INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI®

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

MODEL WEIGHTING ADAPTIVE CONTROL

Sylvain Gendron

Department of Electrical Engineering
McGill University, Montréal

February 1998

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

A ma mère Gabrielle et à la mémoire de Paul, mon père.

Summary

The main exercise of this thesis is the formulation of a mathematical framework for analyzing an existing industrial adaptive control algorithm labeled *Model weighting adaptive control* (MWAC). The algorithm is then analyzed under this framework. The exercise is complemented by a set of algorithmic additions aimed at solving questions that so far had remained open (e.g. the treatment of undermodelling errors). Those solutions, on the other hand build on results derived from the analysis.

A key result for analyzing the algorithm is that when an external excitation is applied (in the form of a control task such as a setpoint change), the adaptive controller behaves, in a short time that follows the application of the excitation, as a linear equation whose parameters are completely known at design time. It follows that during this short period, the input signal provided to the estimation subsystem is at least partially known (except for disturbances) and that the estimation virtually takes place in open loop. Using this information and assuming boundedness of the disturbance signals, it is possible to bound the behaviour of the adaptive system at an early stage.

With the MWAC algorithm, the plant model is formed by making a weighted sum of a finite number of possible plant models. It is shown that, under adequate conditions and in a time corresponding to the apparent plant delay, the plant model will "jump" to a neighborhood of the true plant. The size of this neighborhood will depend in part on how sharply the bad models are discriminated from the good models. On the other hand, disturbances will smooth the weight map towards a uniform

distribution. The sharpness or smoothness of the weight map can be measured online by computing the sum of the square root of all the weights in the set. The remarkable property of this measure is that an upper bound on the distance between the true plant and its model can be found which an affine function of the measure.

The effect of external disturbances such as measurement errors can be reduced by an external excitation of sufficient magnitude. This is not true however of disturbances caused by undermodelling errors which are almost always present to a lesser or greater degree. Two solutions are proposed to counteract this undesirable effect. The first method consists in bandpass filtering the input/output data in such a way that the frequency content of the data is consistent with data obtained from some first order plus delay (FOPD) model. The second method adjusts the sampling period online such that a compromise between satisfying the FOPD assumption and the coarseness of the control is obtained.

Résumé

L'exercice principal de cette thèse consiste à formuler un cadre d'analyse mathématique pour un algorithme adaptatif industriel appelé *Commande adaptative par modèles pondérés* (MWAC, l'acronyme en anglais). L'algorithme est ensuite analysé à l'intérieur de cadre. Des modifications à l'algorithme original sont ensuite proposées dans le but de résoudre des questions qui, à ce jour, demeuraient sans réponse (tel le traitement des erreurs de sous-modélisation). Par ailleurs, les modifications proposées découlent des résultats obtenus à l'analyse.

Un des principaux résultats de cette analyse est l'observation que dans une courte période de temps suivant l'application d'une excitation externe, le controlleur adaptatif se comporte tel une équation linéaire dont tous les paramètres sont connus avant la mise en marche du système. Il en découle que, pendant une courte période de temps, le signal fourni à l'estimateur est partiellement connu (nonobstant les perturbations) et que l'estimation du modèle se déroule pratiquement en boucle ouverte. En utilisant cette information et en supposant une borne sur l'amplitude des perturbations, il est possible d'obtenir une borne sur le comportement à court terme de l'algorithme.

L'algorithme MWAC compose un modèle du procédé en faisant une somme pondérée de modèles possibles. Il est démontré que, sous des conditions adéquates et à l'intérieur d'un temps correspondant au retard apparent du procédé, le modèle se déplace rapidement vers l'intérieur d'un voisinage de la dynamique du véritable procédé. La taille de ce voisinage dépend en partie de l'acuité avec laquelle le système peut trancher

entre les mauvais et les bons modèles. A l'opposé, les perturbations tendent à uniformiser la répartition des poids associés aux modèles. L'acuité de la distribution des poids peut être mesurée en calculant la somme de la racine carrée des poids. La propriété remarquable de cette mesure est qu'il est possible de trouver une borne supérieure à la distance entre la dynamique du procédé et son modèle qui est une fonction affine de la mesure.

L'effet des perturbations externes telles les erreurs de mesure peut être réduit en utilisant une excitation externe de grandeur suffisante. Telle n'est pas le cas, toutefois, pour ce qui est des erreurs causées par une sous-modélisation systématique du procédé. Conséquemment, nous proposons deux méthodes afin de contrer cet effet indésirable. La première méthode consiste à filtrer les données du procédé par un filtre passe-bande de telle sorte que le contenu fréquentiel des données soit cohérent avec celui d'un modèle de premier-ordre-avec-retard (POAR). La deuxième méthode consiste à ajuster la période d'échantillonage afin d'obtenir un compromis acceptable entre l'hypothése d'un système POAR et la précision de la commande.

Acknowledgments

First and foremost, I have to say that this work would never have taken place without the opportunity offered to me by the Pulp and Paper Research Institute of Canada (Paprican). I therefore want to thank Jim Rogers, Ron Crotogino, Alain Roche and all those who were involved in making this possible. I also want to thank them for their patience in allowing me the time to complete this work.

Next I want to thank my supervisor, Pierre Bélanger, for his trust, his foresight and the many suggestions he contributed to this work.

Many people have contributed in one way or another to my thesis. I am blessed to work at Paprican under the supervision of Alain Roche. I will never thank him enough for his support. Special thanks go to my colleagues for many interesting and valuable conversations: Guy Dumont, Normand Legault, Tony Holko, Bruce Allison, James Owen, Lin Lin, Jean Barrette, Jean Valiquette, Marc Champagne.

During my time at McGill University, I have had the chance to brush elbows with some terrific people who have helped me more than once: Kaouthar Benameur, Mark Readman, Finn Wredenhagen, Hamid Taghirad, Seddik Djouadi, Michael Glaum.

I am especially indebted to one chemical engineer who has been an invaluable source of information: my friend Michel Perrier. Among other things Michel introduced me to the notions of robust control and predictive control[1].

[1] In exchange of which, I taught him how to put on a decent necktie.

Finally, for putting up with me during difficult periods, making sure our household was in order, for her support, I thank my companion Ginette Vigneault. I also thank her along with Gabrielle and Nicolas for a wonderful, loving family life.

Remerciements

En premier lieu, je dois souligner que je n'aurais pu faire ce travail si l'Institut Canadien de Recherches sur les Pâtes et Papiers (Paprican) ne m'en avait donné la possibilité. Je me dois de remercier Jim Rogers, Ron Crotogino, Alain Roche et tous ceux qui ont rendu ceci possible. Je veux aussi les remercier pour leur patience en me permettant de prendre le temps requis pour compléter ce travail.

Je tiens ensuite à remercier mon superviseur, Pierre Bélanger pour sa confiance, sa vision et ses suggestions.

De nombreuses personnes ont contribué à mon travail de diverses façons. J'ai une chance exceptionnelle de travailler à Paprican sous la supervision d'Alain Roche. Je ne le remercierai jamais assez pour son appui. Des remerciements spéciaux vont à mes collègues pour de nombreuses conversations enrichissantes: Guy Dumont, Normand Legault, Tony Holko, Bruce Allison, James Owen, Lin Lin, Jean Barrette, Jean Valiquette, Marc Champagne.

Durant mon séjour à l'Université McGill, j'ai eu la chance de cotoyer nombres de personnes qui m'ont aidé plus d'une fois: Kaouthar, Mark Readman, Finn Wredenhagen, Hamid Taghirad, Seddik Djouadi, Michael Glaum.

J'ai une dette toute spéciale envers un ingénieur chimique qui fut une source intarissable d'information tout au long de mes études doctorales: mon ami Michel Perrier. Entre autres choses, Michel m'a fait connaitre les notions de commande robuste et de commande prédictive[2].

[2]En échange de quoi, je lui ai appris à faire un noeud de cravate digne de ce nom.

Finalement, pour m'avoir supporté dans les périodes difficiles, pour s'être assuré du bien être de notre famille, pour son soutien, je veux remercier ma compagne, Ginette Vigneault. De plus, je la remercie ainsi que Gabrielle et Nicolas pour une merveilleuse vie de famille.

Claim of originality

This thesis provides a framework and an approach for analysing an existing industrial adaptive control technique. Previous analyses of similar algorithms have been restricted to the asymptotic behaviour of such algorithms. The features of the analysis and the novelties introduced here are:

- It is focused on the short-term behaviour of the controller as opposed to an asymptotic analysis.
- An equivalence is found (which is valid only in the early period following the application of an external excitation) between the nonlinear, time-varying control equation and a linear, time-invariant difference equation which is completely known from prior knowledge and user-selected parameters.
- A central result is that, in a time that corresponds to the apparent time delay of the plant, the plant model "jumps" inside a neighborhood of the true plant. Bounds on the size of this neighborhood are given.
- An outcome of the above is that the proximity of the model to the true plant can be assessed online through a simple measure.
- Persistence of excitation conditions are not required and the only assumption on signals is that the closed loop system is externally excited by signals encountered under normal operation (e.g. a step setpoint change).
- Formulae are derived for helping a system designer select appropriate parameter values for a given problem.
- Two techniques are proposed for offseting the effects of undermodelling errors.

TABLE OF CONTENTS

Summary iii

Résumé v

Acknowledgments vii

Remerciements ix

Claim of originality xi

LIST OF FIGURES 4

LIST OF TABLES 6

Notation 7

CHAPTER 1. Introduction 9

1. Adaptive control and multiple models 13

Multiple models and LQG control 15

Multiple model adaptive control (MMAC) 16

Jump parameter systems adaptive control 19

Outline of the thesis 20

CHAPTER 2. Model Weighting Estimation 22

1. Prior Knowledge 22

A family of plant models 24

2. Off Line Model Weighting Estimation 26

A suboptimal policy 28
3. Online Model Weighting Estimation 29
Controller synthesis 31
Parameter estimation 31
Data filtering 32
4. Division by Zero 33
Initialization 34
5. Examples 35
Summary 38

CHAPTER 3. Model Estimation Using MWAC 39
1. The Approach 39
1.1. The user-selected parameters 42
1.2. Assumptions on the plant, plant uncertainty and noise 43
1.3. Apparent plant delay 44
Truncation operator 45
Disturbance 47
2. Framework and definitions 48
Local linear invariance 49
More definitions 50
3. Analysis of MWAC 51
3.1. Basic relations 51
3.2. Local linear invariance of the controller 53
3.3. Characterization of model adequacy 57
4. Tracking the model properties 61
4.1. The effects of undermodelling and additive noise 62
4.2. The base line weights distribution 65
Summary 70

CHAPTER 4. Implementation of MWAC 71

1. The initial invariance of the controller 71
2. Locking the estimates 74

CHAPTER 5. Treatment of undermodelling errors 78
1. Filtering the data 78
1.1. Designing the filter 80
Example 1 revisited (First method) 81
2. Adjustment of the sampling period 83
Example 1 revisited (Second method) 85
3. Merits and inconveniences of the proposed methods 86

CHAPTER 6. Conclusions 89
1. Future research 90

APPENDIX A. Supplemental proofs 92

APPENDIX B. Pseudo computer code for implementing μ-modification . . 97

APPENDIX C. Detailed calculations for Theorem 4.3 99

REFERENCES 103

LIST OF FIGURES

2.1	Bounds of $\mathcal{F}$ in the frequency domain	25
2.2	Structure of the adaptive controller	30
2.3	MWAC Example: Step response of the two plants	35
2.4	MWAC Example: Closed loop response for the two plants	37
2.5	MWAC Example: Adapted gains for the first plant	37
2.6	MWAC Example: Adapted gains for the second plant	38
3.1	Flow diagram of the error signals	52
4.1	Modified flow diagram for the error signals	72
4.2	Logic diagram for locking the estimates	76
5.1	Nyquist plot showing the inconsistencies of approximating a high order plant by a low order model	79
5.2	Bode plots of the plant and filter	82
5.3	Handling undermodelling errors: Input/output data for the first method (filter)	83
5.4	Handling of undermodelling errors: Adapted gains for the first method (filter)	84
5.5	Handling of undermodelling errors: Input/output data for the second method (sampling rate adjustments)	86

5.6 Handling of undermodelling errors: Blow up of the previous figure for the initial and final time segments 87

5.7 Handling of undermodelling errors: Additional feedback loop . 87

LIST OF TABLES

2.1 Table of tuning parameters for the examples. 36

Notation

$\mathbb{R}$		The set of real numbers
$\mathbb{Z}$		The set of integer number
$x(t)$	$x : \mathbb{Z} \to \mathbb{R}$	Discrete time signal
q^{-1}	$q^{-1}x(t) = x(t-1)$	Backward shift operator
$F(q^{-1})$		Filter, rational function in q^{-1}
$x^F(t)$	$F(q^{-1})x(t)$	Discrete time signal filtered by $F(q^{-1})$
z^{-1}		z-domain variable
$X(z^{-1})$	$\sum_{n=0}^{\infty} x(n)z^{-n}$	z-transform of $x(n)$
$X(e^{-j\omega})$		z-transform of $x(n)$ evaluated on the unit circle
$[X(z^{-1})]_t$	$\sum_{n=0}^{t} x(n)z^{-n}$	Truncated z-transform of $x(n)$ up to time index t
$\|[X(z^{-1})]_t\|_{p,\lambda}$	$\left(\frac{1}{2\pi}\int_0^{2\pi} \|X(\lambda^{-1}e^{-j\omega})\|^p \, d\omega\right)^{1/p}$	Weighted p-norm $(1 \leq p < \infty)$ of truncated signal $x(n)$
$\|[X(z^{-1})]_t\|_{\infty,\lambda}$	$\sup_{\omega\in[0,2\pi]} \|X(\lambda^{-1}e^{-j\omega})\|$	Weighted ∞-norm of truncated signal $x(n)$
$X^t(z^{-1})$		"Frozen" z-transform: The z-transform that a time-varying operator would assume if its parameters were frozen from time t to ∞.

SNR		Signal-to-noise ratio
$\mathcal{T}_I$	$[0, d] \subset \mathbb{Z}$	The interval of integers from 0 to d

In this thesis, the equation labels are located on the left-hand-side of the equations and must be interpreted as follows:

$$(s.c.e)$$

where s is the section number, c the chapter number and e the equation number.

CHAPTER 1

Introduction

In the manufacturing industries, a typical process has many inputs, many outputs. Its outputs do not react linearly to input adjustments. It does not react quite exactly the same way from one day to the next. The measurements (when available) of the desired properties of the end product are often flawed by errors. Furthermore, the properties of the raw material the process is fed are not completely known and they vary. Yet, as has been known for a long time, a little feedback from the measurements to the inputs performs minute miracles to bring order to this apparently chaotic business.

Many industrial processes nowadays perform poorly if at all without the benefits of feedback control. By closing a loop on an incompletely or imprecisely described process, the latter is coaxed into producing an end product with properties that are closer to specifications than those properties obtained without the feedback loop. More generally, feedback attenuates the majority of difficulties that stem from a lack of process understanding.

Up to a limit.

If the model is too imprecise, then closing the loop on that process may in fact deteriorates its performance. It may even become unstable. During the short history of control technology, the necessary handling of model precision or model *uncertainty* has launched the exploration of increasingly formal treatments.

Halfway through this century, the desire for better feedback performance in the face of an unknown or imprecise process model has spurred researchers to develop a number of control approaches that went beyond the fixed calculation of input signals from feedback error signals. The common thread of the approaches proposed in that period was to add the ability to "learn" about the process from its measured reaction to the computed inputs. Thus, in the 50s, was born the field of adaptive control [2]. From its modest but courageous beginnings, this field exploded in the following decades with the development of computer technology, theoretical achievements and reports of industrial successes. The theoretical and practical aspects of modern adaptive control technology are too numerous to fit in this short introduction but those issues that are pertinent to this thesis will be discussed shortly. For more details the reader will be refered to the cited publications. The reader will also find in the literature many excellent textbooks on the subject.

Although many of the general facts cited at the beginning about feedback control have been known for a long time, it is only in the late 70s - early 80s that these notions have been cast in a formal mathematical framework. The seminal work of Zames [47] in this area has sparked a flurry of developments from many quarters. These developments have evolved into a theory known under various names such as $\mathbf{H}^\infty$ or *Robust control theory.* The centerpiece of this theory consists in solving the following problem: given that over all frequencies, the process and its model deviate by no more than some quantity (or some bounding frequency function), what linear time-invariant feedback control will simultaneously satisfy a given performance specification and guarantee internal stability of the closed-loop system. Despite its simple formulation, the above problem is difficult to solve and typically yields high-order solutions. Approximate solutions exist however, and one of these solutions, namely the *internal model control* approach has received wide acceptance from the process industries [33].

Obviously, *adaptive controllers* and *robust controllers* approach the problem of dealing with uncertainty from vastly different angles. Fundamental research that

would provide rules to determine when to favour one solution over the other is an open research area [37]. In fact, even within classes of solutions, there really exists no hard rule for choosing one particular solution and this decision depends largely on the context and the designer's experience and judgment. This implicitely involves subjective evaluation criteria such as easiness and familiarity. In this thesis, we are largely biased by the implementation issues and consider that if solutions A and B solve a given problem but if solution A is easier to understand, easier to implement and easier to maintain than solution B, then solution A is better. A closed-loop restatement of Occam's razor[1].

From this standpoint, the robust control approach has an edge since its associated difficulties are handled at the design stage only and, in the end, yields a linear, time-invariant controller. Adaptive control, with its inherent nonlinear and time-varying handling of the process, is more likely to require attention even after it is put in operation.

With few exceptions, the congeniality aspect has been neglected by the general adaptive control solutions proposed in the literature. There appear to be at least two fundamental issues that are causing this: 1) The very general objectives of mainstream adaptive control techniques and 2) the conflicting natures of identification and control. The former is closely related to the issue of providing in a sufficiently general way an adaptive controller with prior knowledge about the problem to be solved. The latter has been known for a long time and has given way to *dual control* techniques [7]. For an in-depth discussion of the above issues see [2] and references therein. It follows that the implementation (and actually also the analysis) of most existing adaptive control techniques cannot escape a certain level of complexity. One way around this problem consists in building a software front-end which shelters the untrained user from the adaptive controllers nitty-gritty implementation details (and possibly also the controller from an untrained user) [3], [1], [2]. Such systems however are a far cry from simple three-term controllers which still dominate the process industries.

[1]William of Occam: A 13th century English scholastic philosopher known to have stated that "It is vain to do with more what can be done with less"

About prior knowledge, Astrom [2] notes that "For specific applications, it is possible to make adaptive controllers that make effective use of prior knowledge". The adaptive control technique proposed by Gendron et al [19] makes an explicit use of prior knowledge by surveying the general characteristics of the processes of a single industry, namely the pulp and paper industry:

- Long time delays, non-minimum phase behaviours
- Open loop stable and well damped
- Nonlinear
- Subjected to random disturbances

Note that these process features are not the exclusive claims of pulping and papermaking processes. The underlying assumption of that technique is that a large number of pulp and paper processes can approximately be described by a first order plus delay model. This representation has the advantage of only requiring three parameters to be estimated by the designer. Furthermore, these parameters can be quickly estimated from simple step response tests, a technique at least as old as the venerable Ziegler-Nichols test ([4], pp 231-232). The only novelty here as far as the designer is concerned is that of providing a possible range of uncertainty for the estimated parameters instead of a single estimate.

Not surprisingly, the algorithm labelled "Model Weighting Adaptive Control" (MWAC) has two of the expected functions of any adaptive controller: a fixed, model-based control equation and an estimation component which adjusts the model from input-output measurements. Thus, functionally, it does not differ from other methods found within a generic class of adaptive solutions. However, by the direct approach it takes towards adaptation, the method has some congenial virtues not found in most other adaptive algorithms. One manifestation of this forthrightness is that the direct translation of the complete set of equations into computer code is limited to a few lines (≈ 20) in any modern programming language. Furthermore, it does not require special supervisory functions. The complete method is later described in Chapter 2.

The reference [**19**] also describes an application of the method to pulp brightness control. Since then, the MWAC algorithm has been successfully applied by mill-resident engineers to the control of white liquor causticizing [**38**], to the pH control of cooking liquor [**24**] and effluents [**15**].

MWAC was born in industry. And because of its intuitive, direct approach to adaptation, it was applied nearly immediately after its inception without further examination or analysis. There is a need however for formally comprehending the mechanisms and circumstances that make it work, to assess its limitations.There is also a need for understanding how it relates to other adaptive control methods.

This is the subject of this thesis.

In what follows we propose a mathematical framework for analyzing the problem and use this framework to gain insight into the salient features of the algorithm. It happens that MWAC bears some characteristics which are similar to other adaptive algorithms found in the literature. In the following section, we review some of these algorithms, underline the similarities and compare the differences. We conjecture that some of the results found in the later parts of the thesis directly apply to those algorithms with similarities to MWAC.

Also, it is our objective (that will, from now, be unspoken) to leave the simplicity of the original algorithm untouched by the mathematical treatment we will now attach to it. We leave it to the reviewers and readers of this thesis to decide if we have succeeded or not in that respect.

1. Adaptive control and multiple models

As mentioned earlier, adaptive control already has a rich history. Its history, however is far from over. In fact, the very notion of what is implied by calling a system *adaptive* has been reexamined ([**35**], [**48**], [**20**] and [**27**]). The difficulty of nailing down a universally acceptable definition seems to be caused by the separation between the external manifestations of adaptive control and the internal mechanisms that implement the adaptation [**35**]. As pointed out by Zames [**48**], one designer's

adaptive design may appear as a fixed nonlinear design to another. The most satisfying definition the author of this thesis has found is the one given by Zames [48]:

> A *nonadaptive* controller is one designed on the basis of *a priori* information, i.e., which is available at the outset. An *adaptive* controller makes use of *a posteriori* information to achieve better performance than could be obtained with a nonadaptive one.

We claim that under this definition, the algorithm presented in [19] (MWAC) is truly adaptive. This will be further discussed once we have gone through the analysis of the algorithm.

Regardless of the definition of adaptation, there exists a number of algorithms in the literature that are recognized as being adaptive. One may consult [5] or [35] to get a description of an appreciable subset of such algorithms. Although many features of any adaptive algorithm are shared by MWAC, we will now only look at those which bear enough ressemblance to MWAC to justify comparisons.

To enable such comparisons, we begin by outlining the adaptation mechanism used by MWAC. As mentioned earlier, the user provides the MWAC algorithm with a range of possible values for the parameters of a first order plus delay system. These intervals are then partitioned. A family $\mathcal{F}$ of models is then formed from the set of first order plus delay models which pick their parameters from the Cartesian product of the partitioned intervals. Let the members of $\mathcal{F}$ be the set of indexed first order plus delay models $\{P_i\}$. Next let $e_i(t)$ be the difference between the plant output and the particular model P_i of $\mathcal{F}$. Consider the familiar weighted 2-norm

$$\|e_i(t)\|^2_{2,\lambda} = \sum_{k=0}^{t} \lambda^{t-k} e_i^2(k) \qquad 0 < \lambda \leq 1$$

as being a measure of the ability of P_i to reproduce the true plant behaviour. The parameter λ exponentially discounts the contribution of old data to the norm . An estimate of the plant model is then obtained by computing the sum

$$\tilde{P} = \sum_i w_i(t)\ P_i$$

where

$$w_i(t) = \frac{\left(\mu + \|e_i(t)\|^2_{2,\lambda}\right)^{-1}}{\sum_n \left(\mu + \|e_n(t)\|^2_{2,\lambda}\right)^{-1}} \tag{1.1.1}$$

where μ is some small positive quantity inserted to prevent possible divisions by zero.

The online model $\tilde{P}$ is then at every time step replaced in a model-based control equation in the *certainty equivalence* fashion.

Multiple models and LQG control. Using, as a starting point, a bounded family of plant models to represent the uncertainty about the exact behaviour of a plant is not new. In 1965, Magill [**30**] published an article on state estimation where it was assumed that the plant belonged to a finite set of possible state space representations. This idea was later extended and thoroughly analyzed by Lainiotis and a complete account of this work can be found in [**25**], [**26**] and references therein. The basic idea of this line of work derives from the *Linear-Quadratic-Gaussian*(LQG) estimation theory. Consider the state space system

$$\begin{aligned} \mathbf{x}(t+1) &= A_i\mathbf{x}(t) + B_iu(t) + w(t) \\ y(t) &= C_i\mathbf{x}(t) + v(t) \end{aligned}$$

where $w(t)$ and $v(t)$ are Gaussian, zero-mean, independent sequences with covariance $\mathbf{W}$ and $\mathbf{V}$ respectively. The optimal estimator of the state given the past measurements is computed by the Kalman filter [**12**], i.e.,

$$\begin{aligned} \hat{\mathbf{x}}(t+1|t) &= A_i\hat{\mathbf{x}}(t|t-1) + B_iu(t) + K(t)(y(t) - C_i\hat{\mathbf{x}}(t|t-1)) \\ K(t) &= (A_iP(t)A_i^T + W)\, C_i^T\, (C_i(A_iP(t)A_i^T + V)C_i^T + W)^{-1} \\ P(t+1) &= (I - K(t)C_i)(A_iP(t)A_i^T + W) \end{aligned}$$

Now consider the innovations $z(t) = y(t) - C_i\hat{\mathbf{x}}(t|t-1)$. The sequence of innovations is orthogonal and since the system is assumed Gaussian, then the conditional probability

density function of $z(t)$ is also Gaussian and given by

$$p_{t-1}(z(t)) = \frac{1}{\sqrt{2\pi|\mathrm{D}_i(t)|}} \exp[-\frac{1}{2} z^T(t)\mathrm{D}_i^{-1}(t)z(t)] \tag{1.1.2}$$

where $D_i(t) = C_i P(t) C_i^T + V$ is the covariance matrix of the innovations.

On the control side, the LQG controller which solves

$$\min E\left\{\sum_{n=0}^{\infty} \mathrm{x}^T(n)Q_1\mathrm{x}(n) + u^T(n)Q_2u(n)\right\}$$

for the above system is computed from

$$u(t) = -L_i(t)\,\hat{\mathrm{x}}(t|t-1) \tag{1.1.3}$$

where

$$L_i(t) = \left(B_i^T S_i(t) B_i + Q_2\right)^{-1} B_i^T S_i(t) \tag{1.1.4}$$

and where $S_i(t)$ solves another Riccati equation.

Multiple model adaptive control (MMAC). Now suppose that the triplet $\alpha_i = (A_i\ , B_i\ , C_i)$ actually belongs to a finite number N of possible values. Then the adaptation is based on the *a posteriori* probabilities of the α_i's. Let $\mathrm{Y}_t \triangleq \{y(0), y(1), \cdots, y(t)\}$ be the set of past measurements. Then from Bayes' rule, we have, as in [30]

$$\mathrm{p}[\alpha_i|\mathrm{Y}_t] = \frac{\mathrm{p}[\mathrm{Y}_t|\alpha_i]\mathrm{p}[\alpha_i]}{\sum_{n=1}^{N} \mathrm{p}[\mathrm{Y}_t|\alpha_n]\mathrm{p}[\alpha_n]}$$

We combine the above with

$$\mathrm{p}[\mathrm{Y}_t|\alpha_i] = \mathrm{p}[y(t)|\mathrm{Y}_{t-1}; \alpha_i]\ \mathrm{p}[Y_{t-1}|\alpha_i] \tag{1.1.5}$$

Now if the true plant were given by the triplet α_i, then $\mathrm{p}[\mathrm{Y}_t|\alpha_i]$ could be recursively computed from (1.1.2) and (1.1.5).

Alternatively, we could, as [25], go directly to a recursive form by taking out only the effect of the last measurement, i.e.

$$\text{(1.1.6)} \qquad \mathrm{p}[\alpha_i|\mathrm{Y}_t] = \mathrm{p}[\alpha_i|y(t)\,;\mathrm{Y}_{t-1}] = \frac{\mathrm{p}[y(t)|\alpha_i;\mathrm{Y}_{t-1}]}{\sum_{n=1}^{N}\mathrm{p}[y(t)|\alpha_n;\mathrm{Y}_{t-1}]\;\mathrm{p}[\alpha_n|\mathrm{Y}_{t-1}]}\mathrm{p}[\alpha_i|\mathrm{Y}_{t-1}]$$

At this point, the developers of this technique make the following two approximations:

- They introduce N Kalman filters (one for each triplet) and assume that $\mathrm{p}[\mathrm{Y}_t|\alpha_i]$ can be computed from (1.1.2) and (1.1.5) for **all** triplets in the set (although this is true for at most one triplet).
- For lack of more knowledge about the probabilities $\mathrm{p}[\alpha_i]$, a uniform distribution is assumed, i.e. $\mathrm{p}[\alpha_i] = 1/N\ i = 1,\cdots,N$.

It follows that the pdf of α_i based on the past $t+1$ measurements is the product of the of previous conditional pdf's of the innovations or, more correctly, the *pseudo-innovations* [25]. Hence

$$\begin{aligned}\mathrm{p}[\alpha_i|\mathrm{Y}_t] &= \frac{\prod_{k=0}^{t} p_k^{(i)}(z_i(k))}{\sum_{n=1}^{N}\prod_{k=0}^{t} p_k^{(n)}(z_n(k))} \\ &= \frac{\prod_{k=0}^{t}|D_i(k)|^{-1/2}\;\exp[-\frac{1}{2}\sum_{k=0}^{t} z_i^T(k)D_i^{-1}(k)z_i(k)]}{\sum_{n=1}^{N}\prod_{k=0}^{t}|D_n(k)|^{-1/2}\;\exp[-\frac{1}{2}\sum_{k=0}^{t} z_n^T(k)D_n^{-1}(k)z_n(k)]} \qquad \text{(1.1.7)}\end{aligned}$$

In the scalar case, (1.1.7) reduces to

$$\text{(1.1.8)} \qquad \mathrm{p}[\alpha_i|\mathrm{Y}_t] = \frac{\Pi_i^{-1}(t)\exp\left[-\frac{1}{2}\sum_{k=0}^{t} z_i^2(k)/D_i(k)\right]}{\sum_{n=1}^{N}\Pi_n^{-1}(t)\exp\left[-\frac{1}{2}\sum_{k=0}^{t} z_n^2(k)/D_n(k)\right]}$$

where

$$\Pi_i(t) = \prod_{k=0}^{t} D_i^{-1/2}(k)$$

The MMAC architecture then consists of generating N separate optimal control signals $u_i(t)\ \ i = 1,\cdots,N$, i.e. each one being optimally tuned for the corresponding state-space system in the set. The actual input applied to the process is then obtained

by tying the individual inputs through the conditional pdf's, i.e.

$$u(t) = \sum_{i=1}^{N} \mathrm{p}[\alpha_i|\mathrm{Y}_t]\ u_i(t)$$

From (1.1.3) and (1.1.4), this means that the MMAC controller is summarized by the equation

$$u(t) = -\sum_{i=1}^{N} \mathrm{p}[\alpha_i|\mathrm{Y}_t]\ \left(B_i^T S_i(t) B_i + Q_2\right)^{-1} B_i^T S_i(t) \hat{\mathbf{x}}(t|t-1) \tag{1.1.9}$$

It seems that the most important application area of MMAC has been the control of military aircrafts ([**6**], [**17**], [**42**], [**31**]) but it has found applications in other contexts as well ([**32**], [**9**], [**34**]). The article by Athans and coworkers [**6**] contain many useful comments on implementation issues. In particular, some processing is applied to the pdf's prior to their utilization: their value is not allowed to go below some value in order for the estimator not to "freeze" into some state and they are low-pass filtered to prevent oscillating estimates (which may occur under some conditions) to affect the feedback control.

An analysis of the asymptotic properties of MMAC systems can be found in [**23**], [**44**] and [**16**]. These studies are actually concerned with the *almost sure* convergence of one pdf to 1. The analysis of [**22**] mentions stability problems associated with MMAC and tries to analyze this problem from a deterministic viewpoint.

Obviously, the *pdf's* of MWAC play a role similar to the *weights* of MWAC. The above equation shows however a structural difference between the two algorithms. MMAC uses the pdf's *a posteriori* i.e. after the optimal input signals have been computed to synthesize a single input signal while MWAC uses the weights *a priori* to compose a single plant model from which the controller is synthesized and the plant input computed. Due to the nonlinear operations involved in these controller constructions, there is no reason to believe that, in general, the MMAC and MWAC approaches are equivalent despite the similarities.

The exponential function in (1.1.8) stems only from the Gaussian assumption on the perturbations. If we reliquinsh this assumption, the exponential function can also be simply seen as a nonlinear function used to enforce the required discrimination between the "bad" and "good" models of the set. The approximation

$$e^{-x} \approx \frac{1}{1+x}$$

leads to

$$\mathrm{p}[\alpha_i|\mathrm{Y}_t] \approx \frac{\left(\Pi_i(t) + \sum_{k=0}^{t} z_i^2(k)\, \frac{\Pi_i(t)}{2D_i(k)}\right)^{-1}}{\sum_{n=0}^{N}\left(\Pi_n(t) + \sum_{k=0}^{t} z_n^2(k)\, \frac{\Pi_n(t)}{2D_n(k)}\right)^{-1}} \tag{1.1.10}$$

The above approximation allows further comparisons with the MWAC weight assignment law (1.1.1). It shows that the time weighting multiplier λ^{t-k} of the weighted 2-norm used by MWAC is replaced in MMAC by the weight

$$\frac{\Pi_i(t)}{2D_i(k)}$$

which does in fact accomplish a scaling of the instantaneous deviations $(z_i^2(k))$ using the variance $D_i(k)$ of the pseudo innovations. Furthermore, the offset $\Pi_i(t)$, since it is a long-term, converging statistics associated to the triplet α_i, can be seen as a mean of associating a default distribution of the weights (or pdf's) based on the estimated statistics instead of the uniform default offset μ provided by MWAC.

Jump parameter systems adaptive control. An interesting extension of the above problem is when the plant actually "jumps" randomly between the triplets α_i. If the jumps are Markovian and the transition probabilities

$$\mathrm{P}[\alpha(t) = \alpha_i|\alpha(t-1) = \alpha_j] = \phi_{ij}$$

are known, this additional information improves the estimation of the conditional pdf's.

This problem has been thoroughly investigated in the context of continuous time by [**13**], [**14**] and references therein and in the context of discrete time by [**45**]. In continuous time, the formal solution to estimating the conditional pdf's is derived from the Wonham filter [**46**] and is given in the form of two stochastic differential equations.

The issue of interest however is not so much the estimation aspect but rather how the control signal is generated from the pdf's. In [**14**], the structure of computation is the same as MWAC in the sense that the pdf's are used *a priori*, i.e. first a composite plant model is formed from the pdf's and this composite model is then used to synthesize the controller and compute the input signal. In [**14**], the solution is derived in continuous time and it is assumed that we have full state information. If we transpose the results of [**14**] to the discrete time context we have used so far, first we get

$$\hat{A} = \sum_{i=1}^{N} \mathrm{p}[\alpha(t) = \alpha_i | \mathrm{Y}_t]\, A_i \qquad \hat{B} = \sum_{i=1}^{N} \mathrm{p}[\alpha(t) = \alpha_i | \mathrm{Y}_t]\, B_i$$

and the control is obtained from

$$u(t) = -\left(\hat{B}^T \hat{S}(t) \hat{B} + Q_2\right)^{-1} \hat{B}^T \hat{S}(t) \mathrm{x}(t) \tag{1.1.11}$$

where $\hat{S}(t)$ is the solution to a Riccati equation whose parameters are $\hat{A}$ and $\hat{B}$. The reader is invited to compare (1.1.11) with (1.1.9).

On the other hand, the structure of computation in [**45**] is of the *a priori* form and can thus be considered an application of MMAC to the jump-Markov problem.

Outline of the thesis. MWAC borrows concepts from the above classes of solutions. We conjecture that some of the results found here may also apply to those solutions as well.

In what follows, we first review the design principles behind MWAC as they were presented in earlier work [**19**]. In that same chapter, we also introduce a slight modification to the existing algorithm. This modification is introduced to simplify the

analysis of the algorithm. This analysis is carried out in the latter part of Chapter 3. At the beginning of that same chapter, we define the framework within which the analysis takes place. Chapters 4 and 5 propose modifications to the original algorithm to enhance its performance. This is followed by a final discussion in the concluding chapter.

CHAPTER 2

Model Weighting Estimation

In this chapter, we discuss the general concept of model weighting estimation. Before we review the on line algorithm described in [19], we introduce an off line model weighting estimation algorithm that will serve as a basis for discussing the features of the on line version. At the end of the present chapter, we introduce a modification to the original algorithm that will greatly facilitate the analysis of Chapter 3 while preserving the key features of the original algorithm.

1. Prior Knowledge

Model weighting estimation, as described in [19], addresses the problem of controlling plants with a stable first order plus delay representation, i.e.

$$P_0(s) = \frac{ge^{-Ds}}{1+\tau s} \tag{1.2.1}$$

where the gain (g), delay (D) and time constant (τ) are known only up to a certain degree of precision which is given by uncertainty intervals

$$g \in [\underline{g}, \overline{g}] \tag{1.2.2}$$

$$D \in [\underline{D}, \overline{D}] \tag{1.2.3}$$

$$\tau \in [\underline{\tau}, \overline{\tau}] \tag{1.2.4}$$

Obviously, the model 1.2.1, because of its low order, has some shortcomings. However, the dynamics of many industrial processes are often *dominated* by a representation of the form 1.2.1. Processes involving mixing, heat transfer, plug flow, material transport or simple endothermic chemical reactions often fall in this category. In general, the model weighting adaptive control algorithm is *implemented* on a plant that belongs to the general *family* of plant models

$$\mathcal{P} \triangleq \{P(s)|\ |P(j\omega)| \leq |P_0(j\omega)| \cdot |1 + l_m(j\omega)|\} \tag{1.2.5}$$

where $1 + l_m(s)$ is a multiplicative term whose magnitude overbounds the uncertainty we assume knowing about the members of $\mathcal{P}$. Equations (1.2.1) to (1.2.5) represents the *prior knowledge* about the plant. Under the most realistic conditions, this uncertainty stems from a) the parametric uncertainty (1.2.2, 1.2.3 and 1.2.4) and b) undermodelling errors. In the synthesis of the control algorithm, we will ignore b) but later, we will thoroughly investigate the effect of undermodelling and its influence on the choice of tuning parameters.

The exposition and analysis of the algorithm is performed in discrete time. The discrete time version of (1.2.1) is given by

$$P(z^{-1}) = g\frac{(1-\alpha^m) - (\alpha - \alpha^m)z^{-1}}{1-\alpha z^{-1}} z^{-d-1} \tag{1.2.6}$$

where

$$\begin{aligned} \alpha &= e^{-T/\tau} \\ d &= \text{int}\frac{D}{T} \\ m &= d + 1 - \frac{D}{T} \end{aligned}$$

and T is the sampling period. Alternatively, (1.2.6) may be written

$$P(z^{-1}) = gA(z^{-1})(\theta z^{-d-1} + (1-\theta)z^{-d-2}) \tag{1.2.7}$$

where

$$A(z^{-1}) = \frac{1-\alpha}{1-\alpha z^{-1}}$$
$$\theta = \frac{1-\alpha^m}{1-\alpha}$$

A family of plant models. We now make two simplifying design choices: 1) we assume that the modeling errors are dominated by gain and delay uncertainty and pick a value τ in the interval $[\underline{\tau}, \overline{\tau}]$ as an estimate of the dominant time constant of the plant; 2) we overbound the discretized delay interval (1.2.3) by integer multiples of the sampling period and include in the interval all the integers between its bounds given by

$$\underline{d} = \inf_{d \in \mathbb{Z}} \frac{D}{T} \text{ and } \overline{d} = \sup_{d \in \mathbb{Z}} \frac{\overline{D}}{T}$$

The errors introduced by these two choices are particular cases of undermodelling errors and will be discussed later. These choices lead to the simplified model (From (1.2.6))

$$P(z^{-1}) = g\frac{1-\alpha}{1-\alpha z^{-1}} z^{-d-1} \tag{1.2.8}$$

Let $\mathbf{D} \triangleq [\underline{d}, \overline{d}] \subset \mathbb{Z}$ and $L_d = \text{card } (\mathbf{D})$ and let $\mathbf{G}$ be a partition of $[\underline{g}, \overline{g}]$ with elements g_i which pick their index from $[1, \cdots, L_g] \subset \mathbb{Z}$ where $L_g = \text{card } (\mathbf{G})$ and

$$g_1 = \underline{g} \text{ and } g_{L_g} = \overline{g}$$

REMARK 1.1. *We assume that the sign of the gain of the plant never changes, i.e. sgn* $(\underline{g})$*= sgn*$(\overline{g})$*. This avoids the potentially troublesome situation of having* $g = 0$.

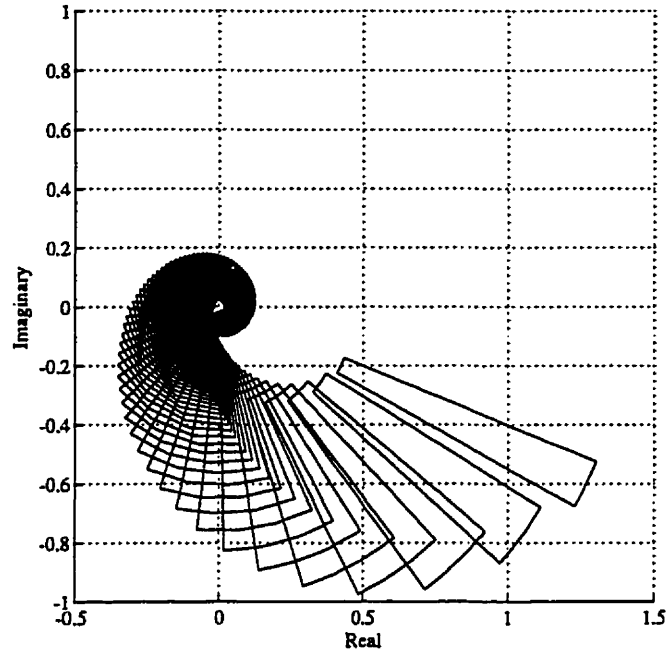

FIGURE 2.1. Bounds on frequency response. The graph shows the surfaces spanned by $\mathcal{F}$ at individual frequencies.

The model (1.2.8) together with the uncertainty intervals form a family $\mathcal{F}$ of possible models, i.e.

$$\mathcal{F} \triangleq \left\{ g\frac{1-\alpha}{1-\alpha z^{-1}}z^{-d-1} \;\middle|\; g \in \mathbf{G} \times d \in \mathbf{D} \right\} \tag{1.2.9}$$

The elements of $\mathcal{F}$ are labeled $P_{ij}(q^{-1})$ and they pick their indices from $i \in [1, \cdots, L_g]$ and $j \in [1, \cdots, L_d]$ respectively. Figure 2.1 show the Nyquist plot of $\mathcal{F}$ and indicate the surface spanned by $\mathcal{F}$ at individual frequencies.

Following [19], the controller selects the plant model from the convex hull of $\mathcal{F}$, i.e.

(1.2.10)

$$\mathbf{Co}(\mathcal{F}) \triangleq \{P_m(z^{-1}) = \sum_i \sum_j w_{ij} P_{ij}(z^{-1}) | P_{ij} \in \mathcal{F}; 0 \leq w_{ij} \leq 1; \sum_i \sum_j w_{ij} = 1\}$$

Hence the model has the form

$$P_m(z^{-1}) = \frac{1-\alpha}{1-\alpha z^{-1}} \cdot \sum_j \gamma_j z^{-j-1}$$

with some constraints on the parameters γ_j. It is interesting to notice that the above model in fact encompasses a richer set of plants than the original first-order-plus-delay description. This property however, will not be examined in this work.

2. Off Line Model Weighting Estimation

We seek the minimizer P_m (belonging to $\mathbf{Co}(\mathcal{F})$) of the objective function

$$\mathbf{J} = \|P - P_m\|_2^2 = \frac{1}{2\pi}\int_0^{2\pi} |P(e^{-j\omega}) - P_m(e^{-j\omega})|^2 d\omega \tag{2.2.1}$$

Since $P_m \in \mathbf{Co}(\mathcal{F})$, it satisfies

$$P_m(e^{-j\omega}) = \sum_i \sum_j w_{ij} P_{ij}(e^{-j\omega})$$

and $0 \leq w_{ij} \leq 1$ and $\sum_i \sum_j w_{ij} = 1$.

To simplify notation, we reorganize the indexing of $\mathcal{F}$ to use a single index running from 1 to N $(= L_d \times L_g)$, the total number of models in $\mathcal{F}$. Hence, we may write P_m as

$$P_m(e^{-j\omega}) = \sum_i w_i P_i(e^{-j\omega}) = W^T P_{\mathcal{F}}$$

where W and $P_{\mathcal{F}}$ are column vectors containing the weights and the models frequency response respectively. Let $\mathbf{1}^T$ be the N-dimensional vector $\mathbf{1}^T = [1\ 1\ \cdots\ 1]$. We may write the model error as

$$P - P_m = W^T(P\mathbf{1} - P_{\mathcal{F}})$$

Let the inner product $< \cdot, \cdot >$ be

$$< A(e^{-j\omega}), B(e^{-j\omega}) > \triangleq \frac{1}{2\pi}\int_0^{2\pi} A(e^{-j\omega})\overline{B}^T(e^{-j\omega}) d\omega$$

The objective function can thus be written

$$\begin{aligned} \mathbf{J} &= < P - P_m, P - P_m > \\ &= W^T \frac{1}{2\pi} \int_0^{2\pi} (P\mathbf{1} - P_{\mathcal{F}}) \, \overline{(P\mathbf{1} - P_{\mathcal{F}})}^T d\omega \, W \\ &= W^T \Sigma W \end{aligned} \tag{2.2.2}$$

where

$$\Sigma = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1N} \\ s_{21} & s_{22} & \cdots & s_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ s_{N1} & s_{N2} & \cdots & s_{NN} \end{bmatrix}$$

The entries of Σ are given by

$$s_{ik} \triangleq \frac{1}{2\pi} \int_0^{2\pi} (P - P_i) \cdot \overline{(P - P_k)} \, d\omega$$

The optimal set of weights is given by the following theorem.

THEOREM 2.1. *The weight vector W^* minimizing the objective function $\mathbf{J} = W^T \Sigma W$ subject to the constraint $W^T \mathbf{1} = 1$ is given by*

$$W^* = (\mathbf{1}^T \Sigma^{-1} \mathbf{1})^{-1} \Sigma^{-1} \mathbf{1} \tag{2.2.3}$$

and the minimum value of $\mathbf{J}$ is given by

$$\mathbf{J}^* = \left(\mathbf{1}^T \Sigma^{-1} \mathbf{1}\right)^{-1} \tag{2.2.4}$$

PROOF. Using the Lagrange multiplier Λ, we form

$$\mathbf{H} = W^T \Sigma W + \Lambda (W^T \mathbf{1} - 1)$$

Taking the gradient of $\mathbf{H}$ with respect to W and setting to zero we get

$$\nabla_W \mathbf{H} = 2\Sigma W + \Lambda \mathbf{1} = 0$$

from which we get

$$W = -\frac{\Lambda}{2}\Sigma^{-1}\mathbf{1} \tag{2.2.5}$$

Replacing in $W^T\mathbf{1} = 1$ yields

$$-\frac{\Lambda}{2} = (\mathbf{1}^T\Sigma^{-1}\mathbf{1})^{-1}$$

Using (2.2.5) yields the result for W^*, i.e.

$$W^* = (\mathbf{1}^T\Sigma^{-1}\mathbf{1})^{-1}\Sigma^{-1}\mathbf{1}$$

and the minimum value for $\mathbf{J}$ is then

$$\mathbf{J}^* = (W^*)^T\Sigma W^* = (\mathbf{1}^T\Sigma^{-1}\mathbf{1})^{-1}$$

□

Note that if Σ were diagonal, the optimal policy for the choice of an individual weigth w_i would simply be

$$w_i^* = \frac{1/s_{ii}}{\sum_k 1/s_{kk}} \tag{2.2.6}$$

There exists a standard result of statistics equivalent to the above when one estimates some quantity using the weighted sum of N independent noisy channels [**21**], [**28**].

A suboptimal policy. To limit computation and data storage requirements Gendron *et al* [**19**] suggested for on line model weighting adaptation to only consider the diagonal elements of the error norm matrix. It is interesting to compute how such a policy would deteriorate the performance of the off line model. Let Σ be written

$$\Sigma = \Sigma_D + \nu_\Sigma$$

where Σ_D is a diagonal matrix with its diagonal equal to the diagonal of Σ and ν_Σ contains the non-diagonal elements of Σ. The suboptimal weighting policy is then

$$W_D = (1^T\Sigma_D^{-1}1)^{-1}\Sigma_D^{-1}1 \tag{2.2.7}$$

and the corresponding value of the objective function is given by the following theorem.

THEOREM 2.2 (Suboptimal policy). *The value of the objective function J_D corresponding to the suboptimal policy (2.2.7) is given by*

$$J_D = J_D^{opt} + (J_D^{opt})^2 R \tag{2.2.8}$$

where $J_D^{opt} = (1^T\Sigma_D^{-1}1)^{-1}$ and $R = 1^T\Sigma_D^{-1}\nu_\Sigma\Sigma_D^{-1}1$.

PROOF. We have

$$J_D = W_D\Sigma W_D = \frac{1^T\Sigma_D^{-1}\Sigma\Sigma_D^{-1}1}{(1^T\Sigma_D^{-1}1)^2}$$

but $\Sigma = \Sigma_D + \nu_\Sigma$ and thus

$$\begin{aligned} J_D &= (1^T\Sigma_D^{-1}1)^{-1} + (1^T\Sigma_D^{-1}1)^{-2}\cdot 1^T\Sigma_D^{-1}\nu_\Sigma\Sigma_D^{-1}1 \\ &= J_D^{opt} + (J_D^{opt})^2 R \end{aligned}$$

□

The value J_D^{opt} is the optimal value the objective function would take if Σ were truly diagonal (i.e. $= \Sigma_D$).

3. Online Model Weighting Estimation

We now describe the adaptive controller based on online model weighting adaptation as introduced in [19]. Figure 2.2 shows the structure of the adaptive control system. The plant model is estimated online and the estimated model modifies the controller settings. Thus the structure is that of an *indirect adaptive controller.* We

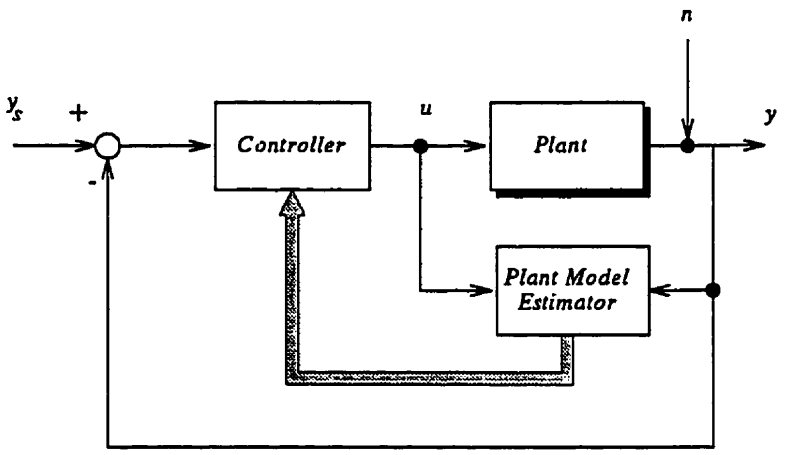

FIGURE 2.2. Structure of the adaptive controller. The structure is that of an indirect certainty equivalence controller.

adopt a certainty equivalence policy by which the controller bears a pre-established structure and desired performance but whose parameters are a function of the unknown model parameters which are replaced at every sampling instant by their current estimates.

We return to the two-indexing system where the model gains are drawn from **G** and the delays from **D**. Let g_i and d_j be the members of **G** and **D** respectively. The plant model is then written

$$P_m(z^{-1}) = \sum_{i=1}^{L_g}\sum_{j=1}^{L_d} w_{ij}P_{ij}(z^{-1}) = \sum_{i=1}^{L_g}\sum_{j=1}^{L_d} w_{ij}g_i\frac{1-\alpha}{1-\alpha z^{-1}}z^{-d_j-1}$$

$$= \frac{1-\alpha}{1-\alpha z^{-1}}\sum_{j=1}^{L_d}\gamma_j z^{-d_j-1} \tag{3.2.1}$$

where

$$\gamma_j = \sum_{i=1}^{L_g} w_{ij}g_i \tag{3.2.2}$$

The steady-state gain of the model is therefore $\gamma = \sum_{j=1}^{L_d}\gamma_j$.

Controller synthesis. We make use of the simple pole-placement controller synthesis proposed in [19]. The desired closed-loop system is specified to be

$$R(z^{-1}) = \frac{(1-\beta)z^{-1}}{1-\beta z^{-1}} \sum_{j=1}^{L_d} (\gamma_j/\gamma) z^{-d_j} \tag{3.2.3}$$

where β is the desired closed-loop pole and the only performance-related tuning parameter. The open-loop zeros of the plant model are carried over to the closed-loop in order to avoid cancelling zeros outside the unit circle. Using Equations 3.2.1 and 3.2.3, the controller equation is given by

$$C(z^{-1}) = \frac{1-\beta}{\gamma(1-\alpha)} \frac{1-\alpha z^{-1}}{1-\beta z^{-1} - (1-\beta)\sum_j (\gamma_j/\gamma) z^{-d_j-1}} \tag{3.2.4}$$

As mentioned earlier, the controller parameters are functions of the unknown quantities γ_j and γ (which are direct functions of w_{ij}). Following the certainty equivalence policy, the unknown quantities are estimated online and the estimated values are inserted at every sampling instant in the design (Equation 3.2.4). In what follows, it is explained how the model weighting adaptive method approaches the on line estimation of w_{ij}.

Parameter estimation. In the off line case, the discrepancy between the dynamics of the plant and the dynamics of a model P_{ij} was measured as the 2-norm

$$\sigma_{ij} = \frac{1}{2\pi} \int_0^{2\pi} |P - P_{ij}|^2 d\omega$$

Assuming no disturbance, the computation of σ_{ij} amounts to driving P and P_{ij} by a common input, a pulse of unit amplitude, measuring the deviation e_{ij} between the plant output and the model output and computing the 2-norm of e_{ij}, i.e.

$$\sigma_{ij} = \|e_{ij}\|_2^2 = \sum_{k=0}^{\infty} e_{ij}^2(k)$$

In the on line case, we proceed similarly but replace the input signal by the signal generated by the controller and replace the 2-norm of the error signal by the truncated

2-norm

$$\sigma_{ij}(t) = \|e_{ij}(t)\|_{2,\lambda}^2 = \sum_{k=0}^{t} \lambda^{t-k} e_{ij}^2(k)$$

Following [19], the estimation scheme simply consists of setting the weight of a model to the inverse of $\sigma_{ij}(t)$ and normalizing

$$\hat{w}_{ij}(t) = \frac{1/\sigma_{ij}(t)}{\sum_{l,m} 1/\sigma_{lm}(t)} \tag{3.2.5}$$

Note the similarity between (3.2.5) and (2.2.6) (Reminder: (2.2.6) uses the one-index convention while (3.2.5) uses the two-index convention).

The coefficient λ is a scalar bounded by $0 < \lambda \leq 1$ which puts more or less emphasis on the more recent data. Its role is entirely similar to that played by the "forgetting factor" in conventional recursive least squares estimation ([21], [41]).

REMARK 3.1. *It is possible, as suggested by* [19], *to use a slightly more general weight assignment law, i.e.*

$$\hat{w}_{ij}(t) = \frac{1/\|e_{ij}(t)\|_{2,\lambda}^{\phi}}{n_w}$$

with the normalizing coefficient

$$n_w = \sum_l \sum_m 1/\|e_{lm}(t)\|_{2,\lambda}^{\phi}$$

where parameter ϕ may be used to increase the discrimination between the members of $\mathcal{F}$. In the following analysis, we will under most circumstances investigate the case $\phi = 2$ but in most cases, the generalization to other values of ϕ will require little modifications and will be commented wherever appropriate.

Data filtering. In the on line case, the error e_{ij} is a function of the plant/model mismatch $P - P_{ij}$ but also of any disturbance showing up on the output of P. It may thus be desirable to filter the input/output data prior to using it for model estimation. In particular, if a dc offset is present, we want the data filter to remove this

component. Hence we will deal with filtered versions of the i/o data, i.e.

$$y^F(t) = F(q^{-1})y(t)$$

$$u^F(t) = F(q^{-1})u(t) \tag{3.2.6}$$

where, to remove dc offsets, the linear filter $F(q^{-1})$ has a zero at $q = 1$. The model error signal is

$$e_{ij}(t) = F(q^{-1})(y(t) - P_{ij}(q^{-1})u(t)) = y^F(t) - P_{ij}(q^{-1})u^F(t)$$

The filter is of the form

$$F(q^{-1}) = F'(q^{-1}) \cdot 1 - q^{-1}$$

Obviously, the filter plays an important role in shaping the input signal driving the estimator. Unless otherwise specified, the filter we will use when analyzing the algorithm will simply be the first difference operator, i.e. $F'(q^{-1}) = 1$.

4. Division by Zero

A division by zero occurs in Equation 3.2.5 if one or more error norms are exactly equal to zero. This may occur, in particular, if one member of $\mathcal{F}$ is an exact match to the true plant and there is no noise in the system. In [19], it was suggested to add a small positive offset to all error norms to ensure that all divisions remain bounded. This approach was found to be adequate in practice. Here we extend this idea further by allowing all models whose error signal is no greater than the disturbance level of the plant to be considered as yielding zero-level error norms.

Let η_t be the norm of the error signal of a model that *would be* a perfect match to the plant. Then, before computing the weights, the error norms are modified as follows:

$$\sigma_{ij}(t) = \sigma_{ij}(t)\mathbf{I}_{\sigma_{ij}(t)>\eta_t} + \mu \tag{4.2.1}$$

where **I** is the indicator function, i.e.

$$\mathbf{I}_A = \begin{cases} 1 & \text{if condition } A \text{ is satisfied} \\ 0 & \text{if not} \end{cases}$$

In [19], μ was some arbitrarily small positive quantity. Instead of carrying small μ quantities, we chose instead to assign to $\hat{w}_{ij}(t)$, the limit of (3.2.5) as μ tends to zero, i.e.

$$\hat{w}_{ij}(t) = \lim_{\mu \to 0} \frac{1/\sigma_{ij}(t)}{\sum_{l,m} 1/\sigma_{lm}(t)} \tag{4.2.2}$$

Appendix B gives a simple computer algorithm that can implement the above two steps (Equations 4.2.1 and 4.2.2). We leave it to a subsequent chapter to discuss the merits of actually using (4.2.2) on line instead of the straight offseting proposed in [19]. However, the analysis of the algorithm is greatly simplified by (4.2.1) and (4.2.2). The above two-step procedure is illustrated by the model initialization.

Initialization. Suppose the algorithm is initialized at $t = 0$. The error norms prior to startup are $\sigma_{ij}(t) = 0 \; \forall i, j$ and $t < 0$. Then for an arbitrary $\eta_t > 0$, we have

$$\hat{w}_{ij}(t) = \lim_{\mu \to 0} \frac{1/\mu}{\sum_{l=1}^{L_g} \sum_{m=1}^{L_d} 1/\mu} = \frac{1}{L_g \cdot L_d} = \frac{1}{N} \triangleq \tilde{w}_{ij} \text{ for } t < 0 \tag{4.2.3}$$

for all $i \in [1, L_g] \subset \mathbb{Z}$ and all $j \in [1, L_d] \subset \mathbb{Z}$. The corresponding γ_j parameters are

$$\hat{\gamma}_j(t) = \sum_{i=1}^{L_g} \tilde{w}_{ij} g_i = \frac{\sum_{i=1}^{L_g} g_i}{L_d L_g} \triangleq \tilde{\gamma}_j \text{ for } t < 0 \tag{4.2.4}$$

and the initial steady state gain is

$$\gamma(t) = \sum_{j=1}^{L_d} \tilde{\gamma}_j = \frac{\sum_{i=1}^{L_g} g_i \cdot L_d}{L_g L_d} = \frac{\sum_{i=1}^{L_g} g_i}{L_g} = \tilde{\gamma} \text{ for } t < 0 \tag{4.2.5}$$

In the sequel, the "tilded" variables refer to the above initial values.

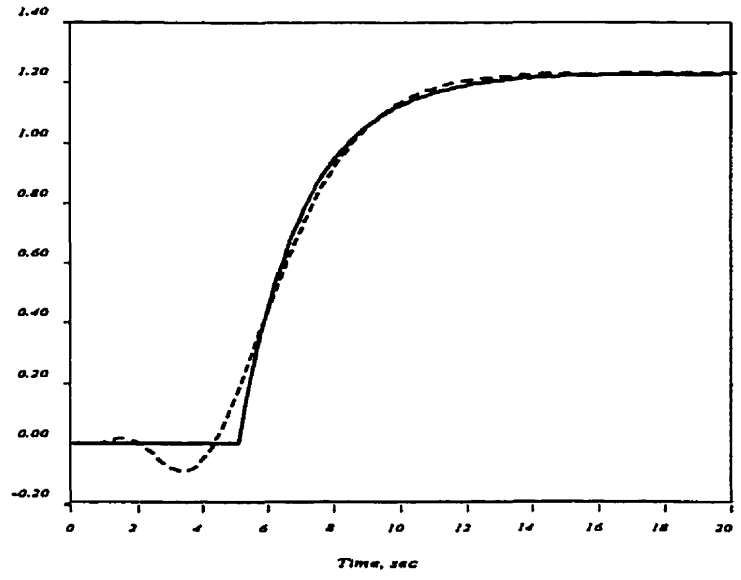

FIGURE 2.3. Step response of systems P_1 (black) and P_2 (dashed).

5. Examples

To help clarify the ideas discussed in this chapter and set the stage for the analysis to follow, we give a few examples of online model weighting adaptive control. Consider the two continuous-time systems:

$$P_1(s) = 1.23\frac{e^{-5s}}{2s+1} \qquad P_2(s) = -1.23\frac{(s-1)(s-2)^2}{(s+1)^4(s+2)^2}$$

Although they appear structurally quite different, a plot of their step response (Figure 2.3 show that these two systems have "similar" behaviors. In fact the first could be interpreted as a first-order-plus-delay approximation to the high-order system (P_2).

We want to change the output of those systems by one unit by feedback control using only approximate knowledge of the system. If we make use of the model weighting adaptive technique described in the previous section and select the tuning parameters shown in Table 2.1, we get , as expected very different results.

Parameter	Symbol	Value
Sampling period	T	1
Partitioned gain set	$\mathbf{G}$	[0.5 0.75 1.0 1.25 1.5]
Partitioned delay set	$\mathbf{D}$	[1 2 3 $\cdots$ 10]
Assumed open loop time constant	τ	2
Closed loop pole	β	0.5
Forgetting factor	λ	0.95
Data filter	$F(q^{-1})$	$1-q^{-1}$
Disturbance detection threshold	η_t	0

TABLE 2.1. Table of tuning parameters for the examples.

Figure 2.4 shows the behavior of both systems under closed loop. In the case of P_1, the response is quite close to what would be expected from a controller designed with complete knowledge of the system. It gives a smooth response with a slight overshoot caused by an initial underestimation of the true plant gain. In the case of P_2, the response indicates a narrower stability margin of the closed loop system Obviously, this is caused by the additional poles and right-half-plane zeroes of P_2 whose effect cannot be replicated by any member of $\mathcal{F}$.

If we examine the behavior of the estimated model parameters over the simulation length, we notice that in the case of P_1 (Figure 2.5) the gains γ_j associated to each delay (Equation 3.2.2) are initially uniformly distributed. As the plant input changes with no visible response from the plant output, the first parameters γ_1, γ_2, ... are sequentially set to zero. When the plant finally responds, then the distribution of the γ's quickly changes to a sharp peak for the model with the correct delay and the model gain is approximately that of the true plant.

On the other hand, for the plant $P_2(s)$ controlled with the same adaptive feedback controller, the response, albeit stable is not as good as that of $P_1(s)$. The adapted gains $\hat{\gamma}_j$ behave initially as with P_1 (see Figure 2.6). However, after approximately 4-5 samples, the bulk of the response has taken place and the estimator has processed most of the information available for deciding what model(s) is(are) more accurate. The result is expectedly less decisive as it is with $P_1(s)$ as the final distribution of the

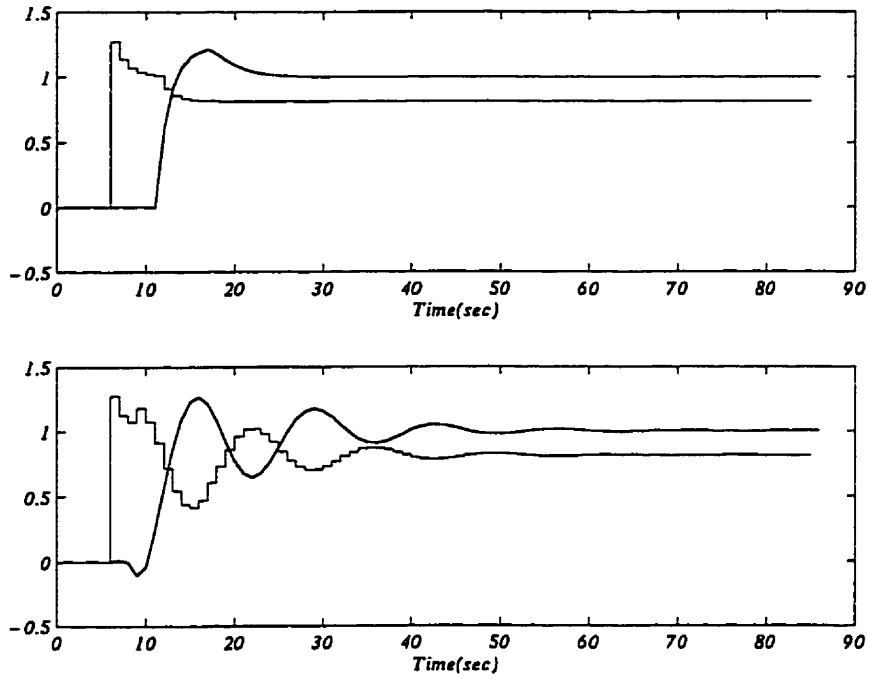

FIGURE 2.4. Response of the closed loop systems for the plant a) $P_1(s)$ and b) $P_2(s)$. The stepwise curve is the plant input and the continuous curve, the output.

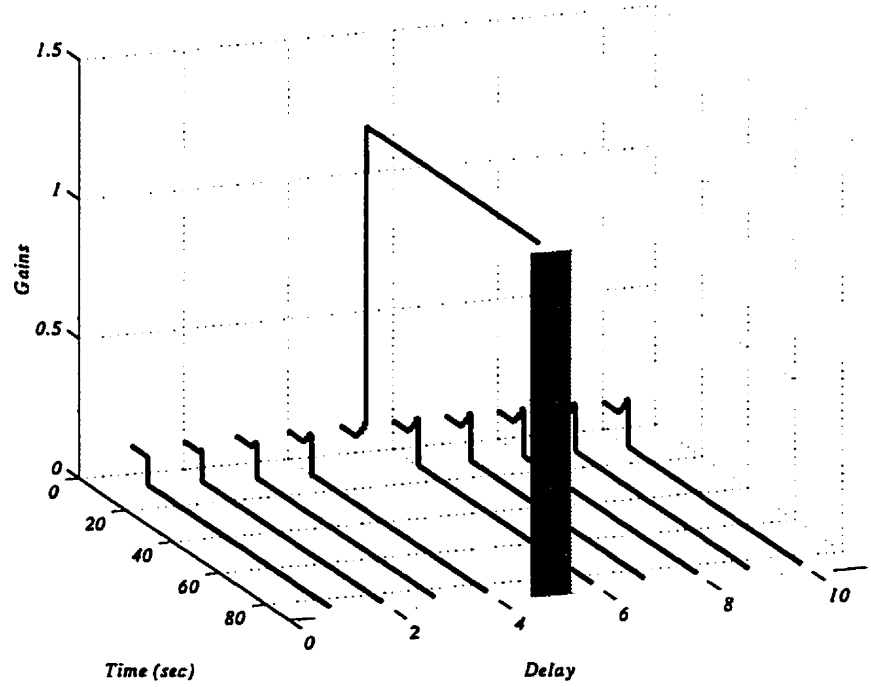

FIGURE 2.5. Adaptation of the gains γ_j over the simulation period for plant P_1.

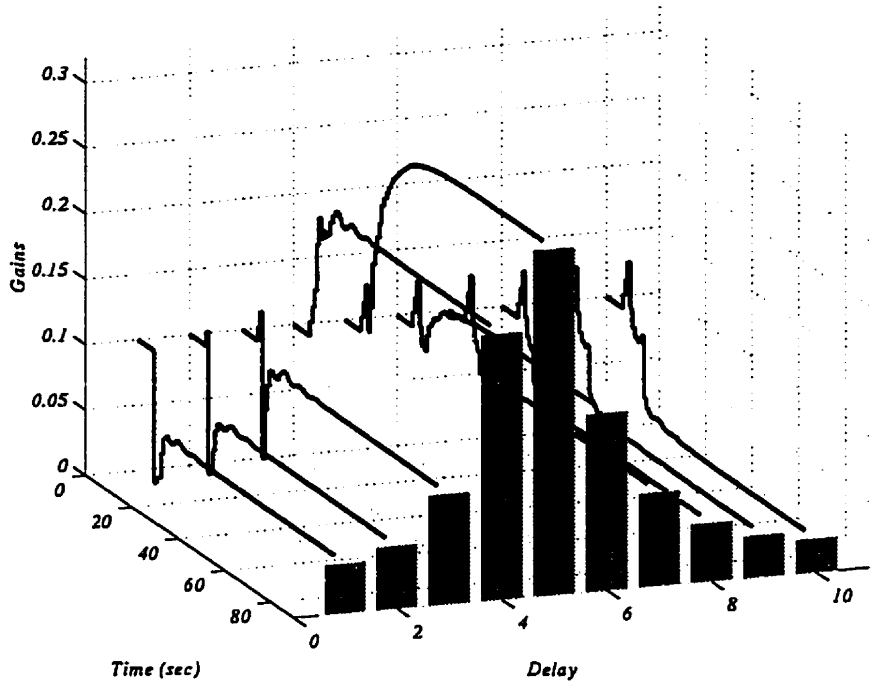

FIGURE 2.6. Adaptation of the gains γ_j over the simulation period for plant P_2.

gains is maximum for a delay of 5 sampling intervals and smoothly decays on both sides of the peak.

Summary

In this section, we have reviewed the basic ideas behind MWAC. The examples have shown that:

(i) The algorithm can lock itself very rapidly on an appropriate model.

(ii) Despite the resemblance between their step responses, the closed-loop performance of two systems using MWAC with identical tuning parameters can be significantly different.

In the following chapter, we theoretically investigate the algorithm and show that these observations are completely predictable from the theory.

CHAPTER 3

Model Estimation Using MWAC

1. The Approach

In this chapter, we analyze the model estimation properties of MWAC. The purpose of model estimation is to acquire information in order to refine the plant-related information which may be readily available prior to initiating the estimation algorithm. The principal virtue of prior knowledge is to limit the search space and possibly increase the efficiency of the search algorithm.

As was described in the previous chapter, the MWAC estimator does assume some prior knowledge about the plant's general behaviour. For instance, it is assumed that the true plant belongs to some neighborhood of the set of proposed models and that all its poles are within the unit circle. The members of this set $\mathcal{F}$ are all first order plus delay models. All members of $\mathcal{F}$ are similar except for their gain and delay which are chosen to cover a range of possible plant gains and delays. The construction of the boundaries of $\mathcal{F}$ (the gain and delay intervals and the dominant time constant) may possibly have been determined by prior experimentation with the plant. As stated in Chapter 2, the partitioned family $\mathcal{F}$ embodies the prior knowledge (The partitioning is introduced only for making the estimation computations feasible) and the search space is the convex hull of $\mathcal{F}$, $\mathbf{Co}(\mathcal{F})$. The members of $\mathbf{Co}(\mathcal{F})$ have the z-domain

form:

$$P_m(z^{-1}) = \sum_{i=1}^{L_g} \sum_{j=1}^{L_d} w_{ij} P_{ij}(z^{-1}) \tag{1.3.1}$$

where the models P_{ij} stem from the partition of the gain and delay intervals. As described in Chapter 2, the weights w_{ij} are modified online so that P_m is made to ressemble P in some sense. We introduce the "frozen time" model

$$P_m^t(z^{-1}) = \sum_i \sum_j \hat{w}_{ij}(t) P_{ij}(z^{-1})$$

where $\hat{w}_{ij}(t)$ is the time-varying adjustable weight associated to model P_{ij}. We use the term "frozen time" model to mean the frequency response the model would take on if its time-varying parameters were frozen from the current instant t to infinity.

To judge the resemblance of P_m^t with P, we need some measure of plant/model discrepancy, written $\delta(P, \hat{P}_m^t)$. This will likely be some frequency domain norm such as

$$\begin{aligned} \delta(P, P_m^t) &= \|P - P_m^t\|_p \\ \delta(P, P_m^t) &= \|F(P - P_m^t)\|_p \end{aligned}$$

where p will either be 2 or ∞ and where F is some linear filter. Because of its dependence on $\hat{P}_m^t$, $\delta(P, P_m^t)$ is also a frozen time function.

To make the earlier statement about the purpose of model estimation more precise, we say that the refinement of the plant model consists of shrinking the initial uncertainty $\delta(P, \hat{P}_m^0)$ through the acquisition and transformation of plant-related information which comes in the form of input/output data.

The analysis of parameter estimation algorithms generally focuses on the asymptotic behaviour of the estimates and related variables ([**12**], [**29**], [**21**]). This approach is natural since there exists a wealth of results on convergence for both deterministic and stochastic frameworks. The issue of the quality of the input/output data

is an inescapable offshoot of convergence analysis. In short, it is desirable that the data provided to the estimation scheme is information-rich in the sense that the application of the algorithm to this data will provide effective shrinking of the model uncertainty. When the estimation is done using some form of recursive least-squares algorithm, it is customary ([**21**], [**35**]) to require the plant input/output data to be persistently exciting (PE) such that, as the number of data points grows to infinity, the parameter error consistently shrinks to zero (in the deterministic case). In the stochastic case, this condition causes the parameter error to converge to zero *almost surely*, (i.e. except on a set of measure-zero sample paths) and *almost everywhere* (i.e. on all of the parameter space with the exception of a set of measure zero)([**43**], [**36**]). The persistent excitation condition translates into special requirements on the shape of the manipulable signals that are *external* to the adaptive loop. In practice, these requirements may consists of adding a PRBS signal to the plant input or making sure that the external signals have a minimum number of spectral lines [**11**]. It is known [**10**] that for some common adaptive control schemes, one cannot do without the PE requirement.

In practice, the signals which are external to the control loop may not fulfill the PE condition. For instance, setpoint or load step changes provide a burst of information about the plant dynamics in the moments that follow the appearance of the change but as soon as the transient disappears into the plant noise, plant/model dynamic mismatches become unobservable.

We set out in this work to examine the model tracking properties of MWAC when the excitation signals are limited to the ones that are likely to be found when the MWAC algorithm is put into application (commonly step setpoint and/or disturbance signals). As mentioned earlier, such signals provide plant information for only short bursts. In this context, the behaviour of the parameter estimates as $t \rightarrow \infty$ is of auxiliary interest since the determinant amount of model error shrinking is accomplished in a short interval following the application of the external excitation. So instead

of the asymptotic analysis, we will focus the analysis of the algorithm described in Chapter 2 on its behaviour over finite intervals.

We have seen that when the system is initialized, the initial or "default" controller is computed from the sum of all models of $\mathcal{F}$ uniformly weighted. If we excite the system (by changing the setpoint for instance) but the plant does not respond, or, produces an output indistinguishable from the noise for some portion of time following the application of the excitation, this data provides some information on the delay but none on the plant gain and thus allows only partial discrimination between the members of $\mathcal{F}$. An interesting and crucial consequence of this is that during this finite portion of time the controller modifies its settings in such a way that there exists a known linear, time-invariant transfer function that reproduces exactly the input/output behaviour of the nonlinear, time-varying controller, a behaviour later dubbed *local linear invariance.* It follows that this linear transfer function shapes the exogenous signals into the signals fed to the estimation subsystem at least for some short period. Knowing the properties of the exogenous signals, it is thus possible to bound the information provided to the estimator and compute a bound on the modeling error of the estimated model.

The analysis presented in this chapter consists of determining how this bound varies as a function of the user-selected parameters and the *a priori* assumptions on the noise and the unmodeled dynamics.

1.1. The user-selected parameters. The user-selected parameters derive in part from the prior knowledge about the plant, i.e. the upper and lower bounds on the plant gain and delay and the dominant open-loop time constant. The user also chooses the sampling period T (the delay uncertainty interval is partitioned accordingly) and the partition of the gain interval. We will assume, for now, a uniform partition of the gain interval, i.e.

$$g_l = \underline{g} + (l-1)\Delta g \quad \text{with } l = 1, \cdots, L_g \quad L_g > 1 \tag{1.3.2}$$

where Δg ($\triangleq (\overline{g} - \underline{g})/(L_g - 1)$) is the *partitioning resolution.*

The choice of the overbounding signal η_t in (4.2.1) should reflect the fact that if the amplitude of a model error signal is indistinguishable from the noise level, this model should be prejudiced as being a good candidate for the plant model.

A sensible value of λ in the error norm $\|e_{ij}(t)\|^2_{2,\lambda}$ must also be selected. The value of λ is bounded by $(0, 1]$. A small value rapidly discounts old data while a value close to 1 sustains the influence of old data on the current estimates. The parameter λ here plays a role similar to the *forgetting factor* found in recursive-least-squares algorithms. Hence we keep this designation in the rest of the document.

Finally, the user must select the desired performance of the closed-loop system. Using the pole-placement algorithm described in Chapter 2, this boils down to choosing the closed-loop pole β in Equation 3.2.3.

REMARK 1.1. *From Equations 4.2.3 and 4.2.4, $\tilde{\gamma}_j$ and $\tilde{\gamma}$ are functions of user-selected parameters.*

REMARK 1.2. *As briefly mentioned in Chapter 2 (page 32), it is also possible to select the value of an extra parameter, the discrimination parameter ϕ to base the estimation on a power of $\|e_{ij}(t)\|_{2,\lambda}$ different than 2.*

1.2. Assumptions on the plant, plant uncertainty and noise. It is important to bear in mind that in this analysis, the plant is LTI and thus that it is fixed but unknown. A consequence of this is that, once the model estimator has perceived that the model is close enough to the true plant, it may freeze its parameters. From that instant on, the whole closed-loop system is then also LTI. We erect the above statement to the

ASSUMPTION 1.1 (LTI plant). *The true plant is stable, linear, time-invariant and strictly proper.*

The above then means that the plant admits a representation of the form

$$y_p(t) = \sum_{k=1}^{\infty} p(k)u(t-k) \tag{1.3.3}$$

where u and y_p are the plant input and output signals respectively. The sequence $p(k)$ is the *discrete impulse response* of the plant. The latter thus admits the z-domain representation

$$P(z^{-1}) = \sum_{k=1}^{\infty} p(k) z^{-k} \tag{1.3.4}$$

with a bounded frequency response on $z = e^{j\omega}$, $\omega \in [0, 2\pi]$.

1.3. Apparent plant delay. Delay-like behaviours in chemical processes appear because of at least four different causes: 1) true transport delay, 2) finite, high-order dynamics and 3) distributed-parameter systems and 4)right-half plane zeroes. The first is found for instance on a paper machine where new fibers are added at the wet-end of the machine to increase the basis weight of the sheet of paper. The new fibers have to travel all the way from the wet-end to the dry-end of the paper machine before their additional weight may be sensed by the basis weight gauge. The second cause is more common. For example, if a stream goes through a series of mixing tank, a sudden change in the composition of the stream entering the first tank will show up in the stream leaving the last tank only after a time interval which depends on the flow and the tank volumes. The third situation occurs in continuous-feed reactors (e.g. bleaching tower) whose dynamics can be described by partial differential equations. The solution to these equations typically yield non-rational functions of the Laplace variable s and have phase responses which vary with the reactor dimensions. Finally, right-half plane zeroes cause a process output to initially move in a direction opposite to its final response thus "delaying" the plant response. This property of right-half plane zeroes is in fact at the center of the Padé approximation of time delays.

We thus need some practical definition of the "true plant delay" which reconciles all possible plant delay incarnations. Bélanger [8] (p. 122) suggests a definition based on the step response of the plant. There the plant delay is defined to be the time required for the step response to reach 50% of its final value. This is a simple and intuitive definition of the plant delay. The specific percentage is actually arbitrary and

we may rewrite this definition using an arbitrary percentage $\mathbf{r}_s$ which then becomes another user-selected parameter. We could formally write

$$d \triangleq \max_t \text{ for which } |s(t)| = \left|\sum_{k=0}^{t} p(k)\right| < \mathbf{r}_s \text{ holds}$$

We could also generalize the above by introducing an exponential weight, i.e.

$$d \triangleq \max_t \text{ for which } \left|\sum_{k=0}^{t} \lambda^{-k} p(k)\right| < \mathbf{r}_s \text{ holds}$$

It will turn out that the analysis carried out in this chapter will largely be set in the frequency domain. It will therefore be convenient to adopt a frequency-domain definition of the apparent plant delay which is less intuitive but is closely connected to the above definitions.

Truncation operator. Equation 1.3.4 may be interpreted as the frequency response of the plant based on a complete knowledge of its impulse response. Since we assume that we do not know precisely the impulse reponse of the plant but that we progressively improve our knowledge, we will often encounter *partial* frequency responses based on a truncated impulse response, we denote this partial frequency response as

$$\left[P(z^{-1})\right]_t \triangleq \sum_{k=1}^{t} p(k) z^{-k} \tag{1.3.5}$$

Any norm functional applied to this partial frequency response will be denoted

$$|[P(e^{-j\omega})]_t|_{p,\lambda}$$

In particular, we note that there is a close connection between the ∞-norm of the partial frequency response and the definition of the apparent delay. More precisely

THEOREM 1.1.

$$|[P(e^{-j\omega})]_t|_{\infty,\lambda} \leq \mathbf{r}_s \Longrightarrow |\sum_{k=0}^{t} \lambda^{-k} p(k)| \leq \mathbf{r}_s$$

PROOF. We have

$$|[P(e^{-j\omega})]_t|_{\infty,\lambda} = \sup_{\omega} \left| \sum_{k=1}^{t} p(k)\lambda^{-k} e^{-jk\omega} \right|$$

By definition, the ∞-norm is greater or equal to the magnitude of the frequency response at any frequency. It is true in particular at $\omega = 0$. It follows that

$$\begin{aligned} \mathbf{r}_s &\geq |[P(e^{-j\omega})]_t|_{\infty,\lambda} \\ &\geq \left| \sum_{k=0}^{t} \lambda^{-k} p(k) \right| \end{aligned}$$

□

This leads to the following definition of the apparent plant delay.

DEFINITION 1.1 (Apparent plant delay). *The apparent time delay d of a transfer function P is the time required for the ∞-norm of the truncated transfer function $[P]_t$ to be greater than a given fraction $\mathbf{r}_s$, i.e.*

$$d \triangleq \max_t \textit{ for which } |[P]_t|_{\infty,\lambda} \leq \mathbf{r}_s \textit{ holds}$$

The following proposition provides useful results on the norms of the partial frequency responses.

PROPOSITION 1.1. *Let $Q(z^{-1}) = Q_1(z^{-1})\, Q_2(z^{-1})$ where Q, Q_1 and Q_2 admit a Laurent series expansion. Then we have*

$$|[Q]_t|_{\infty,\lambda} \leq |[Q_1]_{t_1}|_{\infty} |[Q_2]_{t_2}|_{\infty} \mathbf{B}(t,\lambda)$$

where $t_1 \geq t$, $t_2 \geq t$ and $\mathbf{B}(t,\lambda)$ is the fixed function

$$\mathbf{B}(t,\lambda) = \frac{1}{2\pi} \int_0^{2\pi} \left| \frac{1 - (e^{j\theta}/\lambda)^{t+1}}{1 - e^{j\theta}/\lambda} \right| d\theta$$

PROOF. Given in Appendix. □

Disturbance. The actual measured output is the noise-corrupted signal

$$y(t) = y_p(t) + n(t)$$

where $n(t)$ is a disturbance which belongs to a class of signals that we now characterize. A standard way to characterize noise in a deterministic framework consists in assuming that the value of $n(t)$ is uniformly bounded, i.e.

ASSUMPTION 1.2 (Noise assumption 1).

$$|n(t)| \leq \delta_n \; \forall \; t \in [0, \infty) \tag{1.3.6}$$

The above noise assumption is simple and convenient but often leads to overly pessimistic results. For instance, applying the input/output data filter $F(q^{-1}) = 1 - q^{-1}$ (Eq. 3.2.6) to $y(t)$ yields

$$y^F(t) = y_p^F(t) + n^F(t)$$

The signal used for estimation is thus corrupted by a noise signal of twice the original magnitude, i.e.

$$|n^F(t)| \leq 2\delta_n$$

The above bound is uniformly satisfied with equality sign only when the noise signal jumps back and forth between plus and minus δ_n at every time interval. This is highly unrealistic. Furthermore, the above noise formulation does not permit the treatment of the disturbance signal as a source of plant/model mismatch information and restricts the role of $n(t)$ to that of a nuisance. We also consider the less general but very useful class of noise signal

ASSUMPTION 1.3 (Noise assumption 2).

$$|n^F(t)| \leq \delta_n r_n^t \tag{1.3.7}$$

where $0 < r_n < 1$.

The above class of noise includes perturbations caused by non-zero initial conditions. It also includes particular forms of noise that simultaneously corrupt the output and provide valuable input/output information. For instance,(1.3.7) can accomodate step disturbances of magnitude δ_n by letting $r_n \to 0$.

2. Framework and definitions

In this section, we establish the framework for analyzing the MWAC algorithm. We first assume that the controller is set in operation at some time t_0. Assume that prior to t_0, the plant input was moving about some steady operating point u^* and write the input signal as the deviations from u^*, i.e.

$$u(t) = u^* + \delta_u(t)$$

The plant output is then given by

$$\begin{aligned} y(t) &= \sum_{k=1}^{\infty} p(k)u(t-k) + n(t) \\ &= \sum_{k=1}^{t_0} p(k)u(t-k) + \sum_{k=t_0+1}^{\infty} p(k)u(t-k) + n(t) \\ &= \sum_{k=1}^{t_0} p(k)\delta_u(t-k) + u^* \sum_{k=1}^{\infty} p(k) + \sum_{k=t_0+1}^{\infty} p(k)\delta_u(t-k) + n(t) \end{aligned}$$

Let $y^* = u^* \sum_{k=1}^{\infty} p(k)$. We then have

$$\delta_y(t) = y(t) - y^* = \sum_{k=1}^{t_0} p(k)\delta_u(t-k) + \underbrace{\sum_{k=t_0+1}^{\infty} p(k)\delta_u(t-k)}_{\text{initial conditions}} + n(t) \tag{2.3.1}$$

The initial conditions term in the above equation is decaying with t and can thus be assimilated to a disturbance term compatible with Assumption 1.3 for some δ_n and r_n.

Finally, recalling the property of the data filter $F(1) = 0$ (page 33) we note that u^* and y^* have no bearing on parameter estimation. Hence we may consider, with no loss of generality, that $u^* = y^* = 0$ (and thus $u(t) = \delta_u(t)$ and $y(t) = \delta_y(t)$).

The above discussion leads to the following assumption:

ASSUMPTION 2.1 (Rest conditions). *When the adaptive controller is set into operation at $t = t_0$, the system has been at rest for some time, i.e.*

$$u(t) = 0 \qquad \text{for } t < t_0$$

Local linear invariance. The introduction of the adaptive controller is made to coincide with the application at $t = t_0$ of an external excitation: a step setpoint change of magnitude y_s. We know that for some time following the application of this excitation, the plant output is "small" (in a sense made precise later) because of the presence of a time delay (real or apparent) in the plant dynamics. Let d be this apparent time delay.

We show later that for the time interval $[t_0, t_0 + d]$, the actual adaptation of the controller parameters causes the output of the nonlinear, time-varying controller to coincide exactly with the output computed from a linear invariant difference equation driven by the same input signal as the controller. Furthermore, the parameters of the difference equation are known a priori. We now give a formal definition of what we mean by *local linear invariance* (LLI).

DEFINITION 2.1 (Local linear invariance). *An operator $H(u(t))$ is said to be locally linear invariant over a finite time interval if during this interval, its output may be replicated exactly by the solution of an invariant linear difference equation driven by the same input.*

REMARK 2.1. *Obviously, the above concept is somewhat related to the input signal. In the present work, we restrict ourselves to step-like signals and do not seek further generalization of the LLI concept. We conjecture that this idea can be made to encompass a larger, more general class of signals but leave this for future work.*

To distinguish the input/output behaviour of a LLI operator from its associated LTI operator, we will write them as

$$y(t) = H(u(t)) \equiv \mathsf{H}^{\mathcal{T}}(q^{-1})u(t)$$

where $\mathcal{T}$ is the time interval when $H(\cdot)$ and $\mathsf{H}^{\mathcal{T}}$ are equivalent and q^{-1} is the usual shift operator.

More definitions. We close this section by giving a few definitions that will be handy in the sequel. First note that t_0 may be arbitrarily selected. For simplicity and convenience, we select $t_0 = 0$. It was mentioned earlier that in the time interval $[0, d]$, the controller is LLI. A consequence of that is that it is easy to obtain the (noise-corrupted) shape of the plant input signal for this time segment. This input serves as the initial excitation to the estimation subsystem. We thus label the above time interval, the *initial excitation* interval, i.e.

$$\mathcal{T}_I \triangleq [0, d]$$

It will be convenient to partition $\mathcal{F}$ into subsets with equal time delays, i.e.

$$\mathcal{F}_k \triangleq \{P_{ij} \in \mathcal{F} \mid g_i \in \mathbf{G}, d_j = k\} \text{ and } \bigcup_{k \in \mathbf{D}} \mathcal{F}_k = \mathcal{F}$$

Obviously if $d \in [\underline{d}, \overline{d}]$, then there is one such subset (call it $\mathcal{F}_d$) which is of special interest. The remaining models of $\mathcal{F}$ can be regrouped in subsets of models with delay lower than the true delay ($\underline{\mathcal{F}}$) and models with delay higher than the true delay ($\overline{\mathcal{F}}$). More precisely

$$\underline{\mathcal{F}} \triangleq \bigcup_{k<d} \mathcal{F}_k$$

$$\overline{\mathcal{F}} \triangleq \bigcup_{k>d} \mathcal{F}_k$$

The index sets of the above subsets of $\mathcal{F}$ are written $I_{\mathcal{F}_k}$ and are interpreted

$$(i,j) \in I_{\mathcal{F}_k} \Rightarrow P_{ij} \in \mathcal{F}_k$$

3. Analysis of MWAC

3.1. Basic relations. Consider the diagram in Figure 3.1. The diagram shows the dynamic components used to generate the error signal associated to a particular member of $\mathcal{F}$. All blocks are LTI dynamic systems except for the controller block $\bar{C}$ which is nonlinear and time-varying.

However, assume that for the finite time interval $\mathcal{T}_I$, $\bar{C}$ is LLI, i.e. its input/output behaviour is exactly replicated by $\mathsf{F}_u^{\mathcal{T}_I}(q^{-1})$. Hence

$$u(t) = \mathsf{F}_u^{\mathcal{T}_I}(q^{-1})(y_s(t) - y(t))$$

And we may write

$$u(t) = \frac{\mathsf{F}_u^{\mathcal{T}_I}(q^{-1})}{1 + \mathsf{F}_u^{\mathcal{T}_I}(q^{-1})P(q^{-1})}(y_s(t) - n(t)) \qquad \text{for } t \in \mathcal{T}_I$$

and (dropping the q^{-1} argument for concision)

$$\begin{aligned} e_{ij}(t) &= F(P - P_{ij})u(t) + Fn(t) \\ &= (P - P_{ij})\mathsf{F}_u^{\mathcal{T}_I} \cdot \frac{1}{1 + \mathsf{F}_u^{\mathcal{T}_I}P} \cdot (y_s^F(t) - n^F(t)) + n^F(t) \qquad \text{for } t \in \mathcal{T}_I \end{aligned} \tag{3.3.1}$$

Equation 3.3.1 reveals that the error signals are the sum of two components: 1) an "observer" of the plant/model mismatch driven by the exogenous input signals and 2) a disturbance signal which is common to all models. We can then relate the error norm to the various components of the system via the theorem

THEOREM 3.1. *Provided that Assumptions 1.1-2.1 are satisfied then for some $t \geq d$, there exist coefficients K_X, K_F and K_n such that*

$$\|e_{ij}(t)\|_{2,\lambda} = \|[(G - G_{ij})]_t\|_{\infty,\lambda}\, K_F\, K_X + K_n$$

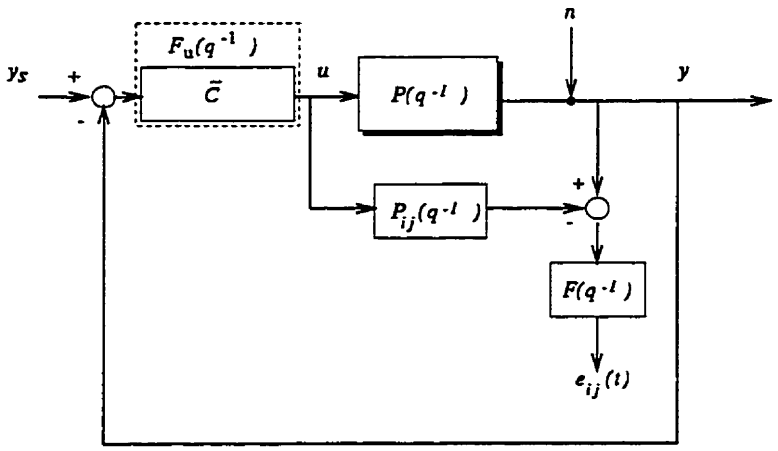

FIGURE 3.1. Flow diagram of the blocks involved in computing the error signal.

where $G = PF_u^{\mathcal{T}_I}$ *is the LLI loop gain,* $G_{ij} = P_{ij}F_u^{\mathcal{T}_I}$ *and where* K_F, K_X *and* K_n *are common to all error signals and are bounded by*

$$\begin{aligned} \frac{1}{1+|[G]_t|_{\infty,\lambda}} \leq \quad & K_F \quad \leq \frac{1}{1-|[G]_t|_{\infty,\lambda}} \\ |y_s| + \underline{|K_n|} \leq \quad & K_X \quad \leq |y_s| + \overline{|K_n|} \\ \underline{|K_n|} \leq \quad & K_n \quad \leq \overline{|K_n|} \end{aligned}$$

where

$$\overline{|K_n|} = -\underline{|K_n|} = \delta_n \sqrt{\frac{\lambda^{t+1} - (\tau_n^2)^{t+1}}{\lambda - \tau_n^2}}$$

PROOF. Proof in appendix. □

For future reference, we will use $\overline{|K_F|}$ and $\overline{|K_X|}$ to denote above upper bounds. Similarly, we will use $\underline{|K_F|}$ and $\underline{|K_X|}$ for the lower bounds.

There is a corollary to the above theorem that will be useful in demonstrating the LLI property of the controller.

COROLLARY 3.1. *Provided that Assumptions 1.1-2.1 are satisfied then we may also write the norm of the error signal as*

$$\|e_{ij}(t)\|_{2,\lambda} = |[G_{ij}]_t|_{\infty,\lambda} K_F' K_X' + |[G]_t|_{\infty,\lambda} K_F' K_X' + K_n'$$

where $K'_F K'_X$ and K'_n have the same bounds as K_F, K_X and K_n in Theorem 3.1.

PROOF. The proof parallels the proof of Theorem 3.1 where G and G_{ij} are treated separately. □

3.2. Local linear invariance of the controller. In the previous section, we have derived equations to represent the effects of the exogenous input and the disturbances on the error norms. These calculations were based on the assumption that in some finite portion of time following the application of the exogenous input, the controller mimicked a known linear invariant system (a behaviour dubbed local, linear invariance - LLI). In this section we find a sufficient condition that validates this assumption.

The first step of this demonstration consists of showing that the members of $\underline{\mathcal{F}}$ are going to be sequentially excited and their associated weights be sequentially set to zero. Sequentially meaning, first the members of $\mathcal{F}_{d_1}$, then the members of $\mathcal{F}_{d_2}$ and so on until $t = d+1$ where d is the apparent time delay that we formally defined in Definition 1.1.

The proof relies on the exogenous signals to provide sufficient information to discriminate inappropriate models. The term "inappropriate" is meant to qualify those models whose error signal rises above a disturbance level which is recognized as being uninformative.

Providing sufficient information formally means that we need to have a sufficient signal to noise ratio **SNR**. We may define the **SNR** in terms of the ratio of the norms of the exogenous and noise signals, i.e.

$$\mathbf{SNR} = \left| \frac{K_X}{K_n} \right|$$

LEMMA 3.1. *Let*

$$\eta_t = \frac{\mathbf{r}_s}{1 - \mathbf{r}_s} \overline{|K_X|} + \overline{|K_n|} \tag{3.3.2}$$

and

$$\underline{\underline{\mathbf{SNR}}} \triangleq \sup_{t\in[\underline{d},d]} \sup_{P_{ij}\in\mathcal{F}} \frac{2(1+\mathbf{r}_s)}{|[G_{ij}]_t|_{\infty,\lambda} - 2\mathbf{r}_s}$$

Then $\mathbf{SNR} > \underline{\underline{\mathbf{SNR}}}$ *implies*

$$\begin{aligned}
\hat{w}_{i1}(t) &= 0 \qquad \textit{for} \quad t \in [d_1, d] \\
\hat{w}_{i2}(t) &= 0 \qquad \textit{for} \quad t \in [d_2, d] \\
&\vdots \qquad\qquad\quad \vdots \\
\hat{w}_{ij}(t) &= 0 \qquad \textit{for} \quad t \in [d_j, d]
\end{aligned}$$

for $\forall\ d_j < d$.

PROOF. From Corrolary 3.1, we have

$$\|e_{ij}(t)\|_{2,\lambda} \geq |||[G_{ij}]_t|_{\infty,\lambda} K'_X K'_F| - |||[G]_t|_{\infty,\lambda} K'_X K'_F| - |K'_n|$$

The first term on the right-hand-side of the above is the one that must make G_{ij} stand above the noise level. The second term is the initial response of the loop transfer function which is deemed to be small (according to the definition of the apparent delay) and which is lumped with the additive noise.

To guarantee that $\|e_{ij}(t)\|_{2,\lambda}$ rises above the noise level we then must have

$$|[G_{ij}]_t|_{\infty,\lambda}\ |K'_F K'_X| > 2(|[G]_t|_{\infty,\lambda} |K'_F K'_X| + |K'_n|)$$

After a few manipulations, this requirement translates into

$$\mathbf{SNR} = \left|\frac{K'_X}{K'_n}\right| > \frac{2}{|K'_F|\ (|[G_{ij}]_t|_{\infty,\lambda} - 2|[G]_t|_{\infty,\lambda})} \triangleq \underline{\mathbf{SNR}}$$

Let

$$\begin{aligned}
\underline{\underline{\mathbf{SNR}}} &\triangleq \sup_{P_{ij}\in\mathcal{F}} \sup_{K_F} \sup_{t\in[\underline{d},d]} \underline{\mathbf{SNR}} \\
&= \sup_{P_{ij}\in\mathcal{F}} \sup_{t\in[\underline{d},d]} \frac{2(1+\mathbf{r}_s)}{|[G_{ij}]_t|_{\infty,\lambda} - 2\mathbf{r}_s}
\end{aligned}$$

Setting the noise level threshold to

$$\begin{aligned}\eta_t &= \sup_{K'_F}\sup_{K'_n}\sup_{t\in[\underline{d},d]} |[G]_t|_{\infty,\lambda}\,|K'_F|\,|K'_X| + |K'_n| \\ &= \frac{\mathbf{r}_s}{1-\mathbf{r}_s}\overline{|K_X|} + \overline{|K_n|}\end{aligned}$$

implies that if $\mathbf{SNR} > \underline{\underline{\mathbf{SNR}}}$, it follows that for $t \in [d_j + 1, d]$, $\|e_{ij}(t)\|_{2,\lambda} > \eta_t$ for $\forall\, P_{ij} \in \bigcup_{l=\underline{d}}^{d_j} \mathcal{F}_l$ and all others are set to μ (Equation 4.2.1). Hence for $t < d$ and $\forall\, P_{ij} \in \bigcup_{l=\underline{d}}^{t-1} \mathcal{F}_l$

$$\begin{aligned}\hat{w}_{ij}(t) &= \lim_{\mu\to 0} \frac{1/(\sigma_{ij}(t)+\mu)}{\sum_{(l,m)\in I_{\cup_{l=\underline{d}}^{t-1}\mathcal{F}_l}} 1/(\sigma_{lm}(t)+\mu) + \sum_{(l,m)\in I_{\cup_{l=t}^{\overline{d}}\mathcal{F}_l}} 1/\mu} \\ &= 0\end{aligned}$$

and for $\forall\, P_{ij} \in \bigcup_{l=t}^{\overline{d}} \mathcal{F}_l$

$$\begin{aligned}\hat{w}_{ij}(t) &= \lim_{\mu\to 0} \frac{1/\mu}{\sum_{(l,m)\in I_{\cup_{l=\underline{d}}^{t-1}\mathcal{F}_l}} 1/(\sigma_{lm}(t)+\mu) + \sum_{(l,m)\in I_{\cup_{l=t}^{\overline{d}}\mathcal{F}_l}} 1/\mu} \\ &= \frac{1}{L_g(\overline{d}-t+1)}\end{aligned}$$

□

The above result is central to the proof of

THEOREM 3.2. *If η_t is given by (3.3.2), $\mathbf{SNR} > \underline{\underline{\mathbf{SNR}}}$ and if Assumption 2.1 is satisfied, i.e.*

$$u(t) = 0 \quad \text{for } t < 0 \tag{3.3.3}$$

then the adaptive controller constructed by modifying on line the parameters of Equation 3.2.4 with the certainty equivalence principle is LLI over $[0, d]$.

PROOF. First we write the adaptive controller in its time-varying form

(3.3.4)

$$u(t) = \frac{1-\beta}{\hat{\gamma}(t)(1-\alpha)}(e(t) - \alpha e(t-1)) + \beta u(t-1) + (1-\beta)\sum_{j=1}^{L_d}\left(\frac{\hat{\gamma}_j(t)}{\hat{\gamma}(t)}\right)u(t-d_j-1)$$

Up until $t = \underline{d}$, $\sigma_{ij}(t) = \mu$ for $\forall P_{ij} \in \mathcal{F}$ and thus $\hat{w}_{ij}(t) = 1/N = \tilde{w}_{ij}$, $\hat{\gamma}_j(t) = \tilde{\gamma}_j$ and $\hat{\gamma}(t) = \tilde{\gamma}$, i.e. all parameters keep their initial values.

Using condition (3.3.3), it is clear that for $t \in [0, \underline{d}]$

$$u(t) = \frac{1-\beta}{\tilde{\gamma}(1-\alpha)}(e(t) - \alpha e(t-1)) + \beta u(t-1) \tag{3.3.5}$$

At $t = \underline{d} + 1 = d_1 + 1$, $u(t) \neq 0$ but from Lemma 3.1, we have $\hat{w}_{i1}(t) = 0$ for $\forall\, P_{ij} \in \mathcal{F}_1$ and $\hat{w}_{ij}(t) = 1/L_g(\overline{d} - \underline{d})$ for $\forall\, P_{ij} \in \bigcup_{l=d_2}^{\overline{d}} \mathcal{F}_l$. This implies that

$$\hat{\gamma}_j(t) = \begin{cases} 0 & \text{if } j = 1 \\ \frac{1}{L_g}\frac{1}{\overline{d}-\underline{d}}\sum_i g_i & \text{for all others} \end{cases}$$

and

$$\begin{aligned} \hat{\gamma}(t) &= \sum_{j=1}^{L_d}\hat{\gamma}_j(t) = 0 + \frac{L_d - 1}{\overline{d} - \underline{d}}\frac{1}{L_g}\sum_i g_i \\ &= \frac{1}{L_g}\sum_i g_i = \tilde{\gamma} \end{aligned}$$

Replacing in (3.3.4), we see that $u(t)$ remains equal to (3.3.5).

At $t = \underline{d} + 2 = d_2 + 1$, we have from Lemma 3.1 that $\hat{w}_{ij}(t) = 0$ for all models in $\bigcup_{l=d_1}^{d_2}\mathcal{F}_1$ and $\hat{w}_{ij}(t) = 1/L_g(\overline{d} - \underline{d} - 1)$ for all models in $\bigcup_{l=d_3}^{\overline{d}}\mathcal{F}_l$. It follows that at $t = \underline{d} + 2$, we have

$$\hat{\gamma}_j(t) = \begin{cases} 0 & \text{if } j = 1 \text{ or } 2 \\ \frac{1}{L_g}\frac{1}{\overline{d}-\underline{d}-1}\sum_i g_i & \text{for all others} \end{cases}$$

and

$$\begin{aligned}\hat{\gamma}(t) &= \sum_{j=1}^{L_d} \hat{\gamma}_j(t) = 0 + 0 + \frac{L_d - 2}{\overline{d} - \underline{d} - 1} \frac{1}{L_g} \sum_i g_i \\ &= \frac{1}{L_g} \sum_i g_i = \bar{\gamma}\end{aligned}$$

Replacing in (3.3.4) shows that $u(t)$ remains equal to (3.3.5).

Repeating the above argument up to $t = d$ proves the result. □

3.3. Characterization of model adequacy. A definition of the adequacy of a plant model is bound to be attached to some norm applied to the difference between the dynamics of the model and that of the actual plant. The actual choice may be dictated by the end use of the model. In this case, the end use is the design of a feedback controller. Hence we may decide that a model is adequate if the closed-loop system that it yields meet some performance specifications.

In this context, the *sensitivity* function of the closed-loop system is an appealing mathematical device to study: its a direct measure of disturbance rejection capability and it provides an indication of the relative stability of the closed loop system. It thus simultaneously provides an indicator of both performance and robustness of the closed-loop system.

Recall from Section 3 that the closed-loop specification (i.e. the transfer function between the setpoint and the output) is given by

$$R(z^{-1}) = \frac{(1-\beta)z^{-1}}{1-\beta z^{-1}} \sum_{j=1}^{L_d} (\gamma_j/\gamma) z^{-d_j}$$

Note that the specification is a function of the model. Hence we write the frozen-time specification $R^t(z^{-1})$ as

$$R^t(z^{-1}) = \underbrace{\frac{(1-\beta)z^{-1}}{1-\beta z^{-1}}}_{B(z^{-1})} \underbrace{\sum_{j=1}^{L_d} (\hat{\gamma}_j(t)/\hat{\gamma}(t)) z^{-d_j}}_{\hat{\Gamma}^t(z^{-1})/\hat{\gamma}(t)}$$

The controller is synthesized by

$$C^t(z^{-1}) = \frac{R^t(z^{-1})}{1 - R^t(z^{-1})} \frac{1}{P_m^t(z^{-1})}$$

from which it follows that the "frozen-time" sensitivity function is given by

$$S^t(z^{-1}) = \left(1 - R^t(z^{-1})\right) \frac{1}{1 + R^t(z^{-1}) \left((P(z^{-1}) - P_m^t(z^{-1})/P_m^t(z^{-1})\right)}$$

Noting that $R^t = F_u A\hat{\Gamma}\tilde{\gamma}/\hat{\gamma}$ and $P_m^t = A\hat{\Gamma}$, we can further simplify the above to

$$S^t(z^{-1}) = \left(1 - R^t(z^{-1})\right) \frac{1}{1 + (\tilde{\gamma}/\hat{\gamma}(t)) \ (G(z^{-1}) - G_m^t(z^{-1}))} \tag{3.3.6}$$

From which may obtain the bound

$$|[S^t]_t|_\infty \leq |[S_0^t]_t|_\infty \cdot \frac{1}{1 - |\tilde{\gamma}/\hat{\gamma}(t)| \ |[G - G_m^t]_t|_\infty} \tag{3.3.7}$$

$$\leq |[S_0^t]_t|_\infty \cdot \frac{1}{1 - |\tilde{\gamma}/\hat{\gamma}(t)| \ |[G - G_m^t]_t|_{\infty,\lambda} \cdot \mathbf{B}(t, \lambda^{-1})} \tag{3.3.8}$$

(From Proposition 1.1)

In the above equation, S_0^t is the frozen time sensitivity function of the closed-loop system if the model is a perfect match to the true plant, i.e. $G = G_m^t$. Equation 3.3.8 also provides a connection between the sensitivity of the closed-loop system and a measure of the plant/plant model discrepancy. Furthermore this discrepancy is expressed directly in terms of the norm used in Theorem 3.1.

We now need to answer the following questions:

(i) How does the model error norm $|[G - G_m^t]_t|_{\infty,\lambda}$ relate to the error norm of the individual members of $\mathcal{F}$, i.e. $|[G - G_{ij}]_t|_{\infty,\lambda}$?

(ii) Is it possible to determine on-line how much shrinking of the model error norm is being accomplished?

A partial answer is provided by the following theorem which establishes a connection between the model error norm and the error norm of the individual members of $\mathcal{F}$.

THEOREM 3.3.

$$|[G - G_m^t]_t|_{\infty,\lambda} \leq \frac{\mathbf{W}(t)}{\sqrt{\sum_{i,j} (|[G - G_{ij}]_t|_{\infty,\lambda} + \rho)^{-2}}} - \rho$$

where

$$\mathbf{W}(t) = \sum_{i,j} \sqrt{\hat{w}_{ij}(t)}$$

and

$$\rho = \frac{K_n}{K_X K_F}$$

PROOF. First we make use of the convexity of the norm operator to bound $|[G - G_m^t]_t|_{\infty,\lambda}$, i.e.

$$\begin{aligned} |[G - G_m^t]_t|_{\infty,\lambda} &= |[\sum_{i,j} \hat{w}_{ij}(t)(G - G_{ij})]_t|_{\infty,\lambda} \\ &\leq \sum_{i,j} \hat{w}_{ij}(t)|[G - G_{ij}]_t|_{\infty,\lambda} \quad \text{(Jensen's Inequality)} \end{aligned} \tag{3.3.9}$$

From Theorem 3.1, we may write

$$\hat{w}_{ij}(t) = \frac{(|[G - G_{ij}]_t|_{\infty,\lambda} K_F K_X + K_n)^{-2}}{n_w} \tag{3.3.10}$$

where n_w is the normalizing factor

$$n_w = \sum_{i,j} (|[G - G_{ij}]_t|_{\infty,\lambda} K_F K_X + K_n)^{-2}$$

We rearrange (3.3.10) into

$$|[G - G_{ij}]_t|_{\infty,\lambda} = \frac{(\hat{w}_{ij}(t)\, n_w)^{-1/2} - K_n}{K_X K_F}$$

and replace in (3.3.9) to get

$$\sum_{i,j} \hat{w}_{ij}(t)|[G - G_{ij}]_t|_{\infty,\lambda} = \frac{(n_w)^{-1/2}}{K_X K_F} \sum_{i,j} \hat{w}_{ij}(t)^{1/2} - \frac{K_n}{K_X K_F}$$

Let $\mathbf{W}(t) \triangleq \sum_{i,j} \hat{w}_{ij}^{1/2}$ and replace the definition of n_w in the above. This leads to

$$\begin{aligned}\sum_{i,j} \hat{w}_{ij}(t)|[G - G_{ij}]_t|_{\infty,\lambda} &= \frac{\mathbf{W}(t)}{K_X K_F \sqrt{\sum_{i,j}(|[G - G_{ij}]_t|_{\infty,\lambda} K_X K_F + K_n)^{-2}}} - \frac{K_n}{K_X K_F} \\ &= \frac{\mathbf{W}(t)}{\sqrt{\sum_{i,j}(|[G - G_{ij}]_t|_{\infty,\lambda} + \rho)^{-2}}} - \rho\end{aligned}$$

where $\rho = K_n / K_X\, K_F$. □

The above theorem is a central result of this thesis. We can further simplify it through

COROLLARY 3.2.

$$|[G - G_m^t]_t|_{\infty,\lambda} \leq \mathbf{W}(t)\underline{\nu} + (\mathbf{W}(t) - 1)\rho$$

where $\mathbf{W}$ *and* ρ *are defined as in Theorem 3.3 and*

$$\underline{\nu} \triangleq \inf_{\mathcal{F}} |[F_u(P - P_{ij})]_t|_{\infty,\lambda}$$

PROOF. It follows immediately from Theorem 3.3 that

$$\begin{aligned}|[G - G_m^t]_t|_{\infty,\lambda} &\leq \frac{\mathbf{W}(t)}{\sqrt{\sum_{i,j}(|[G - G_{ij}]_t|_{\infty,\lambda} + \rho)^{-2}}} - \rho \\ &\leq \mathbf{W}(t)|[G - G^*]_t|_{\infty,\lambda} + (\mathbf{W}(t) - 1)\rho\end{aligned}$$

where $G^* = F_u \cdot P^*$ and P^* is an arbitrary member of $\mathcal{F}$. Select P^* such that

$$\underline{\nu} \triangleq |[G - G^*]_t|_{\infty,\lambda} \leq |[G - G_{ij}]_t|_{\infty,\lambda} \quad \forall\, P_{ij} \in \mathcal{F}$$

□

Theorem 3.3 and its corollary provides the following clues:

- The variable $\mathbf{W}(t)$ is a measure of the "discrimination" between the individual members of $\mathcal{F}$. For instance, when one and only one model in $\mathcal{F}$ reproduces exactly the plant output, its weight is equal to 1, all other weights are equal to

zero and $\mathbf{W}(t) = 1$. At the other end of the scale, when no particular member of $\mathcal{F}$ provides a significantly better approximation of the true plant and all weights are approximately equal to $1/N$ then $\mathbf{W}(t) \approx \sqrt{N}$.

- If $\mathbf{W}(t)$ can be made as close as possible to 1 by proper selection of the controller parameters, then $[G]_t$ is no further away from G_m than it is from the nearest member of $\mathcal{F}$ and the effect of noise (as embodied by ρ in Theorem 3.3) becomes negligible.

To rephrase item 1 above, if one considers the set of weights $\hat{w}_{ij}$ as a two-dimensional "map", then $\mathbf{W}(t)$ is a measure of the sharpness or flatness of this map. We now determine how this measure varies with the user-selected parameters.

4. Tracking the model properties

As we have seen in the previous section, it is important in order to establish bounds for tracking the performance of the adaptation mechanism to resort to the infinite norm of frequency functions. We thus found it useful to express the model error norm $\|e_{ij}(t)\|_{2,\lambda}$ in terms of the infinite norm of the difference between the transfer function of the true plant and the associated member of $\mathcal{F}$ (Theorem 3.1). To study the effect of the user-selected parameters, we will find it useful to express $\|e_{ij}(t)\|_{2,\lambda}$ in terms of the 2-norm of $P - P_{ij}$ (with frequency weighting F_u). For this we need a result provided by a theorem similar to Theorem 3.1.

THEOREM 4.1. *Provided that Assumptions 1.1-2.1 are satisfied then, for some $t \geq d$ there exist coefficients k_X,K_F and K_n such that*

$$\|e_{ij}(t)\|_{2,\lambda} = |[(G - G_{ij})]_t|_{2,\lambda}\, K_F\, k_X + K_n$$

where K_F and K_n are bounded as in Theorem 3.1 and k_X is bounded by

$$|y_s| - \delta_n \frac{1 - (r_n/\sqrt{\lambda})^{t+1}}{1 - r_n/\sqrt{\lambda}} \leq k_X \leq |y_s| + \delta_n \frac{1 - (r_n/\sqrt{\lambda})^{t+1}}{1 - r_n/\sqrt{\lambda}}$$

PROOF. Proof in Appendix. □

Next we see that the effect of undermodeling, i.e. the effect caused by using a elementary low order model (or a set of such models) and the effect of the additive noise is essentially the same: a flattening of the weight map that would be computed in a disturbance-free setting.

4.1. The effects of undermodelling and additive noise. We start by defining and quantifying what we mean by undermodelling in order to suit the requirements of our analysis. Let

$$\mathcal{G} \triangleq \left\{ G_0 = g_0 \frac{1-\beta}{1-\beta z^{-1}} z^{-d-1} \,\middle|\, g_0 \in \mathbb{R} \right\}$$

and pick some $G_0^* \in \mathcal{G}$.

It follows that we can bound the error norm the following way

$$\begin{aligned}
\|e_{ij}(t)\|_{2,\lambda} &= |[G - G_{ij}]_t|_{2,\lambda} K_F k_X + K_n \quad \text{from Theorem 4.1} \\
&= |[G - G_0^* + G_0^* - G_{ij}]_t|_{2,\lambda} K_F k_X + K_n \\
&\leq (|[G - G_0^*]_t|_{2,\lambda} + |[G_0^* - G_{ij}]_t|_{2,\lambda}) K_F k_X + K_n \\
&\leq (|[G - G_0^*]_t|_2 + |[G_0^* - G_{ij}]_t|_{2,\lambda}) K_F k_X + K_n \quad \text{since } \lambda < 1 \\
&\leq (\|G - G_0^*\|_2 + |[G_0^* - G_{ij}]_t|_{2,\lambda}) K_F k_X + K_n \\
&\qquad \text{since} |[\,\cdot\,]_t|_2 \text{ is monotone increasing with respect to } t
\end{aligned}$$

We conclude that there exists a coefficient K_{G_0} bounded by $\|G - G_0^*\|_2$ such that

$$\|e_{ij}(t)\|_{2,\lambda} = (|[G_0^* - G_{ij}]_t|_{2,\lambda} + K_{G_0}) K_F k_X + K_n$$

and we may write the weight estimates as

$$\begin{aligned}
\hat{w}_{ij}(t) &= \left(\sum_l \sum_m \left(\frac{\|e_{ij}(t)\|_{2,\lambda}}{\|e_{lm}(t)\|_{2,\lambda}} \right)^2 \right)^{-1} \\
&= \left(\sum_l \sum_m \left(\frac{|[G_0^* - G_{ij}]_t|_{2,\lambda} + K_{G_0} + (K_F k_X)^{-1} K_n}{|[G_0^* - G_{lm}]_t|_{2,\lambda} + K_{G_0} + (K_F k_X)^{-1} K_n} \right)^2 \right)^{-1} \qquad (4.3.1)
\end{aligned}$$

The above form of the weight estimates is very convenient for evaluating the effects of disturbances on the adaptation mechanism. Let

$$K_D \triangleq K_{G_0} + (K_F k_X)^{-1} K_n$$

be the combined effect of neglected dynamics and additive noise. From (4.3.1), we see that the weight estimates are a function of two agents with opposite aims: one *discriminating* agent which discriminates between the members of $\mathcal{F}$ according to the norm $|[G_0^* - G_{ij}]_t|_{2,\lambda}$ and one *uniformizing* agent, the offset K_D which steers the weights towards a uniform distribution. The following theorem makes this more explicit.

THEOREM 4.2. *If* $|[G_0^* - G_{ij}]_t|_{2,\lambda} \neq 0 \ \ \forall P_{ij} \in \mathcal{F}$, *then*

$$\hat{w}_{ij}(t) = \frac{\left(\alpha_{ij}\sqrt{\hat{w}_{ij}^0(t)} + (1-\alpha_{ij})\frac{1}{\sqrt{N}}\right)^2}{\sum_l \sum_m \left(\alpha_{lm}\sqrt{\hat{w}_{lm}^0(t)} + (1-\alpha_{lm})\frac{1}{\sqrt{N}}\right)^2} \tag{4.3.2}$$

where

$$\hat{w}_{ij}^0(t) = \frac{1/|[G_0^* - G_{lm}]_t|_{2,\lambda}^2}{n_w^0}$$

$$n_w^0 = \sum_l \sum_m |[G_0^* - G_{ij}]_t|_{2,\lambda}^{-2}$$

$$\alpha_{ij} = \frac{1}{1 + \rho_D \sqrt{\hat{w}_{ij}^0(t)}}$$

$$\rho_D = K_D \cdot \sqrt{n_w^0}$$

PROOF. Let $\rho_{lm} = (K_F K_X)^{-1} K_n / |[G_0^* - G_{lm}]_t|_{2,\lambda}$ and

$$\rho_D^2 = \sum_l \sum_m \rho_{lm}^2$$

Then we have

$$K_D = \frac{\rho_D}{\sqrt{n_w^0}}$$

Replacing in (4.3.1), we get

$$\begin{aligned}
\hat{w}_{ij}(t) &= \left(\sum_l \sum_m \left(\frac{|[G_0^* - G_{ij}]_t|_{2,\lambda} + \rho_D/\sqrt{n_w^0}}{|[G_0^* - G_{lm}]_t|_{2,\lambda} + \rho_D/\sqrt{n_w^0}} \right)^2 \right)^{-1} \\
&= \left(\sum_l \sum_m \left(\frac{\sqrt{n_w^0}|[G_0^* - G_{ij}]_t|_{2,\lambda} + \rho_D}{\sqrt{n_w^0}|[G_0^* - G_{lm}]_t|_{2,\lambda} + \rho_D} \right)^2 \right)^{-1} \\
&= \left(\sum_l \sum_m \left(\frac{(\hat{w}_{ij}^0(t))^{-1/2} + \rho_D}{(\hat{w}_{lm}^0(t))^{-1/2} + \rho_D} \right)^2 \right)^{-1} \\
&= \frac{\left(\sqrt{\hat{w}_{ij}^0(t)}/1 + \rho_D \sqrt{\hat{w}_{ij}^0(t)} \right)^2}{\sum_l \sum_m \left(\sqrt{\hat{w}_{ij}^0(t)}/1 + \rho_D \sqrt{\hat{w}_{lm}^0(t)} \right)^2}
\end{aligned}$$

Multiplying the numerator and the denominator of the above by $(1 + \rho_D/\sqrt{N})^2$ and performing a few manipulations produces the result. □

The coefficient ρ_D in the above theorem is a measure of the relative importance of the disturbance level with respect to discrimination provided by the set of error norms. The set of weights $\hat{w}_{ij}^0$ is the weight distribution when no form of disturbance is present in the system. Theorem 4.2 establishes a convex link between $\sqrt{\hat{w}_{ij}^0(t)}$ and $N^{-1/2}$ simultaneously for all pairs (i, j) of gains and delays through the coefficients α_{ij}. This clearly shows that unmodeled dynamics and noise are "flattening" the noise-free weights $\hat{w}_{ij}^0$. It is not difficult to see that

$$\lim_{\rho_D \to \infty} \hat{w}_{ij}(t) = \frac{1}{N} \tag{4.3.3}$$

Although a convex sum links $\sqrt{\hat{w}_{ij}^0(t)}$ to $1/\sqrt{N}$, the relation between the weights $\hat{w}_{ij}(t)$ and the noise-free weights $\hat{w}_{ij}^0(t)$ for an arbitrary noise level ρ_D remain a complex function of ρ_D. We may approximate the relation however by looking at the

asymptotic behaviour of the coefficients α_{ij}, i.e.

$$\alpha_{ij} = \begin{cases} 1 & \text{when } \rho_D \ll (\hat{w}_{ij}^0(t))^{-1/2} \\ \frac{1}{\rho_D \sqrt{\hat{w}_{ij}^0(t)}} & \text{when } \rho_D \gg (\hat{w}_{ij}^0(t))^{-1/2} \end{cases}$$

and the breakpoint occurs approximately when $\rho_D \sqrt{\hat{w}_{ij}^0(t)} = 1$ or

$$\rho_D = \frac{1}{\sqrt{\hat{w}_{ij}^0(t)}}$$

It follows that the largest weight is the one which is most affected by noise, then the second largest and so on, while the noise-free weights which are close to zero require a large noise level for their associated α_{ij} coefficient be affected.

Finally we conclude that to completely describe the behaviour of the adaptive algorithm, it is sufficient to investigate the transfer function error norms $|[G_0^* - G_{ij}]_t|_{2,\lambda}$ since they play the dual role of establishing the disturbance-free weighting of the models **and** determining the abilitity of the model set to *resist* the flattening effect of the disturbances.

4.2. The base line weights distribution. From the previous section, we know that the final form of the plant model and thus of the closed-loop system is completely determined by the set of time-varying norms $|[G_0^* - G_{ij}]_t|_{2,\lambda}$ and the noise level as expressed by ρ_D. From the former we may calculate the "base line" weights distribution $\hat{w}_{ij}^0(t)$ and the interplay between $\hat{w}_{ij}^0(t)$ and ρ_D flattens the base line distribution. The exact value of ρ_D is unknown but from the assumptions about the noise levels and the true plant dynamics, we may compute an upper bound on ρ_D and thus evaluate "worst-case" scenarios about the final weights distribution. On the other hand, the norms $|[G_0^* - G_{ij}]_t|_{2,\lambda}$ are entirely function of user-selected parameters. In the following section, we examine the effect of these parameters on the base line weights distribution.

We have seen in a previous section that in the instants following the application of the setpoint, the controller is LLI up to the time instant d. The plant input signal

$u(t)$ for this time instant is then the solution to this LLI closed-loop system. This was the basis for computing the error norm bounds in Theorems 3.1 and 4.1. By causality, this input signal is also the one driving the plant and the models for the instants immediately following the time delay d, i.e. for $t = d+1, d+2, \cdots$. The system thus enters a short, finite phase where the plant and model sets are driven in virtual open loop. For this phase we may, recalling that

$$G_0^* = F_u P_0^* = \frac{g_0^*}{\tilde{\gamma}} \frac{1-\beta}{1-\beta z^{-1}} z^{-d-1}$$

compute $|[G_0^* - G_{ij}]_t|_{2,\lambda}$ for three specific subsets of $\mathcal{F}$, i.e. for those models with too short delays (i.e. $\underline{\mathcal{F}}$), too large delays (i.e. $\overline{\mathcal{F}}$) and $\mathcal{F}_d$.

For $t > d$, the error norms are given by (details given in Appendix C)

$$|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 = \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 \left[g_i^2 \lambda^{t-d} \frac{\lambda^{d-d_j} - (\beta^2)^{d-d_j}}{\lambda - \beta^2} + (g_0^* - g_i \beta^{d-d_j})^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2} \right] \quad \text{for } P_{ij} \in \underline{\mathcal{F}} \tag{4.3.4}$$

$$|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 = \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i)^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2} \quad \text{for } P_{ij} \in \mathcal{F}_d \tag{4.3.5}$$

$$|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 = \begin{cases} \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2} & \text{if } t \le d_j \\ \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 \left[(g_0^*)^2 \lambda^{t-d_j} \frac{\lambda^{d_j-d} - (\beta^2)^{d_j-d}}{\lambda - \beta^2} \right. & \\ \left. + (g_0^* \beta^{d_j-d} - g_i)^2 \frac{\lambda^{t-d_j} - (\beta^2)^{t-d_j}}{\lambda - \beta^2} \right] & \text{if } t > d_j \end{cases} \quad \text{for } P_{ij} \in \overline{\mathcal{F}} \tag{4.3.6}$$

From these equations, we could compute the weights distribution at $t = d+1, t = d+2, \cdots$ for any combination of parameter values. These weights however, would turn out to be a fairly complicated function of all these parameters. We prefer to simplify the above expressions so that the role of individual parameters is more transparent. In particular, we are interested the effect of the gain partitioning resolution Δg.

First, we introduce an assumption constraining the true steady state plant gain and "true" plant delay to be within the limits of the *a priori* intervals.

ASSUMPTION 4.1 (True plant gain and delay). *We assume that*

$$g_0^* \in [\underline{g}, \overline{g}]$$

and

$$d \in \mathbf{D}$$

Note that since g_0^* is not necessarily a member of the partitioned gain interval $\mathbf{G}$, we do not know *a priori* how far it is from *any* member of $\mathbf{G}$. Let δ^* be the minimum distance from g_0^* to any member of $\mathbf{G}$, i.e.

$$\delta^* \triangleq \min_{g_i \in \mathbf{G}} |g_0^* - g_i|$$

From Assumption 4.1, we know that there exists a member of $\mathbf{G}$ such that

$$\delta^* \leq \frac{\Delta g}{2} \tag{4.3.7}$$

and another member of $\mathbf{G}$ adjacent to the former such that

$$|g^* - g_k| = \Delta g - \delta^* \tag{4.3.8}$$

Next, notice that (4.3.4), (4.3.5) and (4.3.7) are all decaying functions of time unless λ or β are greater than 1 (Assume they never are). It is thus important to discriminate between the members of $\mathcal{F}$ early after $t = d$ has elapsed when the above norms are at their peak. Let $t = d + 1$ be the time instant at which we evaluate the

baseline distribution and let $S_d^0(t)$ be the sum of the weights of the members of $\mathcal{F}_d$.

$$
\begin{aligned}
S^0(d+1) &= \frac{\sum_{(l,m)\in I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}}{\sum_{(l,m)\in I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2} + \sum_{(l,m)\in I_{\mathcal{F}}\setminus I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}} \\
&= \left(1 + \frac{\sum_{(l,m)\in I_{\mathcal{F}}\setminus I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}}{\sum_{(l,m)\in I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}}\right)^{-1} \\
&= \left(1 + \frac{\sum_{(l,m)\in I_{\underline{\mathcal{F}}}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2} + \sum_{(l,m)\in I_{\overline{\mathcal{F}}}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}}{\sum_{(l,m)\in I_{\mathcal{F}_d}} |[G_0^* - G_{lm}]_{d+1}|_{2,\lambda}^{-2}}\right)^{-1} \qquad (4.3.9)
\end{aligned}
$$

Replacing (4.3.4), (4.3.5) and (4.3.7 in the above, and doing some simplifications, we come to the result of

THEOREM 4.3. *If Assumption 4.1 is true, then*

$$
S_d^0(d+1) > \left(1 + \Delta G^2 \frac{L_g}{(L_g-1)^2} \frac{\frac{1}{2}\left[\frac{1}{\underline{g}^2} + \frac{1}{\overline{g}^2}\right] \frac{1}{\lambda^{d-\underline{d}}} \frac{1-\lambda^{d-\underline{d}}}{1-\lambda} + \frac{\overline{d}-d}{\underline{g}^2}}{8}\right)^{-1}
$$

where $\Delta G = \overline{g} - \underline{g}$

PROOF. We begin by noting that Equation 4.3.4 is dominated by the first term within the brackets. Thus it is always possible to write

$$
|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 > \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 g_i^2 \lambda^{t-d} \frac{\lambda^{d-d_j} - (\beta^2)^{d-d_j}}{\lambda - \beta^2} \text{ for } P_{ij} \in \underline{\mathcal{F}}
$$

It follows that the sum over $\underline{\mathcal{F}}$ of the inverted error norms at $t = d+1$ is bounded by

$$\begin{aligned}
\sum_{(i,j)\in I_{\underline{\mathcal{F}}}} |[G_0^* - G_{ij}]_{d+1}|_{2,\lambda}^{-2} &< \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \sum_{(i,j)\in I_{\underline{\mathcal{F}}}} \frac{1}{g_i^2}\frac{1}{\lambda}\frac{\lambda-\beta^2}{\lambda^{d-d_j}-(\beta^2)^{d-d_j}} \\
&= \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{1}{\lambda}\sum_{i=1}^{L_g}\frac{1}{g_i^2}\sum_{d_j=\underline{d}}^{d-1}\frac{\lambda-\beta^2}{\lambda^{d-d_j}-(\beta^2)^{d-d_j}} \\
&= \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{1}{\lambda}\sum_{i=1}^{L_g}\frac{1}{g_i^2}\sum_{d_j=\underline{d}}^{d-1}\frac{\lambda}{\lambda^{d-d_j}}\frac{\overbrace{1-\beta^2/\lambda}}{\underbrace{1-(\beta^2/\lambda)^{d-d_j}}_{<1}} \\
&< \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{L_g}{2}\left[\frac{1}{\underline{g}^2}+\frac{1}{\overline{g}^2}\right]\frac{1}{\lambda^{d-\underline{d}}}\frac{1-\lambda^{d-\underline{d}}}{1-\lambda} \qquad (4.3.10)
\end{aligned}$$

where the last inequality is made possible through Jensen's inequality ([**39**], p. 128).

Using (4.3.7) and (4.3.8), we may bound the sum over $I_{\mathcal{F}_d}$ by

$$\left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \sum_{i=1}^{L_g}\frac{1}{(g_0^*-g_i)^2} > \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2\left(\frac{1}{(\delta^*)^2}+\frac{1}{(\Delta g-\delta^*)^2}\right)$$

Taking the derivative of the above bound and setting to zero shows that the bound is at a minimum when $\delta^* = \Delta g/2$. Hence

$$\sum_{i=1}^{L_g} |[G_0^* - G_{ij}]_{d+1}|_{2,\lambda}^{-2} > \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{8}{\Delta g^2} \quad \text{for } d_j = d \qquad (4.3.11)$$

Finally, we have, over $\overline{\mathcal{F}}$

$$\begin{aligned}
\sum_{(i,j)\in I_{\overline{\mathcal{F}}}} |[G_0^* - G_{ij}]_{d+1}|_{2,\lambda}^{-2} &= \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{L_g}{(g_0^*)^2}(\overline{d}-d) \\
&> \left(\frac{\tilde{\gamma}}{1-\beta}\right)^2 \frac{L_g}{\underline{g}^2}(\overline{d}-d) \qquad (4.3.12)
\end{aligned}$$

Replacing (4.3.10), (4.3.11) and (4.3.12) in (4.3.9) yields the result. □

It follows from Theorem 4.3 that we may set the weights of the models with innapropriate delays arbitrarily close to zero by choosing a fine enough resolution of

the gain interval, i.e.

$$\lim_{L_g \to \infty} S_d^0(d+1) = 1$$

Summary

In this chapter, we have established that in the first instants that follow the application of an external excitation, if the signal-to-noise ratio is sufficiently high, the nonlinear controller behaves equivalently to a known linear invariant system (see Lemma 3.1 and Theorem 3.2). This result has the following implication: the signal which is fed to the estimator is (at least) partially known and this information can be used to track an upper bound on the performance of the adjusted model (see Theorem 3.3 and Corollary 3.2).

Finally we showed how the performance upper bound is influenced by the choice of the tuning parameters, the signal-to-noise ratio and the amount of undermodeling (i.e. the approximation of a possibly high order plant by a first order plus delay model) (Theorems 4.2 and 4.3).

CHAPTER 4

Implementation of MWAC

Model weighting adaptive controllers have already been implemented in a number of industrial applications [19], [38], [24]. These implementations reproduce the algorithm given in [19]. In this chapter, we discuss two implementation issues of MWAC that stem from the analysis of previous chapters and lead to an improved algorithm.

1. The initial invariance of the controller

In Chapter 2, we introduced a modification to the algorithm presented in [19]. This modification worked in two step: 1) Set the models with error norms lower than some threshold be assigned a common "floating" value μ and compute the weight assignment law 2) When all possible cancellations of the common μ have been performed, let $\mu \rightarrow 0$.

The purpose of this modification was to ensure that the weights of the models with delay lower than the actual plant delay are sequentially set exactly to zero. The consequence is that the input/output behaviour of the nonlinear, time-varying controller is exactly that of a known LTI filter over a short period of time (and only over this period of time) following the application of the external excitation. As we saw in Chapter 3 the modification contributed significantly in simplifying the analysis of the adaptive system.

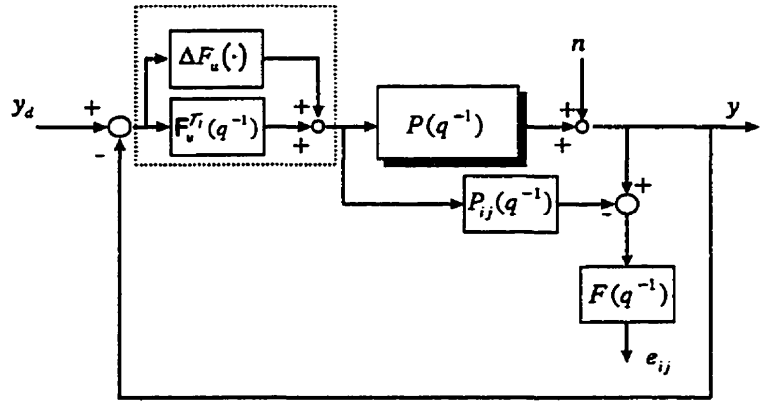

FIGURE 4.1. Modified flow diagram of the blocks involved in computing the error signal.

With the original algorithm, it has been observed experimentally that the weights of the models with the smallest delays are indeed set close to zero (as opposed to *exactly* equal to zero) when an external signal of sufficient magnitude excites the system (see examples of Chapter 2).

So, although it is possible to write an algorithmic procedure for implementing online the μ-modification of Chapter 2, it is worth questioning how significantly the extra code and additional computing time would contribute to the peformance of the actual online algorithm. Furthermore, we may wonder if a deviation from the assumed behaviour of the controller necessitates a different analysis than what was proposed in Chapter 3.

We therefore examine what is the net effect of the controller operator being only close to the LLI filter $F_u^{\mathcal{I}_I}(q^{-1})$. Consider the diagram of Figure 4.1. This is just a modified version of the diagram of Figure 3.1.

In the diagram, the extra branch $\Delta F_u(\cdot)$ is a bounded operator acting on the control error. This operator makes up the difference between the control operator

$\tilde{C}(\cdot)$ and $\mathsf{F}_u^{\mathcal{T}_I}(q^{-1})$. If all other variables are defined as before, we then have the following result.

PROPOSITION 1.1. *The control error $e_{ij}(t)$ associated to a model P_{ij} of $\mathcal{F}$ is given by*

$$e_{ij}(t) = \frac{\mathsf{F}_u^{\mathcal{T}_I}(P - P_{ij})}{1 + F_u P}(y_d^F(t) - (n')^F(t)) + (n')^F(t)$$

where

$$n'(t) = n(t) - \Delta v(t)$$

$$\Delta v(t) = -\frac{P - P_{ij}}{1 + \mathsf{F}_u^{\mathcal{T}_I} P_{ij}} \Delta u(t)$$

and

$$\Delta u(t) = \Delta F_u(y_d(t) - y(t))$$

PROOF. From Figure 4.1, we have

$$(y_d(t) - (Pu(t) + n(t)))\,\mathsf{F}_u^{\mathcal{T}_I} + \Delta F_u(y_d(t) - y(t)) = u(t)$$

from which we get

$$\begin{aligned} u(t) &= \frac{\mathsf{F}_u^{\mathcal{T}_I}}{1 + \mathsf{F}_u^{\mathcal{T}_I} P}(y_d(t) - n(t)) + \frac{1}{1 + \mathsf{F}_u^{\mathcal{T}_I} P}\Delta F_u(y_d(t) - y(t)) \\ &= \frac{\mathsf{F}_u^{\mathcal{T}_I}}{1 + \mathsf{F}_u^{\mathcal{T}_I} P}(y_d(t) - n(t)) + \frac{1}{1 + \mathsf{F}_u^{\mathcal{T}_I} P}\Delta u(t) \end{aligned}$$

Now since

$$e_{ij}(t) = (P - P_{ij})u^F(t) + n^F(t)$$

we then have

$$\begin{aligned} e_{ij}(t) &= \frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P}(y_d^F(t)-n^F(t)) + \frac{P-P_{ij}}{1+\mathsf{F}_u^{\mathcal{T}_I}P}\Delta u^F(t) + n^F(t) \\ &= \frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P}(y_d^F(t)-n^F(t)) - \left(-\frac{P-P_{ij}}{1+\mathsf{F}_u^{\mathcal{T}_I}P_{ij}}\Delta u^F(t)\right)\frac{1+\mathsf{F}_u^{\mathcal{T}_I}P_{ij}}{1+\mathsf{F}_u^{\mathcal{T}_I}P} + n^F(t) \end{aligned}$$

Let

$$\Delta v(t) = -\frac{P-P_{ij}}{1+\mathsf{F}_u^{\mathcal{T}_I}P_{ij}}\Delta u(t)$$

It follows that

$$\begin{aligned} e_{ij}(t) &= \frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P}(y_d^F(t)-n^F(t)) - \frac{1+\mathsf{F}_u^{\mathcal{T}_I}P_{ij}}{1+\mathsf{F}_u^{\mathcal{T}_I}P}\Delta v^F(t) + n^F(t) \\ &= \frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P}(y_d^F(t)-n^F(t)) + \left(\frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P} - 1\right)\Delta v^F(t) + n^F(t) \\ &= \frac{\mathsf{F}_u^{\mathcal{T}_I}(P-P_{ij})}{1+\mathsf{F}_u^{\mathcal{T}_I}P}(y_d^F(t)-n^F(t)+\Delta v^F(t)) - \Delta v^F(t) + n^F(t) \end{aligned}$$

Letting $n'(t) = n(t) - \Delta v(t)$ yields the result. □

Hence the net effect of $\Delta F_u(\cdot) \neq 0$ for the initial time interval $\mathcal{T}_I$ is exactly the same as the effect of having additional noise and then may be treated as such by the analysis of Chapter 3. We therefore conjecture that for external excitations of magnitude worthy of consideration, the conclusions of Chapter 3 are equivalent whether or not the μ-modification is implemented.

This brings us to the second issue of this chapter. We will now see that the above conjecture is further relevant if we introduce a mechanism for locking the model parameters when an adequate model has been found.

2. Locking the estimates

In earlier versions of MWAC ([**18**],[**19**]), no attempt was made to lock the parameters at any time and the weights were allowed to drift back to a uniform distribution

when the information provided by the external signals disapeared. Hence if a set-point change excited the system, the model used for feedback was based on whatever signal-to-noise ratio this change provided.

From Theorem 3.3 and its corrollary, we learned that there is a direct connection between the flatness of the weight map and the proximity of the model to the true plant. A large signal-to-noise ratio sharpens the map of model weights while a low signal-to-noise ratio flattens it. We also saw that a meaningful figure of the map flatness is given by $\sum_{ij}\sqrt{\hat{w}_{ij}(t)}$ which is easily calculated online. This suggests a mechanism for locking the parameter estimates based on this measure.

Consider the diagram in Figure 4.2. The model estimator uses the input/output data from the plant and the weight map is computed as before. At every sampling instant, the current map flatness $\mathbf{W}(t)$ is computed, i.e.

$$\mathbf{W}(t) = \sum_{ij}\sqrt{\hat{w}_{ij}(t)}$$

and compared to the smallest $\mathbf{W}$ so far. Formally, define the later as

$$\underline{\mathbf{W}}(t) = \inf_{k\in[0,t]}\mathbf{W}(k) = \bigwedge_{k=0}^{t}\mathbf{W}(k)$$

Hence at every sampling instant, we produce the following logic variable,

$$\mathbf{I}_{\mathbf{W}(t)<\underline{\mathbf{W}}(t-1)}$$

and test this variable to see if it is desirable to update the model or not.

$$\text{if } \mathbf{I}_{\mathbf{W}(t)<\underline{\mathbf{W}}(t-1)} = \begin{cases} 1 & \begin{cases} \text{Update model and let} \\ \underline{\mathbf{W}}(t) = \mathbf{W}(t) \end{cases} \\ 0 & \text{Do nothing} \end{cases}$$

The value of $\mathbf{W}(t)$ is bounded by 1 and $\sqrt{N}$. The upper bound corresponds to the case where input/output data does not yield enough information to discriminate between any member of $\mathcal{F}$ and is therefore also the initial value of $\mathbf{W}(t)$. As the

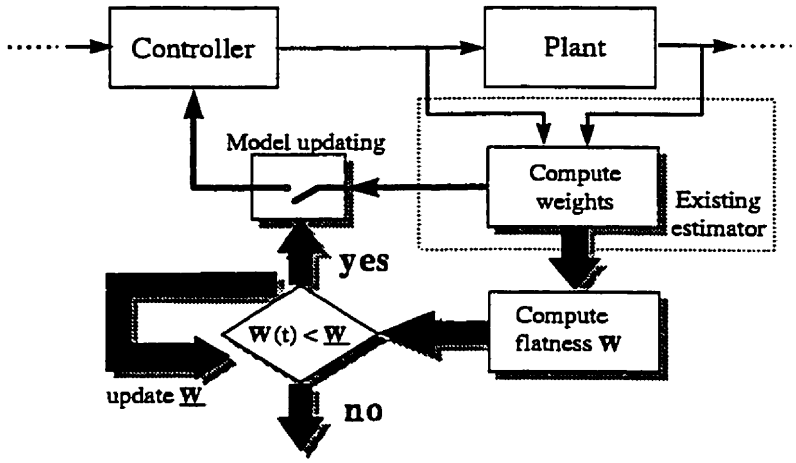

FIGURE 4.2. Additional logic to lock the model parameters.

closed-loop system carries out a number of control assignments, setpoint changes, rejection of disturbance of varying signal-to-noise ratios, $\mathbf{W}(t)$ will be steered towards 1 with similarly varying success. If the actual plant is LTI, then the above logic guarantees that the current model was updated when the signal-to-noise ratio was richest in information. The closed-loop system hence becomes "piecewise-LTI", i.e. it is invariant between two instants when the information supplied to the estimator was deemed rich enough to trigger an update of the model. In between these instants, control tasks carrying poor information are then treated with a model obtained under better conditions.

The above logic however requires absolutely that the plant be LTI since it causes $\underline{\mathbf{W}}(t)$ to be a monotonic decreasing function and if the plant dynamics change, the model may be locked to an innapropriate value with no possibility for the new values of $\mathbf{W}(t)$ to trigger an update of the model.

We may remove this problem by, at every step, allow the value of $\underline{\mathbf{W}}(t)$ to "leak" towards the case where any previous information-rich data is simply ignored, i.e. $\sqrt{N}$.

Mathematically, we can write the update operation of $\underline{\mathbf{W}}(t)$

$$\underline{\mathbf{W}}(t) = \lambda_{leak}\, X(t) + (1 - \lambda_{leak})\sqrt{N}$$

where

$$X(t) = \begin{cases} \underline{\mathbf{W}}(t-1) & \text{if } \mathbf{W}(t) \geq \underline{\mathbf{W}}(t-1) \\ \mathbf{W}(t) & \text{if } \mathbf{W}(t) < \underline{\mathbf{W}}(t-1) \end{cases}$$

where

$$\lambda_{leak} = e^{-T/T_{leak}}$$

and where T_{leak} is a user-selected time constant. Large values of T_{leak} are chosen when the true plant dynamics change slowly while short values of T_{leak} are chosen when the true plant dynamics may vary rapidly.

In this chapter we introduced a mechanism for locking the model parameters. This mechanism is one of the two novelties added to the existing MWAC algorithm from this thesis. In the following chapter, we introduce the second addition which simultaneously deals with the uncertainty on the dominant time constant of the plant and undermodelling errors.

CHAPTER 5

Treatment of undermodelling errors

In this chapter, we introduce two alternative methods for treating the problem of imprecise knowledge of the dominant time constant and undermodelling errors. It was shown in Chapter 3 that the distance between the model and the true plant is directly related to the flatness of the weight map. A sharp peak indicates close proximity to the true plant while a soft peak does not guarantee any proximity. On the other hand, we know that a sharp peak is achieved if 1) the signal-to-noise ratio is high and 2) the true plant belongs or is close to a member of $\mathcal{F}$. The first requirement depends on the control tasks that the closed-loop system is offered. In the previous chapter, we introduced a model-locking mechanism that ensures that the current model was obtained under the best conditions encountered so far. The second requirement is generally satisfied only over a certain bandwidth. Real industrial processes are typically high-order systems while the members of $\mathcal{F}$ are simple first order plus delay systems. The two methods proposed in this chapter rely on making the estimator believe that it is reading data from a first order plus delay plant.

1. Filtering the data

We expect the estimation algorithm to be applied to processes with high-order dynamics and possibly with no true delay except for the apparent delay caused by the combined effects of a number of unaccounted-for poles and zeros. If we consider

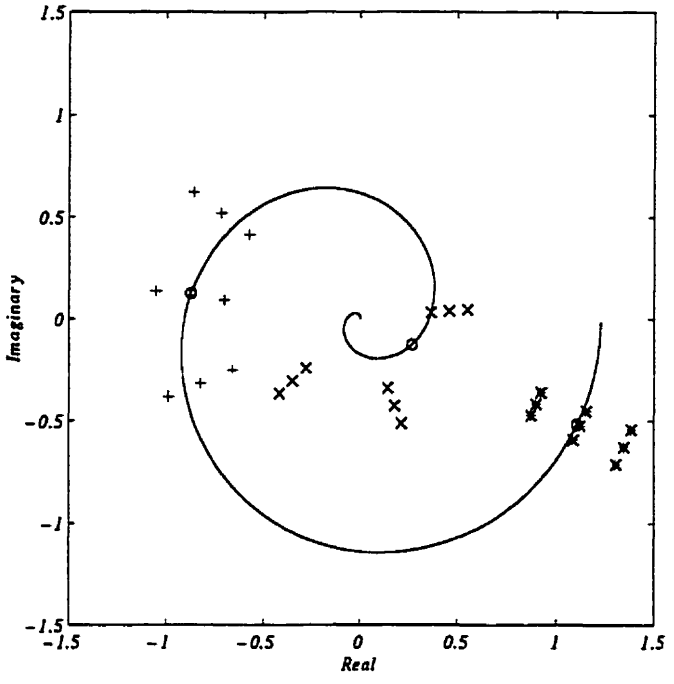

FIGURE 5.1. Nyquist plot of example plant. For three different frequencies, a set of first-order plus delay models are shown. The "*","x" and "+" indicate the FOPDs at three different frequencies. The circles are the value of P_2 at those frequencies.

a plant with such dynamics as a frequency function, the deviation from the idealized first-order-plus-delay (FOPD) plant may be interpreted as an inconsistency of its behaviour over the frequency scale. For instance the actual plant P might behave approximately as the FOPD plant $P^{(1)}$ over the frequency range $F^{(1)}$, as the FOPD plant $P^{(2)}$ over the frequency range $F^{(2)}$ and as the FOPD plant $P^{(3)}$ over $F^{(3)}$ where $F^{(1)} \neq F^{(2)} \neq F^{(3)}$ and so on. We could say that the plant dynamics are inconsistent with the FOPD model if $P^{(1)} \neq P^{(2)} \neq P^{(3)}$. This is illustrated in Figure 5.1 for the plant

$$P_2(s) = -1.23\frac{(s-1)(s-2)^2}{(s+1)^4(s+2)^2}$$

which was used in the examples of Chapter 2. In this case, the plant behaves very closely to the same FOPD model up to the cross-over frequency. This is no longer true however at high frequencies. Now recall, that the filter applied to the data so

far is

$$F(q^{-1}) = 1 - q^{-1}$$

i.e. a first-difference operator which emphasises the high frequency range where the model is inconsistent with respect to any FOPD approximation.

The first method thus consists of reducing the frequency window through which the estimator reads the data such the the true plant is consistent with some FOPD model.

1.1. Designing the filter. We seek to design a bandpass filter with a -6db pass band p_ω centered on some frequency ω_0. The filter should fulfill some basic requirements:

- The pass band should be in a frequency range within the bandwidth of the true process. At design time, we can make use of the prescribed dominant plant time constant τ to estimate the bandwidth of the plant as $\omega_b = 1/\tau$.
- The central frequency ω_0 of the filter should be not too far from the cross-over frequency of the plant since good closed-loop behaviour is typically associated to good modeling of this critical frequency area. We could specify ω_0 by letting it be some fraction r_0 $(0 < r_0 \leq 1)$ of the bandwidth of the plant, i.e. $\omega_0 = r_0\omega_b$
- The width of the pass-band should be specified by as few parameters as possible so as not burden the user with too many selectable parameters.
- It should satisfy our original requirement, i.e. $F(1) = 0$.

The continuous-time filter

$$\tilde{F}(s) = \frac{s}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{1.5.1}$$

satisfies all of the above requirements. It has a single peak

$$\|F(j\omega)\|_\infty = \frac{1}{2\zeta\omega_0}$$

which occurs at $\omega = \omega_0$. Given that ω_0 is selected from the estimated bandwidth of the plant, its -6db pass-band is a linear function of ζ only, i.e.

$$p_\omega = 2\sqrt{3}\omega_0\zeta$$

Using the bilinear transformation

$$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$$

we form the discrete time equivalent of (1.5.1)

$$F(z^{-1}) = \frac{1}{\alpha_0}\frac{1-z^{-2}}{1+\alpha_1 z^{-1}+\alpha_2 z^{-2}} \tag{1.5.2}$$

where

$$\begin{aligned}
\alpha_0 &= \frac{2}{T} + 2\zeta\omega_0 + \frac{T\omega_0^2}{2} \\
\alpha_1 &= \left(2T\omega_0^2 - \frac{4}{T}\right)/\alpha_0 \\
\alpha_2 &= \left(\frac{2}{T} - 2\zeta\omega_0 + \frac{T\omega_0^2}{2}\right)/\alpha_0
\end{aligned}$$

Example 1 revisited (First method). In the examples of Chapter 2, the adaptive closed-loop control of plant $P_2(s)$ was shown to be significantly closer to instability than the adaptive closed-loop control of plant $P_1(s)$ (which was a true FOPD plant) although the step response of these plants are arguably similar. We now repeat the simulation with all parameters being the same as in Table 2.1 with the exception of the data filter which we specify next.

The estimated dominant time constant is 2 secs (see Table 2.1). From this we evaluate the cut-off frequency to be

$$\omega_c = 1/2 \quad \text{[rad/sec]}$$

We set the central frequency of the filter to be $r_0 \times \omega_c$ where we set $r_0 = 0.5$. Hence

$$\omega_0 = 1/4 \quad \text{[rad/sec]}$$

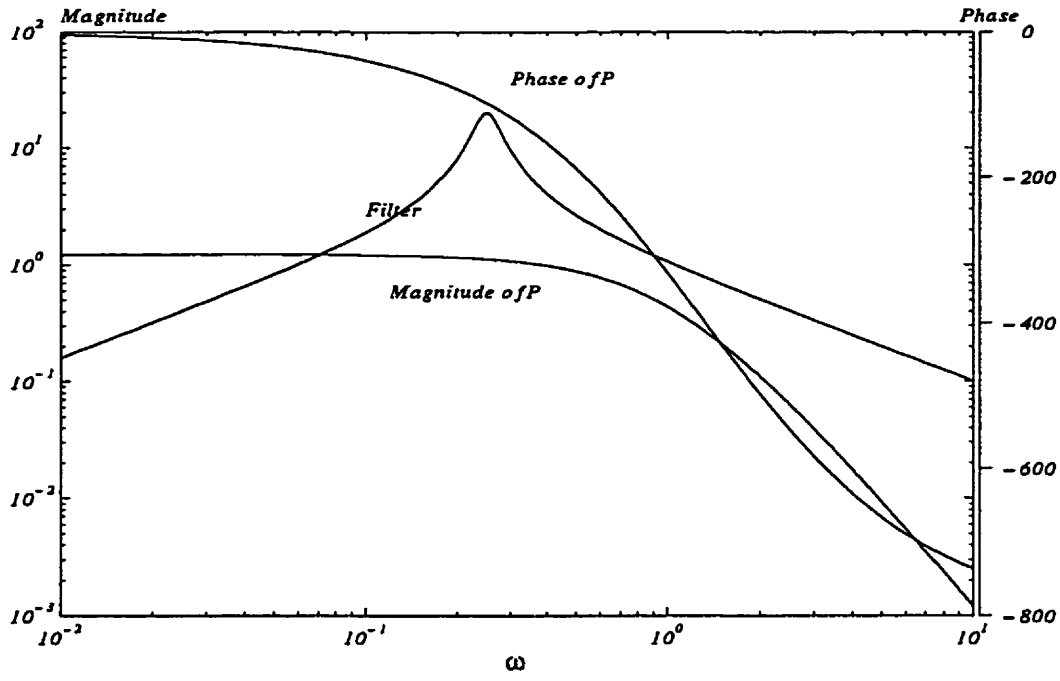

FIGURE 5.2. Bode plot of $P_2(s)$ and $|F(s)|$.

Finally, we set the damping coefficient ζ to be equal to 0.1 which corresponds to a pass-band of

$$p_\omega = 2\sqrt{3}\,\frac{1}{4}\,0.1 = 0.0866 \qquad \text{[rad/sec]}$$

Figure 5.2 shows the location of the filter pass-band with respect to the phase and gain of $P_2(s)$.

Besides the modified input/output filter, we also include in the following example, the modification introduced in Chapter 4 for locking the estimates. The simulation run is shown in Figure 5.3. The closed-loop system experiences a series of setpoint changes. As before, there is a short "learning phase" transient at the beginning but it is smoother than the original transient. Following this learning phase, the closed-loop system settles quickly to its final form as shown by the $\mathbf{W}(t)$ indicator.

The plot of the $\hat{\gamma}_j$ parameters as a function of time (Figure 5.4) shows that with the modified data filter, the estimation sub-system considerably increases its segregation of the members of $\mathcal{F}$.

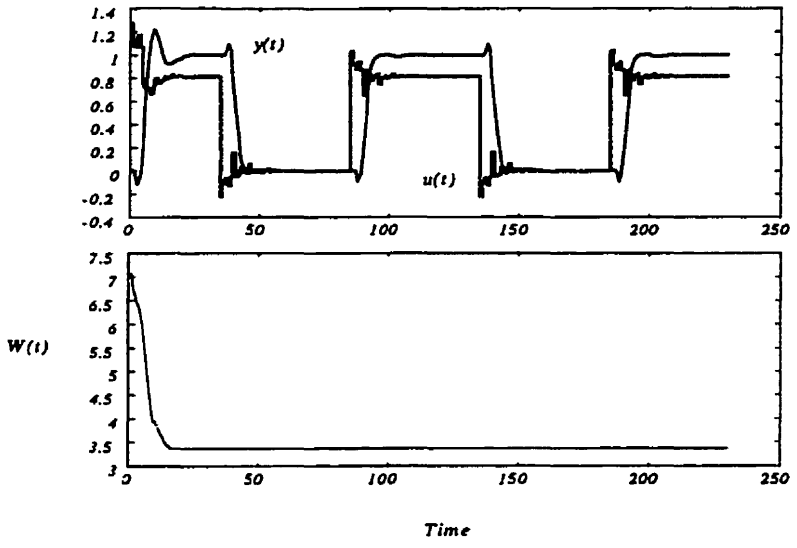

FIGURE 5.3. Top figure: Input and output signals. Bottom figure: Variable $\mathbf{W}(t)$ used for locking the parameter estimates.

2. Adjustment of the sampling period

A sampled high-order dynamic system appears to be of high order only if it is sampled quickly enough. In fact, if the sampling period is long enough, a stable, high order plant appears at the sampling instants as a pure delay with a gain. On the other hand, too long a sampling period limits the control to coarse closed-loop performance. We seek a compromise sampling period that will make the deviations from a pure first-order-plus-delay system small enough to have negligible influence on model precision but without making the control too coarse.

We readily have an online indicator $\underline{\mathbf{W}}(t)$ of the proximity of the model to the true plant. We can make use of that information to adjust the size of the sampling period until we reach a pre-specified target $\underline{\mathbf{W}}_{sp}$ just as a regular feedback loop. This loop differs from a conventional feedback loop in many ways however: since the fluctuations

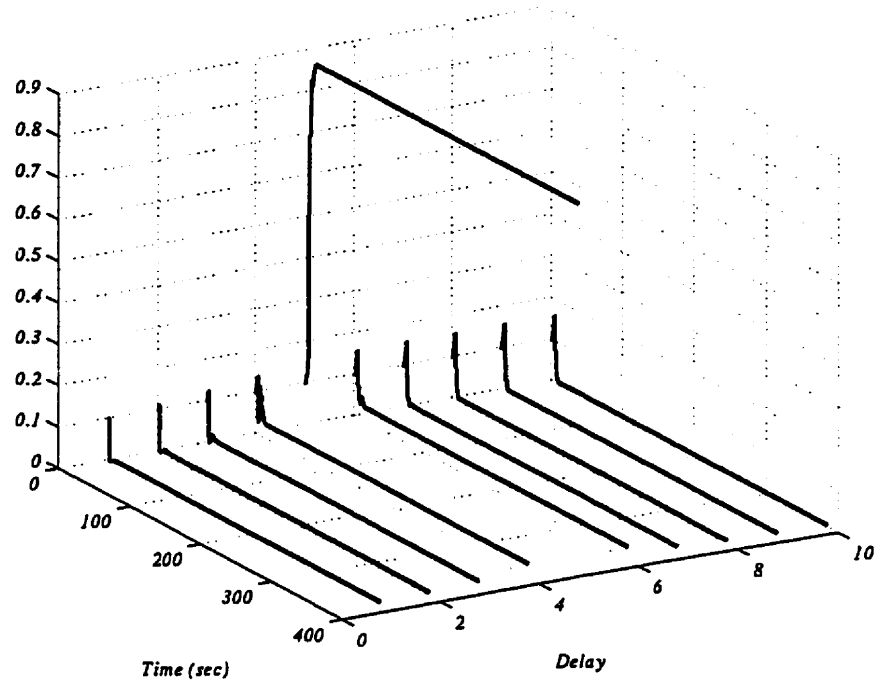

FIGURE 5.4. Parameters $\hat{\gamma}_j$ as a function of time.

of $\underline{\mathbf{W}}(t)$ are triggered by the occurence and size of the external excitations (over which we assume to have no prior knowledge), the adjustment of the sampling period should be implemented on an occasional basis when, for instance, it is detected that fresh data has caused $\underline{\mathbf{W}}(t)$ to decrease. On the other hand, if the adaptive system is not sufficiently excited for some time, $\underline{\mathbf{W}}(t)$ may remain constant over this period but when an important external excitation occurs, it produces many rapid decreases of $\underline{\mathbf{W}}(t)$. Thus to make a sensible use of $\underline{\mathbf{W}}(t)$ as a feedback variable, we need to consider the following:

- When $\underline{\mathbf{W}}(t)$ has decreased, we wait for some pre-determined amount of time T_W before implementing the adjustment of the sampling period. If $\underline{\mathbf{W}}(t)$ decreases **before** T_W has elapsed, we reset the timer and wait for T_W to elapse again. This ensures that we have secured a new, stable value of $\underline{\mathbf{W}}(t)$ before implementing the sampling period adjustment.
- The partition of the gain interval is independent of the adjustment of the sampling period but $\underline{\mathbf{W}}(t)$ is computed from the weight of every member of

$\mathcal{F}$. To make $\underline{\mathbf{W}}(t)$ independent of the gain partition, we can first compute the total weight associated to the individual delays in $[\underline{d}, \overline{d}]$, i.e.

$$\hat{w}_k = \sum_{(i,j)\in I_{\mathcal{F}_k}} \hat{w}_{ij}(t)$$

then compute the modified $\mathbf{W}(t)$, $\mathbf{W}'(t)$ as

$$\mathbf{W}'(t) = \sum_{k=\underline{d}}^{\overline{d}} \sqrt{\hat{w}_k}$$

and the modified indicator $\underline{\mathbf{W}}'(t)$

$$\underline{\mathbf{W}}'(t) = \inf_{k\in[0,t]} \mathbf{W}'(k) = \bigwedge_{k=0}^{t} \mathbf{W}'(k)$$

Say that with the above modifications implemented that the occurences of the sampling period adjustments are indexed with the superscript (n) $n = 1, \cdots,$ then the adjustment we choose is of the integral-type, i.e.:

$$T^{(n+1)} = T^{(n)} + k_T(\underline{\mathbf{W}}_{sp} - \underline{\mathbf{W}}^{(n)})$$

Example 1 revisited (Second method). We repeat the example of Chapter 2 but applying the online adjustment of the sampling period. Apart from this modification and the modification for locking the parameter estimates (see Chapter 4) , the simulation data is the same as that of Chapter 2. We need two additional parameter values for T_W and k_T. In the following simulation, these were set to be

$$T_W = 20 \text{ [secs]} \qquad k_T = 1$$

During this simulation, we let the adaptive system experience a series of exogenous excitations in the form of setpoint changes. The continuous-time plots of the input/output signals are shown in Figure 5.5.

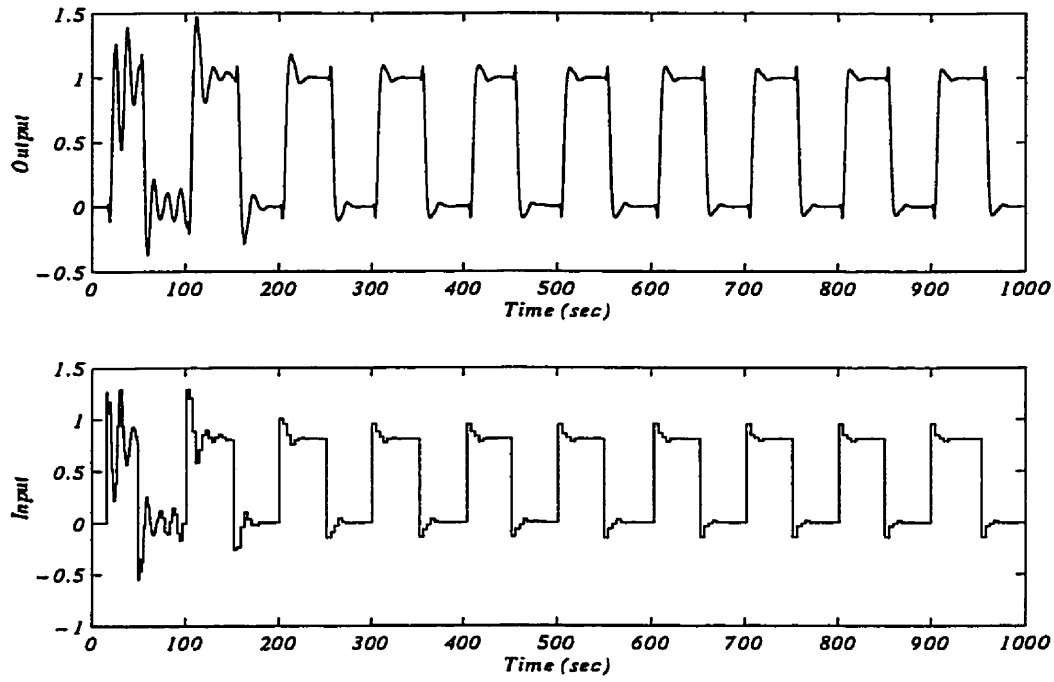

FIGURE 5.5. Continuous-time plots of the input/output signals. Adaptive control of the plant P_2 with sampling period adjustments.

The initial performance of the feedback system is similar to the original behaviour of Chapter 2 but after a few setpoint changes, the control performance is much improved. Figure 5.6) shows a blown-up version of the simulation run for the initial time segment and the final time segment to emphasize the changes made to the sampling period. We observe that in the final segment, the minimum-phase behaviour of the plant is invisible at the sampling instants.

Finally, Figure 5.7 shows the variables of the sampling period adjustment feedback system. The controlled variable $\underline{\mathbf{W}}'(t)$ asymptotically reaches its target value 1.2 while the sampling period increases from its initial value of 1 sec to a value just over 4 secs by the end of the simulation run.

3. Merits and inconveniences of the proposed methods

In this chapter, we presented two methods to render the unmodeled deviations from the ideal first-order-plus-delay representation less visible to the estimation subsystem. The first method uses a passband filter whose bandwidth may be reduced

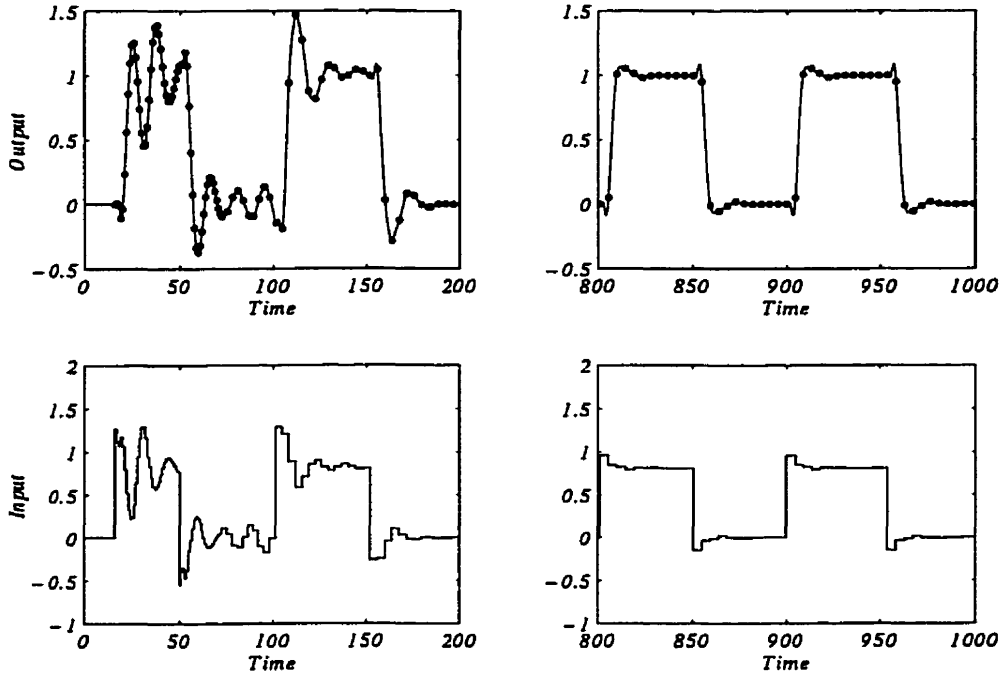

FIGURE 5.6. Blow-up of the initial ($0 \leq t \leq 200$) and final ($800 \leq t \leq 1000$) time segments.

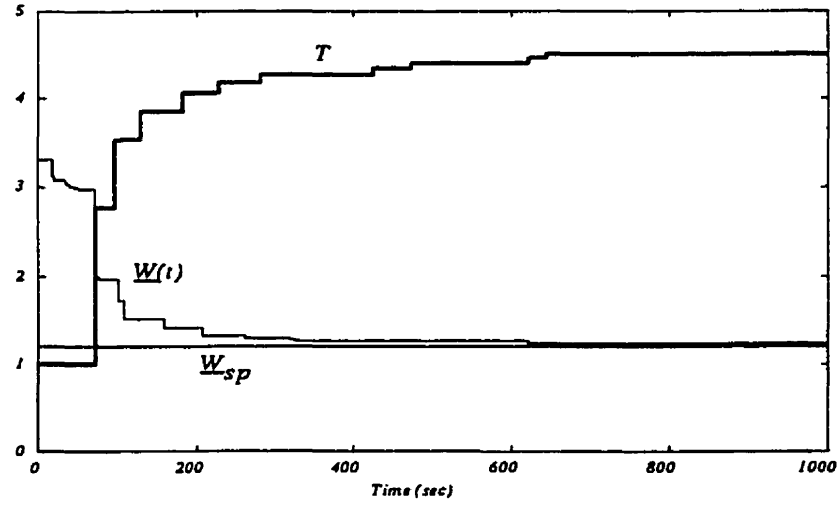

FIGURE 5.7. Feedback adjustment of the sampling period during the simulation.

by the user when it is felt that undermodeling errors may be important. The second method progressively increases the sampling period until the effects of the undermodeling errors are reduced to manageable proportions.

The first method has the advantage of simplicity since it involves no extra feedback loop and since most parameters may be determined from the original data supplied by the user. The second method adds an extra feedback loop and requires the determination of new tuning constants but is likely to be less sensitive to errors induced by mis-estimating the dominant time constant of the plant.

It is difficult at this point to make a firm statement on which particular method is best for an industrial implementation. Simulation under realistic conditions or industrial experience may provide some answers but we leave this question open for the time being.

CHAPTER 6

Conclusions

Pilot-plant and industrial applications of the MWAC technique have revealed that the algorithm can detect very rapidly what model in the set is most appropriate for the current circumstances when a control task such as a setpoint change is requested from the closed loop system. This observation is consistent with the theory developed in Chapter 3. Theorem 3.3 says that in a finite time following the occurence of an external excitation, the model used to compute the control finds itself in a neigborhood of the true plant. The size of this neighborhood depends on two factors: 1) the resolution of the partition of $\mathcal{F}$ and 2) the signal-to-noise ratio. The first factor promotes the sharpness of noise-free, base line weight distribution. The sharper this distribution is, the tighter the neigborhood to the true plant is. Moreover, the sharper the distribution, the more resistance it offers to the uniformizing effects of disturbances (Theorem 4.2). On the other hand, a high signal-to-noise ratio also defeats the uniformizing effect of those disturbances which are external to the plant (i.e. input disturbances, measurement errors). Finally, Theorem 4.3 says that the sharpness of the distribution will peak in a time approximately equal to the apparent time delay of the true plant. The bounds computed in Theorems 3.3, 4.2 and 4.3 allows the designer to estimate an adequate partitioning resolution for a given application.

Furthermore, this jumping mode of convergence is a feature of MWAC (and related algorithms - see Introduction) that significantly differs from adaptive methods

based on recursive least-squares methods or gradient-like methods ([5], [21]). The latter are essentially integrators which accumulate the input/output information and will assymptotically make the effect of the disturbances vanish. This difference in convergence mode might serve as a basis for selecting an adaptive control method for a given application. For instance, the choice would depend on how the information is generated: by sudden, infrequent bursts or on a continuous basis.

In Chapter 3, we have limited the form of external excitation to that of step changes on the setpoint. It might be argued that this limits the general applicability of the results of Chapter 3. However, the tracking of setpoint step changes (along with rejection of step disturbances) is one of the most important industrial control tasks and one by which the performance of controllers is traditionaly assessed. Hence its relevance. On the other hand, it is straightforward to replace the step functions used in Chapter 3 by more general signals as long as they preserve the initial LLI behaviour of the controller. We leave this task for future work.

1. Future research

As mentioned earlier, a high signal-to-noise ratio alleviates in part the effect of input or output disturbances. It does nothing however for plant disturbances that stem from approximating a (possibly) high-order plant with a first order plus delay model. We proposed two methods in Chapter 5 to circumvent this problem: 1) by filtering the data with a bandpass filter whose bandwidth depends on the importance of the undermodelling effects and 2) by progressively adjusting the sampling period so that a compromise is achieved between satisfying the first order plus delay requirement and the achievable closed loop speed of response. This was combined with a scheme for locking the parameter estimates when it is detected that these estimates have been obtained under good conditions (Chapter 4). These additions improve the behaviour of the MWAC algorithm but they also open the door to further investigation and experimentation. For example, the method for locking the estimates introduced in Chapter 4 effectively breaks up the time scale in a countably infinite number of

segments for which the plant dynamics may be assumed to be piecewise or locally linear, time-invariant. This view could give way to a general definition of a class of time-varying plant dynamics.

Along these lines, it would also be desirable to formally redo the analysis with random disturbances using the wealth of existing techniques for analyzing such signals. The present work really was concerned by the finite time approximation of the true plant. In a more general, long term convergence analysis, the asymptotic properties of the statistics of random signals make them an asset to the analysis.

In the present work (and all previous) on MWAC, the plant has been limited to a single input/single output process with stable poles. The case when the plant has a pole at $z = 1$ or outside the unit circle should now be investigated.

Another natural extension of this work also is to examine its implementation on multi input/multi output (MIMO) plants. The application of adaptive techniques to MIMO plants is always a delicate affair just from the point of view of identifiability. We believe however that MWAC offers some advantages over other techniques. For instance, it is reasonable to think that for a typical MIMO system, not all input/output relations require the same degree of adaptation. By increasing or decreasing the range of uncertainty of the parameters, the MWAC controller automatically modulates the amount of adaptation applied to each individual relation in the system.

There is no fundamental reason to believe also that the algorithm should behave differently when the controller structure is different from the simple pole-placement algorithm used in this work. This, however, should be verified.

APPENDIX A

Supplemental proofs

This appendix contains proofs of intermediate results found in the main text. These are:

- Proposition 1.1
- Theorem 3.1
- Theorem 4.1

Proposition 1.1 *Let $Q(z^{-1}) = Q_1(z^{-1})\ Q_2(z^{-1})$ where Q, Q_1 and Q_2 admit a Laurent series expansion. Then we have*

(i)

$$|[Q]_t|_{\infty,\lambda} \leq |[Q_1]_{t_1}|_\infty |[Q_2]_{t_2}|_\infty \mathbf{B}(t,\lambda)$$

(ii)

$$|[Q]_t|_\infty \leq |[Q_1]_{t_1}|_{\infty,\lambda} |[Q_2]_{t_2}|_{\infty,\lambda} \mathbf{B}(t,\lambda^{-1})$$

where $t_1 \geq t$, $t_2 \geq t$ and $\mathbf{B}(t,\lambda)$ is the fixed function

$$\mathbf{B}(t,\lambda) = \frac{1}{2\pi} \int_0^{2\pi} \left| \frac{1-(e^{j\theta}/\lambda)^{t+1}}{1-e^{j\theta}/\lambda} \right| d\theta$$

PROOF. Obviously we have

$$[Q]_t = [[Q_1]_{t_1}\ [Q_2]_{t_2}]_t$$

where $t_1 \geq t$ and $t_2 \geq t$. It follows that

$$|[Q]_t| = \left| \sum_{k=0}^{t} q(k) \sqrt{\lambda}^{-k} e^{-j\omega k} \right|$$

where

$$q(k) = \frac{1}{2\pi} \int_0^{2\pi} [Q_1]_{t_1} [Q_2]_{t_2} e^{j\theta k} d\theta \tag{0.A.1}$$

Hence

$$\begin{aligned} |[Q]_t| &= \left| \sum_{k=0}^{t} \frac{1}{2\pi} \int_0^{2\pi} [Q_1]_{t_1} [Q_2]_{t_2} e^{j\theta k} d\theta\ \sqrt{\lambda}^{-k} e^{-j\omega k} \right| \\ &= \left| \frac{1}{2\pi} \int_0^{2\pi} [Q_1]_{t_1} [Q_2]_{t_2} \sum_{k=0}^{t} \left(\lambda^{-1/2} e^{j(\theta-\omega)} \right)^k d\theta \right| \\ &\leq |[Q_1]_{t_1}|_\infty\ |[Q_2]_{t_2}|_\infty \frac{1}{2\pi} \int_0^{2\pi} \left| \frac{1 - \left(\lambda^{-1/2} e^{j(\theta-\omega)} \right)^{t+1}}{1 - \left(\lambda^{-1/2} e^{j(\theta-\omega)} \right)} \right| d\theta \end{aligned}$$

The above integral computes the average value of its integrand over the unit circle. Since the integrand is periodic and ω only shifts the integrated function, the average is independent of ω and result **1)** follows. If one evaluates $q(k)$ over a circle of radius $\lambda \neq 1$ instead of (0.A.1), then result **2)** is proved using an argument which parallels that of result **1)** *mutatis mutandis.* □

Before proving Theorem 3.1, we need preliminary results provided by the following lemmas, the first of which is just a version of Parseval's theorem slightly modified for truncated series.

LEMMA 0.1. *If*

$$[Y(z^{-1})]_t = \sum_{k=0}^{t} y(k) z^{-k}$$

then

$$\|y(t)\|_{2,\lambda}^2 = \frac{\lambda^t}{2\pi}\int_0^{2\pi} \left|[Y(z^{-1})]_t\right|^2_{z=\sqrt{\lambda}e^{j\omega}}\, d\omega$$

PROOF. We evaluate $[Y]_t$ on the circle $z = \sqrt{\lambda}e^{j\omega}$, i.e.

$$[Y(z^{-1})]_t = \sum_{k=0} y(k)\sqrt{\lambda}^{-k}e^{-j\omega k}$$

and its conjugate

$$[Y(z^{-1})]_t^* = \sum_{k=0} y(k)\sqrt{\lambda}^{-k}e^{j\omega k}$$

It follows that

$$\left|[Y(z^{-1})]_t\right|^2 = [Y(z^{-1})]_t\,[Y(z^{-1})]_t^* = \sum_{k=0}^{t}\sum_{l=0}^{t} y(k)y(l)\sqrt{\lambda}^{-(k+l)}e^{-j\omega(k-l)}$$

Integrating the above over $[0, 2\pi]$ and multiplying by $\lambda^t/2\pi$ yields the desired sum.

$$\|y(t)\|_{2,\lambda}^2 = \frac{\lambda^t}{2\pi}\int_0^{2\pi} \left|[Y(z^{-1})]_t\right|^2_{z=\sqrt{\lambda}e^{j\omega}}\, d\omega$$

□

We now express the 2-norm of y when y is the output of a linear system P driven by an input u.

LEMMA 0.2. *If*

$$Y(z^{-1}) = P(z^{-1})U(z^{-1})$$

then **a)**

$$\|y(t)\|_{2,\lambda}^2 \le |[P]_t|_{2,\lambda}^2\ \ |[U]_t|_{\infty,\lambda}^2$$

and **b)**

$$\|y(t)\|_{2,\lambda}^2 \le |[P]_t|_{\infty,\lambda}^2\ \ |[U]_t|_{2,\lambda}^2$$

PROOF. From Lemma 0.1, we have

$$\|y(t)\|_{2,\lambda}^2 = \frac{\lambda^t}{2\pi}\int_0^{2\pi} \left|[Y(z^{-1})]_t\right|^2_{z=\sqrt{\lambda}e^{j\omega}} d\omega \leq \frac{\lambda^t}{2\pi}\int_0^{2\pi} \left|[P(z^{-1})]_t\right|^2 \left|[U(z^{-1})]_t\right|^2 d\omega$$

Applying Hölder's inequality ([40], p.63 with $p = 1$ and $q = \infty$) to the above yields both results. □

We are now ready to prove Theorem 3.1.

Theorem 3.1 *Provided that Assumptions 1.1-2.1 are satisfied then, for any t there exist coefficients K_X,K_F and K_n such that*

$$\|e_{ij}(t)\|_{2,\lambda} = \left|[(G - G_{ij})]_t\right|_{\infty,\lambda} K_F\, K_X\, + K_n$$

where $G = PF_u$ is the LLI loop gain, $G_{ij} = P_{ij}F_u$ and where K_F, K_X and K_n are common to all error signals and are bounded by

$$\begin{aligned}
\frac{1}{1+\left|[G]_t\right|_{\infty,\lambda}} \leq\ & K_F\ \leq \frac{1}{1-\left|[G]_t\right|_{\infty,\lambda}} \\
-\delta_n\sqrt{\frac{\lambda^{t+1}-(r_n^2)^{t+1}}{\lambda - r_n^2}} \leq\ & K_n\ \leq \delta_n\sqrt{\frac{\lambda^{t+1}-(r_n^2)^{t+1}}{\lambda - r_n^2}} \\
|y_s|\lambda^t - \delta_n\sqrt{\frac{\lambda^{t+1}-(r_n^2)^{t+1}}{\lambda - r_n^2}} \leq\ & K_X\ \leq |y_s|\lambda^t + \delta_n\sqrt{\frac{\lambda^{t+1}-(r_n^2)^{t+1}}{\lambda - r_n^2}}
\end{aligned}$$

PROOF. Consider the equation

$$e_{ij}(t) = (G - G_{ij})\cdot(y_s^F(t) - n^F(t))\cdot\frac{1}{1+G} + n^F(t) \tag{0.A.2}$$

where $y_s^F(t)$ is an impulse of magnitude y_s and $n^F(t)$ is a noise signal satisfying Assumption 1.3. If $X(z^{-1})$ is the z-transform of $y_s^F(t) - n^F(t)$ then from Lemma 0.2 b), we have

$$\|(G - G_{ij})(y_s^F(t) - n^F(t))\|_{2,\lambda} \leq \left|[G - G_{ij}]_t\right|_{\infty,\lambda}\cdot\left|[X]_t\right|_{2,\lambda}$$

and

$$[X]_t = \sum_{k=0}^{t} (y_s^F(k) - n^F(k))\sqrt{\lambda}^{-k} e^{-j\omega k}$$

which, using the assumptions, leads to

$$y_s\lambda^t - \|n^F(t)\|_{2,\lambda} \leq |[X]_t|_{2,\lambda} \leq y_s\lambda^t + \|n^F(t)\|_{2,\lambda}$$

The 2-norm of n^F is bounded by

$$\begin{aligned}
\sum_{k=0}^{t} \left(n^F(k)\right)^2 \lambda^{t-k} &\leq \delta_n^2 \sum_{k=0}^{t} \left(r_n^2\right)^k \lambda^{t-k} \\
&= \delta_n^2 \lambda^t \sum_{k=0}^{t} \left(\frac{r_n^2}{\lambda}\right)^k \\
&= \delta_n^2 \frac{\lambda^{t+1} - (r_n^2)^{t+1}}{\lambda - r_n^2}
\end{aligned}$$

which establishes the bounds for K_X and K_n, the additive noise term.

The bound on K_F is obtained by noting that $(1+G)^{-1}$ is obtained by having G within a unity-feedback loop and then using the Small gain Theorem. □

From Lemma 0.2 we see that there is a companion theorem to Theorem 3.1.

Theorem 4.1 *Provided that Assumptions 1.1-2.1 are satisfied then, for any t there exist coefficients k_X,K_F and K_n such that*

$$\|e_{ij}(t)\|_{2,\lambda} = |[(G - G_{ij})]_t|_{2,\lambda}\, K_F\, k_X \; + K_n$$

where K_F and K_n are bounded as in Theorem 3.1 and k_X is bounded by

$$|y_s| - \delta_n \frac{1 - (r_n/\sqrt{\lambda})^{t+1}}{1 - r_n/\sqrt{\lambda}} \leq k_X \leq |y_s| + \delta_n \frac{1 - (r_n/\sqrt{\lambda})^{t+1}}{1 - r_n/\sqrt{\lambda}}$$

PROOF. The proof is exactly similar to that of Theorem 3.1 except that we use result a) of Lemma 0.2 instead of result b). □

APPENDIX B

Pseudo computer code for implementing μ-modification

In this chapter, we give a pseudo computer code listing for implementing the limit function of Equation 4.2.2.

```
'
'     This pseudo code implements the mu modification
'     The subroutine expects a vector s() of dimension N to be
'     the vector to which the limit must be applied.
'     The subroutine returns the modified values of s() in the same
'     vector.
'     The vectors s_modified() and previous_s() are local, intermediate
'     variables.

mu = 1.0-E-10

for j=1 to N
        if s(j) < threshold then
                mu_flag(j) = TRUE  'Check those values which must be
        else                       'replaced by mu
                mu_flag(j) = FALSE
        end if
next j

epsilon = 1.0E-20;
Max=-1.0E+30
```

APPENDIX B. PSEUDO COMPUTER CODE FOR IMPLEMENTING μ-MODIFICATION

```
while (convergence > epsilon)
        for j=i to N
                if mu_flag(j) = TRUE        ' Modify only those below
                        s_modified(j)=mu    ' the threshold
                else
                        s_modified(j)=s(j)+mu
                end if
                sum=sum+s_modified(j)
        next j
        for j=i to N

                s_modified(j)=s_modified(j)/sum

                buffer=abs(s_modified(j)-previous_s(j)) ' Compare with previous
                if (buffer>Max) then Max=buffer         ' Look for maximum variation
                previous_s(j) = s_modified(j)           ' Store for next iteration
        next j
        convergence = Max        ' Convergence test is based on maximum variation over vector s
        mu=mu/10                 ' Reduce mu

end while
for j=i to N
        s(j)=s_modified(j)       '  Store back modified vector s()
next j
```

APPENDIX C

Detailed calculations for Theorem 4.3

This appendix gives the details for Equations 4.3.4, 4.3.5 and 4.3.7 which are subsequently used in the proof of Theorem 4.3. The following calculations compute the norm of the difference between transfer functions of the form

$$G_0^* = \left(\frac{g_0^*}{\tilde{\gamma}}\right) \frac{1-\beta}{1-\beta z^{-1}} z^{-d-1}$$

and

$$G_{ij} = \left(\frac{g_i}{\tilde{\gamma}}\right) \frac{1-\beta}{1-\beta z^{-1}} z^{-d_j-1}$$

The impulse response coefficients of the above transfer functions are given by

$$h_0^*(k) = \begin{cases} 0 & \text{for } k \leq d \\ g_0 * \frac{1-\beta}{\tilde{\gamma}} \beta^{k-d-1} & \text{for } k > d \end{cases}$$

$$h_{ij}(k) = \begin{cases} 0 & \text{for } k \leq d_j \\ g_i \frac{1-\beta}{\tilde{\gamma}} \beta^{k-d_j-1} & \text{for } k > d_j \end{cases}$$

respectively.

For $P_{ij} \in \underline{\mathcal{F}}$

$$|[G_0^* - G_{ij}]_t|_{2,\lambda} = \underbrace{\sum_{k=0} d\lambda^{t-k} h_{ij}^2(k)}_{\text{PART A}} + \underbrace{\sum_{k=d+1}^{t} \lambda^{t-k}(h_0^*(k) - h_{ij}(k))^2}_{\text{PART B}}$$

PART A.

$$\begin{aligned}
\sum_{k=d_j+1}^{d} \lambda^{t-k} h_{ij}^2(k) &= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 g_i^2 \sum_{k=d_j+1}^{d} \lambda^{t-k}(\beta^2)^{k-d_j-1} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 g_i^2 \lambda^{t-d} \sum_{k=d_j+1}^{d} \lambda^{d-k}(\beta^2)^{k-d_j-1} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 g_i^2 \lambda^{t-d} \sum_{k=0}^{d-d_j-1} \lambda^{d-d_j-1-k}(\beta^2)^{k} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 g_i^2 \lambda^{t-d} \frac{\lambda^{d-d_j} - (\beta^2)^{d-d_j}}{\lambda - (\beta)^2}
\end{aligned}$$

PART B.

$$\begin{aligned}
\sum_{k=d+1}^{d} \lambda^{t-k}(h_0^*(k) - h_{ij}(k))^2 &= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 \sum_{k=d+1}^{t} \lambda^{t-k}(g_0^*\beta^{k-d-1} - g_i\beta^{k-d_j-1})^2 \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i\beta^{d-d_j})^2 \sum_{k=d+1}^{t} \lambda^{t-k}(\beta^2)^{k-d-1} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i\beta^{d-d_j})^2 \sum_{k=0}^{t-d-1} \lambda^{t-d-1-k}(\beta^2)^{k} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i\beta^{d-d_j})^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2}
\end{aligned}$$

For $P_{ij} \in \mathcal{F}_d$

$$\begin{aligned}
|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 &= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i)^2 \sum_{k=d+1}^{t} \lambda^{t-k}(\beta^2)^{k-d-1} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i)^2 \sum_{k=0}^{t-d-1} \lambda^{t-d-1-k}(\beta^2)^{k} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^* - g_i)^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2}
\end{aligned}$$

For $P_{ij} \in \overline{\mathcal{F}}$

There are two cases to consider in this section: $t \leq d_j$ and $t > d_j$.

$t \leq d_j$.

$$\begin{aligned}
|[G_0^* - G_{ij}]_t|_{2,\lambda}^2 &= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \sum_{k=d+1}^{t} \lambda^{t-k}(\beta^2)^{k-d-1} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \sum_{k=0}^{t-d-1} \lambda^{t-d-1-k}(\beta^2)^{k} \\
&= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \frac{\lambda^{t-d} - (\beta^2)^{t-d}}{\lambda - \beta^2}
\end{aligned}$$

$t > d_j$.

$$\begin{aligned}
|[G_0^* - G_{ij}]_2|_{2,\lambda}^2 = & \underbrace{\left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \sum_{k=d+1}^{d_j} \lambda^{t-k}(\beta^2)^{k-d-1}}_{\text{PART A}} \\
& \underbrace{+ \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 \sum_{k=d_j+1}^{t} \lambda^{t-k}(g_0^*\beta^{k-d-1} - g_i\beta^{k-d_j-1})^2}_{\text{PART B}}
\end{aligned}$$

PART A.

$$\left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \sum_{k=d+1}^{d_j} \lambda^{t-k}(\beta^2)^{k-d-1} = \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \lambda^{t-d_j} \sum_{k=0}^{d_j-d-1} \lambda^{d_j-d-1-k}(\beta^2)^k$$

$$= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*)^2 \lambda^{t-d_j} \frac{\lambda^{d_j-d} - (\beta^2)^{d_j-d}}{\lambda - \beta^2}$$

PART B.

$$\left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 \sum_{k=d_j+1}^{t} \lambda^{t-k} \quad (g_0^*\beta^{k-d-1} - g_i\beta^{k-d_j-1})^2 =$$

$$\left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*\beta^{d_j-d} - g_i)^2 \sum_{k=d_j+1}^{t} \lambda^{t-k}(\beta^2)^{k-d_j-1}$$

$$= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*\beta^{d_j-d} - g_i)^2 \sum_{k=0}^{t} \lambda^{t-d_j-1-k}(\beta^2)^k$$

$$= \left(\frac{1-\beta}{\tilde{\gamma}}\right)^2 (g_0^*\beta^{d_j-d} - g_i)^2 \frac{\lambda^{t-d_j} - (\beta^2)^{t-d_j}}{\lambda - \beta^2}$$

REFERENCES

[1] K. E. Arzen, *An architecture for expert system based feedback control*, Automatica **25** (1989), no. 6, 813–827.

[2] K. J. Astrom, *Adaptive control around 1960*, IEEE Control Systems magazine **16** (1996), no. 3, 44–49.

[3] K. J. Astrom, J. J. Anton, and K. E. Arzen, *Expert control*, Automatica **22** (1986), no. 3, 277–286.

[4] K. J. Astrom and B. Wittenmark, *Computer controlled systems - theory and design*, 2 ed., Prentice-Hall, Englewood Cliffs, New Jersey, 1990.

[5] K.J. Astrom and B. Wittenmark, *Adaptive control*, Addison-Wesley, Reading, Mass., 1989.

[6] M. Athans, D. Castanon, K.-P. Dunn, C. S. Greene, W. H. Lee, N. R. Sandell Jr., and A. S. Willsky, *The stochastic control of the f-8c aircraft using a multiple model adaptive control (MMAC) method- part i: Equilibrium flight*, IEEE Transactions on Automatic Control **AC-22** (1977), no. 5, 768–780.

[7] Y. Bar-Shalom and E. Tse, *Dual effect, certainty equivalence and separation in stochastic control*, IEEE Transactions on automatic control **AC-19** (1974), 494–500.

[8] P. R. Bélanger, *Control engineering: A modern approach*, Saunders College Publishing, 1995.

[9] B. W. Bequette, H. Kaufman, R. J. Roy, and C. Yu, *Recent advances in the control of drug delivery systems*, AIChE 1992 Annual Meeting (345 East 47th Street, New York,NY 10017), AIChE, AIChE, November 1992.

[10] R. R. Bitmead, *Persistence of excitation conditions and the convergence of adaptive schemes*, IEEE Transactions on information theory **IT-30** (1984), no. 2, 183–191.

[11] S. Boyd and S. S. Sastry, *Necessary and sufficient conditions for parameter convergence in adaptive control*, Automatica **22** (1986), 629–639.

[12] P. E. Caines, *Linear stochastic systems*, John Wiley and sons, 1988.

[13] P. E. Caines and H. F. Chen, *Optimal adaptive LQG control for systems with finite state process parameters*, IEEE Transactions on automatic control **AC-30** (1985), no. 2, 185–189.

[14] P. E. Caines and J.-F. Zhang, *Adaptive control for jump parameter systems via non-linear filtering*, Preprints of the 31st IEEE CDC conference, IEEE, IEEE, December 1992.

[15] M. Champagne, *Personal communication*, 1996.

[16] S. Dasgupta and L.C. Westphal, *Convergence of partitioned adaptive filters for systems with unknown biases*, IEEE Transactions on automatic control **AC-28** (1983), no. 5, 614–615.

[17] J. R. Elliott, *NASA's advanced control law program for the f-8 digital fly-by-wire aircraft*, IEEE Transactions on Automatic Control **AC-22** (1977), no. 5, 753–757.

[18] S. Gendron, A. P. Holko, and M. Amjad, *Simple adaptive digital dead-time compensators for low-order SISO processes*, Ninth IFAC symposium on system identification and parameter estimation, IFAC, IFAC, 1991, pp. 1012–1024.

[19] S. Gendron, M. Perrier, J. Barrette, M. Amjad, A. Holko, and N. Legault, *Deterministic adaptive control of SISO processes using model weighting adaptation*, International journal of control **58** (1993), no. 5, 1105–1123.

[20] G. C. Goodwin, *Can we identify adaptive control?*, Proceedings of the 1991 ECC, ECC, July 1991, pp. 1714–1725.

[21] G. C. Goodwin and K. S. Sin, *Adaptive filtering, prediction and control*, Prentice-Hall Inc, 1984.

[22] C. S. Greene and A. S. Willsky, *An analysis of the multiple model adaptive control algorithm*, Proceedings of the 19th CDC conference, IEEE, IEEE, December 1980, pp. 1142–1145.

[23] R. M. Hawkes and J. B. Moore, *Performance bounds for adaptive estimation*, Proceedings of IEEE **64** (1976), no. 8, 1143–1150.

[24] R. Kuhne, *Personal communication*, 1993.

[25] D. G. Lainiotis, *Partitioning: A unifying framework for adaptive systems, i: Estimation*, Proceedings of the IEEE **64** (1976), no. 8, 1126–1143.

[26] ———, *Partitioning: A unifying framework for adaptive systems, ii: Control*, Proceedings of the IEEE **64** (1976), no. 8, 1182–1198.

[27] L. Lin, *Fast identification for robust adaptive control - a metric complexity approach*, Ph.D. thesis, McGill University, 1993.

[28] L. Ljung, *System identification: Theory for the user*, Information and system sciences, Prentice-Hall Inc, Englewood Cliffs, New Jersey, 1987.

[29] L. Ljung and T. Soderstrom, *Theory and practice of recursive identification*, The MIT Press, 1983.

[30] D. T. Magill, *Adaptive optimal estimation of sampled stochastic processes*, IEEE Transactions on Automatic Control **AC-10** (1965), no. 4, 434–439.

[31] P. S. Maybeck and D. L. Pogoda, *Multiple-model adaptive controller for the STOL f-15 with sensor/actuator failures*, Proceedings of the 28th CDC conference, IEEE, IEEE, December 1989, pp. 1566–1572.

[32] R. L. Moose, H. F. VanLandingham, and P. E. Zwicke, *Digital set point control of nonlinear stochastic systems*, IEEE Transactions on Industrial Electronics and Control Instrumentation **IECI-25** (1978), no. 1, 39–45.

[33] M. Morari and E. Zafiriou, *Robust process control*, Prentice Hall, 1989.

[34] R. Murray-Smith and T. A. Johansen, *Multiple model approaches to modelling and control*, Taylor and Francis, 1997.

[35] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems*, Prentice-Hall Inc, 1989.

[36] K. Nassiri-Toussi and W. Ren, *On the convergence of least squares estimates in white noise*, IEEE Transactions on automatic control **39** (1994), no. 2, 364–368.

[37] J. Owen, *Performance optimization of highly uncertain systems in H infinity*, Ph.D. thesis, McGill University, 1992.

[38] M. Pudlas, *Personal communication*, 1997.

[39] L. Rade and B. Westergren, *Beta mathematics handbook*, CRC Press, 1990.

[40] W. Rudin, *Real and complex analysis*, McGraw-Hill Inc., 1987.

[41] S. L. Shah and W. R. Cluett, *Recursive least squares based estimation schemes for self-tuning control*, Canadian journal of chemical engineering **69** (1991), no. 1, 89–96.

[42] G. Stein, G. L. Hartmann, and R. C. Hendrick, *Adaptive control laws for F-8 flight test*, IEEE Transactions on Automatic control **AC-22** (1977), no. 5, 758–762.

[43] J. Sternby, *On consistency of the method of least squares using martingale theory*, IEEE Transactions on automatic control **AC-22** (1977), no. 3, 346–352.

[44] J. K. Tugnait, *Convergence analysis of partitioned adaptive estimators under continuous parameter uncertainty*, IEEE Transactions on automatic control **AC-25** (1980), no. 3, 569–572.

[45] K. Watanabe and S. G. Tzafestas, *Multiple-model adaptive control for jump-linear stochastic systems*, International journal of control **50** (1989), no. 5, 1603–1617.

[46] W. M. Wonham, *Some applications of stochastic differential equations to optimal non-linear filtering*, SIAM Journal of Control and optimization **2** (1965), no. 3, 347–369.

[47] G. Zames, *Feedback and optimal sensitivity: model reference transformations, multiplicative seminorms and approximated inverses*, IEEE Transactions on Automatic control **AC-26** (1981), no. 2, 301–320.

[48] G. Zames and L. Y. Wang, *What is an adaptive learning system?*, Proceedings of the CDC Conference, IEEE, December 1990, pp. 2861–2864.