challenging problem in the GPC design for the lifted systems. Bearing in mind this constraint, we will put forward a solution to the GPC problem for the most general multirate system, i.e., the non-uniformly sampled systems, by grouping output samples appropriately in the polynomial domain. To our best knowledge, this is the first effort to propose an optimal GPC solution using lifted models.

2. *Use of the direct design method.*

MPC is a class of algorithms based on discrete-time models and performance indices. However, discrete-time control algorithms in fact are operating with A/D and D/A converters in a continuous-time environment. Poor inter-sample behavior may arise due to the fact that the design is based on performance solely at the sampling instants.

To improve inter-sample performance, we will design multirate GPC controllers directly based on a continuous-time performance criterion.

Analysis of the lifted controllers for multirate systems

1. *Robust stability of the dual-rate GPC controllers in the presence of model-plant mismatch.*

For a model-based control scheme such as GPC, it is important to analyze the stability robustness of the closed-loop system when there exists model-plant mismatch. In this thesis, we will carry out the analysis in the frequency domain by obtaining a polynomial domain solution to the dual-rate GPC problem. We also intend to find some conditions on the stability robustness of multirate closed-loop systems.

2. *Study of the multirate discretization of analog controllers.*

Digital control design is often accomplished by discretizing existing analog controllers in the single-rate setting. For multirate systems, this discretization becomes a potentially important method. In this work, we will study the discretization error between the real analog controller and its approximation in the frequency domain.

1.5 About the thesis

1.5.1 Outline of the thesis

This thesis mainly extends the MPC/GPC design methods from the single-rate setting to multirate settings by using the lifting technique, and handles the so-called causality constraint in the extension. It also incorporates some results on non-uniformly/multirate sampled systems, such as the modeling from the continuous time to the lifted discrete time, the identification of the continuous time delay via interactor matrices of lifted models, and the multirate discretization of analog controllers.

The thesis is organized as follows. As an introduction to the widely-used GPC algorithm, Chapter 2 gives a general review for GPC in a single-rate setting; and describes its three main elements in details: (i) the prediction model (ii) the objective function and (iii) derivation of the control law.

Chapter 3 presents an optimal and causal GPC solution to the non-uniformly sampled system and contains two parts. In the first part, the non-uniformly sampling and updating strategy is introduced, and a lifted discrete model for the non-uniformly sampled system is derived from the continuous-time model. The causality structure for the feedthrough terms

in the lifted controllers is studied, and a sufficient condition is proposed for the lifted models to preserve observability and controllability. In the second part, a new GPC algorithm for non-uniformly sampled systems is obtained, which results in causal optimal controllers and works effectively in a simulation example.

Chapter 4 deals with the direct sampled-data GPC design for a special case of multirate systems, the dual-rate systems. The GPC problem is formulated as a sampled-data problem with the performance index expressed in terms of continuous-time signals to capture the inter-sample behavior. Design procedures are summarized and an explicit solution is given. Its advantage over the indirect GPC design is emphasized by an example.

As a model-based predictive control scheme, the natural question that arises is: how do dual-rate GPC controllers affect robust stability of closed-loop systems in the presence of model-plant mismatch? Such analysis will be carried out in Chapter 5. Using certain tools, it is concluded that if the input control rate is fixed, the robustness of the dual-rate GPC algorithm can be improved as the sampling ratio decreases.

In Chapter 6, two interesting issues on multirate systems are studied. One is the multirate discretization of analog controllers. Discussion related to this issue and the results of interest are considered in the frequency domain. The other is the continuous time delay identification problem, and some results show that there exists relationship between continuous time delays and interactor matrices of lifted models.

The significant contributions of this thesis are outlined in Chapter 7 and future research directions are indicated.

Since this thesis is written in a paper format, some degree of overlap could not be avoided. For example, the original GPC proposed by Clarke *et al.* [24] in 1987 is mentioned more than once.

1.5.2 Notation

In this thesis, sampled-data systems are studied and both continuous and discrete time signals are involved. Referring to Figure 1.1, the following notation is used throughout the thesis: continuous-time signals evolve over time t (real valued), closed by the round brackets, e.g., $u(t)$; discrete-time signals evolve over time k (integer valued), closed by the square brackets, e.g., $u[k]$. This convention applies to signals in a sampled-data system, e.g., $u[k]$ is $u(t)$ sampled at the k -th sampling instant.

For figures representing multirate sampled-data systems, we use continuous arrows for the continuous-time signals and dotted arrows for discrete-time signals; when two sam-

pling rates exist in discrete time, we use high-frequency dots for fast-rate signals and low-frequency dots for slow-rate signals.

When the lifting technique is applied to multirate systems, all the lifted models and the lifted signals will be underlined, if no confusion arises.

Models in this thesis may be represented in both transfer function and state-space framework. If no confusion arises, a polynomial in the transfer function will be denoted as, for example, $a(q^{-1})$, with q^{-1} as the backward shift operator. The forward shift, backward shift and the difference operators are defined as follows:

$$qy[k] = y[k + 1], \quad (1.1)$$

$$q^{-1}y[k] = y[k - 1], \quad (1.2)$$

$$\Delta y[k] = (1 - q^{-1})y[k] = y[k] - y[k - 1]. \quad (1.3)$$

Chapter 2

Review of Single-Rate GPC

Generalized predictive control (GPC) was proposed by Clarke, Mohtadi and Tuffs [24, 25] in 1987. It uses ideas from Generalized Minimum Variance (GMV) [22] and is with “general” purpose for the stable control of different types of practical systems, especially those systems with problems such as (i) non-minimum-phase (ii) unstable (iii) unknown or variable dead-time and (iv) unknown dynamic order. Due to its advantages, GPC has been successfully implemented in many industry applications during the past decades. For these achievements, one can refer to papers such as [9] by Beaumont *et al.*, [26] by Clarke, and [74] by Camacho *et al.*, just to name a few.

The GPC research literature is by now large. For example, to develop GPC in the state-space domain, efforts have been made by Albertos and Ortega [1], Zhu *et al.* [118], Morari and Lee [69], and Ordys and Clarke [75]. To overcome the drawbacks of the discrete-time formulation, continuous-time GPC was presented by Demircioglu and Gawthrop [32], and sampled-data GPC was developed by Masuda *et al.* [62]. To obtain better robust properties in the case of time-delay systems, a different predictor in GPC was used by Normey *et al.* [73, 74]. To guarantee closed-loop stability, a series of work on constrained receding-horizon predictive control (CHGPC) and stable generalized predictive control (SGPC) have been reported by Clarke and Scattolini [28], Kouvaritakis *et al.* [49], and Rossiter and Kouvaritakis [93]. Systematic studies on GPC problems also exist. Bitmead *et al.* [11] analyzed the inherent characteristics of all the MPC algorithms (especially the GPC) from the point of view of Linear Quadratic Regulator (LQR) and Linear Quadratic Gaussian (LQG) control theory. Camacho and Bordons [14] presented the most recent contributions on implementation issues of MPC (mainly on GPC) for the industry community.

As a member in the class of MPC controllers, GPC has many ideas in common with the other MPC controllers; for example, it also possesses three elements: the prediction

model, the objective performance cost function and the way to obtain the control law. In the remaining of this chapter, these elements will be discussed in detail, showing the general procedure to obtain the control law, and its characteristics.

2.1 Prediction model

A prediction model is the corner stone of MPC/GPC controllers. Based on the model, prediction of the future output signals is forecasted. Thus the prediction model should be able to fully capture the process dynamics.

GPC has been developed in both the polynomial and the state-space domains, thus models can take different forms for different purposes. Here we introduce four models which we will use to derive our multirate GPC algorithms in the next several chapters.

1. CARIMA model

The model used in the GPC method of Clarke *et al.* [24, 25] is given in the following CARIMA (Controlled Auto-Regressive and Integrated Moving Average) form:

$$a(q^{-1})y[k] = b(q^{-1})q^{-d}u[k-1] + c(q^{-1})\frac{e[k]}{\Delta}, \quad (2.1)$$

where u and y are the input and output signals, respectively; e is a zero mean white noise; d is the dead-time of the system; a , b and c are polynomials with a and c monic; $\Delta = 1 - q^{-1}$ is the difference operator.

Note here:

- By including an integrator, an offset-free steady state control could be achieved for systems represented by model (2.1).
- The disturbance term in model (2.1) is given by a white noise e colored by c/Δ . Normally the polynomial c is considered to be 1.
- Later in this thesis, we will include the dead-time term q^{-d} into the polynomial $b(q^{-1})$ for the reason of simplicity.
- Usually c is used to model the disturbance signal and hence improve disturbance rejection; or, in some cases, to enhance robustness of the closed loop [27, 12, 8]. However, the design of c is not systematic in general and only a few basic guidelines have been given [63, 68, 91, 116].

A CARIMA model for an n -output, m -input multivariable process can be expressed as:

$$\mathbf{A}(q^{-1})\mathbf{y}[k] = \mathbf{B}(q^{-1})\mathbf{u}[k - 1] + \mathbf{C}(q^{-1})\frac{e[k]}{\Delta}, \quad (2.2)$$

where $\mathbf{A}(q^{-1})$ and $\mathbf{C}(q^{-1})$ are $n \times n$ monic polynomial matrices and $\mathbf{B}(q^{-1})$ is an $n \times m$ polynomial matrix. If a transfer matrix of a multivariable process, say, $\mathbf{G}(q^{-1})$ is given, a matrix fraction description (MFD) of the multivariable process, i.e., $\mathbf{A}(q^{-1})$ and $\mathbf{B}(q^{-1})$ can be obtained by certain methods. The most simple way of accomplishing this task is by making $\mathbf{A}(q^{-1})$ a diagonal matrix with its diagonal elements equal to the least common multipliers of the denominators of the corresponding row of $\mathbf{G}(q^{-1})$. Matrix $\mathbf{B}(q^{-1})$ is then equal to $\mathbf{B}(q^{-1}) = \mathbf{A}(q^{-1})\mathbf{G}(q^{-1})q$. For more details, refer to [14].

2. Transfer function model

The SGPC by Kouvaritakis *et al.* [49, 93] employed a transfer function model, which uses the concept of transfer function $G = b/a$ and has a form as follows:

$$a(q^{-1})y[k] = b(q^{-1})u[k], \quad (2.3)$$

with a, b polynomials, and u, y the input and output signals.

In practical, this transfer function model can be considered as the rearrangement of CARIMA model in (2.1). Rewrite (2.1) as follows (for simplicity, we will omit q^{-1} in the polynomials if no confusion arises):

$$(a\Delta) \left[\frac{y[k]}{c} \right] = (bq^{-d}) \left[\frac{\Delta u[k - 1]}{c} \right] + e[k],$$

we see that c acts to filter the input-output data and hence to remove the high frequency noises.

Define two filtered variables:

$$y'[k] = \frac{y[k]}{c}, \quad \Delta u'[k - 1] = \frac{\Delta u[k - 1]}{c},$$

and two new polynomials:

$$a' = a\Delta, \quad b' = bq^{-d-1},$$

we obtain

$$a'y'[k] = b'\Delta u'[k] + e[k]. \quad (2.4)$$

Note that e is a white noise with zero mean, so the predictions of future output signals of model (2.3) and model (2.4) should have the same forms. For industrial applications, GPC algorithm are normally derived by using this transfer function model [94, 95], for the reason

that models are usually identified from the filtered input-output data. The predicted signal \hat{y}' obtained this way has to be filtered by $c(q^{-1})$ in order to get the real prediction \hat{y} .

3. State-space model

A simple state-space model without consideration of disturbances was used by Masuda *et al.* [62]. This model has the following representation:

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k], \end{aligned} \quad (2.5)$$

where x is the state, and A , B , and C are the system, input and output matrices, respectively. Compared with the transfer function expression, the state-space representation is capable of handling multivariable or MIMO systems in a very straightforward manner; however, the issue of state estimation arises.

4. A state-space equivalent of CARIMA model

A state-space equivalence of the CARIMA model in (2.1) has been shown in [75] by Ordys and Clarke. It is described as follows:

$$\begin{aligned} x[k+1] &= Ax[k] + B\Delta u[k] + Fe[k] \\ y[k] &= Cx[k] + e[k]. \end{aligned} \quad (2.6)$$

Compared with (2.5), the integral action is included in (2.6). In place of the control signal $u[k]$, the increment of control signal $\Delta u[k]$ appears, and correspondingly, one of the eigenvalues of the matrix A equals to one. Note that there exists explicit relations between the matrices A , B , C , F and the polynomials in (2.1). If the absolute values of the eigenvalues of the matrix $(A - FC)$ are less than one, then the state estimation can be given by the steady-state Kalman filter:

$$\hat{x}[k+1] = (A - FC)\hat{x}[k] + B\Delta u[k] + Fy[k]. \quad (2.7)$$

2.2 Objective function

Based on a specified prediction model, cost functions in MPC/GPC controllers are minimized to obtain the control law. The aim of GPC is to minimize a partially constrained quadratic optimal control criterion, which is in terms of the incremental inputs and outputs of the plant. This cost function is:

$$J[k] = E \left\{ \sum_{j=N_1}^{N_2} \delta[j] \{y[k+j] - w[k+j]\}^2 \right.$$

$$\begin{aligned}
& + \sum_{j=1}^{N_u} \lambda[j] \{ \Delta u[k+j-1] \}^2 \Big\}, \\
\text{subject to } \Delta u[k+j] & = 0, \quad j = N_u, N_u + 1, \dots, N_2,
\end{aligned} \tag{2.8}$$

where N_1 and N_2 are the minimum and maximum prediction horizons and N_u is the control horizon; u and y are the input and output signals; w is the reference signal for the system to track; $\delta[j]$ and $\lambda[j]$ are the weighting sequences for the tracking error and the incremental control signal. The expectation in (2.8) indicates that the future control sequence are calculated by data available up to and including time k , and presuming the stochastic disturbance model. Thus the design of GPC is in fact a standard finite horizon optimal control problem.

Note the following:

- The minimum cost horizon N_1 can always be taken as 1. In cases that the dead-time of the plant is already known as at least d sample intervals, this N_1 can be chosen as d or more to minimize the computation. In this thesis, we always set $N_1 = 1$.
- The maximum prediction horizon N_2 should encompass all the response which is significantly affected by the current control. Usually N_2 is chosen as larger than the degree of polynomial $b(q^{-1})$; and more typically, a rather larger value of N_2 is suggested, corresponding more closely to the rise-time of the plant. A default setting for N_2 is 10. In this thesis, N_2 is chosen as different values, according to different models and design purposes.
- The control horizon N_u is chosen to be $N_u \leq N_2$. This means after N_u intervals, the projected incremental controls are assumed to be zero. This is equivalent to placing effectively infinite weights on control changes after some future time. Usually $N_u = 1$ will be adequate for typical industry plants to get reasonable performance. In this thesis, as a tuning parameter, N_u is chosen to be an integer greater than 1 for higher performance. And sometimes we choose $N_2 = N_u = N$ for the reason of simplicity in the calculation.
- GPC can adopt both constant and varying future setpoints. In most cases, the reference signal $w[k+j]$ will be a constant w equal to the current setpoint $w[k]$. In some cases, $w[k+j]$ can be a smooth approximation from the current value $y[k]$ towards the known reference by means of a simple first order lag system:

$$\begin{aligned}
w[k] & = y[k], \\
w[k+j] & = \alpha w[k+j-1] + (1-\alpha)w, \quad j = 1, 2, \dots
\end{aligned} \tag{2.9}$$

where α is a parameter between 0 and 1. It influences the dynamic response of the system, and the closer to 1 this value, the smoother the transition from the current measured variable to the real setpoint w . In this thesis, $w[k+j]$ will be taken as a constant.

- The weighting sequences $\delta[j]$ and $\lambda[j]$ are usually chosen constant or exponentially increasing. In this thesis, $\delta[j]$ is chosen to be 1 and $\lambda[j]$ is set to be a tuning constant λ ; and thus a simplified cost function of (2.8) is:

$$J[k] = \sum_{j=N_1}^{N_2} \{\hat{y}[k+j] - w[k+j]\}^2 + \lambda \sum_{j=1}^{N_u} \{\Delta u[k+j-1]\}^2. \quad (2.10)$$

2.3 Obtaining the control law (constraint-free)

To calculate the control sequence

$$\Delta u[k+i-1], \quad i = 1, 2, \dots, N_u,$$

we need to minimize (2.10) with the knowledge of the prediction model. To do this, the key step is to write out the i -step ahead prediction of the future output signal $\hat{y}[k+i|k]$, based on all the information available up to the current time k . We assume that during the minimization procedure, there is no constraint on the control signals, for otherwise there does not exist an analytical solution.

1. Control law for the CARIMA model

The prediction for CARIMA model (2.1) involves the use of two Diophantine equations, which is standard in the theory of prediction of such stochastic processes [24]. The first Diophantine equation is:

$$c(q^{-1}) = E_i(q^{-1})a(q^{-1})\Delta + q^{-i}F_i(q^{-1}), \quad (2.11)$$

where E_i, F_i are polynomials and the degree of $E_i(q^{-1})$ is $i-1$. Referring to (2.1), without loss of generality, assuming the time delay $d=0$, and dropping the operator q^{-1} , we have:

$$y[k+i] = \frac{b}{a}u[k+i-1] + E_i e[k+i] + \frac{F_i}{a\Delta}e[k] \quad (2.12)$$

$$= \frac{F_i}{c}y[k] + \frac{E_i b}{c}\Delta u[k+i-1] + E_i e[k+i], \quad (2.13)$$

where (2.13) is derived by replacing $e[k]$ in (2.12) with (2.1), and $e[k+i]$ in (2.13) is independent from the signals known at time k . The minimum variance prediction of $y[k+i]$

is thus given by:

$$\hat{y}[k+i|k] = \frac{F_i}{c}y[k] + \frac{E_i b}{c}\Delta u[k+i-1]. \quad (2.14)$$

Next we use the second Diophantine equation to write out past and future control increments:

$$E_i(q^{-1})b(q^{-1}) = G_i(q^{-1})c(q^{-1}) + q^{-i}\tilde{G}_i(q^{-1}), \quad (2.15)$$

where G_i is a polynomial of order $i-1$ and is given by

$$G_i = g_{i,0} + g_{i,1}q^{-1} + g_{i,2}q^{-2} + \cdots + g_{i,i-1}q^{-(i-1)}. \quad (2.16)$$

A new form of prediction for $y[k+i]$ is then yielded:

$$\begin{aligned} \hat{y}[k+i|k] &= G_i(q^{-1})\Delta u[k+i-1] + \tilde{y}[k+i|k] \\ \text{where } \tilde{y}[k+i|k] &= \frac{F_i}{c(q^{-1})}y[k] + \frac{\tilde{G}_i}{c(q^{-1})}\Delta u[k-1]. \end{aligned} \quad (2.17)$$

When i varies from $N_1 = 1$ to $N_2 \geq N_u$ with the following definitions:

$$\hat{Y}[k] = \begin{bmatrix} \hat{y}[k+1|k] & \hat{y}[k+2|k] & \cdots & \hat{y}[k+N_2|k] \end{bmatrix}^T, \quad (2.18)$$

$$\tilde{Y}[k] = \begin{bmatrix} \tilde{y}[k+1|k] & \tilde{y}[k+2|k] & \cdots & \tilde{y}[k+N_2|k] \end{bmatrix}^T, \quad (2.19)$$

$$\Delta U[k] = \begin{bmatrix} \Delta u[k] & \Delta u[k+1] & \cdots & \Delta u[k+N_u-1] \end{bmatrix}^T, \quad (2.20)$$

all the i -step ahead predictions can be written in a compact form:

$$\hat{Y}[k] = G \cdot \Delta U[k] + \tilde{Y}[k],$$

where

$$G = \begin{bmatrix} g_{1,0} & & & & \\ g_{2,1} & g_{2,0} & & & \\ \vdots & \ddots & \ddots & & \\ g_{N_u, N_u-1} & \ddots & \ddots & g_{N_u, 1} & \\ \vdots & & \ddots & \vdots & \\ g_{N_2, N_2-1} & \cdots & \cdots & g_{N_2, N_2-N_u+1} & \end{bmatrix},$$

and the matrix G is Toeplitz (according to equation (2.16), the first i coefficient of G_{i+1} will be identical to those of G_i). The performance index (2.10) can thus be written as:

$$J[k] = (\hat{Y}[k] - W[k])^T (\hat{Y}[k] - W[k]) + \lambda \Delta U[k]^T \Delta U[k], \quad (2.21)$$

where

$$W[k] = \begin{bmatrix} w[k+1] & w[k+2] & \cdots & w[k+N_2] \end{bmatrix}^T, \quad (2.22)$$

and the minimization of $J[k]$ with respect to $\Delta U[k]$ (at time instant k) results in the following constraint-free optimal control law:

$$\Delta U[k] = (G^T G + \lambda I)^{-1} G^T (W[k] - \tilde{Y}[k]). \quad (2.23)$$

2. Control law for the transfer function model

The prediction for the transfer function model in (2.3) is more straightforward than that for the CARIMA model. The control law can be calculated without the use of the Diophantine equation [93, 95]. Add an integrator to the model (2.3), i.e.,

$$(a(q^{-1})\Delta) \cdot y[k] = b(q^{-1}) \cdot \Delta u[k],$$

and assume

$$\begin{aligned} A(q^{-1}) &= a(q^{-1})\Delta = 1 + A_1 q^{-1} + A_2 q^{-2} + \cdots + A_l q^{-l}, \\ b(q^{-1}) &= b_0 + b_1 q^{-1} + b_2 q^{-2} + \cdots + b_l q^{-l}, \end{aligned}$$

with l the order of the system, (2.3) can be rewritten as:

$$\begin{aligned} y[k] &= b_0 \Delta u[k] + b_1 \Delta u[k-1] + \cdots + b_l \Delta u[k-l] \\ &\quad - A_1 y[k-1] - A_2 y[k-2] - \cdots - A_l y[k-l]. \end{aligned}$$

Then the i -step ahead prediction of future output based on information available at time k is:

$$\begin{aligned} \hat{y}[k+i|k] &= M_i^{[i]} \Delta u[k] + \cdots + M_0^{[i]} \Delta u[k+i] \\ &\quad + N_2^{[i]} \Delta u[k-1] + \cdots + N_l^{[i]} \Delta u[k-l+1] \\ &\quad + D_1^{[i]} y[k] + \cdots + D_l^{[i]} y[k-l+1], \end{aligned} \quad (2.24)$$

where $(\cdot)^{[i]}$ ($i = 1, 2, \dots, N_2$) is used to denote parameters for the i -step ahead prediction, and:

$$M_1^{[1]} = b_1, \quad M_0^{[1]} = b_0, \quad (2.25)$$

$$N_\alpha^{[1]} = b_\alpha, \quad \alpha = 2, 3, \dots, l, \quad (2.26)$$

$$D_\beta^{[1]} = -A_\beta, \quad \beta = 1, \dots, l, \quad (2.27)$$

$$M_\alpha^{[\gamma]} = b_\alpha^{[1]} + \sum_{k=1}^{\alpha} D_k^{[1]} M_{\alpha-k}^{[\gamma-k]}, \quad \alpha = 0, 1, \dots, \gamma, \quad \gamma = 2, 3, \dots, N_2,$$

$$N_\alpha^{[\gamma]} = N_{\alpha+\gamma-1}^{[1]} + \sum_{k=1}^{\gamma-1} D_k^{[1]} N_\alpha^{[\gamma-k]} \quad \alpha = 2, 3, \dots, l, \quad \gamma = 1, 2, \dots, N_2,$$

$$D_\alpha^{[\gamma]} = D_{\alpha+\gamma-1}^{[1]} + \sum_{k=1}^{\gamma-1} D_k^{[1]} D_\alpha^{[\gamma-k]} \quad \alpha = 1, 2, \dots, l, \quad \gamma = 2, 3, \dots, N_2.$$

The remaining symbols not covered in (2.25), (2.26) and (2.27) are taken to be zero, e.g, $N_x^{[1]} = 0$ with $x > l$.

Defining $\hat{Y}[k]$, $\Delta U[k]$, $W[k]$ as in (2.18), (2.20), and (2.22), and letting i change from $N_1 = 1$ to $N_2 \geq N_u$, we have a compact form for all the N_2 prediction equations:

$$\hat{Y}[k] = P_1 \Delta U[k] + P_2 \Delta U_p[k] + P_3 Y_p[k],$$

where

$$P_1 = \begin{bmatrix} M_1^{[1]} & M_0^{[1]} & & \\ \vdots & & \ddots & \\ M_{N_u-1}^{[N_u-1]} & & & M_0^{[N_u-1]} \\ \vdots & & & \\ M_{N_2}^{[N_2]} & & & M_{N_2-N_u+1}^{[N_2]} \end{bmatrix},$$

$$P_2 = \begin{bmatrix} N_2^{[1]} & N_3^{[1]} & \dots & N_l^{[1]} \\ N_2^{[2]} & N_3^{[2]} & \dots & N_l^{[2]} \\ \vdots & \vdots & & \vdots \\ N_2^{[N_2]} & N_3^{[N_2]} & \dots & N_l^{[N_2]} \end{bmatrix}, \quad \Delta U_p[k] = \begin{bmatrix} \Delta u[k-1] \\ \vdots \\ \Delta u[k-l+1] \end{bmatrix},$$

$$P_3 = \begin{bmatrix} D_1^{[1]} & D_2^{[1]} & \dots & D_l^{[1]} \\ D_1^{[2]} & D_2^{[2]} & \dots & D_l^{[2]} \\ \vdots & \vdots & & \vdots \\ D_1^{[N_2]} & D_2^{[N_2]} & \dots & D_l^{[N_2]} \end{bmatrix}, \quad Y_p[k] = \begin{bmatrix} y[k] \\ \vdots \\ y[k-l+1] \end{bmatrix}.$$

The optimal constraint-free GPC solution can thus be derived by making the gradient of J equal to zero:

$$\Delta U[k] = (P_1^T P_1 + \lambda I)^{-1} P_1^T (W[k] - P_2 \Delta U_p[k] - P_3 Y_p[k]). \quad (2.28)$$

3. Control law for the state-space model

From model (2.5), the i -step ahead prediction of the future output is as follows:

$$\begin{aligned} \hat{y}[k+i|k] &= C \hat{x}[k+i] \\ &= \begin{cases} CA^i \hat{x}[k] + \sum_{j=1}^i CA^{i-j} Bu[k+j-1], & \text{when } i \leq N_u \\ CA^i \hat{x}[k] + \sum_{j=1}^{N_u} CA^{i-j} Bu[k+j-1]. & \text{when } i > N_u \end{cases} \end{aligned} \quad (2.29)$$

When i varies from $N_1 = 1$ to $N_2 \geq N_u$, all the N_2 predictions can be put into one equations:

$$\hat{Y}[k] = G \cdot U[k] + \Phi \cdot \hat{x}[k],$$

where we have defined:

$$\begin{aligned}
U[k] &= \begin{bmatrix} u[k] & u[k+1] & \cdots & u[k+N_u-1] \end{bmatrix}^T, \\
G &= \begin{bmatrix} CB \\ CAB & CB \\ \vdots & \ddots & \ddots \\ CA^{N_u-1}B & \ddots & & CB \\ \vdots & \ddots & \ddots & \vdots \\ CA^{N_2-1}B & \cdots & \cdots & CA^{N_2-N_u}B \end{bmatrix}, \\
\Phi &= \begin{bmatrix} (CA)^T & (CA^2)^T & \cdots & (CA^{N_2})^T \end{bmatrix}^T,
\end{aligned}$$

and $\hat{Y}[k]$ is the same as in (2.18).

Note here that

$$U[k] = \Delta U[k] + U[k-1],$$

the minimization of $J[k]$ in (2.21) with respect to $\Delta U[k]$ then gives the constraint-free solution:

$$\Delta U[k] = (G^T G + \lambda I)^{-1} G^T (W[k] - G \cdot U[k-1] - \Phi \cdot \hat{x}[k]), \quad (2.30)$$

where the future reference sequence $W[k]$ has been defined in (2.22).

To estimate the state $x[k]$ in (2.30), a full-order observer could be used:

$$\begin{aligned}
\hat{x}[k+1] &= A\hat{x}[k] + Bu[k] + L(y[k] - C\hat{x}[k]), \\
&= (A - LC)\hat{x}[k] + Bu[k] + Ly[k].
\end{aligned} \quad (2.31)$$

Here L is the observer gain and $(A - LC)$ is chosen to be stable.

4. Control law for the state space equivalence of the CARIMA model

For model (2.6), the output signal at time instant $k+i$, $i = 1, 2, \dots, N_2$ is:

$$\begin{aligned}
y[k+i] &= CA^i x[k] + \sum_{j=1}^i CA^{j-1} Fe[k+i-j] \\
&\quad + e[k+i] + \begin{cases} \sum_{j=1}^i CA^{j-1} B \Delta u[k+i-j], & \text{when } i \leq N_u \\ \sum_{j=1}^{N_u} CA^{i-j} B \Delta u[k+j-1]. & \text{when } i > N_u \end{cases}
\end{aligned}$$

Based on the information available up to and at time instant k , the i -step ahead prediction of output can then be derived:

$$\begin{aligned}
\hat{y}[k+i|k] &= CA^i \hat{x}[k] + CA^{i-1} F \hat{e}[k] \\
&\quad + \begin{cases} \sum_{j=1}^i CA^{j-1} B \Delta u[k+i-j], & \text{when } i \leq N_u \\ \sum_{j=1}^{N_u} CA^{i-j} B \Delta u[k+j-1], & \text{when } i > N_u \end{cases}
\end{aligned} \quad (2.32)$$

here $\hat{x}[k]$ and $\hat{e}[k]$ are the estimates of the state $x[k]$ and the disturbance $e[k]$, respectively. Moreover, all the estimation of future disturbances has been set to be zero since e is assumed to be white noise with zero mean. Replacing $\hat{e}[k]$ in (2.32) by (2.6), i.e.,

$$\hat{e}[k] = y[k] - C\hat{x}[k],$$

then (2.32) can be rewritten as:

$$\begin{aligned} \hat{y}[k+i|k] &= CA^{i-1}(A-FC)\hat{x}[k] + CA^{i-1}Fy[k] \\ &+ \begin{cases} \sum_{j=1}^i CA^{i-1}B\Delta u[k+i-j], & \text{when } i \leq N_u \\ \sum_{j=1}^{N_u} CA^{i-j}B\Delta u[k+j-1]. & \text{when } i > N_u \end{cases} \end{aligned} \quad (2.33)$$

Again, when i takes values from $N_1 = 1$ to $N_2 \geq N_u$, all the N_2 predictions of future output signals can be collected into one compact equation:

$$\hat{Y}[k] = G \cdot \Delta U[k] + \Phi \cdot \hat{x}[k] + \Gamma \cdot y[k],$$

where $\hat{Y}[k]$, $\Delta U[k]$ are the same as in (2.18) and (2.20); and matrices G , Φ , and Γ are defined as follows:

$$\begin{aligned} G &= \begin{bmatrix} CB & & & & \\ CAB & CB & & & \\ \vdots & \ddots & \ddots & & \\ CA^{N_u-1}B & \ddots & & & CB \\ \vdots & \ddots & \ddots & & \vdots \\ CA^{N_2-1}B & \dots & \dots & CA^{N_2-N_u}B & \end{bmatrix}, \\ \Phi &= \left[(C)^T \quad (CA)^T \quad \dots \quad (CA^{N_2-1})^T \right]^T (A-FC), \\ \Gamma &= \left[(CF)^T \quad (CF^2)^T \quad \dots \quad (CF^{N_2})^T \right]^T. \end{aligned}$$

At time instant k , with the same W in (2.22), the following optimal constraint-free control law can be obtained:

$$\Delta U[k] = (G^T G + \lambda I)^{-1} G^T (W[k] - \Phi \cdot \hat{x}[k] - \Gamma \cdot y[k]). \quad (2.34)$$

The state $x[k]$ in (2.34) can be directly estimated by the steady-state Kalman filter (2.7).

2.4 Some remarks

Once the future incremental control sequence $\Delta U[k]$ is calculated at time k , only the first element $\Delta u[k]$ will be implemented to the process, i.e., $u[k] = u[k-1] + \Delta u[k]$. At next

sample time, all the procedures will be repeated. Thus GPC is a receding horizon control strategy.

When we derive control laws, we have assumed that there is no constraint on the inputs and outputs. However, in practice, all processes are subject to constraints. For example, the valves are limited by the positions of totally open or closed and by the response rate. Normally, the constraints on a system can be the bounds in the amplitude and in the flow rate of the control signal and limits in the output:

$$\begin{aligned} u_{min} &\leq u[k] \leq u_{max}, \quad \forall k \\ \Delta u_{min} &\leq u[k] - u[k-1] \leq \Delta u_{max}, \quad \forall k \\ y_{min} &\leq y[k] \leq y_{max}, \quad \forall k \end{aligned}$$

When these constraints are added, the minimization of the cost function in (2.10) will become more complex. An explicit solution does not exist in the presence of constraints. To find the optimal solution, the quadratic programming (QP), or the convex optimization methods with LMI can be used.

Many processes are affected by external disturbances caused by the variation of variables that can be measured. And it is possible to extend the original GPC algorithm to include the case of measurable disturbances [14]. Including the disturbances, the CARIMA model changes to:

$$a(q^{-1})y[k] = b(q^{-1})u[k-1] + d(q^{-1})v[k] + c(q^{-1})\frac{e[k]}{\Delta}, \quad (2.35)$$

where $v[k]$ is the measured disturbance at time k and $d(q^{-1})$ is a known polynomial. By using the Diophantine equation, the i -step ahead expected value for $y[k+i]$ can be interpreted as a combination of the free response and forced response. There will be one term in the forced response that depends on the future deterministic disturbances. Thus the prediction of $y[k+i]$ is related with the properties of the disturbance. The final optimal control law has a similar form as that for the case with no measured disturbances; it only differs from (2.23) in that the future “known” disturbances will affect the control law for systems described by (2.35).

Chapter 3

GPC for Non-Uniformly Sampled Systems

3.1 Introduction

Generalized predictive control (GPC) [24, 25] has found wide applications in the process control industry, mainly due to its features such as the time-domain formulation, receding horizon scheme, and constraint-handling capability. Most studies on GPC assume a single-rate sampling scheme [11, 14]. The main purpose of this chapter is to extend GPC algorithms to more general sampling and updating schemes.

One extension from single-rate systems is the class of multirate systems. The relevance and importance of multirate processes in the MPC/GPC framework have been recognized by several researchers over the last decade, for example, Lee *et al.* [54], and Scattolini *et al.* [99]. They investigated multirate MPC/GPC design in the state-space setting, but dealt with the time-varying nature of multirate systems directly in the time domain. To avoid the complex time-varying systems and problems, our approach to multirate GPC extension will be through the use of the lifting technique.

The idea of lifting was due to Kranc in 1957 [50], in which he proposed a switch decomposition technique; this was later developed into the widely used lifting technique by Khar-gonekar *et al.* [47]. There are several advantages in using the lifting technique: First, it is conceptually simple – lifting converts a time-varying multirate system into a time-invariant single-rate system, for which there is a rich source of results. Second, lifted control moves are computed over a larger interval which is usually integer multiples of the sampling and updating periods; this implies that there is a computational advantage in GPC optimization over the time-varying state-space methods in [54, 99]. Third, lifting gives rise to an LTI framework in which stability and robustness analysis of the resultant closed-loop systems

can be done in a relatively easy manner comparing with the time-varying methods. However, the obstacle in using the lifting technique is the so-called *causality constraint*, which enforces certain structural constraint in the controller feedthrough terms. How to handle this causality constraint has been the main challenge in the robust multirate control [19], and it remains to be an unsolved problem in the GPC design (although idea similar to lifting has been applied to multirate MPC/GPC topics [16, 61], the causality constraint in the design has not been taken into account). One of the main objective in this chapter is to propose a solution to tackle the causality problem in the GPC framework with a more general sampling and updating scheme – the non-uniformly sampled systems.

There are three main reasons for non-uniformly sampled systems to arise: First, in typical applications, a distributed computer system is usually used to implement digital control operations, in addition to other tasks such as process monitoring and setpoint optimization; in such task-sharing situations, it is more reasonable and cost-effective to allow non-uniform sampling and updating operations. Second, the non-uniform sampled systems are quite general and include multirate sampled-data systems as special cases, e.g., for a multirate system as shown in Figure 1.1 with a period of $2h$ for S and a period of $3h$ for H , the sampling and updating pattern is periodic with period $T = 6h$, but the sampling (updating) instants are equally spaced. Third, there are advantages in non-uniformly sampled systems over uniformly sampled ones, e.g., Kreisselmeier [51] proposed a non-uniformly sampled (but periodic) scheme which always preserves controllability and observability in discretization. (Recall that a non-pathological sampling condition is required to guarantee controllability and observability in the uniformly sampled case, see, e.g., [20].)

Research activities on non-uniformly sampled systems exist: Except the reference in [51] mentioned earlier which dealt with controllability and observability issues; modeling of slowly sampled, fast and non-uniformly updated systems was studied in [97]; and a receding horizon control problem was investigated in [2], where non-synchronized discrete-time signals were treated. These studies all assumed the sampling and updating patterns are periodic (with a frame period T), but the patterns are special cases of what we will propose here in the chapter.

Briefly, the contributions in this chapter are as follows:

- Using the lifting technique, we derive a state-space model for the non-uniformly and periodically sampled system in discrete time, based on the continuous-time model; a sufficient condition on sampling is given under which the state-space model is both controllable and observable.

- We present a solution to the GPC problem for non-uniformly and periodically sampled systems, treating the causality constraint by grouping output samples appropriately. The solution is based on the input-output approach in [93, 95]. To our best knowledge, this is the first causal and optimal solution proposed for the lifted models.
- Such a causal GPC solution in the lifted domain is *new* even in the special case of multirate systems in which the causality constraint has been the main difficulty in GPC/MPC design using the lifting framework [102].

The rest of the chapter is organized as follows. Section 3.2 studies modeling issues for the non-uniformly and periodically sampled systems, from continuous time to the lifted discrete time. Controllability and observability of the lifted models are also addressed. Section 3.3 forms the main contribution in this paper by formulating the non-uniformly sampled GPC problem and deriving a causal and optimal solution. Section 3.4 presents an illustrative example for the results in Sections 3.2 and 3.3. Finally, in Section 3.5, concluding remarks are given.

3.2 Modeling of non-uniformly sampled systems

In view of Figure 1.1, in this section we define precisely the non-uniform sampler S and zero-order hold H , and derive a state-space model in the lifted domain. For simplicity, we focus on the single-input, single-output (SISO) case.

3.2.1 The sampling and updating scheme

In Figure 1.1, the continuous-time process P is assumed to have the following state-space representation:

$$P : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{cases} \quad (3.1)$$

where $x(t) \in R^{n_c}$ is the state, $u(t) \in R^1$ is the control input, and $y(t) \in R^1$ is the output.

Non-uniformly sampled systems are characterized by the fact that both the control updating instants (when $u[k]$ occur) and the sampling instants (when $y[k]$ occur) need not be equally spaced in time; however, for tractability we adopt the same assumption as in [97, 2] that the whole sampling and updating pattern is periodic over a larger interval T , known as the *frame period*. The notation and arrangement with the sampling and updating are as follows:

- Over the k -th period $[kT, (k+1)T)$, we assume the control signal u is updated non-uniformly m times at time instants $kT + t_i$, $i = 1, 2, \dots, m$. Without loss of generality, we can take $t_1 = 0$ and arrange $t_1 < t_2 < \dots < t_m < T$.
- Over the period $[kT, (k+1)T)$, there are n_i ($n_i \geq 0$) output samples available within the time interval $[kT + t_i, kT + t_{i+1})$, $i = 1, 2, \dots, m$ (denoting $t_{m+1} = T$); these n_i output samples occur at time instants $kT + t_i^j$, $j = 1, 2, \dots, n_i$. Without loss of generality, we arrange these t_i^j in the following order:

$$t_i \leq t_i^1 < t_i^2 < \dots < t_i^{n_i} < t_{i+1}.$$

Thus during each period of T , the control signal u is updated m times, and the output signal y is sampled $p = n_1 + n_2 + \dots + n_m$ times, all non-uniformly. Such a sampling and updating scheme is briefly illustrated in Figure 3.1. We remark that the non-uniform sampling and updating scheme introduced is quite general: Compared with the one used in [2], we do not assume that the sampling and updating instants are integer multiples of some base period.

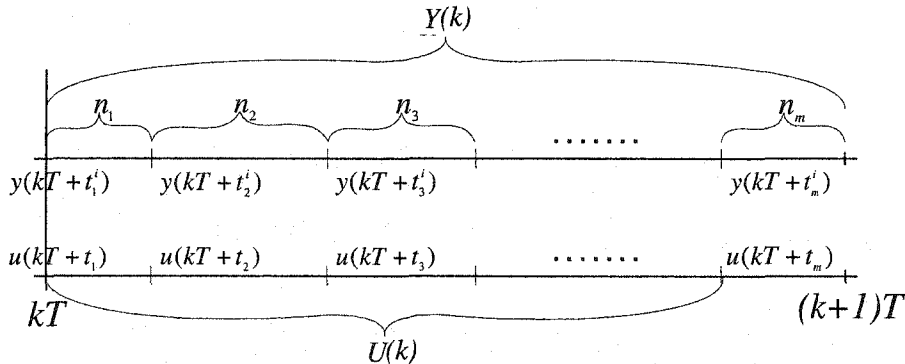


Figure 3.1: The non-uniform sampling and updating scheme

Single-rate GPC problems are usually framed in the discrete-time domain [24, 25, 93, 95]. We will next discuss the lifting technique and lifted discrete-time models for the non-uniformly and periodically sampled systems.

3.2.2 Lifting, lifted models and the causality constraint issues

Looking at the system discussed earlier in discrete time, from $u[k]$ to $y[k]$, it is clearly time-varying, due to the non-uniform sampling and updating scheme. However, if we group every m input values and every p output samples together, we have a $p \times m$ LTI system operating over period T ; this is the idea of lifting.

Let $v[k]$ be a discrete-time signal:

$$v = \{v[0], v[1], v[2], \dots\}. \quad (3.2)$$

For a positive integer n , the n -fold lifted signal, denoted \underline{v} is defined as

$$\underline{v} = \left\{ \left[\begin{array}{c} v[0] \\ v[1] \\ \vdots \\ v[n-1] \end{array} \right], \left[\begin{array}{c} v[n] \\ v[n+1] \\ \vdots \\ v[2n-1] \end{array} \right], \dots \right\}.$$

The map from v to \underline{v} is defined as the lifting operator L_n . The inverse lifting operation, L_n^{-1} , is from \underline{v} to v , defined in the obvious way.

The non-uniformly sampled system from $u[k]$ to $y[k]$ is linear but time-varying. Because the sampling and updating scheme is periodic with period T , we can use lifting to convert the system into an LTI system. Lifting $u[k]$ by L_m and $y[k]$ by L_p and noting that $p = n_1 + n_2 + \dots + n_m$, we get $\underline{u}[k]$ and $\underline{y}[k]$, corresponding to inputs and outputs over the interval $[kT, (k+1)T)$:

$$\underline{u}[k] = \begin{bmatrix} u(kT + t_1) \\ u(kT + t_2) \\ \vdots \\ u(kT + t_m) \end{bmatrix}, \quad \underline{y}[k] = \begin{bmatrix} n_1 \left\{ \begin{array}{c} y(kT + t_1^1) \\ \vdots \\ y(kT + t_1^{n_1}) \end{array} \right\} \\ \vdots \\ n_m \left\{ \begin{array}{c} y(kT + t_m^1) \\ \vdots \\ y(kT + t_m^{n_m}) \end{array} \right\} \end{bmatrix}. \quad (3.3)$$

The lifted system, P_l , maps $\underline{u}[k]$ to $\underline{y}[k]$, and has m inputs and p outputs. From Proposition 1 below we will see that P_l admits a state-space realization in terms of given continuous-time model in (3.1) and the sampling and updating instants; hence P_l is LTI, an advantage of lifting.

Proposition 1 *A state-space model for the lifted system P_l is given by*

$$P_l : \begin{cases} \underline{x}[k+1] = \underline{A} \underline{x}[k] + \underline{B} \underline{u}[k], \\ \underline{y}[k] = \underline{C} \underline{x}[k] + \underline{D} \underline{u}[k], \end{cases} \quad (3.4)$$

where $\underline{x}[k] := x(kT)$, and

$$\underline{A} = e^{AT}, \quad \underline{B} = [B_1 \ B_2 \ \dots \ B_m], \quad (3.5)$$

$$\underline{C} = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{bmatrix}, \quad \underline{D} = \begin{bmatrix} D_1^1 & & & \\ D_2^1 & D_2^2 & & \\ \vdots & \vdots & \ddots & \\ D_m^1 & D_m^2 & \dots & D_m^m \end{bmatrix}, \quad (3.6)$$

with

$$B_i = \int_{T-t_{i+1}}^{T-t_i} e^{A\tau} B d\tau, \quad i = 1, 2, \dots, m, \quad (3.7)$$

$$C_i = \begin{bmatrix} C e^{A t_i^1} \\ \vdots \\ C e^{A t_i^{n_i}} \end{bmatrix}, \quad i = 1, 2, \dots, m, \quad (3.8)$$

$$D_i^i = \begin{bmatrix} \int_0^{t_i^1-t_i} C e^{A\tau} B d\tau + D \\ \vdots \\ \int_0^{t_i^{n_i}-t_i} C e^{A\tau} B d\tau + D \end{bmatrix}, \quad i = 1, 2, \dots, m,$$

$$D_i^j = \begin{bmatrix} \int_{t_i^1-t_{j+1}}^{t_i^1-t_j} C e^{A\tau} B d\tau \\ \vdots \\ \int_{t_i^{n_i}-t_{j+1}}^{t_i^{n_i}-t_j} C e^{A\tau} B d\tau \end{bmatrix}, \quad i = 2, \dots, m; \quad 1 \leq j < i.$$

Proof Solving the state equation in (3.1) from kT to $(k+1)T$, we have

$$x((k+1)T) = e^{((k+1)T-kT)A} x(kT) + \int_{kT}^{(k+1)T} e^{((k+1)T-\tau)A} B u(\tau) d\tau. \quad (3.9)$$

Since

$$u(\tau) = \begin{cases} u(kT + t_1), & \tau \in [kT + t_1, kT + t_2), \\ \vdots \\ u(kT + t_m), & \tau \in [kT + t_m, kT + T), \end{cases}$$

the state at time $(k+1)T$ in (3.9) can be rewritten as:

$$x((k+1)T) = e^{AT} x(kT) + \sum_{i=1}^m \int_{kT+t_i}^{kT+t_{i+1}} e^{[(k+1)T-\tau]A} B d\tau u(kT + t_i).$$

For i from 1 to m , note that $t_1 = 0$, $t_{m+1} = T$, and

$$\int_{kT+t_i}^{kT+t_{i+1}} e^{((k+1)T-\tau)A} B d\tau = \int_{T-t_{i+1}}^{T-t_i} e^{A\tau} B d\tau.$$

Using the definitions in Proposition 1, we get easily:

$$\underline{x}[k+1] = \underline{A} \underline{x}[k] + \underline{B} \underline{u}[k].$$

Also, for the output equation, we have

$$y(kT + t_i^j) = C x(kT + t_i^j) + D u(kT + t_i),$$

where $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n_i$. The state at each time instant can be written as

$$x(kT + t_i^j) = e^{A t_i^j} x(kT) + \sum_{l=1}^{i-1} \int_{t_i^l-t_{l+1}}^{t_i^j-t_l} e^{A\tau} B d\tau u(kT + t_l) + \int_0^{t_i^j-t_i} e^{A\tau} B d\tau u(kT + t_i).$$

Putting these equations together in the definition of $\underline{y}[k]$, we arrive at the output equation in Proposition 1:

$$\underline{y}[k] = \underline{C} \underline{x}[k] + \underline{D} \underline{u}[k],$$

where \underline{C} and \underline{D} are defined as in Proposition 1.

Q.E.D

Note that the upper triangular blocks in \underline{D} in (3.6) are zero; this represents the so-called *causality constraint* in P_l : Certain blocks in the direct feedthrough term must be zero to satisfy causality. Since the model for P_l is derived from the continuous-time model, the causality constraint is automatically satisfied. However, in controller design, the causality constraint on the lifted controllers poses a difficult problem. The lifted controller, to be designed, takes $\underline{y}[k]$ to $\underline{u}[k]$; the causality structure for the direct feedthrough term in this case is summarized in the following proposition.

Proposition 2 *The direct feedthrough terms in the lifted controllers are $m \times p$ matrices mapping $\underline{y}[k]$ to $\underline{u}[k]$, and must satisfy the causality constraint, which takes the following block lower triangular structure:*

$$\left[\begin{array}{cccc} \overbrace{a_1 \ 0 \ \cdots \ 0}^{n_1} & & & \\ \times \ \times \ \cdots \ \times & \overbrace{a_2 \ 0 \ \cdots \ 0}^{n_2} & & \\ \vdots & \vdots & & \\ \times \ \times \ \cdots \ \times & \times \ \times \ \cdots \ \times & \cdots & \overbrace{a_m \ 0 \ \cdots \ 0}^{n_m} \end{array} \right]. \quad (3.10)$$

Here the upper triangular blocks are all zero, \times means a designable element (no restriction). If $t_i^1 = t_i$, $a_i = \times$; otherwise, $a_i = 0$.

This causality constraint must be satisfied by all lifted controllers, for otherwise the controllers cannot be implemented in real time (some control moves would be dependent on future sampled values). Although lifting converts time-varying systems into LTI systems, the causality condition in the lifted controllers poses a hard constraint. How to handle this constraint is the new feature in control design for non-uniformly sampled systems. (Note that since the lifted process models are derived from the continuous-time systems, they automatically satisfy the causality condition.)

We remark that an alternative approach is to model the non-uniformly sampled system in terms of periodically time-varying state-space equations, thus connecting to the MPC/GPC methods studied in [54, 99]. In light of this, the lifting approach we adopted here amounts

to selecting a different state vector and stacking up the inputs and outputs according to the frame period.

3.2.3 Controllability and observability

In Section 3.2.2, we presented a lifted discrete-time model for the non-uniformly sampled system described in Section 3.2.1; the lifted model was given in the state-space form (Proposition 1). A natural question is: Under what condition the lifted model is controllable and observable?

It is well-known that sampling cannot gain controllability and observability; so it is necessary that we assume controllability and observability of the continuous-time model in (3.1). We will give a sufficient condition for the lifted model to preserve the two properties, namely, that the frame period T is *non-pathological* relative to A , which means that no two eigenvalues of A differ by a non-zero integer multiple of $2\pi j/T$ [20].

Theorem 1 *In the discretization process from P in (3.1) to P_l in Proposition 1, assume the frame period T is non-pathological. Then*

1. $(\underline{A}, \underline{B})$ is controllable if (A, B) is controllable;
2. $(\underline{C}, \underline{A})$ is observable if (C, A) is observable.

Proof Discretizing the continuous-time model in (3.1) with period T , we obtain a single-rate model, denoted P_T :

$$P_T : \begin{cases} x[k+1] &= A_T x[k] + B_T u[k], \\ y[k] &= C x[k] + D u[k], \end{cases} \quad (3.11)$$

where $x[k] := x(kT)$, $y[k] := y(kT)$, $u[k]$ is the single-rate control sequence with period T , and

$$A_T = e^{AT}, \quad B_T = \int_0^T e^{A\tau} B d\tau. \quad (3.12)$$

Assume (A, B) is controllable and (C, A) is observable. Because T is non-pathological, it follows that (A_T, B_T) is controllable and (C, A_T) is observable [20]. Next we will use this to prove Theorem 1.

Part 1: In order to show that $(\underline{A}, \underline{B})$ is controllable, we first note from (3.12) and (3.7) that

$$A_T = \underline{A}, \quad B_T = \sum_{i=1}^m B_i, \quad (3.13)$$

since $t_1 = 0$ and $t_{m+1} = T$. Based on (3.5) and (3.13), it is easily seen that for any eigenvalue λ of \underline{A} ,

$$\text{rank} \begin{bmatrix} \underline{A} - \lambda I & \underline{B} \end{bmatrix} \geq \text{rank} \begin{bmatrix} \underline{A} - \lambda I & B_T \end{bmatrix}.$$

The right-hand side equals to n_c , the dimension of \underline{A} , because of controllability of (A_T, B_T) ; hence $(\underline{A}, \underline{B})$ is controllable.

Part 2: Define the observability matrix for $(\underline{C}, \underline{A})$:

$$\Pi := \begin{bmatrix} \underline{C} \\ \underline{C}\underline{A} \\ \vdots \\ \underline{C}\underline{A}^{n_c-1} \end{bmatrix}.$$

Because of the dimension of \underline{C} , this is a $pn_c \times n_c$ matrix. Now we define

$$\Pi_{ij} := \begin{bmatrix} Ce^{At_i^j} \\ Ce^{At_i^j}\underline{A} \\ \vdots \\ Ce^{At_i^j}\underline{A}^{n_c-1} \end{bmatrix}, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n_i.$$

By the definition of \underline{C} in (3.6), it follows that all rows in Π_{ij} are those of Π . Hence

$$\text{rank } \Pi \geq \text{rank } \Pi_{ij}. \quad (3.14)$$

However, since $e^{At_i^j}$ and \underline{A} commute and $e^{At_i^j}$ is invertible,

$$\text{rank } \Pi_{ij} = \text{rank} \begin{bmatrix} \underline{C} \\ \underline{C}\underline{A} \\ \vdots \\ \underline{C}\underline{A}^{n_c-1} \end{bmatrix} e^{At_i^j} = \text{rank} \begin{bmatrix} \underline{C} \\ \underline{C}\underline{A} \\ \vdots \\ \underline{C}\underline{A}^{n_c-1} \end{bmatrix},$$

which equals to n_c because of observability of (C, A_T) . Thus from (3.14), $\text{rank } \Pi \geq n_c$; this implies observability of $(\underline{C}, \underline{A})$. Q.E.D

Note in the proof that the sufficient condition in Theorem 1 also guarantees that the uniformly sampled system with period T is controllable and observable. A question may arise: Do non-uniformly sampled systems have any advantage over uniformly sampled systems in preserving controllability and observability? The answer is positive. Let us illustrate this with an example, looking at controllability only.

Example Consider a controllable continuous-time model with

$$A = \begin{bmatrix} 0 & -\pi \\ \pi & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Take the frame period T to be 3 sec, and $m = 3$. It can be verified that this T is pathological. If the uniform updating pattern is used, we have $t_1 = 0$, $t_2 = 1$, and $t_3 = 2$. From Proposition 1, the lifted (A, B) pair in this case can be computed as

$$(\underline{A}, \underline{B}) = \left(\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0.6366 & -0.6366 & 0.6366 \end{bmatrix} \right),$$

which is clearly uncontrollable. However, with the same conditions that $T = 3$ and $m = 3$, if we use a non-uniform updating pattern, say, $t_1 = 0$, $t_2 = 0.8$, and $t_3 = 1.2$, then we get

$$(\underline{A}, \underline{B}) = \left(\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} -0.1871 & 0.3742 & -0.1871 \\ 0.5758 & 0 & 0.0608 \end{bmatrix} \right),$$

which is now controllable. We comment here that the choice of the sampling instants is critical in preserving controllability properties.

3.3 GPC algorithm for non-uniformly sampled systems

In this section, we study the GPC design problem for the non-uniformly and periodically sampled systems discussed in the preceding section. We will see that conventional GPC algorithms fail to provide causal control laws; and we propose a new GPC solution, taking into account the causality constraint in (3.10). The lifted model in Proposition 1 will be used in setting up the problems.

In view of the lifted model in (3.4), the GPC design is to minimize a cost function of the form

$$J[k] = \sum_{i=1}^{N_2} \left\{ w[k+i] - \hat{y}[k+i|k] \right\}^T \left\{ w[k+i] - \hat{y}[k+i|k] \right\} + \lambda \sum_{i=1}^{N_u} \left\{ \Delta \underline{u}[k+i-1]^T \Delta \underline{u}[k+i-1] \right\}, \quad (3.15)$$

by computing the incremental control moves $\Delta \underline{u}[k+i]$ for $i = 0, 1, \dots, N_u - 1$, subject to the condition that $\Delta \underline{u}[k+i] = 0$ for $i = N_u, N_u + 1, \dots, N_2$. The vector sequence $w[k+i]$ is the output tracking reference; $\hat{y}[k+i|k]$ is the i -step ahead prediction of the future lifted output at present time k . The minimum and maximum prediction horizons are 1 and N_2 , respectively; N_u is the control horizon. The weighting for the error signal between w and \hat{y} is an identity matrix, and for the lifted incremental control signal is a constant diagonal matrix λI . For simplicity, in the following, we assume $N_2 = N_u = N$.

We choose to deal with the GPC design problem in the input-output framework instead of the state-space framework for two reasons: First, this way we avoid estimating the state

vector in which a causality condition arises; second, the method we use later is based on reformulating the GPC problem by introducing delays in certain output channels, which is easy to accommodate in the transfer function domain, but not in the state-space domain (state dimension is increased with the time delays). Including an integrator $1/\Delta$ to the lifted model in (3.4), we can obtain a transfer function representation from $\Delta \underline{u}$ to \underline{y} :

$$\underline{y}[k] = \frac{N(q^{-1})}{d(q^{-1})} \Delta \underline{u}[k], \quad (3.16)$$

where $d(q^{-1})$ is the common denominator (polynomial in q^{-1}) of the form

$$d(q^{-1}) = 1 + d_1 q^{-1} + d_2 q^{-2} + \dots + d_l q^{-l} \quad (3.17)$$

(assuming the order of the system involved is l), $N(q^{-1})$ is a $p \times m$ matrix polynomial of the form

$$N(q^{-1}) = \begin{bmatrix} N_{11}(q^{-1}) & N_{12}(q^{-1}) & \dots & N_{1m}(q^{-1}) \\ N_{21}(q^{-1}) & N_{22}(q^{-1}) & \dots & N_{2m}(q^{-1}) \\ \vdots & \vdots & & \vdots \\ N_{p1}(q^{-1}) & N_{p2}(q^{-1}) & \dots & N_{pm}(q^{-1}) \end{bmatrix} \quad (3.18)$$

with each element being an l -th order polynomial:

$$N_{ij}(q^{-1}) = N_{ij}^0 + N_{ij}^1 q^{-1} + N_{ij}^2 q^{-2} + \dots + N_{ij}^l q^{-l}. \quad (3.19)$$

In the following, we will omit q^{-1} in the polynomials, if no confusion will arise.

3.3.1 Conventional GPC design

First, following [93, 95], we review how the GPC solution is derived, without considering the causality constraint. This solution will be helpful in two ways: First, it illustrates that the conventional GPC solution is not implementable in the lifted framework, because it is noncausal; second, based on this solution, we will propose a causal GPC solution.

Rewrite (3.16) as follows:

$$\begin{aligned} \underline{y}[k] &= N_0 \Delta \underline{u}[k] + N_1 \Delta \underline{u}[k-1] + \dots + N_l \Delta \underline{u}[k-l] \\ &\quad - D_1 \underline{y}[k-1] - D_2 \underline{y}[k-2] - \dots - D_l \underline{y}[k-l]. \end{aligned} \quad (3.20)$$

Here, based on (3.18) and (3.17), we have

$$N_i = \begin{bmatrix} N_{11}^i & N_{12}^i & \dots & N_{1m}^i \\ N_{21}^i & N_{22}^i & \dots & N_{2m}^i \\ \vdots & \vdots & & \vdots \\ N_{p1}^i & N_{p2}^i & \dots & N_{pm}^i \end{bmatrix}, \quad D_i = \begin{bmatrix} d_i & & & \\ & \dots & & \\ & & d_i & \\ & & & \dots \end{bmatrix}_{p \times p}$$

From [93, 95], for the MIMO system in (3.20), $\hat{y}[k+i|k]$, the i -step ahead prediction of future output at current time k , is

$$\begin{aligned}\hat{y}[k+i|k] &= M_i^{[i]}\Delta u[k] + \cdots + M_0^{[i]}\Delta u[k+i] \\ &\quad + N_2^{[i]}\Delta u[k-1] + \cdots + N_l^{[i]}\Delta u[k-l+1] \\ &\quad + D_1^{[i]}\underline{y}[k] + \cdots + D_l^{[i]}\underline{y}[k-l+1],\end{aligned}\quad (3.21)$$

where $(\cdot)^{[i]}$ ($i = 1, 2, \dots, N$) is used to denote parameters for the i -step ahead prediction:

$$M_1^{[1]} = N_1, \quad M_0^{[1]} = N_0, \quad (3.22)$$

$$N_\alpha^{[1]} = N_\alpha, \quad \alpha = 2, 3, \dots, l, \quad (3.23)$$

$$D_\beta^{[1]} = -D_\beta, \quad \beta = 1, \dots, l, \quad (3.24)$$

$$M_\alpha^{[\gamma]} = N_\alpha^{[1]} + \sum_{k=1}^{\alpha} D_k^{[1]} M_{\alpha-k}^{[\gamma-k]}, \quad \alpha = 0, 1, \dots, \gamma, \quad \gamma = 2, 3, \dots, N,$$

$$N_\alpha^{[\gamma]} = N_{\alpha+\gamma-1}^{[1]} + \sum_{k=1}^{\gamma-1} D_k^{[1]} N_\alpha^{[\gamma-k]} \quad \alpha = 2, 3, \dots, l, \quad \gamma = 1, 2, \dots, N,$$

$$D_\alpha^{[\gamma]} = D_{\alpha+\gamma-1}^{[1]} + \sum_{k=1}^{\gamma-1} D_k^{[1]} D_\alpha^{[\gamma-k]} \quad \alpha = 1, 2, \dots, l, \quad \gamma = 2, 3, \dots, N.$$

The remaining symbols not covered in (3.22), (3.23) and (3.24) are taken to be zero, e.g, $N_x^{[1]} = 0$ with $x > l$.

Now we define

$$\begin{aligned}\hat{Y}[k] &= \left[\hat{y}[k+1|k]^T \quad \cdots \quad \hat{y}[k+N|k]^T \right]^T, \\ W[k] &= \left[w[k+1]^T \quad \cdots \quad w[k+N]^T \right]^T.\end{aligned}$$

Letting i in (3.21) vary from 1 to N , we obtain the following compact equations for $\hat{Y}[k]$ and $J[k]$ in (3.15):

$$\begin{aligned}\hat{Y}[k] &= P_1 \Delta U[k] + P_2 \Delta U_p[k] + P_3 Y_p[k], \\ J[k] &= (W[k] - \hat{Y}[k])^T (W[k] - \hat{Y}[k]) + \Delta U[k]^T \Lambda \Delta U[k],\end{aligned}\quad (3.25)$$

where $\Lambda = \lambda I_{mN \times mN}$,

$$P_1 = \begin{bmatrix} M_1^{[1]} & M_0^{[1]} & & \\ \vdots & & \ddots & \\ M_{N-1}^{[N-1]} & & & M_0^{[N-1]} \\ M_N^{[N]} & & & M_1^{[N]} \end{bmatrix}, \quad \Delta U[k] = \begin{bmatrix} \Delta u[k] \\ \vdots \\ \Delta u[k+N-1] \end{bmatrix},$$

$$P_2 = \begin{bmatrix} N_2^{[1]} & N_3^{[1]} & \dots & N_l^{[1]} \\ N_2^{[2]} & N_3^{[2]} & \dots & N_l^{[2]} \\ \vdots & \vdots & \dots & \vdots \\ N_2^{[N]} & N_3^{[N]} & \dots & N_l^{[N]} \end{bmatrix}, \quad \Delta U_p[k] = \begin{bmatrix} \Delta \underline{u}[k-1] \\ \vdots \\ \Delta \underline{u}[k-l+1] \end{bmatrix},$$

$$P_3 = \begin{bmatrix} D_1^{[1]} & D_2^{[1]} & \dots & D_l^{[1]} \\ D_1^{[2]} & D_2^{[2]} & \dots & D_l^{[2]} \\ \vdots & \vdots & \dots & \vdots \\ D_1^{[N]} & D_2^{[N]} & \dots & D_l^{[N]} \end{bmatrix}, \quad Y_p[k] = \begin{bmatrix} \underline{y}[k] \\ \vdots \\ \underline{y}[k-l+1] \end{bmatrix}.$$

Minimizing $J[k]$ with respect to the future control sequence $\Delta U[k]$, we obtain the optimal GPC solution:

$$\Delta U[k] = (P_1^T P_1 + \lambda I)^{-1} P_1^T (W[k] - P_2 \Delta U_p[k] - P_3 Y_p[k]).$$

In the frame period $[kT, (k+1)T)$, only the first element of $\Delta U[k]$, namely, $\Delta \underline{u}[k]$, will be implemented, and it is given by

$$\Delta \underline{u}[k] = \begin{bmatrix} I_{m \times m} & 0 & \dots & 0 \end{bmatrix} (P_1^T P_1 + \lambda I)^{-1} P_1^T (W[k] - P_2 \Delta U_p[k] - P_3 Y_p[k]). \quad (3.26)$$

Since the first element of $Y_p[k]$ is $\underline{y}[k]$, the direct feedthrough term from $\underline{y}[k]$ to $\Delta \underline{u}[k]$ is

$$\underline{D}_c = - \begin{bmatrix} I_{m \times m} & 0 & \dots & 0 \end{bmatrix} (P_1^T P_1 + \lambda I)^{-1} P_1^T P_3 \begin{bmatrix} I_{p \times p} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Clearly, this \underline{D}_c depends on the design horizons, weighting matrices, and dynamic properties of the process; it is a general $m \times p$ matrix. There is no guarantee that it has the block lower triangular structure as shown in (3.10). So such a conventional GPC solution is not implementable in real time.

Our task next is to find a GPC algorithm which results in a causal control law.

3.3.2 Proposed GPC design

To obtain causal GPC control law, the idea we adopt is as follows: During every frame period T , construct a chain of new lifted signals and models, corresponding to the control moves; then apply the conventional GPC design to every one of them.

To see how to handle the causality issue, let us review the conventional algorithm in Section 3.3.1. Over the k -th frame period $[kT, (k+1)T)$, the optimal control law given in (3.26) is the lifted vector $\Delta \underline{u}[k]$, which has m incremental control moves for the whole frame period. As time goes on, the elements in $\Delta \underline{u}[k]$ would be implemented one by one, e.g.,

on time interval $[kT + t_1, kT + t_2)$, we would implement $\Delta u(kT + t_1)$; on the next interval $[kT + t_2, kT + t_3)$, we would implement $\Delta u(kT + t_2)$. However, $\Delta u(kT + t_1)$, $\Delta u(kT + t_2)$, and so on would depend on future measurements occurring later in the frame period, making it impossible for real-time implementation.

In our proposed algorithm, during the k -th frame period $[kT, (k + 1)T)$, each control move will be calculated separately; at each time instant $t = kT + t_i$, we will construct a corresponding lifted output $\underline{y}^i[k]$, consisting of the most recent p measurements (some elements in $\underline{y}^i[k]$ are in fact measured in the last frame period). For example, at time $t = kT + t_1$ ($t_1 = 0$), if $y(kT)$ is not available, i.e., $t_1^1 > 0$, then the new lifted output $\underline{y}^1[k]$ is defined by

$$\underline{y}^1[k] = \begin{bmatrix} y((k-1)T + t_1^1) \\ \vdots \\ y((k-1)T + t_m^{n_m}) \end{bmatrix}, \quad (3.27)$$

where all the elements are listed in their order of occurrence; if $t_1^1 = 0$, that is, $y(kT)$ happens to be available, then

$$\underline{y}^1[k] = \begin{bmatrix} y((k-1)T + t_1^2) \\ \vdots \\ y((k-1)T + t_m^{n_m}) \\ y(kT + t_1^1) \end{bmatrix}. \quad (3.28)$$

Similarly we can define $\underline{y}^i[k]$ corresponding to the control move at $t = kT + t_i$. It is relatively easy to derive a model relating $\underline{y}^i[k]$ to $\Delta \underline{u}[k]$; thus we can solve the standard GPC problem using this model and a similar cost function as in (3.15), implementing $\Delta u(kT + t_i)$ only. Because of the way the chain of new lifted output signals and new models are constructed, it is clear that the optimal $\Delta u(kT + t_i)$, $i = 1, 2, \dots, m$, are causal.

Denote the model from $\Delta \underline{u}[k]$ to $\underline{y}^i[k]$ by M^i ; it can be derived from that in (3.16). For example, to find M^1 , we rewrite (3.16) as

$$\underline{y}[k-1] = \frac{1}{d} N \Delta \underline{u}[k-1] = \frac{1}{d} (q^{-1} N) \Delta \underline{u}[k],$$

Then in the case of $\underline{y}^1[k]$ in (3.27), we have

$$\underline{y}^1[k] = \frac{1}{d} N^1 \Delta \underline{u}[k], \quad N^1 = \begin{bmatrix} q^{-1} N_{11} & \cdots & q^{-1} N_{1m} \\ \vdots & \cdots & \vdots \\ q^{-1} N_{p1} & \cdots & q^{-1} N_{pm} \end{bmatrix}.$$

(For the case of $\underline{y}^1[k]$ in (3.28), the change is obvious.)

Thus, in one frame period T , a chain of m models are obtained. For each model, follow the procedure in Section 3.3.1 to compute the optimal $\Delta \underline{u}^i[k]$:

$$\Delta \underline{u}^i[k] = \begin{bmatrix} \Delta u^i(kT + t_1) \\ \vdots \\ \Delta u^i(kT + t_i) \\ \vdots \\ \Delta u^i(kT + t_m) \end{bmatrix}.$$

However, only the i -th element $\Delta u^i(kT + t_i)$ is implemented. As time goes (i increases from 1 to m), the actual lifted incremental control signal implemented in the k -th frame period is

$$\Delta \underline{u}^*[k] = \begin{bmatrix} \Delta u^1(kT + t_1) \\ \Delta u^2(kT + t_2) \\ \vdots \\ \Delta u^m(kT + t_m) \end{bmatrix}. \quad (3.29)$$

It is clear from the discussions that the proposed GPC control algorithm has the following properties:

1. It is based on a chain of lifted models derived from the original model.
2. The control law is periodic with period T .
3. The control law is causal and therefore can be implemented in real time.

Closed-loop expression for the optimal and causal control in (3.29) can be derived. Consider the frame interval $[kT, (k+1)T)$, based on (3.26), $\Delta u^i(kT + t_i)$ in (3.29) has the following form

$$\Delta u^i(kT + t_i) = K_1^i W[k] + K_2^i \Delta U_p[k] + K_3^i Y_p^i[k], \quad (3.30)$$

where

$$Y_p^i[k] = \begin{bmatrix} \underline{y}^i[k] \\ \vdots \\ \underline{y}^i[k-l+1] \end{bmatrix},$$

K_1^i , K_2^i , and K_3^i are $1 \times pN$, $1 \times m(l-1)$, and $1 \times pl$ matrices, respectively, all depending on the i -th model M^i . Partition these matrices as follows:

$$\begin{aligned} K_1^i &= \begin{bmatrix} K_{11}^i & \cdots & K_{1N}^i \end{bmatrix}, \\ K_2^i &= \begin{bmatrix} K_{21}^i & \cdots & K_{2(l-1)}^i \end{bmatrix}, \\ K_3^i &= \begin{bmatrix} K_{31}^i & \cdots & K_{3l}^i \end{bmatrix}. \end{aligned}$$

Assume that the future reference signal is a constant w along the horizon, and define:

$$F^i = \sum_{j=1}^N \sum_{k=1}^p K_{1j}^i(1, k), \quad G^i = \sum_{j=1}^{l-1} q^{-j} K_{2j}^i, \quad H^i = \sum_{j=1}^l q^{-j+1} K_{3j}^i, \quad (3.31)$$

where $K_{1j}^i(1, k)$ represents the element in the k th column of K_{1j}^i , and q^{-1} is the backward shift operator (on the lifted signals). Then equation (3.30) can be written as

$$\Delta u^i(kT + t_i) = F^i w + G^i \Delta \underline{u}^*[k] + H^i \underline{y}^i[k].$$

Let i vary from 1 to m , and note that

$$\underline{y}^i[k] = O^i \underline{y}[k], \quad (3.32)$$

$$O^i = \left[\begin{array}{c|c} 0 & \text{diag}\{q^{-1}, \dots, q^{-1}\}_{p-n_0-\dots-n_{i-1}} \\ \hline I_{n_0+\dots+n_{i-1}} & 0 \end{array} \right], \quad (n_0 = 1) \quad (3.33)$$

where we have assumed that $t_i^1 = t_i$, i.e., the output signal is available right at the time instant $kT + t_i$, when the control input is updated. Thus we obtain the closed-loop expression for the optimal control in (3.29):

$$\Delta \underline{u}^*[k] = \Psi^{-1} \cdot [\Theta w + \Xi \underline{y}[k]], \quad (3.34)$$

where

$$\Psi = I_m - \begin{bmatrix} G^1 \\ \vdots \\ G^m \end{bmatrix}, \quad \Theta = \begin{bmatrix} F^1 \\ \vdots \\ F^m \end{bmatrix}, \quad \Xi = \begin{bmatrix} H^1 O^1 \\ \vdots \\ H^m O^m \end{bmatrix}. \quad (3.35)$$

(Ψ is assumed to be invertible.) The result can be summarized as follows.

Theorem 2 *Assume that the future reference signal is a constant w along the horizon. The optimal lifted control law is given by its closed-loop form in (3.34), where the term from $\underline{y}[k]$ to $\Delta \underline{u}^*[k]$, $\Psi^{-1} \Xi$, always satisfies the causality structure in (3.10).*

Proof First, we introduce an operator \aleph : If a system P is with the state-space representation $[A, B, C, D]$, then $\aleph(P) = D$.

By the definition of G^i and Ψ in (3.31) and (3.35), it is easy to see that

$$\aleph(\Psi) = \aleph(I_m) - \aleph\left(\begin{bmatrix} G^1 \\ \vdots \\ G^m \end{bmatrix}\right) = I_m - 0_m = I_m,$$

in this way, $\aleph(\Psi^{-1}) = I_m$.

Also, from H^i in (3.31) and O^i in (3.33), we have

$$\begin{aligned}\aleph(H^i O^i) &= \aleph(H^i) \aleph(O^i) \\ &= \left[\times \cdots \times \right]_{1 \times p} \left[\begin{array}{c|c} 0 & 0_{p-n_0-n_1-\dots-n_{i-1}} \\ \hline I_{n_0+n_1+\dots+n_{i-1}} & 0 \end{array} \right]_{p \times p} \\ &= \left[\begin{array}{cccc} \overbrace{\times \cdots \times}^{n_0+n_1+\dots+n_{i-1}} & 0 & \cdots & 0 \end{array} \right]_{1 \times p}.\end{aligned}$$

When i varies from 1 to m , the following result can be obtained:

$$\aleph(\Xi) = \left[\begin{array}{cccc} \overbrace{\star \ 0 \ \cdots \ 0}^{n_1} & & & \\ \times \ \cdots \ \times & \overbrace{\star \ 0 \ \cdots \ 0}^{n_2} & & \\ \vdots & & \ddots & \\ \times \ \cdots \ \times & \cdots & \cdots & \overbrace{\times \ \cdots \ \times}^{n_{m-1}} \ \overbrace{\star \ 0 \ \cdots \ 0}^{n_m} \end{array} \right]. \quad (3.36)$$

This is exactly the same as the causal structure proposed in (3.10). Since $\aleph(\psi^{-1}\Xi) = \aleph(\Psi^{-1})\aleph(\Xi) = \aleph(\Xi)$, we conclude that the new algorithm presented in this paper can always result in a causal GPC controller. It needs to be mentioned that this $m \times p$ matrix will have a little bit change when the assumption $t_i^1 = t_i$ (this is made when O^i is defined) is not true for every $i \in \{1, \dots, m\}$. Some \star s (or maybe all \star s) will be 0. However, no matter what assumptions are made on t_i and t_i^j , the resulted GPC controller is no doubt always causal. Q.E.D

3.4 Example

In this section, we illustrate the GPC algorithm proposed in Section 3.3.2 by an example. Consider a SISO continuous-time model:

$$G(s) = \frac{0.0039(s + 0.7294)}{(s + 0.0708)(s + 0.0042)}. \quad (3.37)$$

We assume that the control signal is updated every 8 sec, while the output signal is sampled every 12 sec; thus the frame period is $T = 24$ sec. Such a multirate system is a special case of the non-uniformly sampled systems discussed in this paper, so all the results in Sections 3.2 and 3.3 are directly applicable. (To the best of our knowledge, no GPC solution exists for such multirate systems in the lifted domain, producing causal control laws.)

First, let us check controllability and observability of the lifted control system. For the continuous-time model in (3.37), a minimal realization is:

$$A = \begin{bmatrix} -0.0750 & -0.0003 \\ 1.0000 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0.0039 & 0.0028 \end{bmatrix}, \quad D = 0.$$

Following Proposition 1, we can obtain the discrete-time lifted model for the multirate system:

$$\underline{A} = \begin{bmatrix} 0.1373 & -0.0032 \\ 10.8301 & 0.9496 \end{bmatrix}, \quad \underline{B} = \begin{bmatrix} 1.6277 & 3.2055 & 5.9969 \\ 80.9051 & 62.1705 & 26.4152 \end{bmatrix},$$

$$\underline{C} = \begin{bmatrix} 0.0039 & 0.0028 \\ 0.0239 & 0.0028 \end{bmatrix}, \quad \underline{D} = \begin{bmatrix} 0 & 0 & 0 \\ 0.1511 & 0.0341 & 0 \end{bmatrix}.$$

It is easy to see that $(\underline{A}, \underline{B})$ is controllable and $(\underline{C}, \underline{A})$ is observable. This is also true by Theorem 1, because $T = 24$ sec is non-pathological.

Next we will design a multirate GPC controller for the uniformly sampled system. Notice here that in one frame period $T = 24$ sec, the control input is updated 3 times. According to the results in Section 3.3.2, a chain of 3 lifted models should be defined. However, in this example (and many industrial processes), the output is sampled at a slower rate: The first sample is taken at $t = kT$ sec, and the next is not available until time instant $t = kT + 12$ sec. Thus the first two incremental control moves, $\Delta u(kT)$ and $\Delta u(kT + 8)$, can be computed together in one time by constructing $\underline{y}^1[k]$; and the third incremental control moves $\Delta u(kT)$, will be computed separately by constructing $\underline{y}^2[k]$. For an illustration, see Figure 3.2.

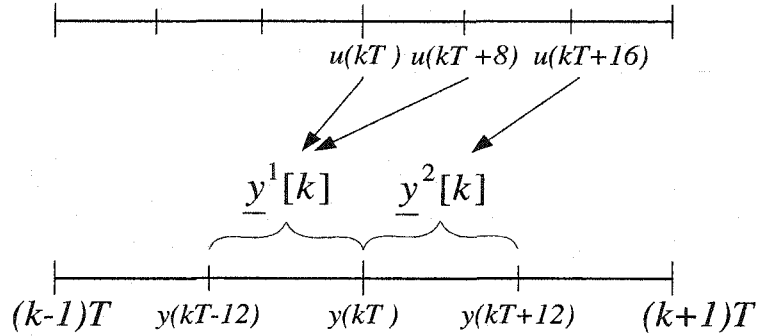


Figure 3.2: An illustration of the design

So in this example, only two lifted models need to be used. These two lifted models, including the integrator, in the input-output framework are

$$\underline{y}^1[k] = \frac{1}{d} N^1 \Delta \underline{u}[k], \quad N^1 = \begin{bmatrix} N_1^1 & N_2^1 & N_3^1 \end{bmatrix},$$

$$\underline{y}^2[k] = \frac{1}{d} N^2 \Delta \underline{u}[k], \quad N^2 = \begin{bmatrix} N_1^2 & N_2^2 & N_3^2 \end{bmatrix},$$

where

$$\underline{y}^1[k] = \begin{bmatrix} y(kT - 12) \\ y(kT) \end{bmatrix}, \quad \underline{y}^2[k] = \begin{bmatrix} y(kT) \\ y(kT + 12) \end{bmatrix}, \quad \Delta \underline{u}[k] = \begin{bmatrix} \Delta u(kT) \\ \Delta u(kT + 8) \\ \Delta u(kT + 16) \end{bmatrix},$$

and

$$\begin{aligned} d &= 1 - 2.0863q^{-1} + 1.2516q^{-2} - 0.1653q^{-3}, \\ N_1^1 &= \begin{bmatrix} 0.2801q^{-1} + 1.0901q^{-2} + 0.1053q^{-3} \\ 0.9131q^{-1} + 0.5612q^{-2} + 0.0011q^{-3} \end{bmatrix}, \\ N_2^1 &= \begin{bmatrix} 0.0338q^{-1} + 1.0887q^{-2} + 0.3529q^{-3} \\ 0.4666q^{-1} + 0.9642q^{-2} + 0.0446q^{-3} \end{bmatrix}, \\ N_3^1 &= \begin{bmatrix} 0.6803q^{-2} + 0.7828q^{-3} + 0.0123q^{-4} \\ 0.1311q^{-1} + 1.1405q^{-2} + 0.2038q^{-3} \end{bmatrix}, \\ N_1^2 &= \begin{bmatrix} 0.9131q^{-1} + 0.5612q^{-2} + 0.0011q^{-3} \\ 0.2801 + 1.0901q^{-1} + 0.1053q^{-2} \end{bmatrix}, \\ N_2^2 &= \begin{bmatrix} 0.4666q^{-1} + 0.9642q^{-2} + 0.0446q^{-3} \\ 0.0338 + 1.0887q^{-1} + 0.3529q^{-2} \end{bmatrix}, \\ N_3^2 &= \begin{bmatrix} 0.1311q^{-1} + 1.1405q^{-2} + 0.2038q^{-3} \\ 0.6803q^{-1} + 0.7828q^{-2} + 0.0123q^{-3} \end{bmatrix}. \end{aligned}$$

The purpose of the GPC design is to minimize the cost function discussed in Section 3.3.2, where the tuning parameters are N_2 , N_u and λ . If we choose

$$N_2 = 6, \quad N_u = 5, \quad \lambda = 0.1, \quad (3.38)$$

the tracking performance of the closed loop with the multirate GPC controller can be simulated, see the solid lines in Figure 3.3.

Both the process output and the control input are shown in Figure 3.3. The simulation time is 800 sec, and the setpoint changes from 0 to 1 at $t = 0$, from 1 to 0 at $t = 203$ sec, and from 0 to 3 at $t = 600$ sec.

Tuning the parameters N_2 , N_u and λ can affect tracking performance, as well as closed-loop robustness. For example, increasing λ – the weighting factor on the control signal – from 0.1 to 5 will result in decrease in the maximum value of the control input, but the price paid is that the tracking becomes more sluggish, see the dotted lines in Figure 3.3. We can also tune N_2 or N_u to be a larger or a smaller number, say, $N_2 = 10$ or $N_u = 1$ to make the closed-loop system more robust in the presence of model-plant mismatch, see, e.g., [8].

Finally, with the tuning parameters in (3.38), we can compute the closed-loop form of the multirate GPC controller. According to (3.30) to (3.34), the lifted optimal control law

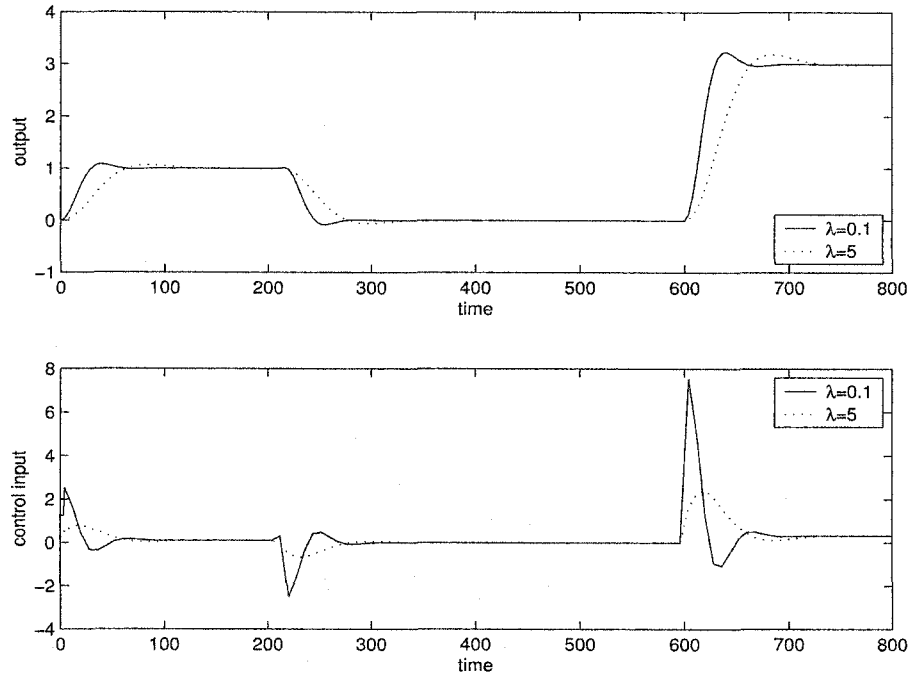


Figure 3.3: Tracking performance of the closed-loop with the multirate GPC controller

is

$$\Delta \underline{u}^*[k] = \begin{bmatrix} \Delta u^1(kT) \\ \Delta u^1(kT + 8) \\ \Delta u^2(kT + 16) \end{bmatrix} = \Psi^{-1} [\Theta w + \Xi \underline{y}[k]].$$

The direct feedthrough term from $\underline{y}[k]$ to $\Delta \underline{u}^*[k]$ is $\Psi^{-1}\Xi$, which is computed to be

$$\Psi^{-1}\Xi = \begin{bmatrix} 1.1321 & 0 \\ 0.0369 & 0 \\ -1.5729 & 1.4937 \end{bmatrix}.$$

This matrix is block lower triangular, satisfying the causality constraint in this case.

3.5 Conclusions

In this chapter, we studied non-uniformly sampled systems, which are characterized by a non-uniform but periodic pattern for control updating and output sampling. Periodicity allows lifted and LTI models to be derived. Moreover, we gave a sufficient condition for the lifted models to inherit properties such as controllability and observability from continuous time.

We also dealt with the GPC problems for non-uniformly sampled systems. It is emphasized that when the lifting technique is used, the so-called causality constraint on lifted

controllers should be taken into account in design. Starting from the conventional GPC design for MIMO systems, we proposed an approach which can handle the causality constraint in the design of GPC controllers.

Chapter 4

Multirate GPC for Sampled-Data Systems

4.1 Introduction

Generalized predictive control (GPC) was first proposed by Clarke *et al.* in 1987 [24, 25]. The control algorithms derived there were based on discrete-time models using a discrete-time performance index. Because discrete-time control algorithms in fact are operating with A/D and D/A converters in a continuous-time environment, poor inter-sample behavior may arise due to the fact that the design is based on performance solely at the sampling instants. To improve inter-sample performance, several pieces of work have been done. Demircioglu and Gawthrop proposed the continuous-time generalized predictive controller (CGPC) in 1991 using continuous-time performance index and implementing continuous-time control law [32]; however it is difficult to implement the continuous-time scheme using a digital computer. Then in 1995, Lauritsen *et al.* proposed to implement continuous-time GPC algorithms using the delta operator [53]; however in the case of a slow sampling rate, the delta operator brings about poor performance. Moreover, in these methods, the predictors are approximated by a truncated Maclaurin series. Therefore, in the case of larger prediction horizon, the approximation error becomes large. Basically, these ways of getting a discrete-time control law works well only when the prediction horizon is not too large or the sampling rate is fast; otherwise poor performance may result.

A preferred approach is the *sampled-data design problem*: Design discrete-time controllers directly based on continuous-time performance criterion [20]. This approach has the advantage of capturing the inter-sample behavior and obtaining optimal discrete-time controllers directly in design. Such sampled-data design problems were considered by Masuda *et al.* in 1997 in the GPC framework [62] and also by Chai *et al.* in 1999 in the

model predictive control (MPC) framework [17], both in a single-rate setting. Their solutions have a common feature: By converting continuous-time performance indices into equivalent discrete-time ones, they reduced the sampled-data design problems to equivalent discrete-time design problems, and hence solvable in discrete time.

The goal in this chapter is to extend their solutions to multirate sampled-data systems where different A/D and D/A converters adopt different operating rates, due to practical constraints. For simplicity, we consider a useful case where the D/A conversion is an integer multiple faster than the A/D conversion, i.e., the manipulated signals are adjusted faster, while the output measurements are sampled slower. Such dual-rate systems arise in process control industry when physical constraints limit the sampling rate of the output, yet the manipulated signal can be updated faster. We will see later that this multirate control scheme delivers better performance than both the conventional discrete-time multirate scheme, and the sampled-data single-rate scheme operating at the slow sampling rate.

This chapter is organized as follows. In Section 4.2 we introduce the multirate sampled-data system we will study and propose the associated GPC problem, taking into account the inter-sample behavior. In Section 4.3 we reduce the multirate GPC problem to a least square problem and derive an explicit solution. In Section 4.4 we give an illustrative example and compare the sampled-data multirate GPC algorithm with: (1) the optimal discrete-time multirate GPC algorithm and (2) the optimal sampled-data single-rate GPC algorithms operating at the fast and slow rates; the results show the advantage of the multirate control scheme. Finally, in Section 4.5 we offer some concluding remarks.

4.2 Problem formulation

Consider the SISO sampled-data system shown in Figure 4.1, an idealized representation of

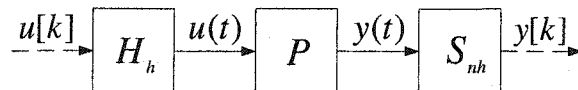


Figure 4.1: Dual-rate sampled-data open-loop system

the practical situation involving fast D/A conversion and slow A/D conversion. Here, P is a continuous-time LTI process with the following state-space representation:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t); \end{aligned} \quad (4.1)$$

H_h is a fast zero-order hold with period h ; and S_{nh} is a slow sampler with period nh (n is an integer). The continuous-time signals involved in Figure 4.1, $y(t)$ and $u(t)$, are

process output and control input, respectively; the discrete-time signals, $u[k]$ and $y[k]$, are digital controller output and sampled output, respectively. In such a dual-rate system, $y[k]$ is available only once every slow period nh ; the control signal $u[k]$ is updated every fast period h ; and the corresponding continuous-time signal $u(t)$ is given by

$$u(t) = u[k], \quad kh \leq t < (k+1)h. \quad (4.2)$$

To capture inter-sample behavior, we use the continuous-time GPC performance criterion

$$J(k) = \int_{(k+N_1)T_s}^{(k+N_2)T_s} [y(\tau) - y_m(\tau)]^2 d\tau + \int_{kT_s}^{(k+N_u)T_s} \lambda [\Delta u(\tau)]^2 d\tau. \quad (4.3)$$

Note here that all the time frames in $J(k)$ are multiples of $T_s = nh$, the slow sampling period; N_1 , N_2 and N_u are the minimum, maximum prediction horizons and control horizon, respectively. (Later, we assume $N_1 = 1$ and $N_2 = N_u = N$.) The positive constant λ is the control weighting. The signal $y_m(t)$ is the reference and is defined to be a step-wise function:

$$y_m(t) = y_m[k], \quad kT_s \leq t < (k+1)T_s. \quad (4.4)$$

In the following, $y_m[k]$ will be chosen as constant for $k \geq 0$, i.e., the control objective is set-point tracking. Finally in (4.3), Δ is the differencing operator:

$$\Delta u(t) = \Delta u[k] = u[k] - u[k-1], \quad kh \leq t < (k+1)h. \quad (4.5)$$

The *SD multirate GPC design problem* is to find a discrete-time control sequence $u[k]$ (or $\Delta u[k]$) to minimize the continuous-time J in (4.3).

To solve the SD GPC design problem, we need a discretized model for P . To handle the dual-rate feature, first we introduce a discrete-time system S_n , the down-sampler by a factor of n , defined as

$$y = S_n x \Leftrightarrow y(k) = x(nk). \quad (4.6)$$

It is easy to verify that $S_{nh} = S_n S_h$; and this fact together with the discretization method in [20] leads from Figure 4.1 to a discrete-time system as shown in Figure 4.2, where P_h is

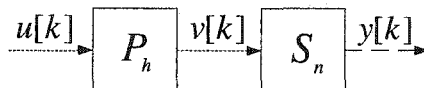


Figure 4.2: Dual-rate discrete-time open-loop system

the discretized P at the sampling period h . The discrete-time system P_h has the following

state-space form (see, e.g., [20]):

$$\begin{aligned} x[k+1] &= A_h x[k] + B_h u[k] \\ v[k] &= C x[k], \end{aligned} \quad (4.7)$$

where $x[k] = x(kh)$, $v[k]$ is the output of P_h , and

$$A_h = e^{Ah}, \quad B_h = \int_0^h e^{A\sigma} d\sigma \cdot B. \quad (4.8)$$

Note that the discrete-time model from $u[k]$ to $y[k]$ in Figure 4.2 is time-varying, because of the presence of the down-sampler S_n . In order to get an LTI model for the purpose of design later, we employ the lifting technique.

Lift u to get the slow-rate signal \underline{u} , as shown in Figure 4.3, and compress the three systems L_n^{-1} , P_h and S_n into one: \underline{P}_h ; it is easy to see that the lifted plant \underline{P}_h is LTI [20] and can be represented by the following state-space model:

$$\begin{aligned} \underline{x}[k+1] &= \underline{A} \underline{x}[k] + \underline{B} \underline{u}[k] \\ y[k] &= \underline{C} \underline{x}[k], \end{aligned} \quad (4.9)$$

where

$$\underline{A} = A_h^n, \quad \underline{B} = \begin{bmatrix} A_h^{n-1} B_h & A_h^{n-2} B_h & \cdots & A_h B_h & B_h \end{bmatrix}, \quad \underline{C} = C, \quad (4.10)$$

and A_h and B_h were defined in (4.8).

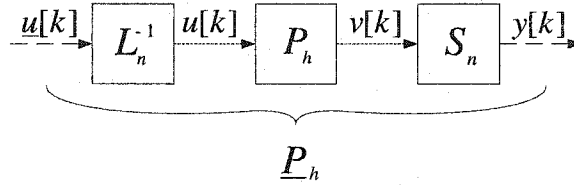


Figure 4.3: Lifted LTI and single-rate model

Note that \underline{P}_h is a LTI and single-rate model at the slow sampling period nh . Later we will convert the SD GPC design problem to a discrete-time lifted GPC problem and obtain an LTI lifted GPC algorithm. We remark that when the lifted GPC controller is designed, it computes $\underline{u}[k]$; and $\underline{u}[k]$ should be inverse lifted before it is implemented to the real process. Thus the inverse lifting operator together with the lifted controller can be thought of as the required controller for the original dual-rate system.

We also remark that the causality constraint (3.10) on the lifted controllers (which we have discussed in Chapter 3) will be automatically satisfied for dual-rate systems studied

here. Refer to Figure 4.3, the lifted controller inputs $y[k]$ and calculates $\underline{u}[k]$, where

$$y[k] = y(k \cdot nh), \quad \underline{u}[k] = \begin{bmatrix} u(k \cdot nh) \\ u(k \cdot nh + h) \\ \vdots \\ u(k \cdot nh + (n-1)h) \end{bmatrix}.$$

It is obvious that the future control signals only depend on information available up to time $k \cdot nh$, thus the design and analysis of the lifted controller for \underline{P}_h can adopt any results for single-rate systems, without the consideration of the causality constraint.

4.3 Dual-rate GPC design for sampled-data systems

The sampled-data GPC problem for single-rate systems was proposed and solved recently [62]; this digital controller resulted in better inter-sample behavior than conventional GPC design. In this section, we will extend this solution to the SD dual-rate GPC problem posed in Section 4.2. The whole procedure can be summarized into several steps.

4.3.1 Discretization at the fast rate

Imagine a fast single-rate system, we derive a GPC design problem using a discrete-time cost function which is equivalent to the continuous-time one in (4.3) [62].

With the assumption that $N_1 = 1$, and $N_2 = N_u = N$, J in (4.3) can be rewritten as follows:

$$\begin{aligned} J &= \int_{knh+nh}^{knh+nNh} [y(\tau) - y_m(\tau)]^2 d\tau + \int_{knh}^{knh+nNh} \lambda [\Delta u(\tau)]^2 d\tau \\ &= \sum_{j=n}^{nN-1} \left\{ \int_{(kn+j)h}^{(kn+j+1)h} [y(\tau) - y_m(\tau)]^2 d\tau \right\} \\ &\quad + \sum_{j=0}^{nN-1} \left\{ \int_{(kn+j)h}^{(kn+j+1)h} \lambda [\Delta u(\tau)]^2 d\tau \right\}. \end{aligned} \quad (4.11)$$

Then for the fast-rate GPC problem, $J(k)$ in (4.3) can be thought of as the performance index at time instant $t = (nk)h$. In this case, the minimum and maximum prediction horizons turn out to be $nN_1 = n$ and $nN_2 = nN$, respectively; and the control horizon is nN . Based on results in [62], define:

$$Y_m^f[k] = \begin{bmatrix} y_m^f[k] & y_m^f[k+1] & \cdots & y_m^f[k+nN-1] \end{bmatrix}^T, \quad (4.12)$$

where

$$y_m(t) = y_m^f[k], \quad kh \leq t < (k+1)h,$$

and

$$X[k] = \begin{bmatrix} x[k]^T & x[k+1]^T & \cdots & x[k+nN-1]^T \end{bmatrix}^T, \quad (4.13)$$

$$U[k] = \begin{bmatrix} u[k] & u[k+1] & \cdots & u[k+nN-1] \end{bmatrix}^T, \quad (4.14)$$

$$\Delta U[k] = \begin{bmatrix} \Delta u[k] & \Delta u[k+1] & \cdots & \Delta u[k+nN-1] \end{bmatrix}^T, \quad (4.15)$$

we get the following discrete-time equivalent cost function $J_f[k]$ at $t = kh$:

$$\begin{aligned} J_f[k] &= X[k]^T \Lambda_Q X[k] + 2X[k]^T \Lambda_S U[k] + RU[k]^T \Lambda_I U[k] \\ &\quad - 2X[k]^T \Lambda_M Y_m^f[k] - 2M_2 U[k]^T \Lambda_I Y_m^f[k] \\ &\quad + h Y_m^f[k]^T \Lambda_I Y_m^f[k] + \lambda h \Delta U[k]^T \Delta U[k]. \end{aligned} \quad (4.16)$$

Here

$$\Lambda_Q = V_p^T \begin{bmatrix} Q & & \\ & \ddots & \\ & & Q \end{bmatrix} V_p, \quad V_p = \begin{bmatrix} \overbrace{0 \cdots 0}^n & I & & \\ & & \ddots & \\ 0 & \cdots & 0 & I \end{bmatrix}, \quad (4.17)$$

$$\Lambda_S = V_p^T \begin{bmatrix} S & & \\ & \ddots & \\ & & S \end{bmatrix} V, \quad V = \begin{bmatrix} \overbrace{0 \cdots 0}^n & 1 & & \\ & & \ddots & \\ 0 & \cdots & 0 & 1 \end{bmatrix}, \quad (4.18)$$

$$\Lambda_I = V^T V, \quad \Lambda_M = V_p^T \begin{bmatrix} M_1^T & & \\ & \ddots & \\ & & M_1^T \end{bmatrix} V, \quad (4.19)$$

and

$$Q = \int_0^h A_d(\tau)^T C^T C A_d(\tau) d\tau, \quad M_1 = \int_0^h C A_d(\tau) d\tau, \quad (4.20)$$

$$R = \int_0^h B_d(\tau)^T C^T C B_d(\tau) d\tau, \quad M_2 = \int_0^h C B_d(\tau) d\tau, \quad (4.21)$$

$$S = \int_0^h A_d(\tau)^T C^T C B_d(\tau) d\tau, \quad (4.22)$$

with the definition that: $A_d(t) = e^{At}$, and $B_d(t) = \int_0^t e^{A\sigma} d\sigma \cdot B$. Notice here that $x[k]$, $u[k]$, $\Delta u[k]$ and $y_m^f[k]$ are all discrete-time signals at the fast rate (with period h): $x[k]$ and $u[k]$ appear in the fast-rate model (4.7), and $y_m^f[k]$ is the fast-rate reference signal; Q , R , S , M_1 and M_2 are matrices which can be calculated using matrix exponentials, see [17] and [20].

Replacing k in (4.13)-(4.16) by nk , we get that $J(k)$ in (4.3) is equal to $J_f[nk]$ as follows:

$$\begin{aligned} J_f[nk] &= X[nk]^T \Lambda_Q X[nk] + 2X[nk]^T \Lambda_S U[nk] + RU[nk]^T \Lambda_I U[nk] \\ &\quad - 2X[nk]^T \Lambda_M Y_m^f[nk] - 2M_2 U[nk]^T \Lambda_I Y_m^f[nk] \\ &\quad + h Y_m^f[nk]^T \Lambda_I Y_m^f[nk] + \lambda h \Delta U[nk]^T \Delta U[nk]. \end{aligned} \quad (4.23)$$

4.3.2 Lifting for conversion to the slow rate

Here we relate fast-rate signals to the lifted slow-rate ones. From the lifted plant model in (4.9) and the fast-rate system in (4.7), we have

$$\begin{aligned}\underline{x}[k] &= x[nk], \\ \underline{u}[k] &= \begin{bmatrix} u[nk] & u[nk+1] & \cdots & u[nk+n-1] \end{bmatrix}^T.\end{aligned}\quad (4.24)$$

Then the compact state $X[nk]$ in (4.23) can be written as

$$X[nk] = \Phi_A \underline{X}[k] + \Phi_B U[nk], \quad (4.25)$$

where

$$\underline{X}[k] = \begin{bmatrix} \underline{x}[k]^T & \underline{x}[k+1]^T & \cdots & \underline{x}[k+N-1]^T \end{bmatrix}^T, \quad (4.26)$$

$$\Phi_A = \begin{bmatrix} \Theta_A & & & \\ & \Theta_A & & \\ & & \ddots & \\ & & & \Theta_A \end{bmatrix}, \quad \Phi_B = \begin{bmatrix} \Theta_B & & & \\ & \Theta_B & & \\ & & \ddots & \\ & & & \Theta_B \end{bmatrix}, \quad (4.27)$$

$$\Theta_A = \begin{bmatrix} I \\ A_h \\ \vdots \\ A_h^{n-1} \end{bmatrix}, \quad \Theta_B = \begin{bmatrix} 0 & & & \\ B_h & 0 & & \\ A_h B_h & B_h & 0 & \\ \vdots & & \ddots & \ddots \\ A_h^{n-2} B_h & A_h^{n-3} B_h & \cdots & B_h & 0 \end{bmatrix}, \quad (4.28)$$

and furthermore, the compact fast-rate future control $U[nk]$ in (4.23) equals to

$$U[nk] = \begin{bmatrix} \begin{bmatrix} u[nk] \\ \vdots \\ u[nk+n-1] \end{bmatrix} \\ \begin{bmatrix} u[n(k+N-1)] \\ \vdots \\ u[nk+nN-1] \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \underline{u}[k] \\ \underline{u}[k+1] \\ \vdots \\ \underline{u}[k+N-1] \end{bmatrix} = \underline{U}[k]. \quad (4.29)$$

As to the compact reference signal $Y_m^f[nk]$ in (4.23), since the control objective is to follow a constant future setpoint, the following fact exists:

$$Y_m^f[nk] = \Phi_C Y_m[k], \quad (4.30)$$

where

$$Y_m[k] = \begin{bmatrix} y_m[k] \\ y_m[k+1] \\ \vdots \\ y_m[k+N-1] \end{bmatrix}, \quad \Phi_C = \begin{bmatrix} 1 & & & & & \\ \vdots & & & & & \\ 1 & & & & & \\ & 1 & & & & \\ & \vdots & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ & & & & 1 & \\ & & & & \vdots & \\ & & & & 1 & \end{bmatrix}. \quad (4.31)$$

Note here that $y_m[k]$ is the slow-rate reference signal in (4.4).

4.3.3 Lifting the cost function

Now we obtain the equivalent discrete-time cost function $J_l[k]$ corresponding to the lifted signals. We have converted the continuous-time $J(k)$ in (4.3) into an equivalent discrete-time ($J_f[nk]$) in Section 4.3.1; in Section 4.3.2 we have also found the relationships between the fast-rate signals and their lifted slow-rate ones. The SD dual-rate design problem is now equivalent to a discrete-time design problem with cost function $J_l[k]$, which can be derived as follows:

1. Use (4.9) repeatedly to obtain the following compact lifted state predictor:

$$\underline{X}[k] = \Phi_l \underline{x}[k] + \Gamma_l U[k], \quad (4.32)$$

with

$$\Phi_l = \begin{bmatrix} I \\ \underline{A} \\ \vdots \\ \underline{A}^{N-1} \end{bmatrix}, \quad \Gamma_l = \begin{bmatrix} 0 & 0 & 0 \\ \underline{B} & 0 & 0 \\ \underline{A}\underline{B} & \underline{B} & 0 \\ \vdots & \ddots & \ddots \\ \underline{A}^{N-2}\underline{B} & & \underline{B} & 0 \end{bmatrix}. \quad (4.33)$$

2. Substitute (4.32) into (4.25) to get

$$X[nk] = \Phi \underline{x}[k] + \Gamma U[k], \quad (4.34)$$

with

$$\Phi = \Phi_A \Phi_l, \quad \Gamma = \Phi_A \Gamma_l + \Phi_B. \quad (4.35)$$

3. Replacing $X[nk]$, $U[nk]$, and $Y_m^f[nk]$ in (4.23) with (4.34), (4.29) and (4.30), respectively, we get the equivalent discrete-time cost function:

$$\begin{aligned}
J_l[k] &= (\Phi \underline{x}[k] + \Gamma \underline{U}[k])^T \Lambda_Q (\Phi \underline{x}[k] + \Gamma \underline{U}[k]) \\
&\quad + 2(\Phi \underline{x}[k] + \Gamma \underline{U}[k])^T \Lambda_S \underline{U}[k] + R \underline{U}[k]^T \Lambda_I \underline{U}[k] \\
&\quad - 2(\Phi \underline{x}[k] + \Gamma \underline{U}[k])^T \Lambda_M \Phi_C Y_m[k] - 2M_2 \underline{U}[k]^T \Lambda_I \Phi_C Y_m[k] \\
&\quad + h Y_m[k]^T \Phi_C^T \Lambda_I \Phi_C Y_m[k] + \lambda h \Delta \underline{U}[k]^T \Delta \underline{U}[k].
\end{aligned} \tag{4.36}$$

4.3.4 Optimal control law

We then minimize $J_l[k]$ over $\Delta \underline{U}[k]$ to derive the optimal SD dual-rate GPC algorithm.

Note that $\underline{U}[k] = \underline{U}[k-1] + \Delta \underline{U}[k]$. Setting the partial derivative of J_l in (4.36) with respect to $\Delta \underline{U}[k]$ to zero and solving for $\Delta \underline{U}[k]$, we get the optimal solution:

$$\Delta \underline{U}[k] = \left\{ (G + \lambda h I)^T \right\}^{-1} \left\{ -H \Phi \underline{x}[k] + F Y_m[k] - G^T \underline{U}[k-1] \right\}, \tag{4.37}$$

where we have defined

$$\begin{aligned}
G &= \Gamma^T \Lambda_Q \Gamma + \Gamma^T \Lambda_S + \Lambda_S^T \Gamma + R \Lambda_I, \\
H &= \Gamma^T \Lambda_Q^T + \Lambda_S^T, \\
F &= (\Gamma^T \Lambda_M + N \Lambda_I) \cdot \Phi_C.
\end{aligned} \tag{4.38}$$

Notice that only the first element of $\Delta \underline{U}[k]$ is implemented in practice, i.e., the optimal control law is actually

$$\Delta \underline{u}[k] = K \left\{ -H \Phi \underline{x}[k] + F Y_m[k] - G^T \underline{U}[k-1] \right\}, \tag{4.39}$$

where $K = \begin{bmatrix} I_n & 0 & \cdots & 0 \end{bmatrix} \left\{ (G + \lambda h I)^T \right\}^{-1}$.

To estimate the state $\underline{x}[k]$ in (4.39), full-order observers in [62] can be used here:

$$\begin{aligned}
\hat{\underline{x}}[k+1] &= \underline{A} \hat{\underline{x}}[k] + \underline{B} \underline{u}[k] + L(y[k] - \underline{C} \hat{\underline{x}}[k]) \\
&= (\underline{A} - L \underline{C}) \hat{\underline{x}}[k] + \underline{B} \underline{u}[k] + L y[k],
\end{aligned} \tag{4.40}$$

where L is the observer gain and $\underline{A} - L \underline{C}$ is chosen to be stable.

To apply the optimal solution to the original sampled-data dual-rate system, first we calculate

$$\underline{u}[k] = \underline{u}[k-1] + \Delta \underline{u}[k]; \tag{4.41}$$

then inverse lift $\underline{u}[k]$ to get

$$\{u[nk], u[nk+1], \dots, u[nk+n-1]\}. \tag{4.42}$$

These n fast control moves are applied in the time interval $[kT_s, (k+1)T_s)$ as follows:

$$u(t) = \begin{cases} u[nk], & nkh \leq t < nkh + h \\ u[nk + 1], & nkh + h \leq t < nkh + 2h \\ \vdots \\ u[nk + n - 1], & nkh + nh - h \leq t < nkh + nh. \end{cases} \quad (4.43)$$

Note that the whole optimization procedure is repeated every slow period $T_s = nh$.

4.4 A design example

In this section, we apply the sampled-data dual-rate GPC algorithm derived in Section 4.3 to an example studied in [62] in the single-rate case. We will compare its performance with (1) that using conventional discrete-time dual-rate scheme, and (2) those using SD single-rate schemes at the fast and slow rates. The continuous-time plant is a third-order system with a state-space model of the form in (4.1) with

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad C^T = \begin{bmatrix} 1 \\ -0.2 \\ 0 \end{bmatrix}. \quad (4.44)$$

4.4.1 Discrete-time dual-rate GPC scheme

This is the case where $y(t)$ is sampled with period nh and $u(t)$ is adjusted with period h , but we are using a discrete-time cost function – a conventional discrete-time approach. The problem is formulated as follows: Find the optimal $\Delta \underline{u}[k]$ (lifted) to minimize:

$$J_d = \sum_{j=1}^{N-1} \{y[k+j] - y_m[k+j]\}^2 + \sum_{j=1}^N \lambda \{\Delta \underline{u}[k+j-1]\}^2, \quad (4.45)$$

Note that in (4.45), the weighting elements, minimum and maximum prediction horizons, and control horizon are all chosen to be the same as those in the continuous-time case in (4.3). $y[k]$ is the slow-rate sampled output in (4.9), $y_m[k]$ is the slow-rate reference signal in (4.4), and $\Delta \underline{u}[k]$ is the lifted incremental control signal.

The solution can be derived similarly to what Masuda *et al.* did in [62], except that here the discrete-time model is a lifted one. Define

$$Y_m[k] = \begin{bmatrix} y_m[k] & \cdots & y_m[k+N-1] \end{bmatrix}^T, \quad (4.46)$$

$$\underline{U}[k] = \begin{bmatrix} \underline{u}[k]^T & \cdots & \underline{u}[k+N-1]^T \end{bmatrix}^T, \quad (4.47)$$

the optimal control law is given as follows:

$$\Delta \underline{u}[k] = K_d \{-H_d \Phi_l \underline{x}[k] + F_d Y_m[k] - G_d \underline{U}[k-1]\}, \quad (4.48)$$

where

$$\begin{aligned}
K_d &= \begin{bmatrix} I_n & 0 & \cdots & 0 \end{bmatrix} (G_d + \lambda I)^{-1} \\
G_d &= \Gamma_l^T (V_p^d)^T \Lambda_Q^d V_p^d \Gamma_l, \\
H_d &= \Gamma_l^T (V_p^d)^T \Lambda_Q^d V_p^d, \\
F_d &= \Gamma_l^T (V_p^d)^T \Lambda_S^d V_p^d,
\end{aligned}$$

with

$$V_p^d = \begin{bmatrix} 0 & I & & \\ & & \ddots & \\ 0 & & & I \end{bmatrix}, \quad V^d = \begin{bmatrix} 0 & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{bmatrix}, \quad (4.49)$$

$$\Lambda_Q^d = \begin{bmatrix} C^T C & & & \\ & \ddots & & \\ & & & C^T C \end{bmatrix}, \quad \Lambda_S^d = \begin{bmatrix} C^T & & & \\ & \ddots & & \\ & & & C^T \end{bmatrix}. \quad (4.50)$$

Notice that two matrices: Φ_l and Γ_l , have been defined in (4.32); and the state $\underline{x}[k]$ is estimated by the observer in (4.40).

4.4.2 Sampled-data single-rate GPC schemes

For the model described by (4.1) with parameters in (4.44), two SD single-rate GPC algorithms are designed: One is the fast-rate case where the sampling period is $h = 4.0$ sec; the other is the slow-rate case where the sampling period is nh ; for simplicity, we choose $n = 2$. Both schemes use the continuous-time cost function in (4.3) at $t = (nk)h = k(nh)$. The fast-rate case has been discussed in Section 4.3.

For the slow-rate case, we can write down the equivalent discrete-time cost function as follows [62]:

$$\begin{aligned}
J_s[k] &= X_s[k]^T \Lambda_Q^s X_s[k] + 2X_s[k]^T \Lambda_S^s U_s[k] + R_s U_s[k]^T \Lambda_I^s U_s[k] \\
&\quad - 2X_s[k]^T \Lambda_M^s Y_m^s[k] - 2N_s U_s[k]^T \Lambda_I^s Y_m^s[k] \\
&\quad + nh Y_m^s[k]^T \Lambda_I^s Y_m^s[k] + \lambda nh \Delta U_s[k]^T \Delta U_s[k].
\end{aligned} \quad (4.51)$$

Here

$$X_s[k] = \begin{bmatrix} x_s[k]^T & \cdots & x_s[k+N-1]^T \end{bmatrix}^T, \quad (4.52)$$

$$U_s[k] = \begin{bmatrix} u_s[k] & \cdots & u_s[k+N-1] \end{bmatrix}^T, \quad (4.53)$$

$$Y_m^s[k] = \begin{bmatrix} y_m^s[k] & \cdots & y_m^s[k+N-1] \end{bmatrix}^T, \quad (4.54)$$

$$\Delta U_s[k] = \begin{bmatrix} \Delta u_s[k] & \cdots & \Delta u_s[k+N-1] \end{bmatrix}^T, \quad (4.55)$$

and $y_m^s[k]$ is the slow-rate reference signal. The signals $x_s[k]$ and $u_s[k]$ appear in the slow-rate discretized model of (4.1):

$$\begin{aligned} x_s[k+1] &= A_s x_s[k] + B_s u_s[k] \\ y_s[k] &= C x_s[k], \end{aligned} \quad (4.56)$$

with $A_s = e^{A \cdot nh}$, $B_s = \int_0^{nh} e^{A\sigma} d\sigma B$. Other matrices in (4.51) are defined as follows:

$$\Lambda_Q^s = (V_p^s)^T \begin{bmatrix} Q_s & & \\ & \ddots & \\ & & Q_s \end{bmatrix} V_p^s, \quad V_p^s = \begin{bmatrix} 0 & I & & \\ & & \ddots & \\ 0 & & & I \end{bmatrix}, \quad (4.57)$$

$$\Lambda_S^s = (V_p^s)^T \begin{bmatrix} S_s & & \\ & \ddots & \\ & & S_s \end{bmatrix} V^s, \quad V^s = \begin{bmatrix} 0 & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{bmatrix}, \quad (4.58)$$

$$\Lambda_M^s = (V_p^s)^T \begin{bmatrix} M_s^T & & \\ & \ddots & \\ & & M_s^T \end{bmatrix} V^s, \quad \Lambda_I^s = (V^s)^T V^s, \quad (4.59)$$

and

$$Q_s = \int_0^{nh} A_d(\tau)^T C^T C A_d(\tau) d\tau, \quad M_s = \int_0^{nh} C A_d(\tau) d\tau, \quad (4.60)$$

$$R_s = \int_0^{nh} B_d(\tau)^T C^T C B_d(\tau) d\tau, \quad N_s = \int_0^{nh} C B_d(\tau) d\tau, \quad (4.61)$$

$$S_s = \int_0^{nh} A_d(\tau)^T C^T C B_d(\tau) d\tau, \quad (4.62)$$

where $A_d(t) = e^{At}$, $B_d(t) = \int_0^t e^{A\sigma} d\sigma B$. Note that in the slow-rate case, the matrices Q_s , R_s , S_s , M_s and N_s are all computed by integrating over the time period $[0, nh]$ instead of $[0, h]$ in the fast-rate case.

The optimal control laws for SD single-rate GPC problems operating at the fast and slow rates can then be obtained by minimizing $J_f[k]$ and $J_s[k]$, respectively, and take similar forms to that in the SD dual-rate case [62]. The observer in (4.40) is used to give the desired state estimation. The responses of these GPC algorithms are simulated and compared below.

4.4.3 Simulation results

In the simulation, the GPC design parameters are chosen as follows: The control weighting is $\lambda = 0.01$; the minimum prediction horizon, the maximum prediction horizon, and the control horizon for both the slow single-rate and the dual-rate GPC are $N_1 = 1$, $N_2 = 5$, and $N = N_2 = 5$, respectively; for the fast single-rate case, they are all multiplied by the factor $n = 2$ to be compatible. For these tuning parameters, if we increase N_2 or decrease

N_u , we will get more robust closed-loop system in the presence of model-plant mismatch, as we have mentioned in Chapter 3. However, the purpose of this chapter is to improve the inter-sample behavior of the closed-loop system, and we will not pay much attention on how to tune these parameters. The poles of the observer are the same as those in [62]: 0.2, 0.3, and 0.4. We simulate the responses from 0 to 50 sec.

Step responses of the closed-loop system with the discrete-time dual-rate GPC scheme is shown in Figure 4.4, where small circles in the system output denote the sampled output. It is clear that considerable inter-sample ripples exist because of the use of the discrete-time cost function.

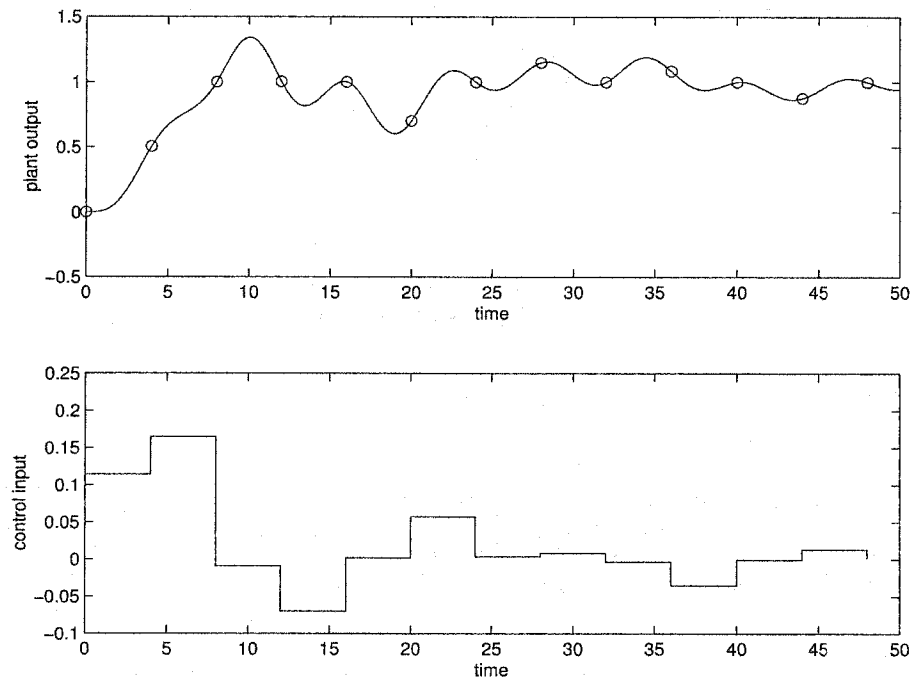


Figure 4.4: Tracking performance and control efforts with discrete-time dual-rate GPC

Step responses of the closed-loop system with the three SD GPC schemes (fast single-rate, dual-rate, and slow single-rate) are given in Figures 4.5 and 4.6. We see that the poor inter-sample behavior in Figure 4.4 is improved a great deal, especially by the fast and dual-rate SD algorithms.

Among the three SD schemes, the fast single-rate design has the best closed-loop tracking performance – the tracking is fast and the overshoot is small; while the slow single-rate design gives a sluggish response with large overshoot. The dual-rate design is not as good as the fast single-rate one, but it is much better than the slow single-rate one. These are also supported by the Table 4.1, where the three equivalent cost functions for the SD designs

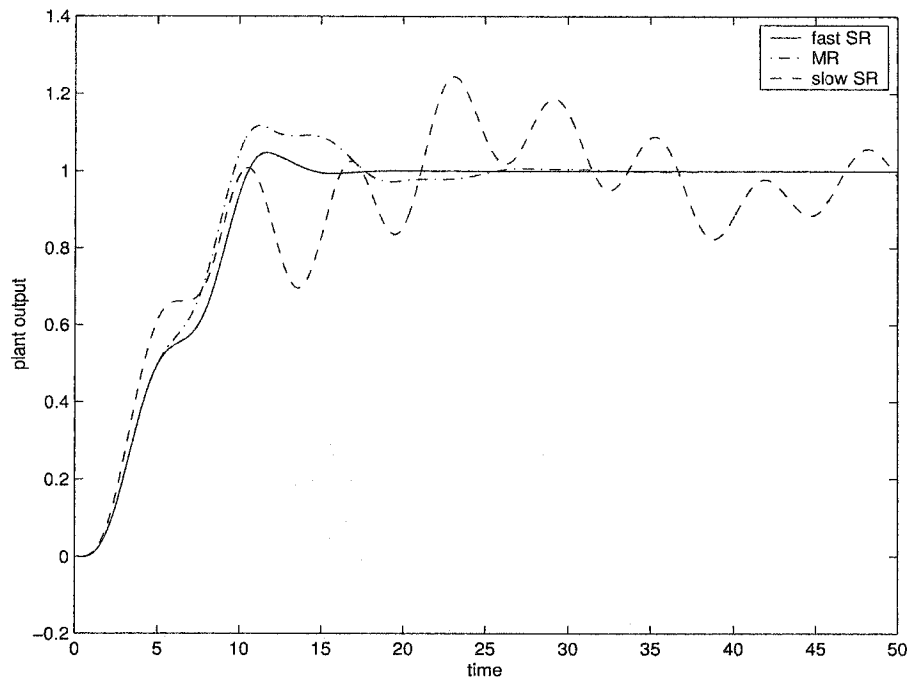


Figure 4.5: Comparison of the system tracking responses with the three SD schemes

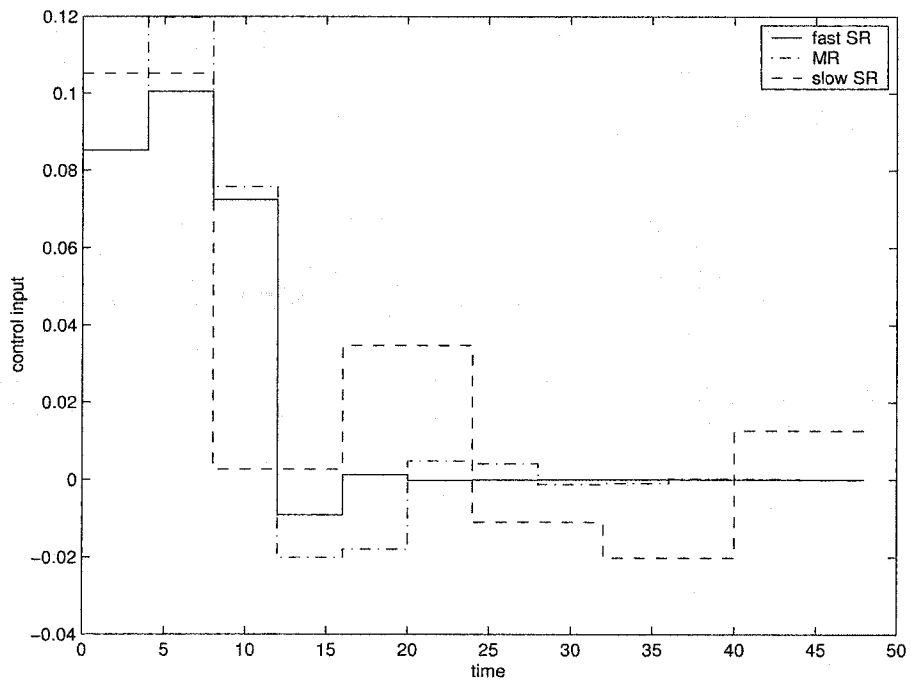


Figure 4.6: Comparison of the control efforts with the three SD schemes

are compared at some sampling instants. From table 4.1, the following inequality always

time k	J_f (fast SD)	J_l (dual-rate SD)	J_s (slow SD)
0	0.06575195080217	0.06575195080217	0.42563648582065
8	0.00030254662026	0.00471099387869	0.28238019454586
16	0.00000440701098	0.00059018694606	0.18125003123815
24	0.00000000109848	0.00003115873481	0.12713442722877
32	0.00000000001982	0.00000158270061	0.07792679268696
40	0.00000000000002	0.00000007477598	0.05648079587473
48	0.00000000000000	0.00000000349837	0.03413331387905

Table 4.1: Comparison of cost functions for the three SD designs

holds:

$$J_f \leq J_l < J_s. \quad (4.63)$$

But J_l is much closer to J_f than J_s .

We remark that in Table 4.1, the difference in the first row (where the sampling instant is 0) is caused by different sampled-data control laws only; the results in other rows are also affected by different initial conditions. Note that the inequality (4.63) exists for those data in the first row. It shows that in cases when practical constraints limit the sampling frequency, the dual-rate design is a much better choice than the slow single-rate design.

We also remark that inter-sample ripples may occur for other reasons: e.g., the observability of the open-loop system is lost due to sampling; or there are poorly damped zeros in the open-loop system that are canceled by the controller ([7]). In this example, the inter-sample ripple is caused by large sampling period (note that the base period is $h = 4$ sec). If a smaller base period, say, $h = 2$ sec is allowed, this inter-sample ripple in both the fast single-rate discrete time design and the dual-rate discrete time design will disappear. However, in chemical process applications, the availability of the output measurement is usually limited. For example, a composition analyzer in a distillation column can only be sampled in a slow rate. In such a case, the sampled-data design would be beneficial. And the dual-rate sampled-data design will result in satisfactory inter-sample behavior for systems where multirate scheme is employed and slow operating period is required.

4.5 Conclusions

This chapter proposed a sampled-data GPC scheme for a special case of multirate systems, where the input control rate is an integer multiple of the output sampling rate. The design example showed that the sampled-data dual-rate GPC scheme can effectively capture the

inter-sample behavior of continuous-time responses, and deal with the limitation on the output sampling rate imposed by possible physical constraints. Future work will focus on extension of this work to more general sampling situations, for which a causality condition (for more details on this, see, e.g., [19]) would arise in the design.

Chapter 5

On the Robust Stability of Dual-Rate GPC Systems

5.1 Introduction

Generalized predictive control (GPC) was proposed by Clarke and co-workers in 1987, and the control algorithm derived was in the polynomial domain [24, 25]. Subsequently, many pieces of work have been completed on the analysis and applications of GPC, for example, the work in [9] and [27]. Success of GPC in industrial applications, especially in the process control industry, is a strong motivation for continuous research in this area [11, 14].

State-space approach to GPC problems has the advantage of ease in treating multivariable systems and has been studied in a series of important work in [1], [118], [12], [69], [75] and [55]. One objective of our research is to derive a state-space GPC solution for multirate systems by using the lifting technique, with the causality condition being considered in the design of the lifted controllers. Note that a causal solution to multirate GPC problems has been presented in Chapter 3, however, it is in the transfer function framework and its state-space counterpart is still an open problem.

In this chapter, we will extend the state-space GPC algorithms developed by Ordys and Clarke [75] to the special fast-control slow-sampling dual-rate systems. This kind of systems have the property that the updating rate is an integer multiple of the sampling rate, and more importantly, they automatically satisfy the so-called causality constraint in the design of the lifted controllers (see Section 4.2). In the following, for simplicity, when we say *dual-rate* systems and controllers, we mean systems described here and controllers designed for them.

We emphasize that this chapter will focus on design and analysis of GPC problems for dual-rate systems. In particular, our objectives are three-fold:

- First, using the lifting technique, we derive a state-space solution to the GPC problem for dual-rate processes in the unconstrained case by extending methods in [75]; the solution is explicit in terms of matrix operations. Our starting point is a model for the process in consideration at the faster rate; i.e., we assume a single-rate model at the input control rate.
- Second, we study stability robustness of such dual-rate GPC controllers where there is model-plant mismatch (MPM), and give some sufficient robustness conditions; these conditions are in terms of the nominal model, the sampling ratio, and the dual-rate controller; and can be checked readily using norm computation in modern robust control theory.
- Finally, we illustrate the results derived in the paper on two examples and show that several meaningful conclusions can be obtained, e.g., if the input control rate is fixed, the robustness of the dual-rate GPC algorithm is improved as the sampling ratio decreases.

We remark that prior work on robustness of GPC algorithms in the single-rate setting exists, see, e.g., [91] and [8]. Both papers made use of the well-known small gain theorem; the former dealt with the robust design of the GPC observer pre-filters for mean-level and dead-beat performance; the later combined the spectral estimation of the model-plant uncertainty bound with robust design methods in the GPC framework. In this chapter, we will examine the stability robustness of the dual-rate GPC algorithms.

This chapter is organized as follows. Section 5.2 sets up the stage for the dual-rate sampled-data systems by importing lifting technique to convert dual-rate systems into single-rate systems. Section 5.3 formulates the dual-rate GPC problem and derives a solution in the state-space domain. Section 5.4 combines the lifting technique with the small gain theorem to analyze the closed-loop stability robustness of the dual-rate GPC controller when there is model-plant mismatch. Section 5.5 presents two illustrative examples for the results in Sections 5.3 and 5.4. Finally, in Section 5.6, concluding remarks are given.

5.2 Dual-rate sampled-data systems with noise

A dual-rate sampled-data system with noise is shown in Figure 5.1, where P and N are the LTI continuous-time process and noise model, respectively; H_h is the zero-order hold with period h , and S_{nh} is the sampler with period nh , with n being a positive integer; ω is the

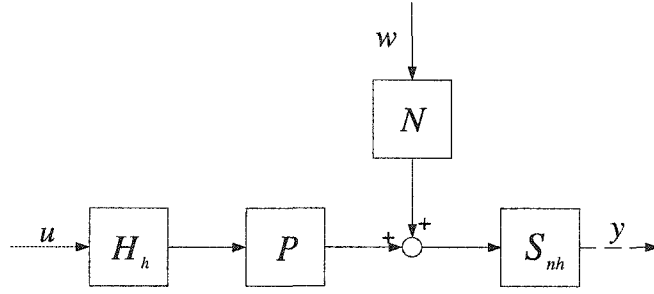


Figure 5.1: Dual-rate sampled-data open-loop system with noise

white noise in continuous time; and u and y are the control signal and output measurement in discrete time.

This typical fast-control, slow-sampling multirate system arises often in process control industry. And later we will show that such dual-rate systems usually lead to better performance than corresponding single-rate systems operating at the slow output sampling rate.

By introducing the down-sampler S_n defined in (4.6), together with the discretization method [20], Figure 5.1 is converted into a discrete-time system shown in Figure 5.2. Here,

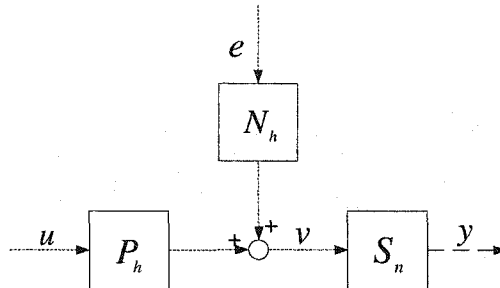


Figure 5.2: Dual-rate discretized system with noise

both P_h and N_h are single-rate discretizations at the sampling period h : P_h is the step-invariant transformation of P , and N_h the impulse-invariant transformation of N . The input to N_h , e , is a white noise in discrete time.

Now we will derive a single-rate discrete-time system from Figure 5.2 using lifting and down-sampling, and then close the loop and introduce the so-called lifted controller. Lift u to get the slow signal \underline{u} , as shown in Figure 5.3, and down-sample the white noise input e to get the noise model N_{nh} at the slow rate. Then bring in the lifted controller \underline{K} which inputs the slow sampled error signal and outputs the lifted control signal \underline{u} . This way we obtain the closed-loop configuration in Figure 5.3.

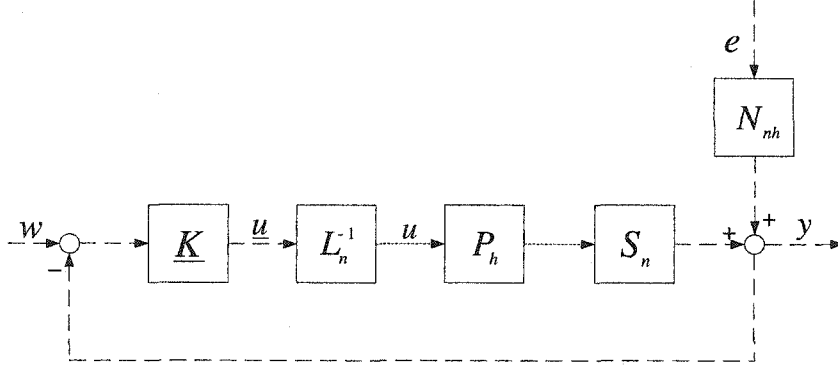


Figure 5.3: Lifted LTI closed-loop system

Note that \underline{K} is single-rate and to be designed, and hence we can require that \underline{K} be LTI. Compress the three systems L_n^{-1} , P_h , and S_n into one, the lifted process model; it is easy to see that this compressed system is also single-rate and LTI. This way we arrive at a closed-loop LTI system operating at the slow single rate nh .

5.3 Dual-rate GPC problem: a state-space solution

In this section, we formulate a dual-rate GPC problem and present a state-space solution. We will augment an integrator in the process model and the model from Δu to y is defined as $P_{dh} = P_h/\Delta$.

Let us start by obtaining state-space models for single-rate systems (such as P_{dh} , or, $n = 1$) modeled by the following CARIMA model [24]:

$$a(q^{-1})y[k] = b(q^{-1})u[k-1] + \frac{c(q^{-1})}{\Delta}e[k]. \quad (5.1)$$

Here, the sampling period is h ; $e[k]$ is assumed to be a Gaussian distributed white noise; and a , b and c are polynomials in q^{-1} :

$$\begin{aligned} a(q^{-1}) &= 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a}, \\ b(q^{-1}) &= b_0 + b_1q^{-1} + \dots + b_{n_b}q^{-n_b}, \\ c(q^{-1}) &= 1 + c_1q^{-1} + \dots + c_{n_c}q^{-n_c}. \end{aligned}$$

It has been shown in [7] and [75] that such a model has a canonical observable state-space realization. To write it down, let us multiply both sides of (5.1) by Δ and then by q raised to a suitable power; thus we obtain an equation involving polynomials of q , the forward shift operator:

$$\tilde{a}(q) \cdot y[k] = \tilde{b}(q) \cdot \Delta u[k] + \tilde{c}(q) \cdot e[k]. \quad (5.2)$$

Write $\tilde{a}(q)$, $\tilde{b}(q)$ and $\tilde{c}(q)$ as follows:

$$\begin{aligned}\tilde{a}(q) &= q^r + \tilde{a}_1 q^{r-1} + \cdots + \tilde{a}_r, \\ \tilde{b}(q) &= \tilde{b}_1 q^{r-1} + \cdots + \tilde{b}_r, \\ \tilde{c}(q) &= q^r + \tilde{c}_1 q^{r-1} + \cdots + \tilde{c}_r.\end{aligned}$$

Then the canonical observable state-space model is

$$\begin{aligned}x[k+1] &= Ax[k] + B\Delta u[k] + Fe[k] \\ y[k] &= Cx[k] + e[k],\end{aligned}$$

with

$$\begin{aligned}A &= \begin{bmatrix} 0 & 0 & 0 & \cdots & \cdots & -\tilde{a}_r \\ 1 & 0 & 0 & \cdots & \cdots & -\tilde{a}_{r-1} \\ 0 & 1 & 0 & \cdots & \cdots & -\tilde{a}_{r-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & -\tilde{a}_2 \\ 0 & 0 & 0 & \cdots & 1 & -\tilde{a}_1 \end{bmatrix}, \quad F = \begin{bmatrix} \tilde{c}_r - \tilde{a}_r \\ \tilde{c}_{r-1} - \tilde{a}_{r-1} \\ \tilde{c}_{r-2} - \tilde{a}_{r-2} \\ \vdots \\ \tilde{c}_2 - \tilde{a}_2 \\ \tilde{c}_1 - \tilde{a}_1 \end{bmatrix}, \\ B &= [\tilde{b}_r \quad \tilde{b}_{r-1} \quad \tilde{b}_{r-2} \quad \cdots \quad \tilde{b}_2 \quad \tilde{b}_1]^T, \\ C &= [0 \quad 0 \quad 0 \quad \cdots \quad 0 \quad 1].\end{aligned}$$

Such a process can be generalized to multi-input, multi-output (MIMO) systems. For a m -output, p -input single-rate system modeled by the CARIMA model

$$a(q^{-1})y[k] = b(q^{-1})u[k-1] + \frac{c(q^{-1})}{\Delta}e[k], \quad (5.3)$$

where $a(q^{-1})$ and $c(q^{-1})$ are $m \times m$ monic polynomial matrices and $b(q^{-1})$ is a $m \times p$ polynomial matrix; similarly, we convert equation (5.3) into

$$y[k] = \frac{\tilde{b}(q)}{\tilde{a}(q)}\Delta u[k] + \frac{\tilde{c}(q)}{\tilde{a}(q)}e[k] \quad (5.4)$$

[$\tilde{a}(q)$ is a scalar function] and write

$$\begin{aligned}\tilde{a}(q) &= q^r + \tilde{a}_1 q^{r-1} + \tilde{a}_2 q^{r-2} + \cdots + \tilde{a}_{r-1} q + \tilde{a}_r \quad (\tilde{a}_i \text{ are scalars}), \\ \tilde{b}(q) &= \tilde{b}_1 q^{r-1} + \tilde{b}_2 q^{r-2} + \cdots + \tilde{b}_{r-1} q + \tilde{b}_r \quad (\tilde{b}_i \text{ are } m \times p), \\ \tilde{c}(q) &= \tilde{c}_0 q^r + \tilde{c}_1 q^{r-1} + \tilde{c}_2 q^{r-2} + \cdots + \tilde{c}_{r-1} q + \tilde{c}_r \quad (\tilde{c}_i \text{ are } m \times m).\end{aligned}$$

Then the observable canonical state-space realization for the MIMO model in (5.4) is

$$x[k+1] = Ax[k] + B\Delta u[k] + Fe[k] \quad (5.5)$$

$$y[k] = Cx[k] + Ge[k], \quad (5.6)$$

where

$$\begin{aligned}
A &= \begin{bmatrix} 0_m & 0_m & 0_m & \cdots & \cdots & -\tilde{a}_r I_m \\ I_m & 0_m & 0_m & \cdots & \cdots & -\tilde{a}_{r-1} I_m \\ 0_m & I_m & 0_m & \cdots & \cdots & -\tilde{a}_{r-2} I_m \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & -\tilde{a}_2 I_m \\ 0_m & 0_m & 0_m & \cdots & I_m & -\tilde{a}_1 I_m \end{bmatrix}, \quad F = \begin{bmatrix} \tilde{c}_r - \tilde{a}_r I_m \\ \tilde{c}_{r-1} - \tilde{a}_{r-1} I_m \\ \tilde{c}_{r-2} - \tilde{a}_{r-2} I_m \\ \vdots \\ \tilde{c}_2 - \tilde{a}_2 I_m \\ \tilde{c}_1 - \tilde{a}_1 I_m \end{bmatrix}, \\
B &= \begin{bmatrix} \tilde{b}_r^T & \tilde{b}_{r-1}^T & \tilde{b}_{r-2}^T & \cdots & \tilde{b}_2^T & \tilde{b}_1^T \end{bmatrix}^T, \\
C &= \begin{bmatrix} 0_m & 0_m & 0_m & \cdots & 0_m & I_m \end{bmatrix}, \\
G &= \tilde{c}_0.
\end{aligned}$$

This state-space model in (5.5) and (5.6) is used to represent the fast single-rate process with sampling period h :

$$y = P_{dh} \Delta u + N_h e.$$

To introduce the GPC problem for the dual-rate system, let us review first the GPC problem for the single-rate system in (5.5) and (5.6): The aim is to minimize the following cost function

$$\begin{aligned}
J_s[k] &= E \left\{ \sum_{i=N_1}^{N_2} \{w[k+i] - y[k+i]\}^T \{w[k+i] - y[k+i]\} \right. \\
&\quad \left. + \lambda \sum_{i=1}^{N_u} \Delta u[k+i-1]^T \Delta u[k+i-1] \right\}, \quad (5.7)
\end{aligned}$$

subject to $\Delta u[k+j] = 0$, $j = N_u, N_u + 1, \dots, N_2$. Thus N_u future control increments, namely,

$$\{\Delta u[k+j] : j = 0, \dots, N_u - 1\}$$

are computed explicitly through the minimization of (5.7) under the condition that only data up to and including time instant k is available. Once these control moves are computed, however, only the first, namely, $u[k] = u[k-1] + \Delta u[k]$, is applied. The process of minimization is repeated at the next sample point.

For the dual-rate GPC problem, we will first obtain a state-space model for the lifted system from $\Delta \underline{u}$ and e to y (refer to Figure 5.3), the starting point being the single-rate fast model in (5.5) and (5.6). Using lifting and down-sampling [20], one can obtain the following state-space model for the lifted LTI system with period nh :

$$\underline{x}[k+1] = \underline{A} \underline{x}[k] + \underline{B} \Delta \underline{u}[k] + \underline{F} e[k] \quad (5.8)$$

$$y[k] = \underline{C} \underline{x}[k] + \underline{D} \Delta \underline{u}[k] + \underline{G} e[k], \quad (5.9)$$

where $\underline{x}[k] = x[nk]$, and \underline{u} is the lifted u , i.e.,

$$\Delta \underline{u}[k] = \begin{bmatrix} \Delta u(k \cdot nh) \\ \Delta u(k \cdot nh + h) \\ \vdots \\ \Delta u[k \cdot nh + (n-1)h] \end{bmatrix}.$$

The matrices in (5.8) and (5.9) can be given in terms of the fast single-rate model in (5.5) and (5.6):

$$\begin{aligned} \underline{A} &= A^n, \quad \underline{B} = \begin{bmatrix} A^{n-1}B & A^{n-2}B & \dots & AB & B \end{bmatrix}, \\ \underline{F} &= A^{n-1}F, \quad \underline{C} = C, \quad \underline{D} = 0, \quad \underline{G} = G. \end{aligned}$$

Here $\underline{D} = 0$ is due to the fact that the fast single-rate system from Δu to y is assumed to be strictly causal.

The GPC problem for the dual-rate system modeled in (5.8) and (5.9) can be stated as follows: Compute the control moves $\Delta \underline{u}[k+j]$, $j = 0, 1, \dots, n_u - 1$, to minimize the GPC index

$$\begin{aligned} J_d[k] &= E \left\{ \sum_{i=n_1}^{n_2} \{w[k+i] - y[k+i]\}^T \Omega \{w[k+i] - y[k+i]\} \right. \\ &\quad \left. + \lambda \sum_{i=1}^{n_u} \Delta \underline{u}[k+i-1]^T \Delta \underline{u}[k+i-1] \right\}, \end{aligned} \quad (5.10)$$

subject to $\Delta \underline{u}[k+j] = 0$, $j = n_u, \dots, n_2$. Once these control moves are computed, however, only the first, namely, $\Delta \underline{u}[k]$, is implemented; this corresponds to implementing n fast control moves at once. The computation is repeated at the slow sample rate.

Here we take $n_1 = 1$. For the dual-rate cost function in (5.10) and the single-rate one in (5.7) to be comparable (and hence we can compare performance of single-rate and dual-rate GPC controllers later), we make the following standing assumptions

- The cost and control horizons satisfy $N_2 = n \cdot n_2$ and $N_u = n \cdot n_u$.
- The weighting matrix for the tracking error in (5.10) is $\Omega = nI_q$.

Now we extend the approach in [75] to give a solution to the dual-rate GPC problem posed above. Let j be a positive integer; from (5.8) and (5.9), we obtain

$$\begin{aligned} y[k+j] &= \underline{C} \underline{A}^j \underline{x}[k] + \sum_{i=1}^j \underline{C} \underline{A}^{j-i} \underline{B} \Delta \underline{u}[k+i-1] + \sum_{i=1}^j \underline{C} \underline{A}^{j-i} \underline{F} e[k+i-1] \\ &\quad + \underline{G} e[k+j]. \end{aligned}$$

Since for future noise $e[k+i]$ ($0 < i < j$), $E\{e[k+i]\} = 0$, we get the j -step ahead prediction of $y[k+j]$ at time k :

$$\hat{y}[k+j|k] = \underline{C} \underline{A}^j \hat{x}[k] + \sum_{i=1}^j \underline{C} \underline{A}^{j-i} \underline{B} \Delta \underline{u}[k+i-1] + \underline{C} \underline{A}^{j-1} \underline{F} \hat{e}[k].$$

The estimation of the present noise $e[k]$ can be easily derived:

$$\hat{e}[k] = \underline{G}^{-1} \{\hat{y}[k] - \underline{C} \hat{x}[k]\} = \underline{G}^{-1} y[k] - \underline{G}^{-1} \underline{C} \hat{x}[k].$$

Thus

$$\begin{aligned} \hat{y}[k+j|k] &= \sum_{i=1}^j \underline{C} \underline{A}^{j-i} \underline{B} \Delta \underline{u}[k+i-1] + (\underline{C} \underline{A}^j - \underline{C} \underline{A}^{j-1} \underline{F} \underline{G}^{-1} \underline{C}) \hat{x}[k] \\ &\quad + \underline{C} \underline{A}^{j-1} \underline{F} \underline{G}^{-1} y[k]. \end{aligned} \quad (5.11)$$

Letting j in (5.11) go from n_1 to n_2 and defining

$$\begin{aligned} Y[k] &= \left[\hat{y}^T[k+n_1|k] \quad \hat{y}^T[k+n_1+1|k] \quad \cdots \quad \hat{y}^T[k+n_2|k] \right]^T, \\ \Delta U[k] &= \left[\Delta \underline{u}^T[k] \quad \Delta \underline{u}^T[k+1] \quad \cdots \quad \Delta \underline{u}^T[k+n_u-1] \right]^T, \end{aligned}$$

we have

$$Y[k] = H \Delta U[k] + \Phi \hat{x}[k] + \Gamma y[k] = H \Delta U[k] + \tilde{Y}[k],$$

where we have used the definition

$$\tilde{Y}[k] = \Phi \hat{x}[k] + \Gamma y[k].$$

The matrices involved are given below:

$$\begin{aligned} H &= \begin{bmatrix} \underline{C} \underline{B} & & & & & \\ \underline{C} \underline{A} \underline{B} & \underline{C} \underline{B} & & & & \\ \vdots & \ddots & \ddots & & & \\ \underline{C} \underline{A}^{n_u-1} \underline{B} & \ddots & \ddots & & \underline{C} \underline{B} & \\ \vdots & \ddots & \ddots & & \vdots & \\ \underline{C} \underline{A}^{n_2-1} \underline{B} & \cdots & \cdots & \cdots & \underline{C} \underline{A}^{n_2-n_u} \underline{B} & \end{bmatrix}, \\ \Phi &= \begin{bmatrix} \underline{C} \\ \underline{C} \underline{A} \\ \vdots \\ \underline{C} \underline{A}^{n_2-1} \end{bmatrix} (\underline{A} - \underline{F} \underline{G}^{-1} \underline{C}), \\ \Gamma &= \begin{bmatrix} \underline{C} \\ \underline{C} \underline{A} \\ \vdots \\ \underline{C} \underline{A}^{n_2-1} \end{bmatrix} \underline{F} \underline{G}^{-1}. \end{aligned}$$

Defining the vector

$$W[k] = \begin{bmatrix} w^T[k + n_1] & w^T[k + n_1 + 1] & \cdots & w^T[k + n_2] \end{bmatrix}^T,$$

we can rewrite the dual-rate GPC cost function in (5.10) in the matrix form:

$$J_d[k] = (W[k] - Y[k])^T \Upsilon (W[k] - Y[k]) + \lambda \Delta U[k]^T \Delta U[k].$$

Here Υ is a block diagonal matrix:

$$\Upsilon = \text{diag}\{\Omega, \Omega, \dots, \Omega\}.$$

From here, we can compute the optimal dual-rate GPC solution:

$$\Delta U[k] = (H^T \Upsilon H + \lambda I)^{-1} H^T \Upsilon^T (W[k] - \tilde{Y}[k]).$$

Comparing with the single-rate results in, e.g., [75], we note that the multirate GPC strategy

$$\{\Delta \underline{u}[k + j] : j = 0, \dots, n_u - 1\}$$

is a sequence of *lifted* control moves. As mentioned before, only the first one, $\Delta \underline{u}[k]$, is implemented; this corresponds to implementing n fast control moves for every period (nh) the output is sampled.

The state estimation is an important part in state-space GPC solutions ([7],[75]). we use a steady-state Kalman filter for this purpose:

$$\hat{\underline{x}}[k + 1] = (\underline{A} - \underline{F} \underline{G}^{-1} \underline{C}) \hat{\underline{x}}[k] + \underline{B} \Delta \underline{u}[k] + \underline{F} \underline{G}^{-1} y[k]. \quad (5.12)$$

With (5.12), the dual-rate state-space GPC solution would be exactly the same as an input-output solution if one would choose to approach the GPC problem from an input-output viewpoint.

5.4 Stability robustness

For a model-based control scheme such as GPC, it is important to analyze the stability robustness of the closed-loop system in the presence of model-plant mismatch. It turns out that it is better to carry out the analysis in the frequency domain, for which we need a polynomial domain solution of the dual-rate GPC problem. We will focus on SISO systems in this section.

5.4.1 CARIMA models

Starting with a CARIMA model in (5.1) for a fast single-rate system, rewrite it as follows:

$$a(q^{-1})\Delta \cdot y[k] = q^{-1}b(q^{-1}) \cdot \Delta u[k] + c(q^{-1}) \cdot e[k]. \quad (5.13)$$

Down-sampling the output by a factor n and lifting the control signal – refer to Figure 5.3, we get a lifted LTI system

$$A(q^{-1})y[k] = B(q^{-1})\Delta u[k - 1] + C(q^{-1})e[k]. \quad (5.14)$$

The underlying periods of all signals involved are the same, namely, nh . Notice here that A and C are scalar polynomials, and B is a $1 \times n$ polynomial matrix.

To compute A , B and C , we first find from (5.13) the two transfer functions for P_{dh} , the system from Δu to y , and N_h , the system from e to y . Then obtain state-space models for the two transfer functions:

$$P_{dh}(q) = \left[\begin{array}{c|c} A_p & B_p \\ \hline C_p & 0 \end{array} \right], \quad N_h(q) = \left[\begin{array}{c|c} A_N & B_N \\ \hline C_N & D_N \end{array} \right].$$

Notice here that we assumed that P_{dh} is strictly causal and hence $D_p = 0$.

It is then not difficult to get the lifted model \underline{P}_{dh} and the down-sampled noise model N_{nh} :

$$\underline{P}_{dh}(q) = \left[\begin{array}{c|cccc} A_p^n & A_p^{n-1}B_p & A_p^{n-2}B_p & \cdots & B_p \\ \hline C_p & 0 & 0 & \cdots & 0 \end{array} \right],$$

$$N_{nh}(q) = \left[\begin{array}{c|c} A_N^n & A_N^{n-1}B_N \\ \hline C_N & D_N \end{array} \right].$$

From here we can obtain the transfer functions for \underline{P}_{dh} and N_{nh} ; then comparing equation

$$y[k] = \underline{P}_{dh}(q)\Delta u[k] + N_{nh}(q)e[k]$$

with (5.14), we can find polynomials A , B and C in the following form:

$$A(q^{-1}) = 1 + A_1q^{-1} + A_2q^{-2} + \cdots + A_rq^{-r}, \quad (5.15)$$

$$B(q^{-1}) = B_1 + B_2q^{-1} + \cdots + B_{r-1}q^{-(r-1)}, \quad (5.16)$$

$$C(q^{-1}) = 1 + C_1q^{-1} + C_2q^{-2} + \cdots + C_rq^{-r}. \quad (5.17)$$

Note again that A_i and C_i are scalars, and B_i are $1 \times n$ matrices.

Next we will describe a solution to the dual-rate GPC problem based on the CARIMA model in (5.14) in the polynomial domain.

5.4.2 Dual-rate GPC solution in the polynomial domain

We have reduced the dual-rate GPC problem into a single-rate one with multiple inputs and a single output, as is described by (5.14). In the following we will give a solution to the GPC problem in the polynomial domain; such a solution is useful later for our robustness analysis.

First, we solve the Diophantine equation:

$$C = E_j A + q^{-j} F_j, \quad (5.18)$$

where j is a positive integer, E_j and F_j are unique polynomial scalars of order $j - 1$ and $r - 1$, respectively. Multiplying (5.14) by q^j and using (5.18), we get

$$y[k + j] = \frac{B}{A} \Delta \underline{u}[k + j - 1] + E_j e[k + j] + \frac{F_j}{A} e[k].$$

From (5.14), replace $e[k]$ by

$$e[k] = \frac{1}{C} [Ay[k] - B\Delta \underline{u}[k - 1]]$$

to get

$$y[k + j] = \frac{E_j B}{C} \Delta \underline{u}[k + j - 1] + \frac{F_j}{C} y[k] + E_j e[k + j].$$

Since $E\{e[k + j]\} = 0$, at time k , the best j -step ahead prediction of the output is then

$$\hat{y}[k + j|k] = \frac{E_j B}{C} \Delta \underline{u}[k + j - 1] + \frac{F_j}{C} y[k]. \quad (5.19)$$

Next, solve the Diophantine equation:

$$E_j B = G_j C + q^{-j} \tilde{G}_j. \quad (5.20)$$

Here G_j is a $1 \times n$ polynomial matrix of order $j - 1$ and is given by

$$G_j = G_{j,0} + G_{j,1}q^{-1} + \dots + G_{j,j-1}q^{-(j-1)},$$

and $G_{j+1,i} = G_{j,i}$ for $i = 0, \dots, j - 1$. Replacing $E_j B$ in (5.19) by (5.20), we obtain the optimal j -step ahead prediction:

$$\begin{aligned} \hat{y}[k + j|k] &= G_j \Delta \underline{u}[k + j - 1] + \frac{\tilde{G}_j}{C} \Delta \underline{u}[k - 1] + \frac{F_j}{C} y[k] \\ &= G_j \Delta \underline{u}[k + j - 1] + \tilde{y}[k + j|k]. \end{aligned} \quad (5.21)$$

We just defined in (5.21)

$$\tilde{y}[k + j|k] = \frac{\tilde{G}_j}{C} \Delta \underline{u}[k - 1] + \frac{F_j}{C} y[k].$$

Then we want to give a solution to the GPC problem in the matrix form, with the performance index defined in (5.10). Letting j vary from $n_1 = 1$ to n_2 in (5.21) and defining

$$\begin{aligned} Y[k] &= \begin{bmatrix} \hat{y}[k+n_1|k] & \hat{y}[k+n_1+1|k] & \cdots & \hat{y}[k+n_2|k] \end{bmatrix}^T, \\ \tilde{Y}[k] &= \begin{bmatrix} \tilde{y}[k+n_1|k] & \tilde{y}[k+n_1+1|k] & \cdots & \tilde{y}[k+n_2|k] \end{bmatrix}^T, \\ \Delta U[k] &= \begin{bmatrix} \Delta \underline{u}^T[k] & \Delta \underline{u}^T[k+1] & \cdots & \Delta \underline{u}^T[k+n_u-1] \end{bmatrix}^T, \end{aligned}$$

we have

$$Y[k] = H\Delta U[k] + \tilde{Y}[k],$$

where H is a matrix defined as follows:

$$H = \begin{bmatrix} G_{1,0} & & & & \\ G_{2,1} & G_{2,0} & & & \\ \vdots & \ddots & \ddots & & \\ G_{n_u, n_u-1} & \ddots & \ddots & G_{n_u, 0} & \\ \vdots & \ddots & \ddots & \vdots & \\ G_{n_2, n_2-1} & \cdots & \cdots & G_{n_2, n_2-n_u} & \end{bmatrix}.$$

From here the optimal GPC control vector is

$$\Delta U[k] = (H^T \Upsilon H + \lambda I)^{-1} H^T \Upsilon^T (W[k] - \tilde{Y}[k]). \quad (5.22)$$

Note again that only $\Delta \underline{u}(k)$, the first element in U , is implemented.

Finally, we want to express the GPC control law in a feedback form. Define the first n rows of matrix $(H^T \Upsilon H + \lambda I)^{-1} H^T \Upsilon^T$ in (5.22) as K and the i -th column of K as K_i , we get

$$\begin{aligned} \Delta \underline{u}[k] &= \sum_{i=1}^{n_2} K_i \{w[k+i] - \tilde{y}[k+i|k]\} \\ &= \sum_{i=1}^{n_2} K_i w[k+i] - \sum_{i=1}^{n_2} K_i \tilde{y}[k+i|k] \\ &= \sum_{i=1}^{n_2} K_i w[k+i] - \sum_{i=1}^{n_2} K_i \frac{\tilde{G}_j}{C} \Delta \underline{u}[k-1] - \sum_{i=1}^{n_2} K_i \frac{F_j}{C} y[k]. \end{aligned}$$

Assume the future reference signal is constant along the horizon: $w[k+i] = w[k]$. The above equation can be written as:

$$\left(C \cdot I + q^{-1} \sum_{i=1}^{n_2} K_i \tilde{G}_i \right) \Delta \underline{u}[k] = C \sum_{i=1}^{n_2} K_i w[k] - \sum_{i=1}^{n_2} K_i F_i y[k]. \quad (5.23)$$

Now we define

$$\begin{aligned}
 R(q^{-1}) &= C \cdot I + q^{-1} \sum_{i=1}^{n_2} K_i \tilde{G}_i, \\
 T(q^{-1}) &= C \sum_{i=1}^{n_2} K_i, \\
 S(q^{-1}) &= \sum_{i=1}^{n_2} K_i F_i,
 \end{aligned}$$

and assume that the square matrix $R(q^{-1})$ is invertible. Then equation (5.23) becomes

$$\Delta u[k] = R^{-1} [T w[k] - S y[k]]. \quad (5.24)$$

This feedback control law gives rise to the closed-loop control configuration shown in Figure 5.4, which is the basis for our subsequent study on stability robustness.

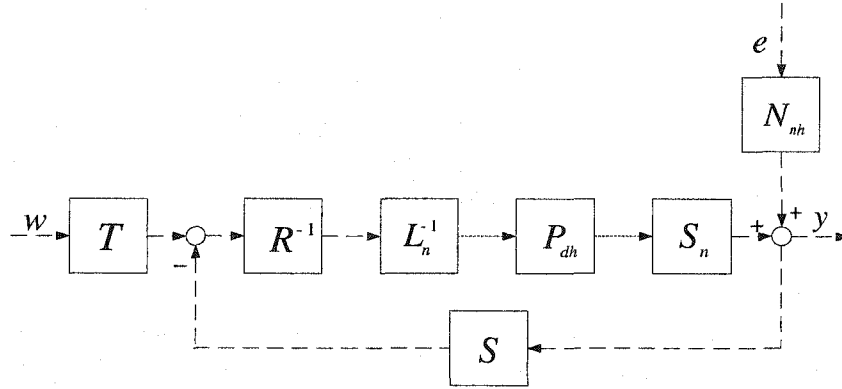


Figure 5.4: Dual-rate GPC closed-loop system

5.4.3 Stability robustness analysis

We have set the stage for studying the stability robustness of the dual-rate GPC control system in Figure 5.4. In the following we assume that a multiplicative uncertainty model is adopted. In Figure 5.5, \hat{P}_{dh} is the *estimated* fast single-rate model from Δu to y ; $(1 + OW_2)\hat{P}_{dh}$ captures the class of uncertain models, where W_2 is a fixed stable weighting filter and O represents the model-plant mismatch or the uncertainty; O is stable and belongs to the class

$$\{O : \|O\|_\infty < 1\}.$$

Robust stability is achieved if the controller stabilizes every admissible plant.

We remark that due to the presence of the down-sampler S_n and inverse lifting L_n^{-1} , the feedback system in Figure 5.5 is a time-varying system. For stability issues, we ignore the

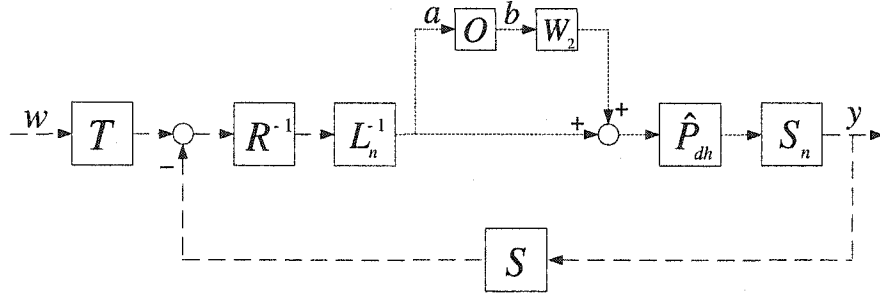


Figure 5.5: Dual-rate GPC closed-loop system with model uncertainty

input. Isolating the O block, we can reconfigure Figure 5.5 into the feedback connection in Figure 5.6, where the feedback loop, the system from b to a is $-MW_2$; it is easy to derive that

$$M = (I + L_n^{-1}R^{-1}SS_n\hat{P}_{dh})^{-1}L_n^{-1}R^{-1}SS_n\hat{P}_{dh}.$$

Applying the small gain condition [33], we have that the GPC controller achieves robust stability if the loop gain $\|MW_2\|_\infty$ is less than 1. Thus the problem reduces to computing

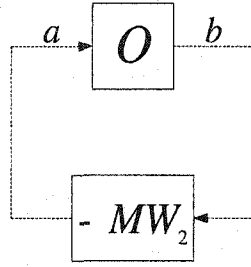


Figure 5.6: Simplified system for robustness analysis

the norm of MW_2 , which in this case is a dual-rate system.

Using the norm preserving property of the lifting, we can relate the norm computation of MW_2 to that for an LTI system. Since $\|MW_2\|_\infty = \|L_nMW_2L_n^{-1}\|_\infty$ and

$$\begin{aligned} L_nMW_2L_n^{-1} &= L_n(I + L_n^{-1}R^{-1}SS_n\hat{P}_{dh})^{-1}L_n^{-1}R^{-1}SS_n\hat{P}_{dh}W_2L_n^{-1} \\ &= [I + R^{-1}SS_nL_n^{-1}(L_n\hat{P}_{dh}L_n^{-1})]^{-1}R^{-1}SS_nL_n^{-1}(L_n\hat{P}_{dh}L_n^{-1})(L_nW_2L_n^{-1}) \\ &= (I + R^{-1}SV\hat{P}_{dh})^{-1}R^{-1}SV\hat{P}_{dh}W_2, \end{aligned}$$

where we defined

$$\hat{P}_{dh} = L_n\hat{P}_{dh}L_n^{-1}, \quad \underline{W}_2 = L_nW_2L_n^{-1}, \quad V = S_nL_n^{-1}.$$

Notice that \hat{P}_{dh} and \underline{W}_2 are LTI systems and V is indeed a static system given by the

matrix

$$V = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix}.$$

Hence

$$\|MW_2\|_\infty = \|(I + R^{-1}SV\hat{P}_{dh})^{-1}R^{-1}SV\hat{P}_{dh}W_2\|_\infty. \quad (5.25)$$

To summarize, the stability robustness property is indicated by the norm computation in (5.25), which amounts to evaluating the ∞ -norm of an LTI system involving \hat{P}_{dh} , W_2 and V , depending on the model, and R and S , depending on the GPC controller.

5.5 Illustrative examples

Two examples are given in this section: In the first one we apply the state-space GPC algorithm developed in Section 5.3 to a multirate MIMO system and compare its performance with those of single-rate control schemes; in the second we study stability robustness in an example based on the result in Section 5.4, as the down-sampling ratio n changes.

Example 1

This example is based on an experimental setup in the Department of Chemical Engineering at the University of Alberta. This is a pilot-scale stirred tank heater whose schematic diagram is simply given in Figure 5.7. This process has two inputs (manipulated variables), the valve positions of the inlet cold water and steam, and two measured outputs, the level and temperature of the water in the tank.

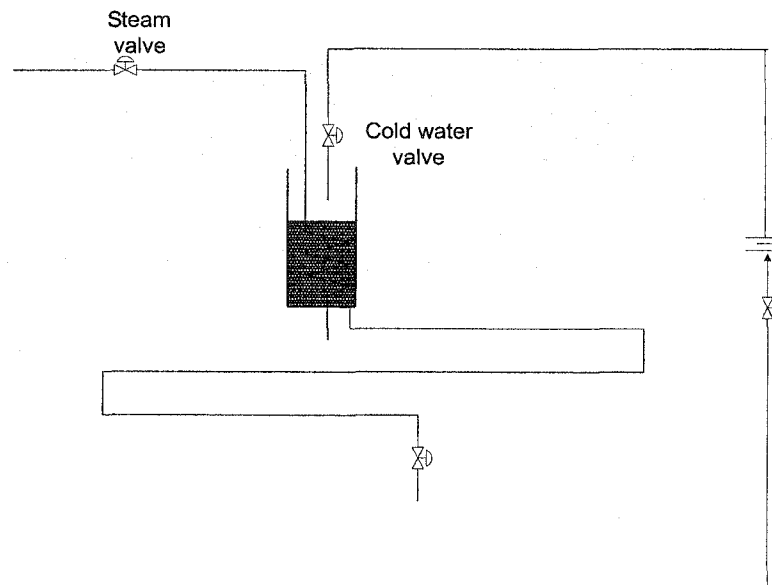


Figure 5.7: Schematic diagram of a stirred tank heater

We will study and compare three GPC design cases for the stirred tank heater:

Case 1 Adjust the control signals and sample the outputs every 4 seconds – the fast single-rate case. A fast single-rate process model (A, B, C, D) can be obtained using certain system identification methods, for example, CVA [52] or N4SID [109, 110]. However, the true noise model is difficult to obtain. So in the state-space GPC problems, we use F as a design parameter. Thus (A, B, C, D, F) is considered to be the fast single-rate state-space model. The GPC performance index used is given by

$$J_f[k] = E \left\{ \sum_{i=1}^{12} \{w[k+i] - y[k+i]\}^T \{w[k+i] - y[k+i]\} + 2 \sum_{i=1}^{12} \Delta u[k+i-1]^T \Delta u[k+i-1] \right\}.$$

Case 2 Adjust the control signals every 4 seconds and sample the outputs every 8 seconds – the dual-rate case with ratio 2. The lifted state-space model $(\underline{A}, \underline{B}, \underline{C}, \underline{D}, \underline{F})$ can be obtained from the fast single-rate model. The associated GPC index is

$$J_d[k] = E \left\{ 2 \sum_{i=1}^6 \{w[k+i] - y[k+i]\}^T \{w[k+i] - y[k+i]\} + 2 \sum_{i=1}^6 \Delta u[k+i-1]^T \Delta u[k+i-1] \right\}.$$

Case 3 Adjust the control signals and sample the outputs every 8 seconds – the slow single-rate case. Based on the fast single-rate model, the slow single-rate state-space model $(A_s, B_s, C_s, D_s, F_s)$ can be obtained. The associated GPC index is

$$J_s[k] = E \left\{ 2 \sum_{i=1}^6 \{w[k+i] - y[k+i]\}^T \{w[k+i] - y[k+i]\} + 4 \sum_{i=1}^6 \Delta u[k+i-1]^T \Delta u[k+i-1] \right\}.$$

Notice that the choices of horizons and weightings are such that the three performance indices are compatible. The setpoint changes are as follows: Setpoints for water level and temperature both change from 0 to 1 mA at time 0 (we use mA to quantify both y_1 and y_2 ; there are simple linear relationships to translate these units to actual physical units); then the setpoint for water level changes to 4 mA at time 1200 sec; and setpoint for temperature changes to 4 mA at time 1400 sec. Also, at time 200 sec, the water level is subjected to a step type disturbance with magnitude 1. There are also filtered integrated white noise added to the process outputs.

Under these conditions, the three GPC problems are solved via state-space algorithms and their performances are compared by simulation. The tracking performance of the three cases over the whole time horizon is presented in Figure 5.8, which shows little difference for the three cases.

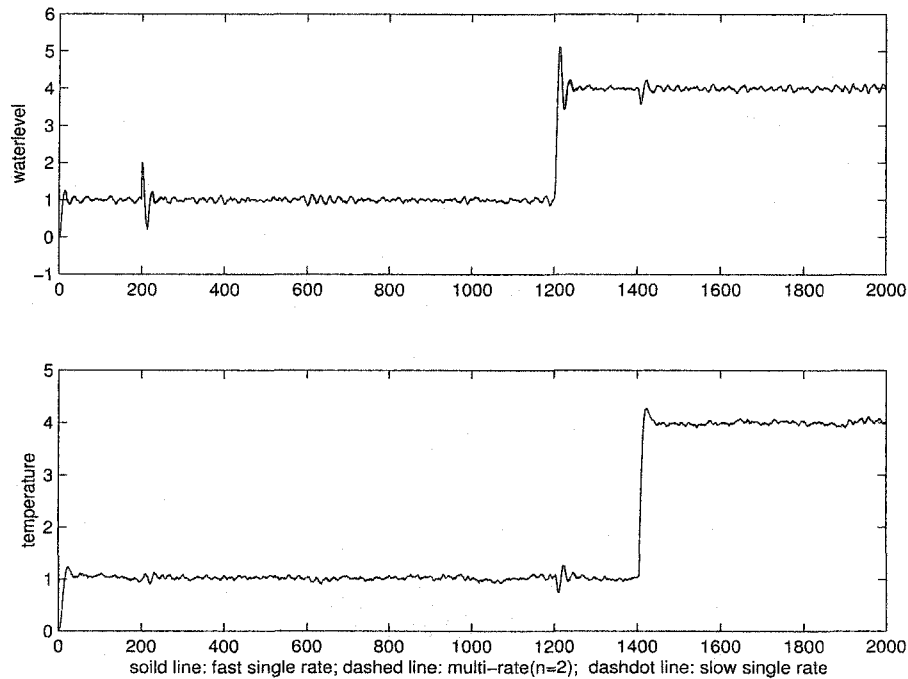


Figure 5.8: Tracking performance over the whole time horizon

Now let us zoom into the period from time 0 to 250 sec, see Figure 5.9 and 5.10. It is clearer that the fast single-rate GPC generates the fastest response with minimum overshoot, the slow single-rate GPC is the worst, while the dual-rate one is in between.

We can also compare the performance indices for the three cases for setpoint tracking to get a rough idea – see Figure 5.11. It appears that the fast single-rate GPC has the smallest performance index at most times, and the dual-rate GPC’s performance is in between the performance of the two single-rate ones. In cases when practical constraints limit the sampling frequency, this example shows that dual-rate control scheme is a better choice than the slow single-rate scheme.

Example 2

Here we study stability robustness using an example from [8]. The continuous-time process is third-order:

$$P(s) = \frac{1}{(s+1)(3s+1)(5s+1)}$$

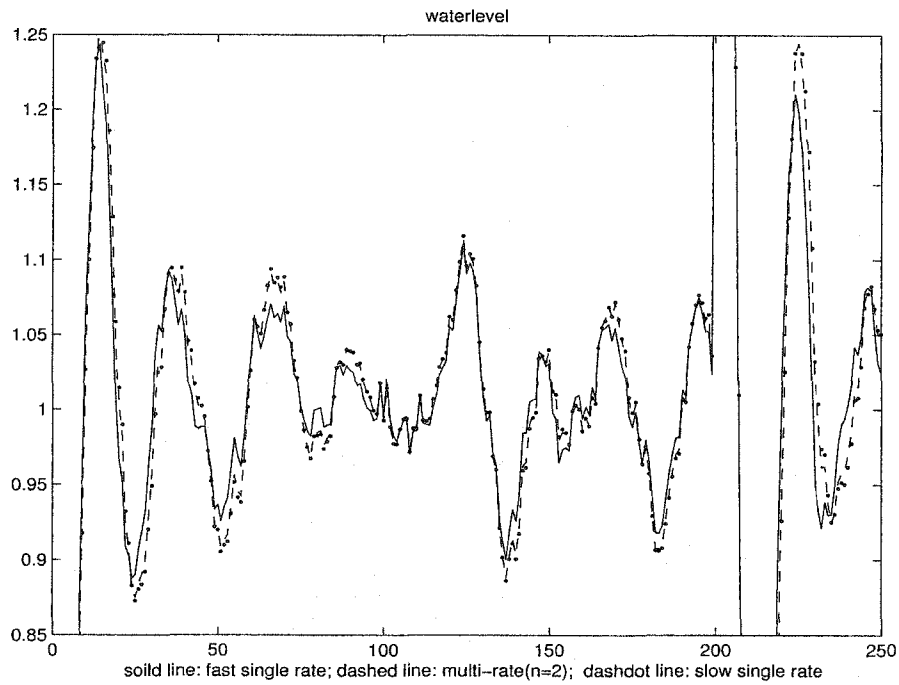


Figure 5.9: Tracking performance of the water level during time period [0, 250] sec

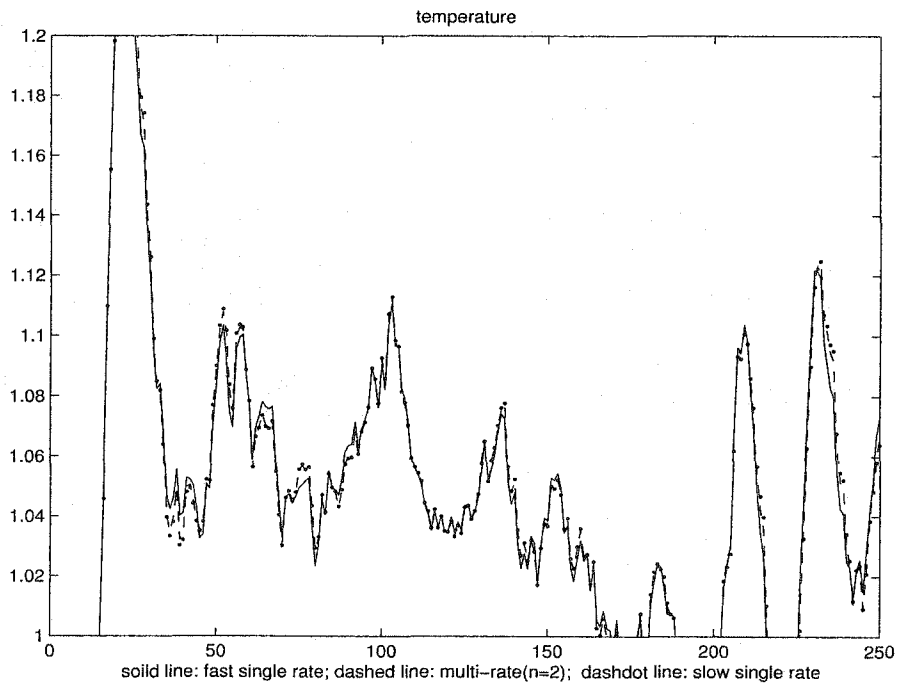


Figure 5.10: Tracking performance of the temperature during time period [0, 250] sec

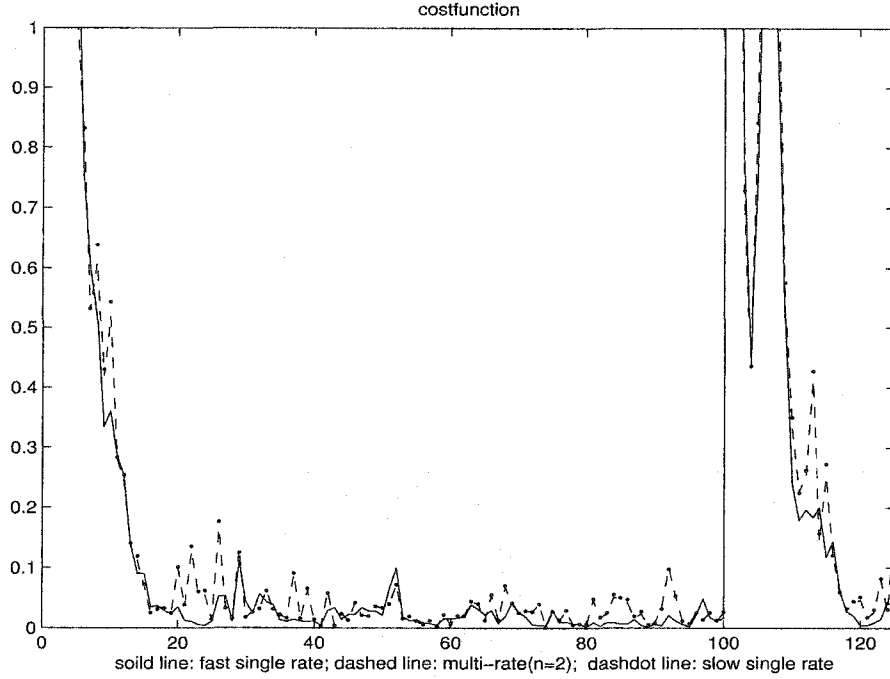


Figure 5.11: Performance index comparison for three cases over time horizon $[0, 250]$ sec

Take the sampling period to be $h = 1$ sec; then the discrete-time model is

$$P_h(q^{-1}) = \frac{0.0077q^{-1} + 0.0212q^{-2} + 0.0036q^{-3}}{1 - 1.9031q^{-1} + 1.1514q^{-2} - 0.2158q^{-3}}. \quad (5.26)$$

This is regarded as the true process model. Using square wave excitation and the least square methods, a first-order model was identified:

$$\hat{P}_h(q^{-1}) = \frac{0.0419q^{-1} + 0.0719q^{-2}}{1 - 0.8969q^{-1}}. \quad (5.27)$$

Notice the model-plant mismatch between P_h and \hat{P}_h . Next we want to embed P_h into the multiplicative uncertainty class

$$\left\{ (1 + OW_2)\hat{P}_h : \|O\|_\infty \leq 1 \right\},$$

in order to find a suitable weighting filter W_2 . This requires that W_2 satisfies the following inequality

$$\left| 1 - \frac{P_h(e^{j\omega})}{\hat{P}_h(e^{j\omega})} \right| \leq |W_2(e^{j\omega})|, \quad \forall \omega \in [0, \pi].$$

By trial and error, we found a suitable weighting function

$$W_2(q^{-1}) = \frac{1 - 0.96q^{-1}}{1 - 0.6q^{-1}}.$$

Figure 5.12 shows the Bode plot of W_2 over-bounding that of the true multiplicative perturbation, namely, $1 - P_h/\hat{P}_h$.

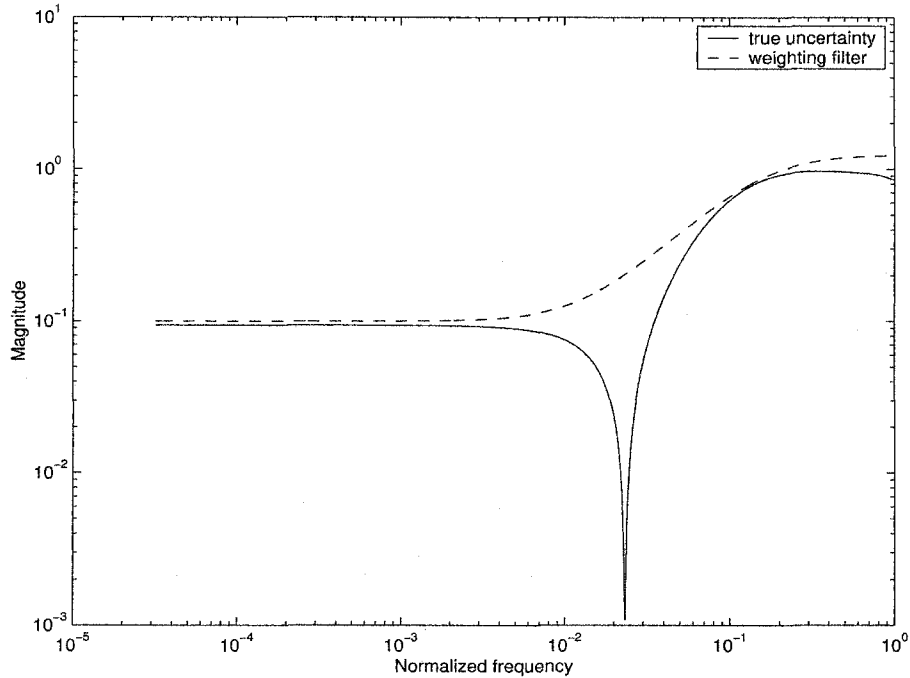


Figure 5.12: Bode plots of W_2 and the true uncertainty

Now we design dual-rate GPC for the nominal model \hat{P}_h . Performance indices in (5.7) and (5.10) are used for the fast single-rate and the dual-rate GPC problems, respectively. The tuning parameters are chosen as follows:

$$N_1 = n_1 = 1, \quad N_2 = N_u = 12, \quad n_2 = N_2/n, \quad n_u = N_u/n, \quad \lambda = 0.5, \quad (5.28)$$

where the larger the N_2 (or the smaller the N_u), the more robust the closed-loop system. But we will not tune them in this example and only the down-sampling ratio n is left as a variable to be changed in the design process: We allow n to take values among 1 (the fast single-rate case), 2, 3, 4, and 6. Another important tuning parameter, the filter polynomial $c(q^{-1})$ – see (5.1), is chosen to be $c(q^{-1}) = 1 - 0.8q^{-1}$. The step tracking performance of the GPC's with different n values, when applied to the nominal model \hat{P}_h is shown in Figure 5.13. We see that as n increases the overshoot gets larger.

Next we want to see the effect of the down-sampling ratio n on the stability robustness properties of the dual-rate GPC's. According to (5.25), the quantity $\|MW_2\|_\infty$ is an indicator of stability robustness: The smaller that quantity, the more robust the closed-loop system. The following table summarizes our calculation:

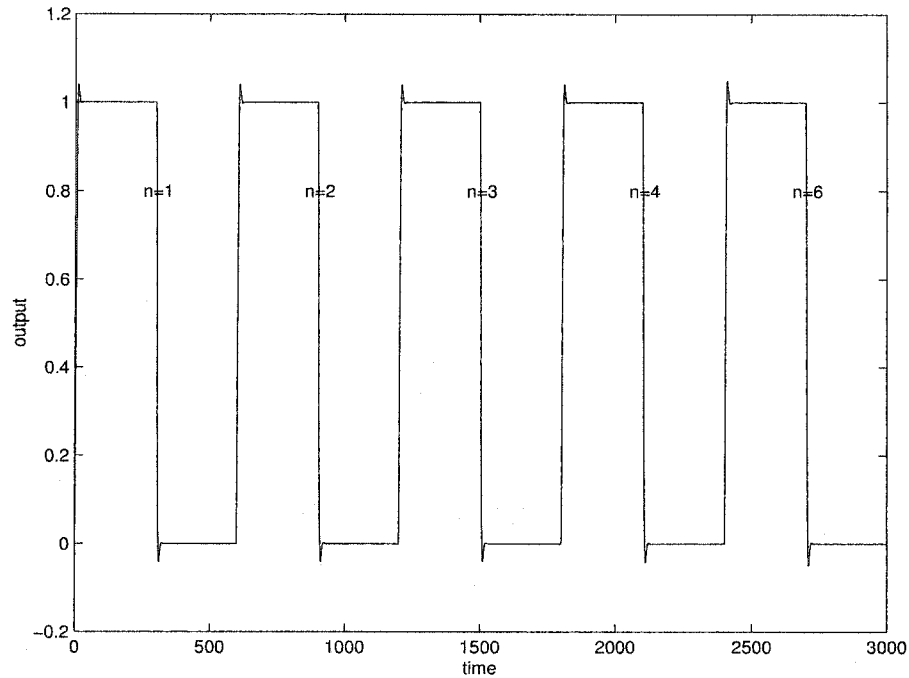


Figure 5.13: GPC tracking response with model \hat{P}_h

ratio n	$\ MW_2\ _\infty$
6	1.4607
4	1.2316
3	1.0131
2	0.8381
1	0.7269

So the closed-loop system becomes much more robust when n decreases. This observation can be verified by the following simulation results. Figures 5.14 and 5.15 show, respectively, the closed-loop tracking responses when the GPC's designed for nominal model \hat{P}_h are applied to the true model P_h in (5.26) and the model with transfer function $(1 + W_2)\hat{P}_h$, one of the extreme cases in the uncertainty class with $O = 1$. In the first situation, we see more oscillation as n increases, indicating closer to instability. In the second situation, the closed-loop system becomes unstable, when $n = 6$.

All these results indicate that the down-sampling ratio n is closely related to the robustness property of the GPC algorithm. The smaller this ratio n , the more robust the dual-rate GPC closed-loop system. An explanation by intuition is that when n decreases, the output is sampled more frequently, and the uncertainty information is fed to the controller more quickly. As a result, the controller can act to keep the closed-loop robust stability more efficiently.

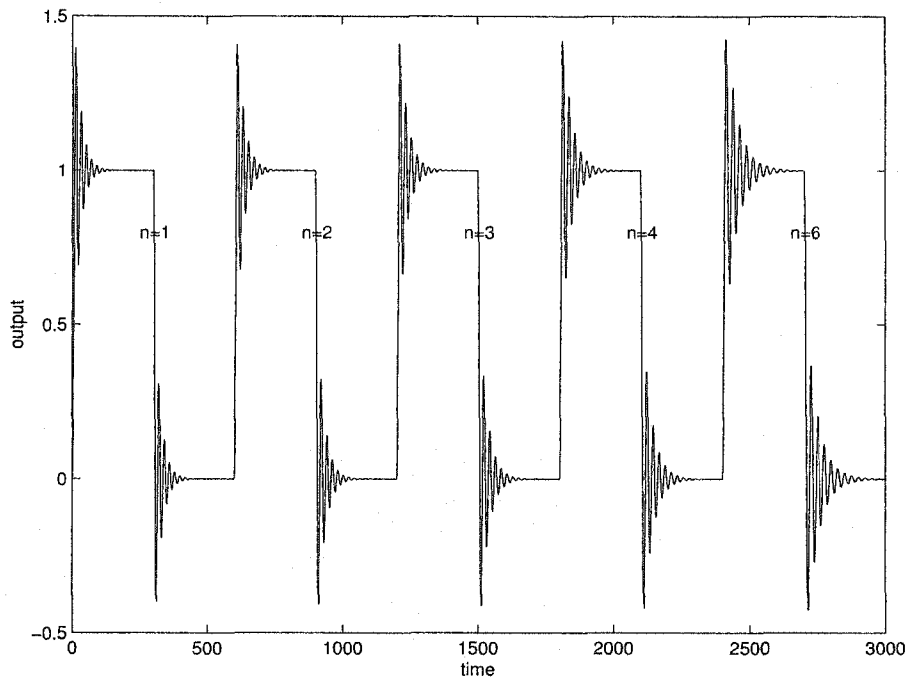


Figure 5.14: GPC tracking response with model P_h

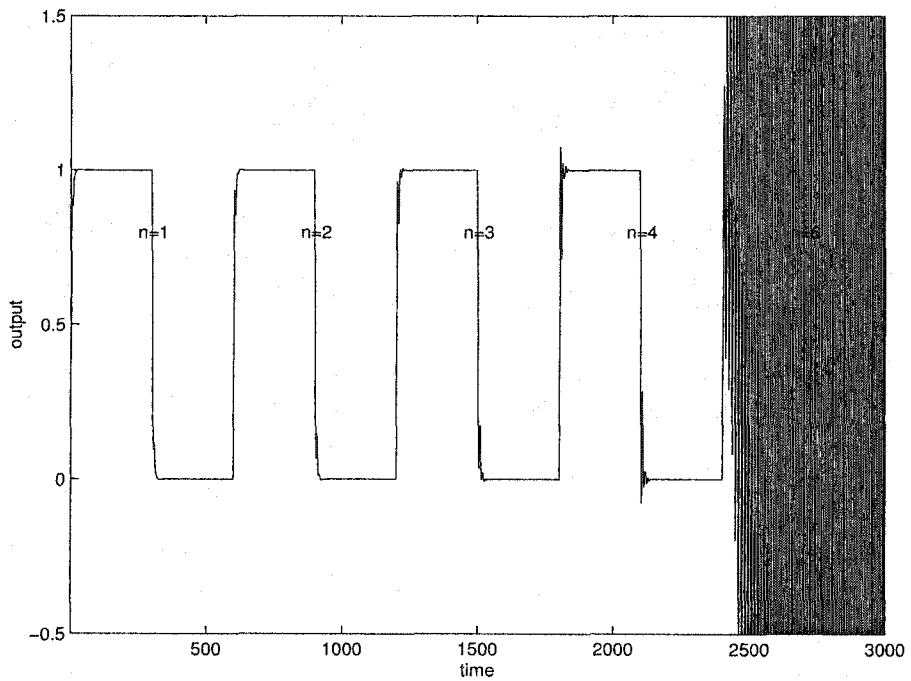


Figure 5.15: GPC tracking response with model $(1 + W_2)\hat{P}_h$

5.6 Conclusions

In this paper, we have studied GPC problems in a special case involving dual sampling rates. It is possible to find explicit formulas for the GPC solutions in both the state-space and polynomial domains, the main tool used being the lifting technique. We have also attempted stability robustness study for the dual-rate GPC algorithms.

We remark that the state-space GPC obtained in this chapter for dual-rate systems is an extension of the single-rate GPC algorithm developed by Ordys and Clarke [75], where the relations between the polynomial and state-space parameters are presented. In particular, the Kalman gain in (2.7) reflects the observer filter $c(q^{-1})$ in the CARIMA model (2.1). This is also true for the lifted model. There have been efforts to design c in the transfer function framework to improve disturbance rejection or to enhance robustness of the closed-loop [27, 12, 8]. However, how to tune the corresponding Kalman gain in the state-space framework for the same purposes are still open problems in the process control area.

Chapter 6

Issues on Multirate Systems

6.1 Introduction

Research on multirate systems can be traced back to the late fifties, when Kranc [50], Jury and Mullin [45], and Kalman and Bertram [46] published a series of early work. Lately, multirate issues have been discussed in the LQG/LQR designs by Berg *et al.* [10], Al-Rahmani and Franklin [4], Chen and Francis [18], and Meyer [65]; the parameterization of all stabilizing controllers by Meyer [64] and Ravi *et al.* [85]; the \mathcal{H}_2 and \mathcal{H}_∞ discrete-time and sampled-data designs handling the causality constraint by Feintuch *et al.* [34], Georgiou and Khargonekar [37], Voulgaris *et al.* [111], and Qiu and Chen [80, 19, 81], and the work by Meyer and Burrus [66], Araki and Yamamoto [6], Hagiwara and Araki [41], Colaneri *et al.* [29], and Sezer and Siljak [100].

A SISO multirate system can be represented by Figure 6.1, where P is a continuous-time plant; H_{mh} and S_{nh} are ideal multirate D/A and A/D converters with period mh and nh , respectively. We remark that m, n are coprime positive integers, for otherwise, the common factors of m and n can be absorbed into h , which is a positive real number and is the so-called base period. For this multirate open-loop system, a multirate controller can

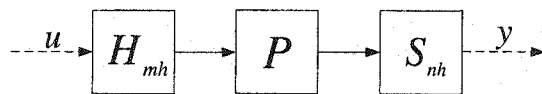


Figure 6.1: A SISO multirate sampled-data system

be designed to close the loop. This controller updates the control signal every period mh , and samples its input signal every nh .

One objective of this chapter is about the design of multirate controllers: Assume an analog controller K has been designed for the continuous-time plant P , how to implement

it on a computer through a multirate scheme? This is called the multirate discretization problem or the multirate digital redesign problem in some references.

We remark that single-rate digital redesign for a well designed continuous-time control system has been widely studied. Traditionally, the single-rate redesign is accomplished by discretizing the analog controller based on any of the conventional discretization methods [7]. However, these methods are open-loop based without consideration of closed-loop stability and performance in the discretization process. Serious performance degradation is observed when sampling rates are limited to be relatively slow. To overcome these disadvantages, the optimal digital redesign technique was proposed by Rafee *et al.* [84]. It has been shown superior to the conventional methods in aspects such as the closed-loop stability and performance, the ability to capture the inter-sample behavior, and so on. Its drawback is that the redesigned digital controllers are sometimes of high order and hence model reduction may be required after the design. This issue was solved by Rabbath *et al.* [82] with the so-called reduced-order plant input mapping technique. It enables the designer to not only constrain the order of the discrete-time controller, but also achieve a satisfactory closed-loop performance.

Multirate discretization of analog controllers is similar to the single-rate cases. Two approaches were mentioned by Rafee *et al.* in [83]: the first is the easy extension to multirate cases of the conventional single-rate discretization methods, such as those based on the step-invariant transformation and bilinear transformation. And the second is based on the optimal matching of the closed-loop step responses of the analog and digital systems. The second approach has been thoroughly studied in [83]. We will only look into the multirate digital redesign by using the first approach in this chapter, focusing on the error analysis in the frequency domain.

Most plant in industry, especially in the process industry, exhibit input-output delays. That is, the effect of a change in the manipulated variable is not felt on the process output until some time has elapsed. Time delay are mainly caused by transport delays or sometimes as the result of processes with dynamics composed of multiple chained lags. The difficulties of controlling processes with significant time delays are well known and are due to the fact that time delays produce a phase lag that deteriorates the phase margin. As a result, how to get the knowledge of the time delay in a continuous time process is important. And another objective of this chapter is to identify the unknown time delay for the continuous-time plant P in Figure 6.1, based on the multirate input-output data.

Identification of the unknown continuous time delay is generally agreed to be a difficult problem. Conventionally, discrete-time approaches will be used to obtain a discrete-time plant model (including a discrete time delay), based on the sampled input and output data. Various identification methods in the discrete time domain have been developed and can be found in the book by Söderström and Stoica [104]. However, in the event of identification of an inherently continuous-time system in terms of a discrete-time equivalent, the question of sampling is not trivial. Also, obtaining a continuous-time model from its identified discrete form has some difficulties ([103]). Furthermore, in the presence of a possible, and often unknown time delay, which may not happen to be an integral multiple of the sampling time, the resulting discrete-time model may have the undesirable non-minimum phase property. For all these reasons, efforts on continuous-time identification approaches have been made in the past decades [108]; but in some circumstances the identified continuous model is biased in the presence of noise [36, 42] except when the output is corrupted only by white noise [21].

Note that both discrete-time and continuous-time identification approaches use sampled input and output data, and most of them are in the single-rate setting. However, single-rate sampled input-output data are sometimes impractical or impossible to obtain, especially in chemical process industry. Identification of the continuous time delay by the multirate input-output data has been discussed by Li *et al.* [59], where a lifted model for the multirate system is first identified, and then the continuous time delay is extracted. In this chapter, instead of the lifted model, an interactor matrix ([113]) of the lifted model is first derived from the available multirate data by using existing techniques and algorithms, and then the continuous time delay is estimated.

The notion of an interactor matrix [113] for a multivariate system can be best understood by relating it to the meaning of the time delay for a univariate process. Interactor matrices are not exactly the reflection of time delays of MIMO systems; for example, a system without time delay can still have an interactor matrix. However, interactor matrices are closely related with time delays in a system; this explains why we manage to get an estimation of the unknown continuous time delay by the interactor matrix of the lifted model in this chapter.

Briefly, this chapter is organized as follows: In Section 6.2, we deal with the multirate digital redesign by extending the single-rate digital redesign methods: discretizing a designed analog controller by step-invariant and bilinear transformation methods. The difference between these two redesigns will be checked in the frequency domain. The continuous

time delay identification problem is discussed in Section 6.3. Properties of the time-delay matrix of the lifted model will be first investigated. Then, with the introduction of interactor matrices, we will show its relationship with the unknown continuous time delay. Results and conclusions in Sections 6.2 and 6.3 are illustrated by examples, respectively. Finally, conclusions are given in Section 6.4.

6.2 Multirate discretization of analog controllers

The multirate digital *redesign* problem can be stated as follows: assume an analog controller K has been designed for continuous-time P in Figure 6.1, then we need to obtain an approximation of K by $H_{mh}K_dS_{nh}$ for some multirate digital K_d . Such a procedure to get K_d from K is called *redesign*. This multirate digital controller K_d can have a form as follows:

$$K_d = S_{mh}KH_{nh}, \quad (6.1)$$

and can be represented by Figure 6.2, where it calculates the control signal and samples the input signal at periods mh and nh , respectively.

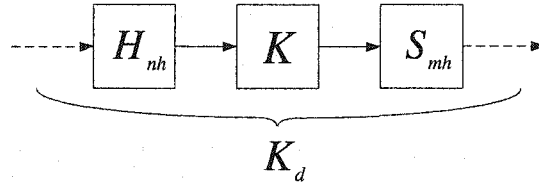


Figure 6.2: Multirate digital K_d

Introducing the discrete-time down-sampler S_m by a factor of m , and the discrete-time zero-order holder H_n defined by:

$$x' = H_n x \Leftrightarrow x'[kn + r] = x[k], \quad r = 0, 1, \dots, n - 1.$$

K_d in (6.1) then becomes:

$$K_d = S_m K_h H_n, \quad (6.2)$$

where K_h is a single-rate discretization of K with sampling period h and any discretization methods for single-rate systems are applicable for it. In this chapter, two commonly used methods will be used: the step-invariant and the bilinear transformation methods.

6.2.1 Error system

Note that for any multirate discretization of K , say, K_d , it is actually implemented by $H_{mh}K_dS_{nh}$ as shown in Figure 6.3. Thus there exists an error, say, $K - H_{mh}K_dS_{nh}$,

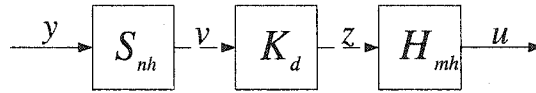


Figure 6.3: Multirate sampled-data implementation of K

between the designed analog K and its sampled-data approximation, see Figure 6.4.

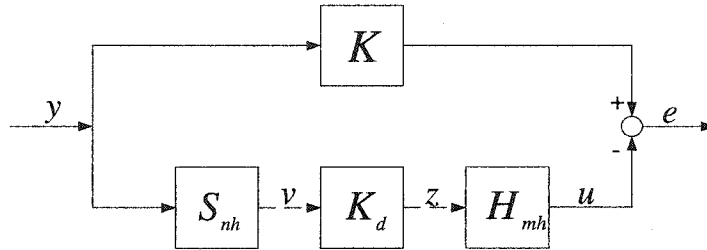


Figure 6.4: The error system

When step-invariant and bilinear transformation methods are applied to K_h (leading to different K_d), the error system in Figure 6.4 will have different responses for the same input y . The smaller the error e , the better the discretization approximation. Thus the error system can be regarded as a way to tell how well an approximation fits the designed analog controller. Since the error system is time-varying and has no transfer function, we will study it in the frequency domain by writing out the relationship between the Fourier transform of signals $\hat{e}(j\omega)$ and $\hat{y}(j\omega)$.

To do so, we first present an equivalent of Figure 6.4 as shown in Figure 6.5; and

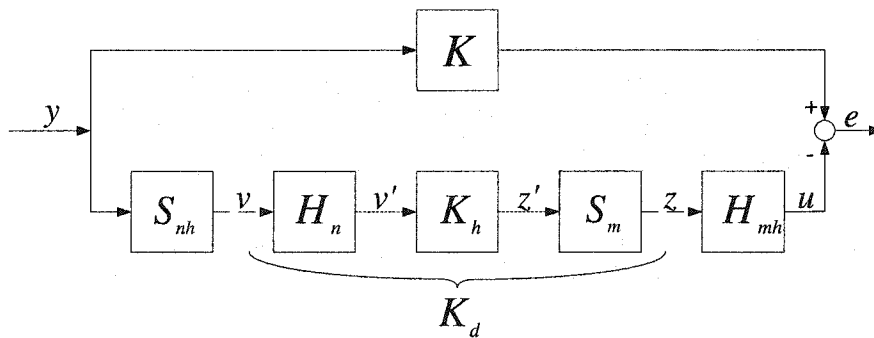


Figure 6.5: The equivalence of Figure 6.4

introduce the following lemma:

Lemma 1 In Figure 6.5,

1. The block S_{nh} maps $y(t)$ to $v[k]$; and their Fourier transforms are related by the

equation

$$\hat{v}(e^{-j\omega h}) = \frac{1}{nh} \hat{y}_e(j\omega). \quad (6.3)$$

Here $\hat{y}_e(j\omega)$ is the periodic extension of $\hat{y}(j\omega)$, i.e.,

$$\hat{y}_e(j\omega) = \sum_{k=-\infty}^{\infty} \hat{y}(j\omega + jk\omega_s),$$

and $\omega_s = \frac{2\pi}{nh}$ is the sampling frequency.

2. The discrete zero-order holder H_n maps $v[k]$ to $v'[k]$, with Fourier transforms given by the equation

$$\hat{v}'(e^{-j\omega h}) = \frac{1 - e^{-j\omega h}}{1 - e^{-j\omega h}} \hat{v}(e^{-j\omega h}). \quad (6.4)$$

3. The down-sampler S_m maps $z'[k]$ to $z[k]$, with Fourier transforms given by the equation

$$\hat{z}(e^{-j\omega m h}) = \frac{1}{m} \sum_{i=0}^{m-1} \hat{z}'(e^{-j\omega h - j\frac{2\pi i}{m}}). \quad (6.5)$$

4. The holder H_{mh} maps $z[k]$ to $u(t)$; and their Fourier transforms are related by the equation

$$\hat{u}(j\omega) = mh \cdot \hat{r}(j\omega) \hat{z}(e^{-j\omega m h}). \quad (6.6)$$

Here $r(t)$ is defined as:

$$r(t) = \begin{cases} \frac{1}{mh}, & 0 \leq t < mh, \\ 0, & \text{elsewhere,} \end{cases}$$

and therefore its Laplace transform and Fourier transform are

$$\hat{r}(s) = \frac{1 - e^{smh}}{smh}, \quad \hat{r}(j\omega) = e^{-j\omega \frac{mh}{2}} \frac{\sin \omega \frac{mh}{2}}{\omega \frac{mh}{2}}.$$

Lemma 1 allows us to write the frequency-domain relationship between $u(t)$ and $y(t)$ in Figure 6.5: from (6.3) to (6.6), we have

$$\hat{u}(j\omega) = mh \cdot \hat{r}(j\omega) \cdot \frac{1}{m} \sum_{i=0}^{m-1} \hat{z}'(e^{-j\omega h - j\frac{2\pi i}{m}}), \quad (6.7)$$

$$\hat{z}'(e^{-j\omega h}) = \hat{k}_h(e^{-j\omega h}) \frac{1 - e^{-j\omega h}}{1 - e^{-j\omega h}} \frac{1}{nh} \hat{y}_e(j\omega). \quad (6.8)$$

Thus for the error system in Figure 6.5, the error $\hat{e}(j\omega)$ is:

$$\hat{e}(j\omega) = \hat{k}(j\omega) \hat{y}(j\omega) - mh \cdot \hat{r}(j\omega) \cdot \frac{1}{m} \sum_{i=0}^{m-1} \hat{z}'(e^{-j\omega h - j\frac{2\pi i}{m}}), \quad (6.9)$$

with $\hat{z}'(e^{-j\omega h})$ defined in (6.8).

Assume that $\hat{y}(j\omega)$ is bandlimited to frequencies less than ω_N , i.e.,

$$\hat{y}(j\omega) = 0 \text{ for } \omega \geq \omega_N.$$

Then

$$\hat{y}_e(j\omega) = \hat{y}(j\omega) \text{ for } \omega < \omega_N,$$

here $\omega_N = \frac{\omega_s}{2}$ is the Nyquist frequency.

For $\omega < \omega_N$, if $m = 1$, i.e., the updating rate of the control signal is an integer (n) multiple of the sampling rate of the output signal, we have:

$$\begin{aligned} \hat{e}(j\omega) &= \hat{k}(j\omega)\hat{y}(j\omega) - h \cdot \hat{r}(j\omega) \cdot \hat{k}_h(e^{-j\omega h}) \frac{1 - e^{-j\omega nh}}{1 - e^{-j\omega h}} \frac{1}{nh} \hat{y}(j\omega), \\ &= \left[\hat{k}(j\omega) - \hat{r}(j\omega) \cdot \hat{k}_h(e^{-j\omega h}) \frac{1 - e^{-j\omega nh}}{1 - e^{-j\omega h}} \frac{1}{n} \right] \hat{y}(j\omega). \end{aligned}$$

This motivates the definition of the *error function*,

$$E(\omega) := \left| \hat{k}(j\omega) - \hat{r}(j\omega) \cdot \hat{k}_h(e^{-j\omega h}) \frac{1 - e^{-j\omega nh}}{1 - e^{-j\omega h}} \frac{1}{n} \right|, \quad (6.10)$$

and the *maximum error*,

$$E_{max} := \max_{\omega < \omega_N} E(\omega). \quad (6.11)$$

Clearly, for inputs that are bandlimited to frequencies less than ω_N , E_{max} is a measure of how closely $H_h K_d S_{nh}$ approximates K . And this error is due to three factors: (i) the presence of the zero-order hold H_{mh} , (ii) the single-rate discretization \hat{k}_h , and (iii) the integer n .

If $m > 1$, due to the properties of the discrete sampler S_m , there does not exist the so-called error function. We can only calculate $\hat{e}(j\omega)$ by making assumptions on the input $\hat{y}(j\omega)$ and using equation (6.9). Except the three factors mentioned above, error $\hat{e}(j\omega)$ is also affected by integer m .

The comparison between the two discretization approximations mentioned, based on the error system in Figure 6.5, will be illustrated next in an example.

6.2.2 Example

In this example, we take the elliptic filter in [20] as the analog controller K in Figure 6.5. It is with zeros $\pm 1.23334j$, $\pm 1.72290j$, poles -0.78280 , $-0.07543 \pm 1.05165j$, -0.379155 ± 0.875369 and gain 0.175407. Figure 6.6 shows its magnitude Bode plot.

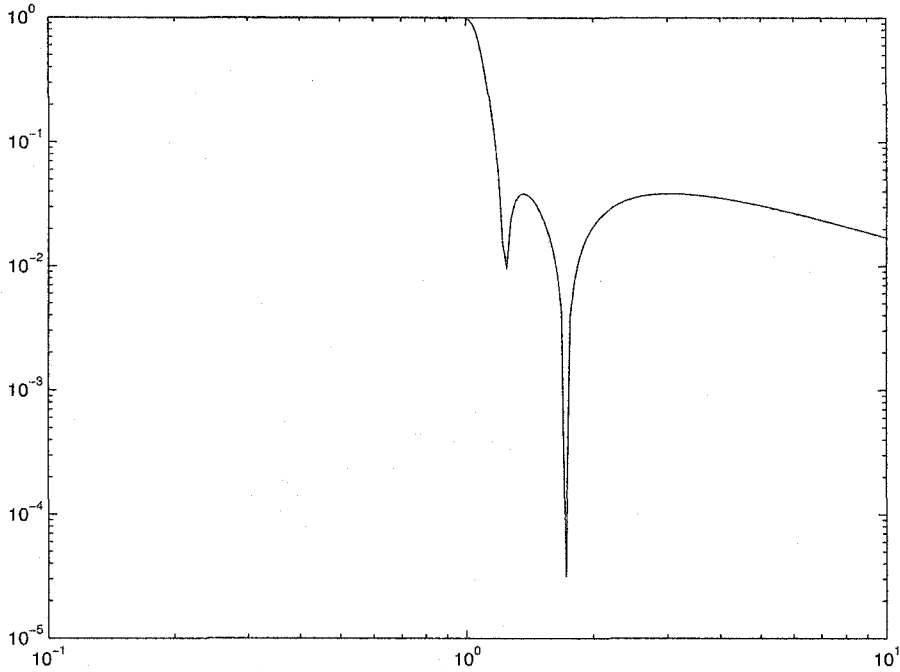


Figure 6.6: Bode plot of the elliptic filter

Now we want to approximate this analog controller by the fast-updating, slow-sampling dual-rate (with ratio n) implementation $H_h K_d S_{nh} = K_h H_n$, and we choose K_h to be the step-invariant transformation and the bilinear transformation, respectively. Based on the error system in Figure 6.5, the error function in (6.10), and the maximum error in (6.11), we will see which approximation is better.

We first try $n = 1$, the most simple case - single-rate discretization. Figure 6.7 shows the graph of $E(\omega)$ under two discretization methods. The error of the bilinear approximation with $\omega_N = 10$ is represented by the solid line and the error of the step-invariant approximation is represented by the dotted line. Also, computation gives that E_{max} for the former transformation is 0.2702 and 0.3058 for the later. All these facts indicate that the bilinear approximation is better than the step-invariant method in single-rate digital redesign.

This is also true when n , the dual-rate ratio, takes increasing integer values, see Figure 6.8, where n varies in the range $(1, 10]$. It is clear that when n increases, errors under two approximation methods tend to be the same. When n is larger than certain value, say, 6, there is almost no difference between these two digital redesign methods. Moreover, for a fixed discretization method and frequency ω , $|E(\omega)|$ will become larger with the increasing of n , see Figure 6.9 for the bilinear approach and Figure 6.10 for the step-invariant transformation.

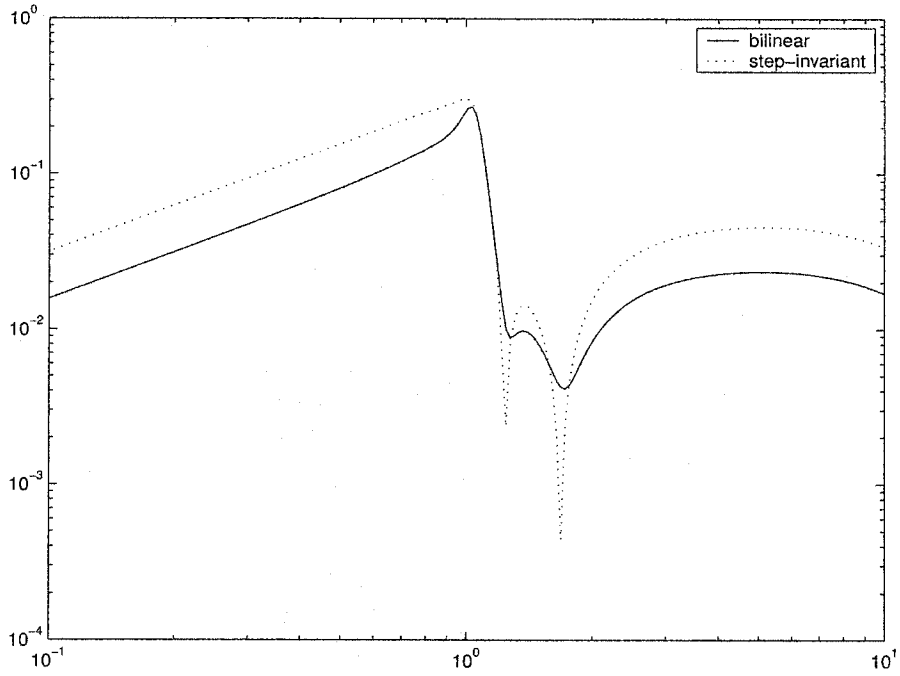


Figure 6.7: Error for single-rate discretization of Elliptic filter

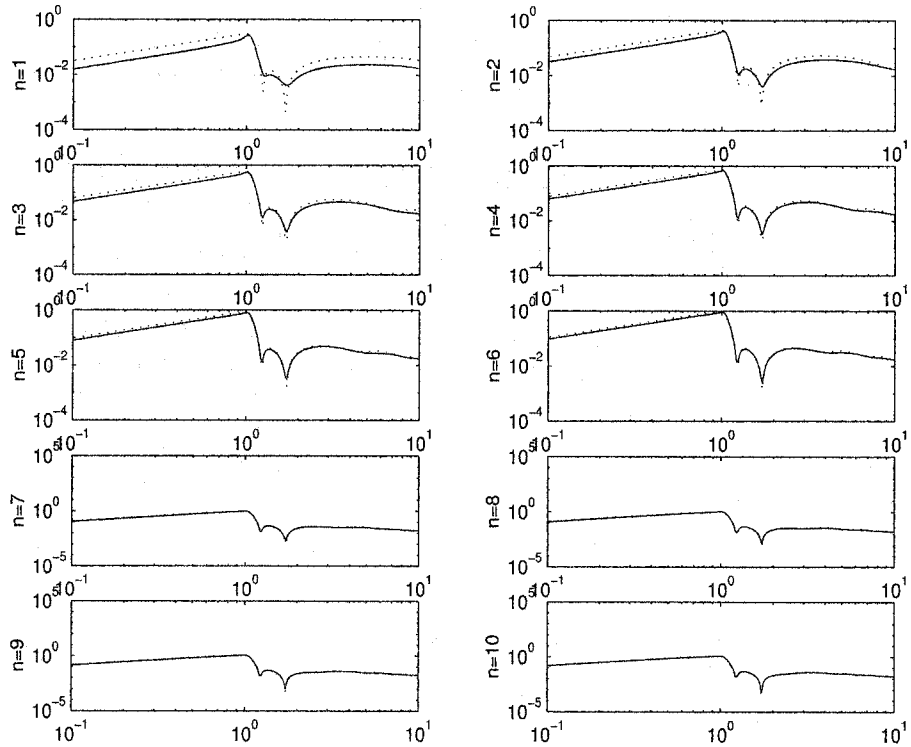


Figure 6.8: Error for dual-rate discretization with different n

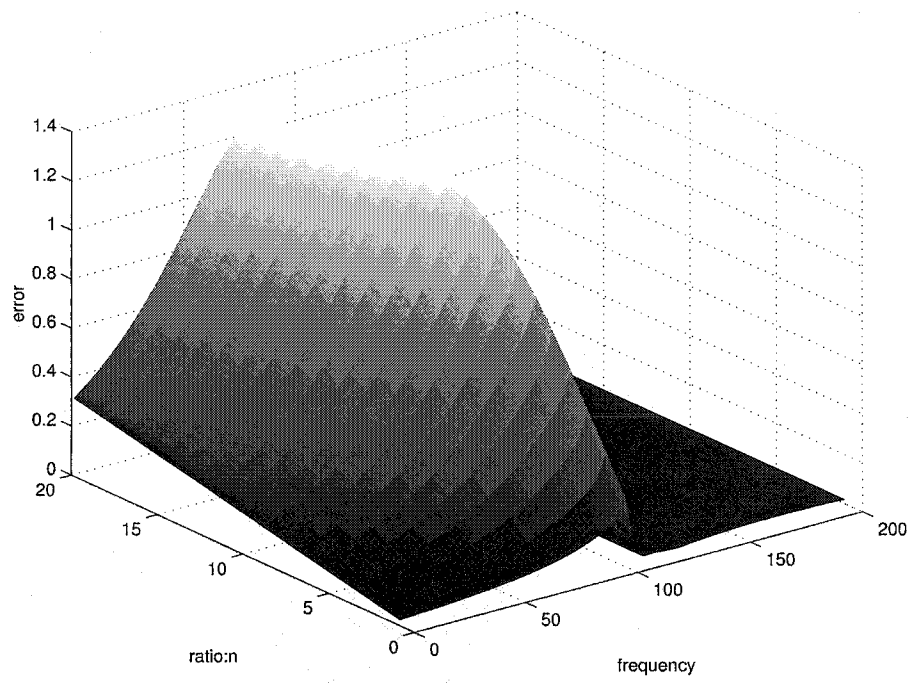


Figure 6.9: Error for bilinear approximation with different n

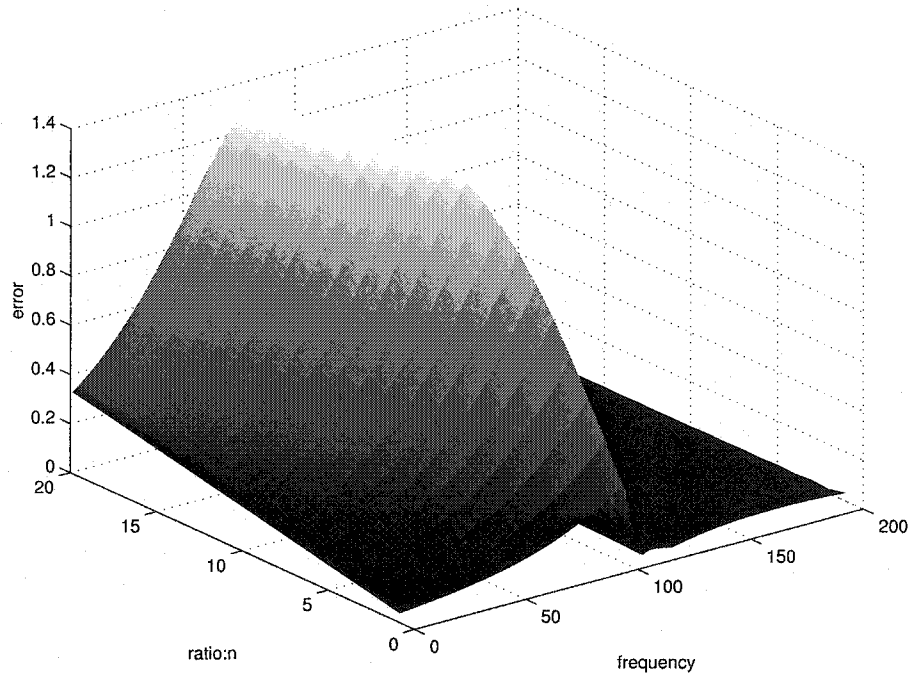


Figure 6.10: Error for step-invariant approximation with different n

n	$E_{bilinear,max}$	$E_{ZOH,max}$
1	0.2702	0.3058
2	0.4136	0.4547
3	0.5552	0.5970
4	0.6884	0.7308
5	0.8108	0.8538
6	0.9207	0.9643
7	1.0164	1.0607
8	1.0996	1.1419
9	1.1691	1.2068
10	1.2221	1.2550
11	1.2583	1.2864
12	1.2780	1.3013
13	1.2816	1.3047
14	1.2752	1.2939
15	1.2566	1.2899
16	1.2586	1.2933
17	1.2617	1.2944
18	1.2629	1.2937
19	1.2625	1.2915
20	1.2610	1.2886

Table 6.1: E_{max} for two approximations with different n

The maximum value of $E(\omega)$ also changes when n takes value in the range $[1, 20]$. Table 6.1 shows this variation for the two dual-rate digital redesign methods.

If $m \neq 1$, i.e., the analog controller is approximated by the general implementation $H_{mh}K_dS_{nh}$, to compare different approximation methods, we need to calculate $\hat{e}(j\omega)$ with assumptions on input $\hat{y}(j\omega)$. Here we take $y(t) = e^{j\omega_0 t}$, i.e., $\hat{y}(j\omega) = 2\pi\delta(\omega - \omega_0)$. By equations (6.9) and (6.8), when ω_0 changes in certain range, $\hat{e}(j\omega)$ over this range can be plotted. See Figure 6.11 and Figure 6.12, where the base period $h = \pi/10$; ω_0 varies in the range $[0.1, 10^{0.9}]$; the solid line represents the bilinear approximation and the dotted line represents the step-invariant approximation.

In Figure 6.11, m is fixed to be 2, and $n = 3, 5, 7, 9$, respectively. Similarly, in Figure 6.12, n is fixed to be 9, and $m = 2, 4, 5, 7$, respectively. Comparing with Figure 6.8, we note that the difference between the integer ratio dual-rate approximation and the multirate approximation is mainly in the high frequency range. This aliasing is introduced by the down-sampler S_m ; and the larger the integer m , the more serious the aliasing.

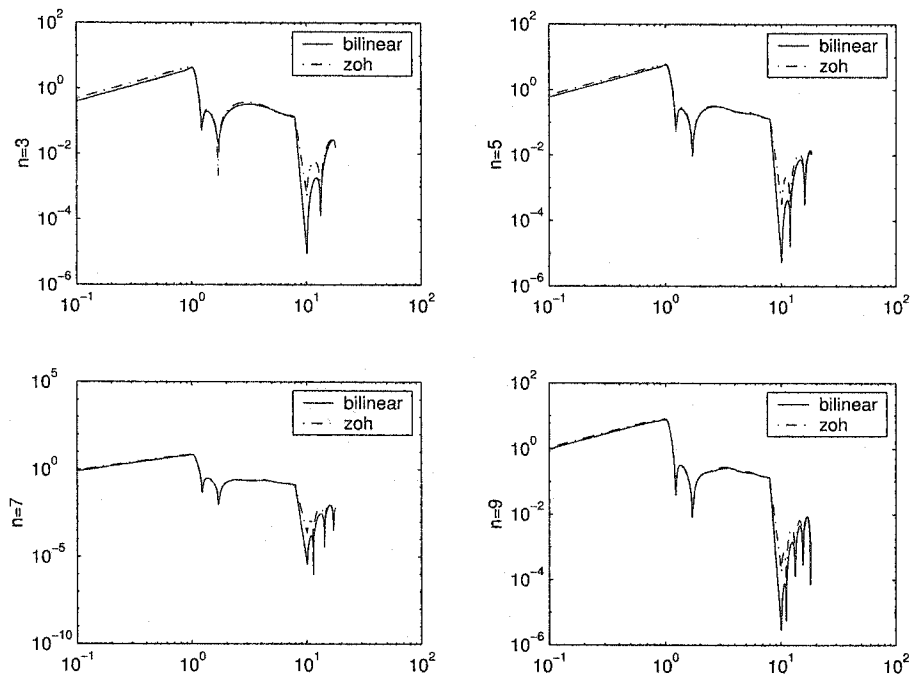


Figure 6.11: Error signal $\hat{e}(j\omega)$ for different n

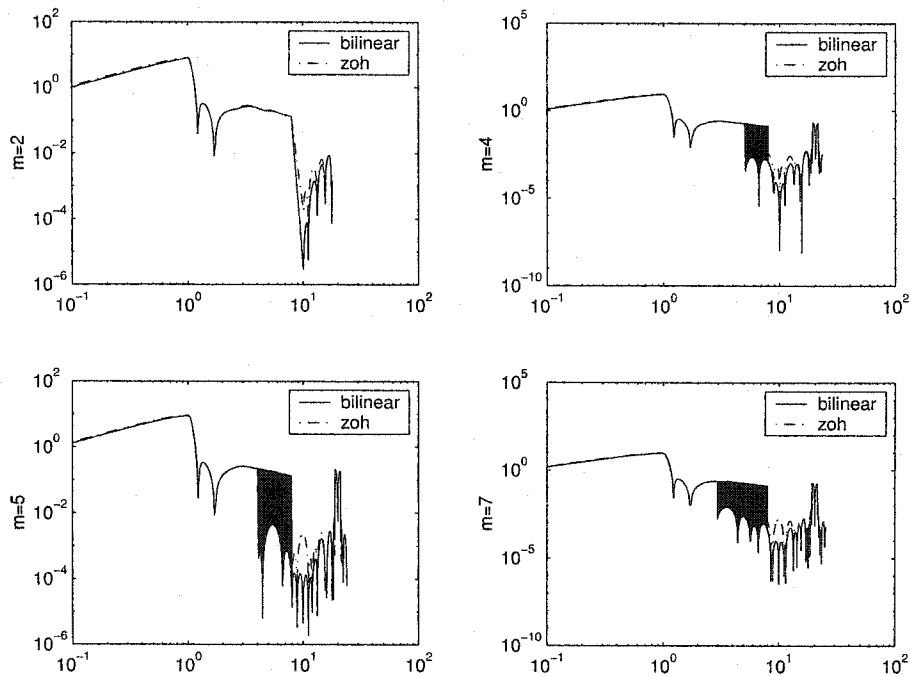


Figure 6.12: Error signal $\hat{e}(j\omega)$ for different m

6.3 Time-delay estimation of multirate systems based on interactor matrices

6.3.1 Properties of time-delay matrices of lifted models

For a SISO multirate system illustrated in Figure 6.1, we first assume the continuous-time plant P is causal and with time delay τ , where τ is a positive real number in the range $((l-1)h, lh]$, and l is a positive integer. Applying the lifting technique, we then get an LTI lifted model \underline{P} which has n inputs and m outputs and a $m \times n$ transfer matrix represented as follows:

$$\underline{P} = \begin{bmatrix} P_{00} & P_{01} & & P_{0,n-1} \\ P_{10} & P_{11} & & P_{1,n-1} \\ \vdots & \vdots & & \vdots \\ P_{m-1,0} & P_{m-1,1} & \cdots & P_{m-1,n-1} \end{bmatrix}. \quad (6.12)$$

For every subsystem P_{ij} ($i = 0, 1, \dots, m-1$, $j = 0, 1, \dots, n-1$), assuming it contains a time delay of l_{ij} , then a time-delay matrix for the lifted \underline{P} is:

$$\Gamma(\underline{P}, l) = \begin{bmatrix} l_{00} & l_{01} & \cdots & l_{0,n-1} \\ l_{10} & l_{11} & \cdots & l_{1,n-1} \\ \vdots & \vdots & & \vdots \\ l_{m-1,0} & l_{m-1,1} & \cdots & l_{m-1,n-1} \end{bmatrix}. \quad (6.13)$$

Here we use $\Gamma(\underline{P}, l)$ to show that this matrix is related to the integer l , which reflects the continuous time delay: $(l-1)h < \tau \leq lh$.

The lifted system \underline{P} maps $\underline{u}[k]$ to $\underline{y}[k]$, where

$$\underline{u}[k] = \begin{bmatrix} u(k \cdot n \cdot mh) \\ u((kn+1) \cdot mh) \\ \vdots \\ u(((k+1)n-1) \cdot mh) \end{bmatrix}, \quad \underline{y}[k] = \begin{bmatrix} y(k \cdot m \cdot nh) \\ y((km+1) \cdot nh) \\ \vdots \\ y(((k+1)m-1) \cdot mh) \end{bmatrix},$$

both having period mnh ; and the subsystem P_{ij} maps $\underline{u}_j[k] = u((kn+j) \cdot mh)$ to $\underline{y}_i[k] = y((km+i) \cdot nh)$. Note that during the first interval $[0, mnh)$, $\underline{u}_j[k]$ occurs at time $j \cdot mh$ and $\underline{y}_i[k]$ at time $i \cdot nh$; so the actual time delay from $\underline{u}_j[k]$ to $\underline{y}_i[k]$, incorporating that due to lifting, is

$$T_{ij} = \tau + jmh - inh. \quad (6.14)$$

By (6.14), if $\tau = 0$, i.e., the continuous-time model has no time delay, then the actual time-delay matrix for the lifted system is:

$$\Xi(\underline{P}) = \begin{bmatrix} 0 & mh & \cdots & (n-1)mh \\ -nh & -nh + mh & \cdots & -nh + (n-1)mh \\ \vdots & \vdots & & \vdots \\ -(m-1)nh & -(m-1)nh + mh & \cdots & -(m-1)nh + (n-1)mh \end{bmatrix}, \quad (6.15)$$

and the corresponding time-delay matrix for \underline{P} is denoted by $\Gamma(\underline{P}, 0)$.

Theorem 3 *Divide the first interval $(0, mn h]$ into mn subintervals: $((k-1)h, kh]$ ($k = 1, 2, \dots, mn$). Then there exists a 1-to-1 correspondence between $\Gamma(\underline{P}, k)$ and k . In other words, given a time-delay matrix $\Gamma(\underline{P}, k)$, the continuous time delay τ can be determined with accuracy h , i.e., $(k-1)h < \tau \leq kh$.*

To prove Theorem 3, we need the following lemmas, where we have assumed that the continuous time delay τ varies in the first interval $(0, mn h]$.

Lemma 2 *For $\Gamma(\underline{P}, k)$, $l_{ix} \leq l_{iy}$ and $l_{xj} \geq l_{yj}$, if $x < y$.*

Lemma 3 *When τ varies from the first subinterval to the mn -th subinterval, every element in $\Gamma(\underline{P}, 0)$ will change only once.*

Lemma 4 *When τ changes from the k -th subinterval to the $(k+1)$ -th subinterval, only one and must be one element in $\Gamma(\underline{P}, k)$ that changes.*

Proof of Lemma 2: During the first interval $(0, mn h]$, we have

$$\begin{cases} l_{ij} = 0, & \text{if } T_{ij} \leq 0, \\ l_{ij} = 1, & \text{if } 0 < T_{ij} \leq mn h, \\ l_{ij} = 2, & \text{if } mn h < T_{ij} \leq 2mn h. \end{cases} \quad (6.16)$$

By (6.14) and (6.16), Lemma 2 is proved. Furthermore, for a specific $\Gamma(\underline{P}, k)$ related with $(k-1)h < \tau \leq kh$, there exist inequality restrictions between every l_{ij} and their neighboring elements, which is shown below:

$$\left[\begin{array}{cccc} l_{00} & \leq & l_{01} & \leq \cdots \leq & l_{0,n-1} \\ \vee & & \vee & & \vee \\ l_{10} & \leq & l_{11} & \leq \cdots \leq & l_{1,n-1} \\ \vee & & \vee & & \vee \\ \vdots & & \vdots & & \vdots \\ \vee & & \vee & & \vee \\ l_{m-1,0} & \leq & l_{m-1,1} & \leq \cdots \leq & l_{m-1,n-1} \end{array} \right]. \quad (6.17)$$

Q.E.D

Proof of Lemma 3: For a complete proof, we first show that every element in $\Gamma(\underline{P}, 0)$ will change. Without loss of generality, we assume that the (i, j) -th element in $\Gamma(\underline{P}, 0)$ is $l_{ij} = x$. By (6.14) and (6.16), the corresponding T_{ij} satisfies:

$$(x-1)mn h < T_{ij} = jmh - inh \leq xmn h.$$

When τ is in the mn -th subinterval $((mn - 1)h, mn]h$, the (i, j) -th element in $\Gamma(\underline{P}, mn)$, say, l'_{ij} , is determined by $T'_{ij} = mn h + j m h - i n h$; here we have used the property that when τ takes different values in one range $((k - 1)h, kh]$ ($k = 1, 2, \dots, mn$), $\Gamma(\underline{P}, k)$ is the same. The limit on T'_{ij} is

$$x m n h < T'_{ij} = m n h + j m h - i n h \leq (x + 1) m n h,$$

i.e., l'_{ij} in $\Gamma(\underline{P}, mn)$ equals to $x + 1$. It is then proved that when τ varies from the first subinterval to the mn -th subinterval, every element in $\Gamma(\underline{P}, 0)$ will change (from x to $x + 1$).

Next, we will prove that any element l_{ij} can only change once. If not, assume l_{ij} changes from x to $x + 1$ and y to $y + 1$, with x and y being two different integers, then

$$\tau + j m h - i n h = x m n h + h,$$

$$\tau' + j m h - i n h = y m n h + h.$$

As a result, it is obtained that $\tau - \tau' = (x - y) m n h$; however, this is impossible, since τ only changes in the range $(0, m n h]$. Q.E.D

Proof of Lemma 4: First, we prove that only one element in $\Gamma(\underline{P}, k)$ will change. If not, say, l_{i_1, j_1} changes from 0 to 1, and l_{i_2, j_2} changes from 1 to 2, then we have

$$i_1 > i_2, \tag{6.18}$$

$$j_1 < j_2, \tag{6.19}$$

$$\tau + j_1 m h - i_1 n h = 0, \tag{6.20}$$

$$\tau + j_2 m h - i_2 n h = m n h, \tag{6.21}$$

where (6.18) and (6.19) can be easily concluded from (6.14), and (6.20) and (6.21) represent the change of l_{i_1, j_1} and l_{i_2, j_2} . From (6.20) and (6.21), we get

$$(j_2 - j_1) m + (i_1 - i_2) n = m n. \tag{6.22}$$

Since m and n are coprime, and $j_2 - j_1$ and $i_1 - i_2$ are positive integer numbers, equation (6.22) cannot be true. In other words, when τ changes from the k -th subinterval to the $(k + 1)$ -th subinterval, only one element in $\Gamma(\underline{P})$ will change.

Next, we prove that when τ changes from the k -th subinterval to the $(k + 1)$ -th subinterval, there must be one element in $\Gamma(\underline{P}, k)$ that changes. If not, $m n$ changes of the elements in $\Xi(\underline{P})$ will occur in other $m n - 1$ times of the interval changes of τ . However, this is conflicting with we just proved above. Q.E.D

Proof of Theorem 3: By Lemmas 3 and 4, when τ varies from the first subinterval to the mn -th subinterval, every element in $\Gamma(\underline{P}, 0)$ will change only once; and for different k , $\Gamma(\underline{P}, k)$ is different. Since $\Gamma(\underline{P}, k)$ corresponds to the range that τ lies, it is proved that given a time-delay matrix $\Gamma(\underline{P}, k)$, τ can be determined with accuracy h , i.e., $(k-1)h < \tau \leq kh$.

Q.E.D

To help understanding Theorem 3, here we illustrate the 1-to-1 correspondence by the case where $m = 2$ and $n = 3$. From (6.15) and (6.16), it is clear that

$$\Xi(\underline{P}) = \begin{bmatrix} 0 & 2h & 4h \\ -3h & -h & h \end{bmatrix} \Rightarrow \Gamma(\underline{P}, 0) = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad (6.23)$$

and during the period $(0, 6h]$, when τ is in different subintervals, the corresponding time-delay matrices are:

$$0 < \tau \leq h \iff \Gamma(\underline{P}, 1) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad (6.24)$$

$$h < \tau \leq 2h \iff \Gamma(\underline{P}, 2) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad (6.25)$$

$$2h < \tau \leq 3h \iff \Gamma(\underline{P}, 3) = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \end{bmatrix}, \quad (6.26)$$

$$3h < \tau \leq 4h \iff \Gamma(\underline{P}, 4) = \begin{bmatrix} 1 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix}, \quad (6.27)$$

$$4h < \tau \leq 5h \iff \Gamma(\underline{P}, 5) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 1 & 1 \end{bmatrix}, \quad (6.28)$$

$$5h < \tau \leq 6h \iff \Gamma(\underline{P}, 6) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 1 & 2 \end{bmatrix}. \quad (6.29)$$

We remark that due to the periodicity of the original multirate system, the time-delay matrix of \underline{P} has the property that if τ takes value in subinterval $(rmnh + (k-1)h, rmnh + kh]$ in the $(r+1)$ -th interval $(rmnh, (r+1)mnh]$, then

$$\Gamma(\underline{P}, rmn + k) = r \cdot E_{m \times n} + \Gamma(\underline{P}, k), \quad (6.30)$$

where r is a positive integer, and $E_{m \times n}$ is an $m \times n$ matrix with all elements being 1. For the case discussed here, if $r = 1$ and $k = 1$, i.e., $6h < \tau \leq 7h$, then (6.30) becomes

$$\Gamma(\underline{P}, 7) = \begin{bmatrix} 2 & 2 & 2 \\ 1 & 1 & 2 \end{bmatrix} = 1 \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + \Gamma(\underline{P}, 1).$$

Theorem 3 shows that for a multirate control system, once m , n and the time-delay matrix $\Gamma(\underline{P}, k)$ are known, the unknown continuous time delay can be estimated with accuracy

h , i.e., $(k-1)h < \tau \leq kh$. A question then arises is: how to get the specific $\Gamma(\underline{P}, k)$? Next we will try to answer this question by introducing the concept of interactor matrices for MIMO systems. We emphasize that a lifted model is with multi-input and/or multi-output, so it has interactor matrices, too.

6.3.2 Interactor matrices

To solve the multivariable deadbeat and minimum variance control problems, Wolovich and Falb [113], Wolovich and Elliott [114], as well as Goodwin and Sin [39] introduced the concept of interactor matrices, which is the generalization of the SISO time delay to the MIMO case.

Theorem 4 *For every $m \times n$ proper, rational polynomial transfer function matrix P , there is a unique, non-singular, $m \times m$ lower triangular polynomial matrix \mathbf{D} , such that $|\mathbf{D}| = q^\alpha$ and*

$$\lim_{q^{-1} \rightarrow 0} \mathbf{D}P = \lim_{q^{-1} \rightarrow 0} \tilde{P} = \mathbf{R} \quad (6.31)$$

where \mathbf{R} is a full rank (full column rank or full row rank) constant matrix, the integer α is defined as the number of infinite zeros of P , and \tilde{P} is the delay-free transfer function (factor) matrix of P which contains only finite zeros. The matrix \mathbf{D} is defined as the **interactor matrix** and can be written as

$$\mathbf{D} = \mathbf{D}_0 + \mathbf{D}_1q + \mathbf{D}_2q^2 + \cdots + \mathbf{D}_dq^d$$

where d is denoted as the order of the interactor matrix and is unique for a given transfer function matrix (Shah et al. [101]; Mutoh and Ortega [71]), and \mathbf{D}_i (for $i = 0, \dots, d$) are coefficient matrices.

The matrix \mathbf{D} also has forms other than the lower triangular one. It can be one of the three forms described in the sequel:

- If \mathbf{D} is of the form, $\mathbf{D} = q^d I$, then it is regarded as a simple interactor matrix.
- If \mathbf{D} is a diagonal matrix, i.e., $\mathbf{D} = \text{diag}(q^{d_1}, q^{d_2}, \dots, q^{d_m})$, then it is regarded as a diagonal interactor matrix.
- Otherwise \mathbf{D} is regarded as a general interactor matrix, which can be a full matrix, an upper triangular matrix (Shah et al. [101]; Huang et al. [43]), a nilpotent matrix (Rogozinski et al. [92]), or a unitary interactor matrix (Peng and kinnaert [76]).

Definition 1 *Instead of taking the lower triangular form, if an interactor matrix in Theorem 4 satisfies*

$$\mathbf{D}^T(q^{-1})\mathbf{D}(q) = I$$

then this interactor matrix is denoted as a unitary interactor matrix.

The unitary interactor matrix is an all-pass factor, as a delay term should be. The existence of the unitary interactor matrix has been established by Peng and Kinnaert [76], where it is also pointed out that the unitary interactor is non-unique. However, it is unique from the algorithm by Peng and Kinnaert [76] and Rogozinski *et al.* [92]. Later in illustrative examples, we will use their algorithms to calculate interactor matrices of lifted models based on multirate input-output data.

6.3.3 Estimation of continuous time delays

In the following, we assume:

- The continuous-time plant P has unknown time delay τ .
- Interactor matrix \mathbf{D} of the lifted model \underline{P} and the corresponding full rank matrix \mathbf{R} are both available.

We will show next that under certain conditions, the continuous time delay τ can be estimated with accuracy h by the knowledge of \mathbf{D} and \mathbf{R} .

For the lifted system \underline{P} with transfer function matrix (6.12), the interactor matrix \mathbf{D} is an $m \times m$ square one, say,

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_{00} & \cdots & \mathbf{D}_{0,m-1} \\ \vdots & & \vdots \\ \mathbf{D}_{m-1,0} & \cdots & \mathbf{D}_{m-1,m-1} \end{bmatrix}.$$

Assume

$$\begin{aligned} P_{ij} &= q^{-l_{ij}} P_{ij}^{(l_{ij})} + q^{-l_{ij}-1} P_{ij}^{(l_{ij}+1)} + \cdots, \\ &= q^{-l_{ij}} (P_{ij}^{(l_{ij})} + q^{-1} P_{ij}^{(l_{ij}+1)} + \cdots), \end{aligned} \quad (6.32)$$

and

$$\begin{aligned} \mathbf{D}_{ij} &= q^{d_{ij}} \mathbf{D}_{ij}^{(d_{ij})} + q^{d_{ij}-1} \mathbf{D}_{ij}^{(d_{ij}-1)} + \cdots + q \mathbf{D}_{ij}^{(1)} + \mathbf{D}_{ij}^{(0)}, \\ &= q^{d_{ij}} (\mathbf{D}_{ij}^{(d_{ij})} + q^{-1} \mathbf{D}_{ij}^{(d_{ij}-1)} + \cdots + q^{-d_{ij}+1} \mathbf{D}_{ij}^{(1)} + q^{-d_{ij}} \mathbf{D}_{ij}^{(0)}), \end{aligned} \quad (6.33)$$

where l_{ij} belongs to $\Gamma(\underline{P}, l)$ in (6.13) and represents the time delay of the subsystem P_{ij} ; and l_{ij}, d_{ij} are both non-negative integers. Then by the definition of interactor matrix in (6.31), the element of \mathbf{R} , i.e., R_{ij} ($i = 0, \dots, m-1; j = 0, \dots, n-1$) is given by:

$$R_{ij} = \lim_{q^{-1} \rightarrow 0} \sum_{r=0}^{m-1} q^{d_{ir}-l_{rj}} [\mathbf{D}_{ir}^{(d_{ir})} P_{rj}^{(l_{rj})} + q^{-1} R_{ij}^{(r)}(q^{-1})], \quad (6.34)$$

where $R_{ij}^{(r)}(q^{-1})$ represents a polynomial in IIR form.

Since we assume the interactor matrix \mathbf{D} and the constant matrix \mathbf{R} are both known, then if we can limit every l_{ij} within certain ranges, i.e., get the time-delay matrix $\Gamma(\underline{P}, l)$, we can determine τ with accuracy h immediately by Theorem 3. So the key step here is to obtain $\Gamma(\underline{P}, l)$ by the knowledge of \mathbf{D} (it can be in any form as described before) and \mathbf{R} . The feasibility of this step still needs to be proved in general. While the procedure will be much more simple when \mathbf{D} has a diagonal form, or a form where every row or column has only one non-zero element.

Theorem 5 *Assume (i) the D/A rate is higher than the A/D rate, i.e., $m < n$; (ii) the lifted model \underline{P} has an interactor matrix \mathbf{D} in the diagonal form, or a form where every row or column has only one non-zero element. Then the continuous time delay τ can be determined with accuracy h , i.e., we can find l such that $(l-1)h < \tau \leq lh$.*

To help proving Theorem 5, let us first look at the following lemma. Here we assume m and n both are integers larger than 1.

Lemma 5 *For a specific time-delay matrix $\Gamma(\underline{P}, l)$ in correspondence with $(l-1)h < \tau \leq lh$,*

1. *the difference between the maximum and the minimum l_{ij} can only be 1 or 2;*
2. *the difference between the maximum and the minimum l_{ij} in one row can only be 0 or 1;*
3. *the difference between the maximum and the minimum l_{ij} in one column can only be 0 or 1;*
4. *the difference between l_{ij} and its four neighbors: $l_{i-1,j}$, $l_{i,j-1}$, $l_{i,j+1}$, and $l_{i+1,j}$, can only be 0 or 1;*
5. *the difference between l_{ij} and its four diagonal neighbors: $l_{i-1,j-1}$, $l_{i-1,j+1}$, $l_{i+1,j-1}$, and $l_{i+1,j+1}$, can only be 0 or 1.*

Proof of Lemma 5:

1. From (6.17) in Lemma 2, the maximum l_{ij} in $\Gamma(\underline{P}, l)$ is $l_{0,n-1}$ and the minimum is $l_{m-1,0}$. Assume $l_{m-1,0} = x$, i.e.,

$$(x-1)mnh < T_{m-1,0} = \tau - (m-1)nh \leq xmnh,$$

then for $T_{0,n-1} = \tau + (n-1)mh$, the following inequality exists:

$$(x-1)mnh + 2mnh - mh - nh < T_{0,n-1} \leq xmnh + 2mnh - mh - nh.$$

By assumptions $m > 1$ and $n > 1$, it is true that

$$mh + nh < mnh,$$

thus we get

$$xmnh < T_{0,n-1} \leq (x+2)mnh - mh - nh.$$

This limit on $T_{0,n-1}$ demonstrates that $l_{0,n-1}$ can only be $x+1$ or $x+2$; in other words, the difference between $l_{0,n-1}$ and $l_{m-1,0}$ can only be 1 or 2.

2. By (6.17), the maximum and the minimum l_{ij} in the i -th row of $\Gamma(\underline{P}, l)$ are $l_{i,n-1}$ and $l_{i,0}$, respectively. Assume $l_{i,0} = x$, i.e.,

$$(x-1)mnh < T_{i,0} = \tau - inh \leq xmnh,$$

then the maximum $T_{i,n-1}$ satisfies

$$xmnh - mh < T_{i,n-1} = \tau + (n-1)mh - inh \leq (x+1)mnh - mh.$$

This indicates that $l_{i,n-1}$ can take a value as x or $x+1$, and correspondingly the difference between $l_{i,n-1}$ and $l_{i,0}$ can only be 0 or 1.

3. Similar as the i -th row, the j -th column of $\Gamma(\underline{P}, l)$ also has the property that the difference between the maximum and the minimum l_{ij} , say, $l_{0,j}$ and $l_{m-1,j}$ can only be 0 or 1.

4. Assume $l_{ij} = x$, i.e., $(x-1)mnh < T_{ij} = \tau + jmh - inh \leq xmnh$, then

$$(x-1)mnh - mh < T_{i,j-1} = \tau + (j-1)mh - inh \leq xmnh - mh,$$

$$(x-1)mnh + mh < T_{i,j+1} = \tau + (j+1)mh - inh \leq xmnh + mh,$$

$$(x-1)mnh + nh < T_{i-1,j} = \tau + jmh - (i-1)nh \leq xmnh + nh,$$

$$(x-1)mnh - nh < T_{i+1,j} = \tau + jmh - (i+1)nh \leq xmnh - nh.$$

These limits indicate that $l_{i,j-1}$ and $l_{i+1,j}$ can take a value as $x - 1$ or x ; $l_{i,j+1}$ and $l_{i-1,j}$ can take a value as x or $x + 1$. In other words, the differences between l_{ij} and its four neighbors can only be 0 or 1.

5. Similarly as above, with the assumption $l_{ij} = x$, the corresponding real time delays of its four diagonal neighbors are limited by:

$$\begin{aligned} (x - 1)mnh + nh - mh &< T_{i-1,j-1} \leq xmnh + nh - mh, \\ (x - 1)mnh + mh + nh &< T_{i-1,j+1} \leq xmnh + mh + nh, \\ (x - 1)mnh - mh - nh &< T_{i+1,j-1} \leq xmnh - mh - nh, \\ (x - 1)mnh + mh - nh &< T_{i+1,j+1} \leq xmnh + mh - nh. \end{aligned}$$

Clearly, the differences between l_{ij} and its four diagonal neighbors can only be 0 or 1.

Q.E.D

Proof of Theorem 5: Assume in the i -th row and the j -th column of \mathbf{D} , \mathbf{D}_{ij} is the only non-zero element; according to the definition for interactor matrices, we have:

$$R_{ir} = \lim_{q^{-1} \rightarrow 0} q^{d_{ij}-l_{jr}} [\mathbf{D}_{ij}^{(d_{ij})} P_{jr}^{(l_{jr})} + q^{-1} R_{ir}^{(j)}(q^{-1})], \quad r = 0, 1, \dots, (n - 1).$$

Thus the range for l_{jr} ($j = 0, 1, \dots, m - 1$) can be determined as follows:

- If $R_{ir} = 0$, then

$$d_{ij} - l_{jr} < 0 \implies l_{jr} > d_{ij}. \quad (6.35)$$

- If $R_{ir} \neq 0$, then

$$d_{ij} - l_{jr} = 0 \implies l_{jr} = d_{ij}. \quad (6.36)$$

Let i varies from 0 to $m - 1$, an initial form of the time-delay matrix Γ is obtained. We call it an initial form because some elements of Γ may not have exact values: they are given in certain ranges as shown in (6.35). To derive an exact Γ , note that the constant $m \times n$ matrix \mathbf{R} is full row rank. That means every row of \mathbf{R} has at least one non-zero element and subsequently every row of Γ has at least one element which can be decided according to (6.36). Assume $l_{\alpha\beta}$ in the α -th row is x , then other elements in the same row with limitations as (6.35) can only be $x + 1$. Otherwise they are conflicting with Lemma 5, where we have proved that the difference between the maximum and the minimum elements in the same row cannot exceed 1.

Once the time-delay matrix $\Gamma(\underline{P}, l)$ is obtained, τ can be estimated immediately with accuracy h by Theorem 3. Q.E.D

We remark that the method to identify the continuous time delay, provided by Theorem 5, has advantages due to the following properties of interactor matrices:

- The interactor matrix (\mathbf{D}_{cl}) of the closed-loop transfer function matrix is the same as the interactor matrix (\mathbf{D}) of the open-loop transfer function matrix ([44]). This is also true for lifted systems.
- The interactor matrix can be calculated independent of the transfer function matrix; only the first several Markov parameter matrices (or impulse response coefficient matrices) are required. In our method, Markov parameter matrices of the lifted model can be obtained by the measured multirate input and output data, using certain existing techniques.

Due to these advantages, when the continuous time plant (including the delay) is unknown, it is still possible for us to calculate the interactor matrix of the closed loop based on the multirate input-output data, and furthermore to estimate the delay.

We also remark that when a multirate system fails to meet the condition $m < n$, but has a diagonal like interactor matrix, the unknown continuous time delay τ is still possible to be identified by taking into account the inequality restrictions in (6.17) and the limitations described in Lemma 5; although it is difficult to prove that τ can definitely be computed. When the control signal is updated at a slower rate than that for sampling the output signal, i.e, $m > n$, \mathbf{R} is of full column rank. It implies that every column of \mathbf{R} has at least one non-zero element and every column of Γ has at least one element with exact value. These values are useful to determine other corresponding elements in their same rows (see the proof for Theorem 5), thus totally $n \times n$ elements in Γ can be fixed. However, there is no guarantee for all other $(m \times n) - (n \times n)$ elements. Fortunately, most of the multirate systems in industry adopt fast-updating and slow-sampling strategy, thus Theorem 5 can be applied to a wide class of industrial systems.

If a multirate system does not have a diagonal like interactor matrix, the estimation of τ becomes more complicated. However, we may try a more direct method. For this purpose, we first introduce the following lemma which is from Chapter 3 in [20], where with reference to a state model, the packed notation

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] \tag{6.37}$$

denotes the transfer function, $D + C(\delta I - A)^{-1}B$. Note that δ stands for the Laplace variable s in the continuous-time context and for the Z -transform variable z in the discrete-time context; and I represents the identity matrix with the same dimension as state matrix A .

Lemma 6 *If a continuous-time system P (with delay τ) has transfer matrix*

$$\hat{p}(s) = \left[\begin{array}{c|c} A & B \\ \hline C & 0 \end{array} \right] e^{-\tau s}, \quad (6.38)$$

then its step-invariant equivalent (with discretization period h) has transfer matrix

$$\hat{p}_h(q^{-1}) = q^{-l} \left[\begin{array}{c|c} A_d & B_d \\ \hline C_d & D_d \end{array} \right], \quad (6.39)$$

where $A_d = e^{hA}$, $B_d = \int_0^h e^{\tau A} d\tau B$, $C_d = C e^{(lh-\tau)A}$, $D_d = C \int_0^{lh-\tau} e^{tA} dt B$, and l is the integer such that the continuous time delay τ lies in the sampling interval $((l-1)h, lh]$.

Now for the multirate system shown in Figure 6.1, denote P_h as the step-invariant equivalent of the continuous P , and notice that $H_{mh} = H_h H_m$ and $S_{nh} = S_n S_h$, a pure discrete-time system is obtained, see Figure 6.13.

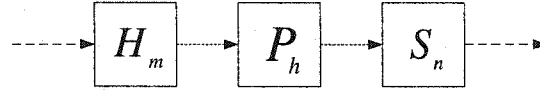


Figure 6.13: Discrete-time equivalent of Figure 6.1

Introducing the lifting and inverse lifting operators, we can get an LTI lifted system $\underline{P} = L_m S_n P_h H_m L_n^{-1}$ as shown in Figure 6.14, where the lifted model \underline{P} can be regarded as a system with n inputs and m outputs, and is LTI with underlying period mnh .

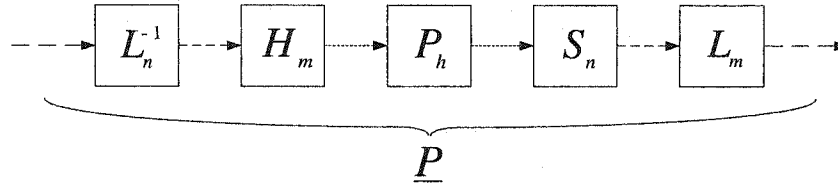


Figure 6.14: Lifted LTI system

To calculate \underline{P} , we first assume P in Figure 6.1 is represented by (6.38). According to Lemma 6, P_h then has a form of (6.39), which is equivalent to the following impulse response transfer function:

$$\hat{P}_h(q^{-1}) = \sum_{i=0}^{\infty} q^{-l} p_i, \quad (6.40)$$

and $p_0 = p_1 = \dots = p_{l-1} = 0$.

Rewrite (6.40) as follows:

$$\hat{P}_h(q^{-1}) = \sum_{i=0}^{mn-1} q^{-i} \hat{P}_i(q^{-mn}), \quad (6.41)$$

where

$$\hat{P}_i(q^{-mn}) = \sum_{k=0}^{\infty} q^{-k \times mn} p_{i+k \times mn},$$

then the transfer matrix of \underline{P} can be written in terms of the impulse sequence p_i in (6.40).

For example, if $m = 2$ and $n = 3$, then

$$\underline{P} = \begin{bmatrix} \hat{P}_0 + q^{-1} \hat{P}_5 & q^{-1} \hat{P}_4 + q^{-1} \hat{P}_3 & q^{-1} \hat{P}_2 + q^{-1} \hat{P}_1 \\ \hat{P}_3 + \hat{P}_2 & \hat{P}_1 + \hat{P}_0 & q^{-1} \hat{P}_5 + q^{-1} \hat{P}_4 \end{bmatrix}; \quad (6.42)$$

and if $m = 3$ and $n = 2$, then

$$\underline{P} = \begin{bmatrix} \hat{P}_0 + q^{-1} \hat{P}_5 + q^{-1} \hat{P}_4 & q^{-1} \hat{P}_3 + q^{-1} \hat{P}_2 + q^{-1} \hat{P}_1 \\ \hat{P}_2 + \hat{P}_1 + \hat{P}_0 & q^{-1} \hat{P}_5 + q^{-1} \hat{P}_4 + q^{-1} \hat{P}_3 \\ \hat{P}_4 + \hat{P}_3 + \hat{P}_2 & \hat{P}_1 + \hat{P}_0 + q^{-1} \hat{P}_5 \end{bmatrix}. \quad (6.43)$$

From preceding derivations we see that the transfer function of \underline{P} is associated with the continuous time delay τ . If \underline{P} is known, it is possible to estimate l by confirming $p_0 = p_1 = \dots = p_{l-1} = 0$ and $p_l \neq 0$; then the continuous time delay τ is immediately limited into the range $(l-1)h < \tau \leq lh$.

6.3.4 Examples

To demonstrate the effectiveness of the determination procedure provided by Theorem 5, and the possibility to estimate τ for systems (i) having $m > n$ but with diagonal like interactor matrix, (ii) only having general form interactor matrix. Three illustrative examples will be given next. We emphasize that the interactor matrix can be computed by different algorithms. Here it is calculated by algorithms provided in [92] and [76]. The continuous process P in Fig. 6.1 is represented by the following transfer function:

$$P(s) = e^{-\tau s} \frac{1}{s+1}. \quad (6.44)$$

Example 1: If the control signal is updated every $2h$ ($m = 2$), and the output is sampled every $3h$ with $h = 1$ sec, i.e., $m < n$; and if the lifted model has diagonal like interactor matrices, τ can definitely be decided according to Theorem 5. Here we will illustrate this by 2 cases based on the information of the interactor matrix \mathbf{D} and the corresponding matrix \mathbf{R} .

Case 1:

$$\mathbf{D} = \begin{bmatrix} 0 & -1 \\ -q & 0 \end{bmatrix}.$$

According to \mathbf{D} , we have

$$\begin{aligned} R_{0j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{01}-l_{1j}} [\mathbf{D}_{01}^{(d_{01})} P_{1j}^{(l_{1j})} + q^{-1} R_{0j}^{(1)}(q^{-1})], \\ R_{1j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{10}-l_{0j}} [\mathbf{D}_{10}^{(d_{10})} P_{0j}^{(l_{0j})} + q^{-1} R_{1j}^{(0)}(q^{-1})], \end{aligned}$$

where $d_{01} = 0$, $d_{10} = 1$, $j = 0, 1, 2$. If

$$R = \begin{bmatrix} \times & \times & 0 \\ \times & \times & \times \end{bmatrix},$$

with \times representing nonzero numbers, then the range for l_{ij} and furthermore the delay τ can be determined as follows:

$$\left. \begin{array}{l} l_{00} = 1, l_{01} = 1, l_{02} = 1, \\ l_{10} = 0, l_{11} = 0, l_{12} > 0, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow 0 < \tau \leq 1 \text{ sec.}$$

Note that l_{12} is originally only limited to the range $l_{12} > 0$, but we can get its exact value, i.e., $l_{12} = 1$ referring to (6.17) and Lemma 5; consequently, τ is determined by Theorem 3 and (6.24).

Similarly, if

$$R = \begin{bmatrix} \times & 0 & 0 \\ \times & \times & 0 \end{bmatrix},$$

then

$$\left. \begin{array}{l} l_{00} = 1, l_{01} = 1, l_{02} > 1, \\ l_{10} = 0, l_{11} > 0, l_{12} > 0, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \end{bmatrix} \Rightarrow 2 \text{ sec} < \tau \leq 3 \text{ sec.}$$

Here we have applied Lemma 5 to determine l_{02} , l_{11} and l_{12} . The minimum l_{ij} is $l_{10} = 0$ and the maximum is l_{02} which satisfies $1 < l_{02} \leq 2 \rightarrow l_{02} = 2$; one neighbor of l_{10} is l_{11} and it should be $0 < l_{11} \leq 1 \rightarrow l_{11} = 1$; the maximum l_{ij} in the second row is l_{12} and it is limited by $0 < l_{12} \leq 1 \rightarrow l_{12} = 1$. According to Theorem 3 and (6.26), τ is easily estimated.

Case 2:

$$\mathbf{D} = \begin{bmatrix} 0 & -q \\ -q & 0 \end{bmatrix}.$$

In this case we have

$$\begin{aligned} R_{0j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{01}-l_{1j}} [\mathbf{D}_{01}^{(d_{01})} P_{1j}^{(l_{1j})} + q^{-1} R_{0j}^{(1)}(q^{-1})], \\ R_{1j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{10}-l_{0j}} [\mathbf{D}_{10}^{(d_{10})} P_{0j}^{(l_{0j})} + q^{-1} R_{1j}^{(0)}(q^{-1})], \end{aligned}$$

where $d_{01} = 1$, $d_{10} = 1$, $j = 0, 1, 2$. If

$$R = \begin{bmatrix} \times & \times & \times \\ \times & 0 & 0 \end{bmatrix},$$

then

$$\left. \begin{array}{l} l_{00} = 1, l_{01} > 1, l_{02} > 1, \\ l_{10} = 1, l_{11} = 1, l_{12} = 1, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 1 & 1 \end{bmatrix} \Rightarrow 4 \text{ sec} < \tau \leq 5 \text{ sec}.$$

Note here τ is determined by Theorem 3 and (6.28). The integers l_{01} and l_{02} can only be 2 for the reason that the former is one diagonal neighbor of l_{10} , and the later is one diagonal neighbor of l_{11} .

If

$$R = \begin{bmatrix} \times & \times & 0 \\ \times & 0 & 0 \end{bmatrix},$$

then

$$\left. \begin{array}{l} l_{00} = 1, l_{01} > 1, l_{02} > 1, \\ l_{10} = 1, l_{11} = 1, l_{12} > 1, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 1 & 2 \end{bmatrix} \Rightarrow 5 \text{ sec} < \tau \leq 6 \text{ sec}.$$

Similarly, here τ is estimated by using Theorem 3 and (6.29). l_{01} and l_{02} can only be 2 because they are diagonal neighbors of l_{10} and l_{11} , respectively. l_{12} is equal to 2 due to its position in the second row: it has the maximum value and cannot exceed 2.

Example 2: For the continuous time plant P , if the system is updating the control inputs every $3h$ and sampling the outputs every $2h$ with $h = 1$ sec, i.e., $m > n$, and has a diagonal like interactor matrix, it is still possible to estimate τ .

Case 1:

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & -1 \\ -q & 0 & 0 \\ 0 & q & 0 \end{bmatrix}.$$

For this case we have

$$\begin{aligned} R_{0j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{02}-l_{2j}} [\mathbf{D}_{02}^{(d_{02})} P_{2j}^{(l_{2j})} + q^{-1} R_{0j}^{(2)}(q^{-1})], \\ R_{1j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{10}-l_{0j}} [\mathbf{D}_{10}^{(d_{10})} P_{0j}^{(l_{0j})} + q^{-1} R_{1j}^{(0)}(q^{-1})], \\ R_{2j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{21}-l_{1j}} [\mathbf{D}_{21}^{(d_{21})} P_{1j}^{(l_{1j})} + q^{-1} R_{2j}^{(1)}(q^{-1})], \end{aligned}$$

where $d_{02} = 0$, $d_{10} = 1$, $d_{21} = 1$, $j = 0, 1$. If

$$R = \begin{bmatrix} \times & 0 \\ \times & \times \\ \times & \times \end{bmatrix},$$

then

$$\left. \begin{array}{l} l_{20} = 0, l_{21} > 0, \\ l_{00} = 1, l_{01} = 1, \\ l_{10} = 1, l_{11} = 1, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \Rightarrow 2 \text{ sec} < \tau \leq 3 \text{ sec}.$$

Here l_{21} can only be 1 because it lies in the same row as $l_{20} = 0$.

Case 2:

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & q \\ 0 & q & 0 \\ q & 0 & 0 \end{bmatrix}.$$

Correspondingly, R_{ij} 's are

$$\begin{aligned} R_{0j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{02}-l_{2j}} [\mathbf{D}_{02}^{(d_{02})} P_{2j}^{(l_{2j})} + q^{-1} R_{0j}^{(2)}(q^{-1})], \\ R_{1j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{11}-l_{1j}} [\mathbf{D}_{11}^{(d_{11})} P_{1j}^{(l_{1j})} + q^{-1} R_{1j}^{(1)}(q^{-1})], \\ R_{2j} &= \lim_{q^{-1} \rightarrow 0} q^{d_{20}-l_{0j}} [\mathbf{D}_{20}^{(d_{20})} P_{0j}^{(l_{0j})} + q^{-1} R_{2j}^{(0)}(q^{-1})], \end{aligned}$$

where $d_{02} = 1, d_{11} = 1, d_{20} = 1, j = 0, 1$. If

$$\mathbf{R} = \begin{bmatrix} \times & \times \\ \times & \times \\ \times & 0 \end{bmatrix},$$

then

$$\left. \begin{array}{l} l_{20} = 1, l_{21} = 1, \\ l_{10} = 1, l_{11} = 1, \\ l_{00} = 1, l_{01} > 1, \end{array} \right\} \Rightarrow \Gamma(\underline{P}) = \begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \Rightarrow 4 \text{ sec} < \tau \leq 5 \text{ sec}.$$

Here l_{01} can only be 2 because it lies in the same row as $l_{00} = 1$ and the same column as $l_{11} = l_{21} = 1$.

Example 3: For the same updating and sampling strategy as in Example 2 for the continuous time plant P , i.e., $m = 3 > n = 2$; if the 3×3 matrix \mathbf{D} is in a general form, there still exists possibility that τ can be identified, although the determination procedure may become a little bit complicated. For example, one estimation of \mathbf{D} turns out to be:

$$\mathbf{D} = \begin{bmatrix} 0 & -0.3940 & -0.9191 \\ -0.9191q & -0.3621q & 0.1552q \\ -0.3940q & 0.8448q & -0.3621q \end{bmatrix},$$

and the corresponding \mathbf{R} is

$$\mathbf{R} = \begin{bmatrix} \times & 0 \\ \times & \times \\ \times & \times \end{bmatrix}.$$

From (6.43), \underline{P} can be rewritten as:

$$\underline{P} = \begin{bmatrix} p_0 & 0 \\ p_0 + p_1 + p_2 & 0 \\ p_2 + p_3 + p_4 & p_0 + p_1 \end{bmatrix} + q^{-1} \begin{bmatrix} p_4 + p_5 + p_6 & p_1 + p_2 + p_3 \\ p_6 + p_7 + p_8 & p_3 + p_4 + p_5 \\ p_8 + p_9 + p_{10} & p_5 + p_6 + p_7 \end{bmatrix} + \dots$$

Writing out the definition equation

$$\lim_{q^{-1} \rightarrow 0} \underline{D}\underline{P} = \mathbf{R}$$

in details, the following equations are enough to determine τ :

$$\begin{aligned} p_0 + p_1 &= 0, \\ -0.3940p_2 - 0.9191(p_2 + p_3 + p_4) &= R_{00}, \\ -0.9191p_0 - 0.3621(p_0 + p_1 + p_2) + 0.1552(p_2 + p_3 + p_4) &= 0, \\ -0.3940p_0 + 0.8448(p_0 + p_1 + p_2) - 0.3621(p_2 + p_3 + p_4) &= 0, \end{aligned}$$

and the result is:

$$p_0 = 0, p_1 = 0, p_2 \neq 0 \implies l = 2 \implies 1 \text{ sec} < \tau \leq 2 \text{ sec}.$$

6.4 Conclusions

In this chapter, we first looked into the multirate digital redesign problem. It is an extension of the single-rate digital redesign methods by discretizing analog controllers with bilinear and step-invariant transformation methods. The example shows that in a fast-updating slow-sampling dual-rate setting, the bilinear method is better than the step-invariant method; but with the increasing of n , this advantage tends to be negligible.

Multirate discretization methods employed in this chapter have disadvantages similar to their single-rate versions: they are open-loop based and cannot guarantee the closed-loop stability and performance. We emphasize that we are not developing novel methods here, but looking into details for systems in the multirate setting. Due to the discrete down-sampler, results in the multirate setting present serious aliasing in high frequencies; that has suggested us to look for some other approaches to handle multirate discretization problems.

We remark that a ZOH equivalence of a continuous-time plant in a non-uniformly sampled system has been obtained in Chapter 3, and the results there are also applicable for the discretization of an analog controller if it is implemented non-uniformly. One question

then arises is: can we non-uniformly discretize such an analog controller by extending the bilinear transformation method from single-rate systems? This is left for the future.

We also presented a novel approach to identify the unknown time delay of a SISO continuous-time process in a multirate system, where the control signal is updated with period mh and the output is sampled with period nh . Based on the multirate operating data, we obtained first an interactor matrix of the lifted model by certain existing algorithms, then the desired estimation with accuracy h . We proved that when integers m and n satisfy certain restrictions and the interactor matrix has simple diagonal like structures, the unknown continuous time delay can be estimated with accuracy h - the base period of the multirate system. The smaller the base period h , the more accurate the estimation.

The effectiveness of our approach is demonstrated by several examples, which also showed that when conditions on m , n and \mathbf{D} mentioned above cannot be satisfied, the unknown continuous time delay is possible to be estimated by certain rules on the properties of time delay matrices of lifted models.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The main contributions of this thesis are:

- Discussion of the GPC problem for non-uniformly/multirate sampled systems by utilizing the lifting technique. We remark that MPC/GPC schemes for multirate systems have been proposed in the literature, e.g., Lee *et al.* [54] and Scattolini and Schiavoni [99]. However, all of these multirate designs are based directly on time-varying system models, and hence complicating the expressions and results. Using the lifting technique, non-uniformly/multirate systems are converted into LTI single-rate cases, the lifted controllers are designed by applying the standard LTI design methods and theories developed for this class of single-rate systems.
- Derivation of a causal GPC solution to the non-uniformly/multirate systems, when the lifting technique is employed. Although ideas similar to lifting have been used in some previous work on multirate MPC/GPC issues, see, for example, Scattolini [98] and Ling and Lim [61], the issue of causality constraint on the lifted controllers has never been discussed or mentioned in the synthesis problem. By appropriately grouping the output samples, we handle this constraint in Chapter 3. To our best knowledge, the results obtained are the first causal and optimal solution for the lifted models.
- Study of non-uniformly sampled systems. Starting from a sampled-data representation of such systems, with a continuous-time model and non-uniform sampler and hold device, the modeling issue is investigated. A sufficient condition is proposed, under which the obtained lifted discrete model will keep the controllability and observability of the original continuous-time model. We emphasize that the non-uniformly sampled

systems in this thesis are more general than those in reference [97, 2], and all results for non-uniformly sampled systems are applicable for multirate systems.

- Development of a sampled-data GPC algorithm for multirate systems. Conventional GPC design is based upon discrete-time model and performance index, and hence the inter-sample behavior is not optimized. Overcoming this disadvantage, Masuda *et al.* [62] proposed a sampled-data GPC scheme for single-rate systems. This scheme is extended to multirate systems in Chapter 4. By using a continuous-time cost function, the optimal solution is finally obtained and results in improved inter-sample behavior and continuous responses. The advantages of this sampled-data multirate GPC scheme are illustrated by a numerical example.
- Development of a state-space GPC algorithm for MIMO multirate systems. State-space representations have the advantage of easily handling MIMO systems, but are non-unique due to different choices of states. In Chapter 5, a multirate state-space GPC is presented based on the single-rate GPC provided by Ordys and Clarke [75]. The advantage is that the relationship between the state-space model and the corresponding CARIMA model is straightforward, particularly, the design of the observer filter $c(q^{-1})$ is reflected in the state-space model as the Kalman gain. The effectiveness of the derived algorithm is demonstrated by an example.
- Analysis of the robust stability of the multirate GPC controller in the presence of the multiplicative and uncertain MPM. The analysis is carried out in the frequency domain by applying small gain theorem to a closed-loop system composed of the lifted LTI model and the designed lifted controller. An example is illustrated to indicate the relationship between the down-sample ratio n and the stability robustness of the closed-loop system.
- Study of the multirate digital redesign problem. Discretization of multirate analog controllers has been carefully studied by Rafee *et al.* [83]. Using \mathcal{H}_2 optimization with a causality constraint, they give a design procedure to guarantee closed-loop stability and performance. In this thesis, we investigate different approaches for the same problem. These are easy extensions to multirate cases of the conventional single-rate discretization methods based on (i) step-invariant transformation, and (ii) the bilinear transformation. The comparison of these two methods are carried out in the frequency domain, based on the discretization error system as shown in Figure 6.5.

- Study of the continuous time delay estimation problem. It is generally a difficult problem to identify the unknown continuous time delay of the model, especially for a system where only multirate sampling strategy can be adopted due to practical constraints. Introducing the interactor matrix in Chapter 6, we connect the continuous time delay estimation problem with the interactor matrix of the lifted models. Some interesting results are presented here and several illustrative examples are given.

7.2 Future work

Towards this end, some of the problems that need to be investigated are:

- Identification of continuous time delays through interactor matrices of lifted models. In Chapter 6, we have shown that the continuous time delay can be determined with accuracy h (h is the base period in a multirate system) by knowledge of the interactor matrix of a lifted model. However, it is illustrated by particular examples and the interactor matrix has a form similar to diagonal, i.e., every row/column has only one non-zero element. Objectives of our future work include the possibility of this estimation by interactor matrices with more general forms, and to prove these results theoretically.
- Development of discretization approaches for an analog controller in the non-uniformly sampled setting. There are many choices to discretize an analog controller in the single-rate setting; step-invariant and bilinear transformations are the most commonly used two methods. Extension of the former method to the non-uniformly sampled case can refer to the modeling issue for non-uniformly sampled systems discussed in Chapter 3, where a lifted discrete time model is derived assuming the continuous-time model is known. Similarly, an analog controller can be discretized. Extension of the later method is not so straightforward. Recall that in using bilinear transformation in the single-rate setting we replace s in controller $K(s)$ with $\frac{2}{h} \cdot \frac{q-1}{q+1}$ (h is the discretization period), thus the natural question that arises is: what is this replacement in the non-uniformly discretization case? If there does not exist such a solution, we propose investigation of other possible approaches.
- Development of causal and state-space GPC controllers for non-uniformly sampled systems based on the lifted models. We remark that a causal and optimal controller for similar problems has been proposed in Chapter 3; however, it is derived in the

polynomial domain. For large systems with multi-input and multi-output, expressions and solutions in the state-space framework are more desirable. To find out the required state-space GPC controllers, our future work will be concerned with solving difficulties such as state estimation, time-delay handling, just to name a few. These state-space controllers may be classical discrete-time ones or sampled-data ones in order to improve the inter-sample behavior of the closed loop.

- Establishing an effective state estimator in the multirate state-space GPC schemes. The state-space GPC derivations in Chapters 2 and 5 have shown that the observer gain F (\underline{F} for the multirate case) reflects the observer polynomial $c(q^{-1})$ ($C(q^{-1})$ for the multirate case) in the CARIMA model. We know that this user-specified filter will affect the closed-loop robustness in the presence of the model-plant mismatch, and the selection guidelines for such filter in the single-rate setting have been discussed by Mohtadi [68], McIntosh *et al.* [63], and Robison and Clarke [91]. Thus F/\underline{F} is in fact the corresponding tuning parameter in the state-space framework and more results on this parameter and the state observer are expected. For example, how to tune the parameter F/\underline{F} ? What rules shall we follow when selecting this F/\underline{F} ? Is the state estimator in (2.7) (for single-rate case) or in (5.12) (for multirate case) effective for most chemical processes in industry? If not, what kind of multirate state filter shall we utilize and develop?
- Analysis and design of stabilizing and robust multirate GPC controllers. We have analyzed the robust stability of the unconstrained GPC for the fast-control, slow-sampling dual-rate systems in Chapter 5, where a closed form expression was derived and the stability analysis was straightforward. Next we will focus on the general multirate systems. Note that for single-rate constrained or unconstrained MPC/GPC systems, various approaches to stability and robust problems have emerged in the past decades. For example, a survey of the related literature before 1999 is given by Camacho and Bordons [14]; an exponential stability of the constrained receding horizon controlled system was established in Lee [56] by using the Lyapunov approach; results on stability and robustness of the receding horizon control with LMI formulations were obtained in Primbs and Nevistić [78, 79]. The extension of these techniques to multirate systems is possible, with the consideration of the causality constraint.
- Development of constrained multirate GPC. In this thesis, the control problem has been formulated by considering all signals to possess an unlimited range. This is not

very realistic because in practice all processes are subject to constraints. Thus a practical issue in our future work will be: how to implement multirate GPC controllers for processes with constrained input and output signals? This implementation will require the solution of quadratic programming problem, that is, an optimization problem with a quadratic objective function and linear constraints. A revision of main QP algorithms has been done by Camacho and Bordons [14]. Recently, convex optimization methods have found several applications in the area of control; and constraints in MPC have been handled in the form of LMI, see, e.g., Kothare *et al.* [48]. Furthermore, to overcome the on-line computation complexity, Rossiter *et al.* [96] have proposed several suboptimal alternatives to the QP optimization to reduce the on-line computational load. All these constraint handling strategies will be considered when we design the multirate constrained GPC controllers.

- Application of the developed GPC algorithms for multirate systems. To verify their effectiveness, experiments will be carried out on different real-time processes and simulated models. We will also look for some on-line tuning techniques for these controllers, in order to improve the closed-loop performance in tracking and disturbance attenuation.

Since GPC is an MPC type of controllers and multirate systems are special cases of non-uniformly sampled systems, our future work also includes extensions of our results on GPC and multirate systems to MPC and non-uniformly sampled systems.

Bibliography

- [1] P. Albertos and R. Ortega, On generalized predictive control: two alternative formulations, *Automatica*, **25**, 753-755, 1989.
- [2] P. Albertos and J. Salt, Receding horizon control of non-uniformly sampled-data systems, *Proc. American Control Conference*, **6**, 4300-4304, 1999.
- [3] J. C. Allwright, *Advances in Model-Based Predictive Control*, chapter on min-max Model-Based Predictive Control. Oxford University Press, 1994.
- [4] H. M. Al-Rahmani and G. F. Franklin, A new optimal multirate control of linear periodic and time-invariant systems, *IEEE Trans. Automat. Control*, **35**, 406-415, 1990.
- [5] P. Ansay and V. Wertz, Model uncertainties in GPC: a systematic two-step design, *Proceedings of the 3th European Control Conference*, Brussels, 1997.
- [6] M. Araki and K. Yamamoto, Multivariable multirate sampled-data systems: state-space description, transfer characteristics, and Nyquist criterion, *IEEE Trans. Automat. Control*, **30**, 145-154, 1986.
- [7] K. J. Åström and B. Wittenmark, *Computer Controlled Systems*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1984.
- [8] P. Banerjee and S. L. Shah, The role of signal processing method in the robust design of predictive control, *Automatica*, **31**, 681-695, 1995.
- [9] A. J. Beaumont, A. D. Noble and A. S. Mercer, Predictive control of transient engine testbed, *Proc. of Control'88*, Oxford, 468-471, 1988.
- [10] M. C. Berg, N. Amit and J. Powell, Multirate digital control system design, *IEEE Trans. Automat. Control*, **33**, 1139-1150, 1988.
- [11] R. R. Bitmead, M. Gevers and V. Wertz, *Adaptive Optimal Control: The Thinking Man's GPC*. Prentice-Hall, Englewood Cliffs, NJ, 1990.
- [12] R. R. Bitmead, M. Gevers and V. Wertz, Adaptive optimal control and GPC: robustness analysis, *Proc. European Control Conf.*, Grenoble, France, 1099-1104, 1991.
- [13] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory*. SIAM books, 1994.
- [14] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer, 1999.
- [15] P. J. Campo and M. Morari, Robust model predictive control, *American Control Conference*, Minneapolis, Minnesota, 1987.
- [16] P. Carini, R. Micheli and R. Scattolini, Multirate self-tuning predictive control with application to a binary distillation column, *Int. J. Syst. Sci.*, **21**, 51-64, 1990.

- [17] L. Chai, Y. Cao, Y. Sun and L. Qiu, Model predictive control for sampled-data systems, *IFAC 14th triennial world congress*, Beijing, P.R. China, 177-182, 1999.
- [18] T. Chen and B. A. Francis, Linear time-varying \mathcal{H}_2 optimal control of sampled-data systems, *Automatica*, **27**, 963-974, 1991.
- [19] T. Chen and L. Qiu, \mathcal{H}_∞ design of general multirate sampled-data control system, *Automatica*, **30**, 1139-1152, 1994.
- [20] T. Chen and B. A. Francis, *Optimal Sampled-data Control Systems*. Springer, London, 1995.
- [21] C. T. Chou, M. Verhaegen and R. Johansson, Continuous-time identification of SISO systems using Laguerre functions, *IEEE Trans. on Signal Processing*, **47**, 349-362, 1999.
- [22] D. W. Clarke and P. J. Gawthrop, Self-tuning control, *Proceedings IEE*, **123**, 633-640, 1979.
- [23] D. W. Clarke, P. P. Kanjilal and C. Mohtadi, A generalized LQG approach to self-tuning control. Part 1: Aspects of design; Part 2: Implementation and simulation, *Int. J. Control*, **41**, 1509, 1985.
- [24] D. W. Clarke, C. Mohtadi and P. S. Tuffs, Generalized predictive control. Part 1: The basic algorithm, *Automatica*, **23**(2), 137-148, 1987.
- [25] D. W. Clarke, C. Mohtadi and P. S. Tuffs, Generalized predictive control. Part 2: Extensions and interpretations, *Automatica*, **23**(2), 149-160, 1987.
- [26] D. W. Clarke, Application of generalized predictive control to industrial processes, *IEEE Control Systems Magazine*, **122**, 49-55, 1988.
- [27] D. W. Clarke and C. Mohtadi, Properties of generalized predictive control, *Automatica*, **25**, 859-875, 1989.
- [28] D. W. Clarke and R. Scattolini, Constrained receding-horizon predictive control, *Proceedings IEE*, **138**(4), 347-354, July 1991.
- [29] P. Colaneri, R. Scattolini and N. Schiavoni, Stabilization of multirate sampled-data linear systems, *Automatica*, **26**, 377-380, 1990.
- [30] C. R. Cutler and B. C. Ramaker, Dynamic matrix control-a computer control algorithm, *Automatic Control Conference*, San Francisco, 1980.
- [31] R. M. C. De Keyser and A. R. Van Cuawenberghe, Extended prediction self-adaptive control, *IFAC Symp. on Identification and System Parameter Estimation*, York, UK, 1317-1322, 1985.
- [32] H. Demircioglu and P. J. Gawthrop, Continuous-time generalized predictive control (CGPC), *Automatica*, **27**, 55-74, 1991.
- [33] J. C. Doyle, B. A. Francis and A. R. Tannenbaum, *Feedback Control Theory*. Maxwell Macmillan Canada, 1992.
- [34] A. Feintuch, P. P. Khargonekar and A. Tannenbaum, On the sensitivity minimization problem for linear time-varying periodic systems, *SIAM J. Control and Optimization*, **24**, 1076-1085, 1986.
- [35] C. E. Garcia, D. M. Prett and M. Morari, Model predictive control: theory and practice-a survey, *Automatica*, **25**(3), 335-348, 1989.
- [36] H. Garnier, P. Sibille and A. Richard, Continuous-time canonical model identification via Poisson moment functionals, *Proc. of the CDC*, 3004-3009, 1995.

- [37] T. T. Georgiou and P. P. Khargonekar, A constructive algorithm for sensitivity optimization of periodic systems, *SIAM J. Control and Optimization*, **25**, 334-340, 1987.
- [38] D. P. Glasson, Development and applications of multirate digital control, *IEEE Contr. Syst. Mag.*, **3**, 2-8, 1983.
- [39] G. Goodwin and K. Sin, *Adaptive Filtering, Prediction and Control*. Englewood Cliffs: Prentice-Hall, 1984.
- [40] C. Greco, G. Menga, E. Mosca and G. Zappa, Performance improvement of self tuning controllers by multistep horizons: the MUSMAR approach, *Automatica*, **20**, 681-700, 1984.
- [41] T. Hagiwara and M. Araki, Design of a stable feedback controller based on the multirate sampling of the plant output, *IEEE Trans. Automat. Control*, **33**, 812-819, 1988.
- [42] B. Haverkamp, C. T. Chou, M. Verhaegen and R. Johansson, Identification of continuous-time MIMO state space models from sampled data, *Proc. of the CDC*, 1539-1544, 1996.
- [43] B. Huang, S. L. Shah and H. Fujii, Identification of the time delay/interactor matrix for MIMO systems using closed-loop data, *Proc. 13th IFAC world congress*, Volume M, 355-360, 1996.
- [44] B. Huang and S. L. Shah, Performance Assessment of Control Loops: Theory and Applications. *Springer-Verlag*, London, 1999.
- [45] E. I. Jury and F. J. Mullin, The analysis of sampled-data control systems with a periodically time-varying sampling rate, *IRE Trans. Automat. Control*, **4**, 15-21, 1959.
- [46] R. E. Kalman and J. E. Bertram, A unified approach to the theory of sampling systems, *J. Franklin Inst.*, **267**, 405-436, 1959.
- [47] P. P. Khargonekar, K. Polla and A. Tannenbaum, Robust control of linear time-invariant plants using periodic compensation, *IEEE Trans. Automat. Control*, **30**, 1088-1096, 1985.
- [48] M. V. Kothare, M. V. Balakrishnan and M. Morari, Robust constrained model predictive control using linear matrix inequality, *Automatica*, **32**, 1361-1379, 1996.
- [49] B. Kouvaritakis, J. A. Rossiter and A. O. T. Chang, Stable generalized predictive control: an algorithm with guaranteed stability, *Proceedings IEE, Part D*, **139**(4), 349-362, 1992.
- [50] G. M. Kranc, Input-output analysis of multirate feedback systems, *IRE Trans. Automat. Control*, **3**, 21-28, 1957.
- [51] G. Kreisselmeier, On sampling without loss of observability/controllability, *IEEE Trans. on Automat. Control*, **44**(5), 1021-1025, 1999.
- [52] W. E. Larimore, Canonical variate analysis in identification, filtering and adaptive control, *Proc. 29th CDC*, 1990,
- [53] M. B. Lauritsen, M. Rostgaard and N. K. Poulsen, Generalized predictive control in the delta-domain, *Proc. 1995 American Control Conf.*, **6**, 3709-3713, 1995.
- [54] J. H. Lee, M. S. Gelormino and M. Morari, Model predictive control of multirate sampled data systems: a state-space approach, *Int. J. Control*, **55**(1), 153-191, 1992.
- [55] J. H. Lee, M. Morari and C. E. Garcia, State-space interpretation of model predictive control, *Automatica*, **30**(4), 707-717, 1994.

- [56] J. W. Lee, Exponential stability of constrained receding horizon control with terminal ellipsoid constraints, *IEEE Trans. Automat. Control*, **45**, 83-88, 2000.
- [57] J. M. Lemos and E. Mosca, A multipredictor-based LQ self-tuning controller, *IFAC Symp. on Identification and System Parameter Estimation*, York, UK, 137-141, 1985.
- [58] D. Li, S. L. Shah, T. Chen and R. S. Patwardhan, System identification and long-range predictive control of multirate systems, *Proc. 1999 ACC*, San Diego, CA, U.S.A., 1999.
- [59] D. Li, S. L. Shah and T. Chen, Identification of fast-rate models from multirate data, *Int. J. Control*, **74**(7), 680-689, 2001.
- [60] S. Li, K. Y. Lim and D. G. Fisher, A state space formulation for model predictive control, *AIChE Journal*, **35**, 241-249, 1989.
- [61] K. V. Ling and K. W. Lim, A state space GPC with extensions to multirate control, *Automatica*, **32**(7), 1067-1071, 1996.
- [62] S. Masuda, A. Inoue, Y. Hirashima and R. M. Miller, Intersample performance improvement in generalized predictive control, *IFAC International Symposium on Advanced Control of Chemical Processes*, Banff, Canada, 139-144, 1997.
- [63] A. R. McIntosh, S. L. Shah and D. G. Fisher, Experimental evaluation of adaptive control in the presence of disturbances and model-plant mismatch, *Workshop on Adaptive Control Strategies for Industrial Use*, Kananaskis, Alberta, June 20-22, 261-282, 1988.
- [64] D. G. Meyer, A parameterization of stabilizing controllers for multirate sampled-data systems, *IEEE Trans. Automat. Control*, **35**, 233-236, 1990.
- [65] D. G. Meyer, Cost translation and a lifting approach to the multirate LQG problem, *IEEE Trans. Automat. Control*, **37**, 1411-1415, 1992.
- [66] R. A. Meyer and C. S. Burrus, A unified analysis of multirate and periodically time-varying digital filters, *IEEE Trans. Circuits and Systems*, **22**, 162-168, 1975.
- [67] C. Mohtadi, *Advanced self-tuning algorithms*. PhD thesis, Oxford University, U. K., 1986.
- [68] C. Mohtadi, On the role of prefiltering in parameter estimation and control, *Workshop on Adaptive Control Strategies for Industrial Use*, Kananaskis, Alberta, June 20-22, 121-143, 1988.
- [69] M. Morari and J. H. Lee, Model predictive control: the good, the bad, and the ugly, *4th Int. Conf. on Chemical Process Control (CPC4)*, Texas, 1991.
- [70] E. Mosca, J. M. Lemos and J. Zhang, Stabilizing I/O receding horizon control, *IEEE Conference on Decision and Control*, 1990.
- [71] Y. Mutoh and R. Ortega, Interactor structure estimation for adaptive control of discrete-time multivariable nondecouplable systems, *Automatica*, **29**(3), 635-647, 1993.
- [72] J. P. Navratil, K. Y. Lim and D. G. Fisher, Disturbance feedback in model predictive control systems, *Proc. IFAC Workshop on Model-Based Process Control*, 63-68, Atlanta, GA, 1988.
- [73] J. E. Normey-Rico and E. F. Camacho, A Smith predictor based generalized predictive controller, *Internal Report GAR 1996/02*, University of Sevilla, 1996.
- [74] J. E. Normey-Rico, J. Gomez-Ortega and E. F. Camacho, A Smith predictor based generalized predictive controller for mobile robot path-tracking, *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, Madrid, Spain, 471-476, 1998.

- [75] A. W. Ordys and D. W. Clarke, A state-space description for GPC controllers, *Int. J. Systems Sci.*, **24**(9), 1727-1744, 1993.
- [76] Y. Peng and M. Kinnaert, Explicit solution to the singular LQ regulation problem, *IEEE Trans. Automat. Control*, **37**, 633-636, 1992.
- [77] V. Peterka, Predictor-based self-tuning control, *Automatica*, **20**(1), 39-50, 1984.
- [78] J. Primbs and V. Nevistić, A new approach to stability analysis of finite receding horizon control without end constraints, *IEEE Trans. Automat. Control*, **45**(8), 1507-1512, 2000.
- [79] J. Primbs and V. Nevistić, A framework for robustness analysis of finite receding horizon control, *IEEE Trans. Automat. Control*, **45**(10), 1828-1838, 2000.
- [80] L. Qiu and T. Chen, \mathcal{H}_2 and \mathcal{H}_∞ designs of multirate sampled-data systems, IMA Preprint Series #1062, University of Minnesota, 1992. (Condensed version in *Proc. 1993 ACC*.)
- [81] L. Qiu and T. Chen, Multirate sampled-data systems: all \mathcal{H}_∞ suboptimal controllers and the minimum entropy controller, *IEEE Trans. Automat. Control*, **44**(3), 537-550, 1999.
- [82] C. A. Rabbath, N. Hori, P. N. Nikiforuk and K. Kanai, Order reduction of PIM-based digital flight control systems, *Proc. 14th IFAC world congress*, Volumn Q, 127-132, Beijing, P.R. China, 1999.
- [83] N. Rafee, T. Chen and O.P Malik, Multirate discretization of analog controllers, *Canadian Conference on Electrical and Computer Engineering*, **2**, 554-557, 1996.
- [84] N. Rafee, T. Chen and O.P Malik, A technique for optimal digital redesign of analog controllers, *IEEE Trans. on Control Systems Technology*, **5**(1), 89-99, 1997.
- [85] R. Ravi, P. P. Khargonekar, K. D. Minto and C. N. Nett, Controller parameterization for time-varying multirate plants, *IEEE Trans. Automat. Control*, **35**, 1259-1262, 1990.
- [86] J. Rawlings and K. Muske, The stability of constrained receding horizon control, *IEEE Trans. Automat. Control*, **38**, 1512-1516, 1993.
- [87] J. Richalet, A. Rault, J. L. Testud and J. Papon, Algorithmic control of industrial processes, *4th IFAC Symposium on Identification and System Parameter Estimation*, Tbilisi URSS, 1976.
- [88] J. Richalet, A. Rault, J. L. Testud and J. Papon, Model predictive heuristic control: application to industrial processes, *Automatica*, **14**(2), 413-428, 1978.
- [89] J. Richalet, S. Abu el Ata-Doss, C. Arber, H. B. Kuntze, A. Jacobash and W. Schill, Predictive functional control. Application to fast and accurate robots, *Proc. 10th IFAC Congress*, Munich, 1987.
- [90] N. L. Ricker, Model predictive control with state estimation, *Ind. eng. Chem. Res.*, **29**, 374-382, 1990.
- [91] B. D. Robinson and D. W. Clarke, Robust effects of a pre-filter in generalized predictive control, *IEEE proceedings-D*, **138**, 2-8, 1991.
- [92] M. Rogozinski, A. Paplinski and M. Gibbard, An algorithm for calculation of a nilpotent interactor matrix for linear multivariable systems, *IEEE Trans. Automat. Control*, **32**(3), 234-237, 1987.
- [93] J. A. Rossiter and B. Kouvaritakis, Constrained stable generalized predictive control, *Proceedings IEE, Part D*, **140**(4), 1993.

- [94] J. A. Rossiter, Notes on multi-step ahead prediction based on the principle of concatenation, *Proc. IMechE*, **207**, 261-263, 1993.
- [95] J. A. Rossiter, *An Introduction to Predictive Control*. Notes of the lecture at the University of Alberta, Jan. 2001.
- [96] J. A. Rossiter, Computationally efficient algorithms for constraint handling with guaranteed stability and near optimality, *Int. J. Control*, **74**(17), 1678-1689, 2001.
- [97] J. Salt, J. Tornero and P. Albertos, Modeling of non-conventional sampled data systems, *Proc. of the 2nd IEEE Conference on Control Applications*, 631-635, 1993.
- [98] R. Scattolini, Self-tuning control of systems with infrequent and delayed output sampling, *Proc. IEE-D*, **135**, 213-221, 1988.
- [99] R. Scattolini and N. Schiavoni, A multirate model based predictive controller, *IEEE Trans. Automat. Control*, **40**(6), 1093-1097, 1995.
- [100] M. E. Sezer and D. D. Siljak, Decentralized multirate control, *IEEE Trans. Automat. Control*, **35**, 60-65, 1990.
- [101] S. Shah, C. Mohtadi and D. Clarke, Multivariable adaptive control without a prior knowledge of the delay matrix, *Systems and Control Letters*, **9**, 295-306, 1987.
- [102] J. Sheng, T. Chen and S. L. Shah, On stability robustness of dual-rate generalized predictive control systems, *Proc. American Control Conference*, **5**, 3415-3420, 2001.
- [103] N. K. Sinha, Estimation of transfer function of continuous system from sampled data, *IEE Proc.*, **119**, 612-614, 1972.
- [104] T. Söderström and P. Stoica, *System identification*. Englewood Cliffs, NJ:Prentice-Hall International, 1989.
- [105] R. Söeterboek, *Predictive Control. A unified approach*. Prentice-Hall, 1992.
- [106] A. K. Tangirala, D. Li, R. S. Patwardhan, S. L. Shah and T. Chen, Issues in multirate process control, *Proc. 1999 ACC*, San Diego, CA, U.S.A., 1999.
- [107] A. K. Tangirala, R. S. Patwardhan, S. L. Shah and T. Chen, Performance comparison of multirate vs single-rate systems, *IFAC Symposium on Advanced Control of Chemical Processes*, Pisa, Italy, 2000.
- [108] H. Unbehauen and G. P. Rao, Continuous-time approaches to system identification – a survey, *Automatica*, **26**, 23-35, 1990.
- [109] P. Van Overschee and B. De Moor, Subspace algorithms for the identification of combined deterministic-stochastic systems, *Automatica*, **30**, 75-93, 1994.
- [110] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems: Theory, Implementation, and Applications*. Kluwer Academic Publishers, 1996.
- [111] P. G. Voulgaris, M. A. Dahleh and L. S. Valavani, \mathcal{H}_∞ and \mathcal{H}_2 optimal controllers for periodic and multirate systems, *Automatica*, 1992.
- [112] K. Warwick and V. Peterka, Optimal observer solution for predictive and LQG optimal control, *Proc. IEE conf. Control'91*, Edinburgh, U. K., 768-772, 1991.
- [113] W. Wolovich and P. Falb, Invariants and canonical forms under dynamic compensation, *SIAM J. Control*, **14**, 996-1008, 1976.
- [114] W. Wolovich and H. Elliott, Discrete models for linear multivariable systems, *Int. J. Control*, **38**(2), 337-357, 1983.

- [115] B. E. Ydstie, Extended horizon adaptive control, *Proc. 9th IFAC World Congress*, Budapest, Hungary, 1984.
- [116] T. W. Yoon and D. W. Clarke, Observer design in receding-horizon control, *International Journal of Control*, **2**, 151-171, 1995.
- [117] A. Zheng and M. Morari, Stability of model predictive control with soft constraints, *Internal Report*, California Institute of Technology, 1994.
- [118] K. Y. Zhu, R. Gorez and V. Wertz, Alternative algorithms for generalized predictive control, *Systems Control Lett.*, **15**, 169, 1990.