CWI Tracts

Managing Editors

J.W. de Bakker (CWI, Amsterdam)
M. Hazewinkel (CWI, Amsterdam)
J.K. Lenstra (CWI, Amsterdam)

Editorial Board

W. Albers (Enschede)
P.C. Baayen (Amsterdam)
R.J. Boute (Nijmegen)
E.M. de Jager (Amsterdam)
M.A. Kaashoek (Amsterdam)
M.S. Keane (Delft)
J.P.C. Kleijnen (Tilburg)
H. Kwakernaak (Enschede)
J. van Leeuwen (Utrecht)
P.W.H. Lemmens (Utrecht)
M. van der Put (Groningen)
M. Rem (Eindhoven)
A.H.G. Rinnooy Kan (Rotterdam)
M.N. Spijker (Leiden)

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

The CWI is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O).

CWI Tract 67

Adaptive control & identification: Conflict or conflux?

J.W. Polderman



Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

1980 Mathematics Subject Classification: 93C40
ISBN 90 6196 377 X
NUGI-code: 811

Copyright © 1989, Stichting Mathematisch Centrum, Amsterdam
Printed in the Netherlands

Preface

This monograph is an (improved) reprint of my doctoral thesis. Among the many people that helped in various ways to write it, I want to thank my supervisors Jan H. van Schuppen and Jan C. Willems, for their constant support. The Centre for Mathematics and Computer Science in Amsterdam, where the research was carried out, is greatly acknowledged for providing a stimulating and sociable working atmosphere.

J.W. Polderman

CONTENTS

How this monograph is organized 2

Chapter I: Introduction 3
1. Introduction 3
1.1 What is adaptive control? 3
1.2 Approaches to adaptive control 12

Chapter II: Mathematical framework 15
1. Introduction 15
1.1 Adaptive controllers 15
1.2 The sets G and H 23
2. G and H for pole assignment: Conflux of identification and control 31
3. G and H for LQ control: Conflict between identification and control 40
4. When is G contained in H? 54

Chapter III: Adaptive control algorithms 60
1. Avoiding singularities in the parameter space 60
2. Adaptive pole assignment 69
2.1 Introduction 69
2.2 Problem statement 70
2.3 Observed state case 71
2.4 Non-observed state case 79
2.5 Simulations 89
3. An adaptive LQ controller for a first order system 91
4. An adaptive pole assignment algorithm for a first order continuous time system 101
4.1 Introduction 101
4.2 The algorithm 101
4.3 Implementation 108

Epilogue 110

Notational conventions 112

References 113

Index 117

HOW THIS MONOGRAPH IS ORGANIZED

This work consists of three chapters. The first chapter is of an introductory nature, the second and the third contain the theoretical material. Each chapter consists of several sections. Some of these sections are subdivided in subsections. The sections do not carry the chapter numbers. The subsections carry the section number and a subsection number. Formulas are numbered as follows. If a section contains a subsection, then the formulas in a subsection carry the subsection number and the formula number, otherwise formulas carry only the formula number. In each (sub)section the numbering starts all over again. If we refer to a formula within the same (sub)section we do so by calling its number, if we refer to a formula outside the present (sub)section, then the chapter number, section number and if necessary the subsection number and finally the formula number is called. Formula numbers always go between brackets. Theorems, lemmas, corollaries, remarks etc. are numbered with the section number, subsection number and their local number. References to theorems etc. are made in the same fashion as to formulas.

Chapter I

Introduction

1. INTRODUCTION

1.1 What is adaptive control?

This monograph is concerned with the problem of adaptive control. In this introductory section we will try to give a non-mathematical description of this problem field. Since the motivation for studying control theory stems from practical real-life problems rather than from elegant abstract mathematical thoughts, we will illustrate the discussion with several examples which in our opinion reflect one or more fundamental features of adaptive control. To that end we will first give a brief description of the notion of dynamical system and of the notion of classical control problem for a dynamical system.

A *dynamical system,* the basic object of study in system theory, is a mathematical model of a dynamical phenomenon, for instance a mechanical system. A dynamical input/output system is a dynamical system in which two entities, called input and output, can be distinguished. This distinction is such that the output is causally dependent on the input. This dependence can be thought of as determined by a set of *laws,* for example the laws of mechanics. The above description is in the spirit of the mathematical definition of a dynamical system as advocated by Willems [61], and the reader is referred to [61] for an elaborate discussion of this definition. Examples of dynamical input/output systems will be given in the sequel.

A *control objective* for a dynamical input/output system is defined by a collection of specifications on the behavior of the system. The rules of the game are then to manipulate the input of the system in such a way that the specifications are met. This can of course be done only if the specifications do not contradict the laws of the system. Well-known examples of control problems are: stabilization and optimal control.

Let us become a little more specific by giving an example.

EXAMPLE 1 (Inverted pendulum) Consider a rod of length l and mass m mounted on a carriage (see [38]). The carriage can move along a horizontal rail and the rod can move in the plane passing through the rail and orthogonal to the surface. The carriage can be moved along the rail to the left or the right

by applying appropriate forces. The measured quantities for this dynamical system are the position of the carriage, the velocity of the carriage, the angle of the rod and the angular velocity of the rod. It is not difficult to understand that the forces on the carriage cause all other variables and should hence be considered as the input of the system. The other variables together form the output. The control problem we want to discuss for this input/output dynamical system is the following: how to generate an input function such that the rod comes to an upright position and remains so. It is not our purpose to discuss possible solutions to this control problem, we merely state that based on the physical laws governing this system, the mass m and the length l, a device can be designed which carries out the control task. Important to our discussion is that the input rule depends on the (physical) laws of the system which we want to manipulate.

□

The solution of every non-trivial control problem will depend on the laws of the system. It is important to note that the controller itself is a dynamical system.
Let us now give a verbal definition of adaptive control.

DEFINITION An adaptive control problem is a control problem where the laws of the system to be controlled are not completely known.

□

The term adaptive stems from the original motivation of adaptive control. Originally adaptive control was intended to be applied to systems of which the laws change (slowly) in time. The controller was then supposed to adapt itself to these changes. The bulk of the existing literature, however, is devoted to the problem of controlling systems with constant but unknown laws. This monograph too is exclusively concerned with systems of which the laws do not change in time. However, there is one type of change in time that we do not want to exclude, namely abrupt changes of the system laws. The reason for this is that it should be possible to connect an adaptive controller to different systems. If we would first use the adaptive controller for one system and then connect it to another one, the controller is supposed to adapt itself to the new system to be controlled. This property reflects a certain universality of adaptive controllers.

REMARK One could define an adaptive controller as a controller which adapts itself to operation conditions. However, this description does not provide a clear distinction between a classical and an adaptive controller. As an example consider Watt's governor for steam engines. This is a device that contracts a valve whenever the velocity gets too high so as to temper the steam pressure and hence the velocity, and opens it as soon as the velocity goes beyond some prescribed threshold. Hence this controller adapts itself to changing operation conditions in the sense that the control action depends on these conditions.

Nevertheless, it should not be considered an adaptive controller, since it may work well for the particular steam engine only. Only if it guarantees good control for *different* engines, should it be called adaptive. This example illustrates the difference between adaptive control and what is usually called constant feedback control.

□

EXAMPLE 1 (continued) Consider the problem of the inverted pendulum. A relevant parameter in the laws of the system is for instance the mass of the rod. If this parameter is not known, we are faced with an adaptive control problem.

□

The main problem in adaptive control is of course that once the laws governing the system are unknown, the appropriate controller is also unknown, since it depends on the unknown laws.
How to cope with this problem?
One way of dealing with the problem of controlling a dynamical system of which the laws are unknown is the following. Carry out a number of experiments on the system and observe the resulting behavior of the system. If the experiments are diverse enough, one may be able to deduce the laws governing the system and then design a controller based on these laws.

EXAMPLE 1 (continued) Suppose the mass of the rod is unknown. By just moving the carriage slightly, the mass of the rod can be approximately calculated from the resulting motion of the rod.

□

This type of solution is usually not considered as part of adaptive control and is often referred to as off-line identification. The unknown laws are determined before the actual control process starts and are not updated anymore. This method can be very useful in many situations, but there may be some drawbacks. For instance, the mass of the rod can be determined with a finite accuracy only. In the case of the inverted pendulum this does not necessarily cause serious trouble, since it can be expected that the controller will not be very sensitive with respect to small errors in m, but for more complicated systems one may wish to check the values of the parameters every now and then, and adjust them if necessary. This will certainly be the case if these parameters change with time. This last statement may sound a little contradictory, since it was assumed that the system laws do not change with time. However, although the theory is based on this assumption it is commonly believed that adaptive controllers that are designed for systems with constant laws, will at least work satisfactorily if the laws change slowly.

A solution in the spirit of adaptive control is as follows. Adjust the control mechanism according to the behavior of the system. This should be done in

such a way that every adjustment leads to an improvement of the quality of the controller.

This is a fairly vague description and there are many ways it could be done. We will describe two possibilities, but first we would like to speculate a little on our own ability of controlling a system adaptively.

EXAMPLE 1 (continued) Let us once more consider the rod, but now no longer mounted on the carriage. Imagine a human being who tries to balance the rod on his hand. This can be seen as an example of an adaptive control problem, since the person tries to control a system of which he does not know the governing laws exactly. Most people will have little difficulty in balancing the rod as long as its weight and length have a reasonable value. So let us indeed assume that the person is capable of keeping the rod standing up on his finger tip. How is our system-theoretic juggler doing this? Is he trying to discover the physical laws the rod obeys? And is he then estimating the relevant parameters such as length and mass occurring in these laws? And after having done that, does he then design a control strategy so as to keep the rod in the desired position? It seems to us that it is very unlikely that something like that really happens, since the person may have no knowledge of control theory at all. It does not belong to our competence to analyze how a human being is able to balance a rod without knowing anything about system theory or physics. However, private experience has shown that in order to balance the rod you have to keep the tip of the rod in view and compensate for movements of the rod (observed by carefully watching the tip) by moving your hand in the appropriate (i.e. same) direction. Initially you will not have a good feeling for how fast and intense the compensating movements of the hand should be, and as a result the rod will make wild and unpredictable movements or it may even fall. These wild movements are undesirable from the stability point of view, but one can imagine that it enables the juggler to gain a good feeling for how to keep the rod upright.
Now if a person is able to balance one rod, he may initially have trouble in balancing another one if its length or its mass differs considerably from the first one. This reflects the phenomenon that he has to adapt himself to the new rod.

□

We will now discuss two approaches to adaptive control.

DIRECT ADAPTIVE CONTROL The type of adaptive control that comes close to the previous example is usually called direct adaptive control. Roughly speaking it consists of a (possibly infinite) family of controllers and of a device that on the basis of the observed behavior of the system to be controlled decides which of the controllers should be brought into operation. If that controller does not result in satisfactory behavior, the device tries another one, and so on, with the aim to end up with an appropriate controller. All this should be

done in a systematic way. By now there exists a vast amount of literature on the direct approach. The term *direct* refers to the fact that the laws of the controller are determined directly without first determining the laws of the system.

INDIRECT ADAPTIVE CONTROL If there exists a direct method, then the reader will not be surprised that also an indirect approach has been developed. In the indirect approach one first tries to pry the system laws out of the observed behavior of the system. And then, based on the "guessed" laws, the corresponding controller is designed. The process of estimating the laws from the observed behavior is usually called *identification.* We will not be specific about what we mean by the "corresponding" controller because several interpretations are possible. The designed controller may not be the right one, since the knowledge of the system laws may be incomplete, especially initially. But as time goes by, knowledge of the system laws gradually increases, and hence the controller can be designed more and more accurately by adjusting it to the knowledge acquired.

The main difference between the two methods just mentioned on the one hand and the off-line identification method on the other hand is that in the first case controller adjustment never stops. This implies that identification and control are done simultaneously. This is often referred to as *on-line identification.* In this work we will deal exclusively with the indirect approach. Approaches other than the two mentioned here are possible, but we will not explain them here. Summarizing, the main features of adaptive control are:

(i) The laws of the system to be controlled are not, or not completely known.
(ii) Control and identification have to be done simultaneously (on-line identification)
(iii) An adaptive controller is universal in the sense that it can be used for a whole class of systems: it adapts automatically to the characteristics of the system to be controlled.

An important problem which arises almost naturally in both the direct and the indirect approach is the *closed-loop identification* problem. This problem plays a major role in our work and we will therefore try to explain it here. It is best illustrated in the indirect method. Recall that this method was based on the estimation of the system laws on the basis of the observed behavior of the system. To determine the laws exactly, the experiments (i.e. trying different controllers) have to be diverse enough. If there is no experimentation at all, we learn nothing about the system; if the experiments are too poor, we learn only a little. It should be clear that if we apply all possible input functions to the system, we will induce all possible input/output relations. In principle these relations will determine the laws exactly. It can however not be expected that an adaptive controller will generate a rich enough range of inputs. For the inputs are not generated arbitrarily, but on the basis of the observed behavior. We will illustrate this by means of an example. It is a simplified version of an

example in [37], Chapter 12.4.

EXAMPLE 2 Suppose we have two slot machines. The first one pays one dollar per game with probability p (>0) and the second one pays one dollar per game with probability q (>0). From a certain point of view it is optimal to play on the machine with the highest probability of winning. Now suppose these probabilities are not known to the gambler and that the only possibility of figuring out these quantities is by playing on the machines. Consider the following strategy, which can be seen as an adaptive controller. First play ten times on the first machine and then ten times on the second one. Then play the twenty-first game on the machine with the highest pay-off thus far. Every next game is played on the "best" machine up to that moment. At first sight this seems to be a very reasonable and to some extent even natural strategy. However, it is not difficult to see that there is a positive probability that we get stuck on the worst machine. This can be seen as follows. Suppose that p is larger than q. Consider the event that the first ten games on the first machine (the best one) were unsuccessful and that on the second machine at least one game paid off. The probability of this event may be very small, still it is larger than zero. If this event really occurs, then from game twenty-one on all games should be played on machine two, because regardless of the outcome of each game, the yield of the first machine will stay zero (because we do not play on it anymore) whereas that of the second machine will always be positive. The reason that this can happen is that there is too little experimentation going on. From probability theory we know that p and q can only be estimated consistently (i.e. exactly) if we play infinitely often on both machines. This implies that experimentation should never stop: one should always be willing to reconsider one's opinion about p and q. In other words every strategy that claims to be able to detect the best machine within a finite number of trials is false.

□

Example 2 may be interpreted as an indirect adaptive control problem. The strategy of estimating p and q on the basis of the previous games and then proceeding on the machine with the highest estimated probability is too naive, since even after an infinite number of games p and q may not be known exactly. The example illustrates that the input of the system (in the example: the input is the choice of the machine) is not chosen arbitrarily, but is determined by the resulting output of the system (in the example: the output is the cumulative yield of each machine). It also shows that the resulting sequence of inputs is not rich enough to reveal the laws of the system (in the example: the laws are determined by p and q). This phenomenon is known as the closed-loop identification problem. The term closed-loop refers to the fact that the inputs are not generated arbitrarily but according to the behavior of the system.

In Example 2 the closed-loop identification problem caused a serious difficulty: the impossibility of determining the system laws, and hence the impossibility of controlling the system optimally. We call this the *conflict* between

identification and control.
There are also control problems where the closed-loop identification problem does not cause these extra difficulties. Again we illustrate this by an example.

EXAMPLE 3 Let there be given a non-zero real number $y(0)$. For every sequence $\{u(k)\}_{k\in\mathbb{N}}$ the sequence $\{y(k)\}_{k\in\mathbb{N}}$ is defined by:

$$y(k+1) = ay(k) + bu(k) \tag{1}$$

We interpret $u(k)$ as the input of the system and $y(k)$ as the output. For every (a,b) the law of the system is given by (1). Suppose that we do not know the true value of (a,b) but that we do know that it is either (2,2) or (1,1) and suppose furthermore that we want to choose the input in such a way that the output becomes identically zero. Assume that the true system law is given by (1,1), this means that the input/output sequence satisfies:

$$y(k+1) = y(k) + u(k) \tag{2}$$

This is clearly an adaptive control problem: we have to control the system without knowing its law exactly. A typical indirect strategy goes as follows. Try the first possibility for the value of (a,b), that is, postulate that:

$$y(k+1) = 2y(k)+2u(k) \tag{3}$$

If this were true, then the only way to make the output $y(1)$ equal to zero, is by choosing:

$$u(0) = -\frac{2}{2}y(0) = -y(0) \tag{4}$$

The resulting output will then be:

$$y(1) = y(0) + u(0) = y(0) - y(0) = 0 \tag{5}$$

And that was exactly what we wanted. It is not difficult to see that once the output is zero for $k=1$, the zero-input will keep it zero for all k, without ever revealing the true value of (a,b)! The surprising conclusion is that although we had a wrong guess about the value of (a,b), we were able to meet the control objective. Moreover, the wrongness of our guess will never be revealed by the behavior of the system.

□

Example 3 shows that there are adaptive control problems for which the lack of experimentation due to the closed-loop identification problem does not obstruct the control objective. In that case we speak about the *conflux* of identification and control.

Example 2 is a special case of an optimal control problem. An optimal control problem is a control problem where the control objective is the maximization of a yield or the minimization of a cost. It can be argued that for adaptive optimal control problems there will always be a conflict between identification

and control. In chapters two and three we will investigate this problem for the Linear Quadratic (LQ) optimal control problem.
Example 3 is a (very) special case of system regulation: controlling the output of the system to zero. The proof that for this type of control problem there is no conflict between identification and control will be given in chapter II. It is one of the major results of this monograph.

We have illustrated the problem of adaptive control by some examples. Let us now have a brief look at it from a research point of view. Classical control theory is based on the assumption that the laws of the system are known exactly. As such adaptive control should be seen as a generalization of classical control theory. Adaptive control is now a rapidly developing area in system theory. The reason for this is twofold. Firstly because of an increasing demand for controlling complex systems of which the laws are not exactly known or slowly changing with time. The second reason is that the actual implementation of adaptive controllers is more often possible than in the past, due to the growing availability of fast digital computers. The second point is particularly important since typically the application of an adaptive controller requires many computations to be carried out simultaneously with the control process.
Whereas the original motivation for adaptive control is of a purely practical nature, its impact on the theoretical aspects of system theory has been enormous, and still forms a challenge to many researchers in the field. Nowadays emphasis is on the development of *robust* adaptive controllers. Controllers that not only have good theoretical properties, but are also applicable in practical situations. This should fill the gap between on the one hand adaptive controllers that work well in practice but are theoretically not completely understood, and on the other hand adaptive controllers that work well in theory but are of no practical value since they rely on unrealistic assumptions. This monograph is of a theoretical nature and we do not make any claim about the applicability of our algorithms. The justification for this kind of work lies in the, in principle unverifiable, statement that if problems such as the closed-loop identification problem play a fundamental role on a theoretical level, then at least they will have some impact in practice. Also it can be argued that a minimum requirement for adaptive algorithms should be that they work well in an idealized theoretical environment.

We would like to conclude this section by describing two examples of successful application of adaptive controllers in practical control problems.

EXAMPLE 4 The ore crusher. An early account of a commercial application of adaptive control is given in Borisson and Syding [8]. There the adaptive control of an ore crusher is described. The task of this machine is to crush incoming ore to a prescribed maximum size, typically 25mm. The crusher is designed in such away that ore which does not meet the specifications after it has been crushed is returned in the crusher. The problem is that if a large amount of ore is fed back, the capacity of the crusher can be exceeded,

resulting in a standstill. The percentage of ore which has to be fed back into the crusher varies from 25 to 75 percent. These variations are due to the changing average sizes of the incoming ore, the crushability of the ore and the condition of the crusher. The control task is the regulation of the amount of incoming ore per time unit in such a way that the capacity of the crusher is used as efficiently as possible while avoiding overload due to the recycled ore. This can be seen as an adaptive control problem since the variations in the percentage of recycled ore induce unpredictable changes in the dynamics of the system. Originally the crusher was controlled by a classical controller. This controller had to be cautious in order to be able to deal with "worst case" situations. As a result the crusher operated only at 77% of its capacity. Adaptively controlled, it operated at 91% of the capacity. This was mainly due to the fact that the adaptive controller could be less cautious and as a result gave much better control in average situations.

□

EXAMPLE 5 Ship steering. In a series of papers, see [1] and the references there, Van Amerongen and several co-workers have reported on the application of adaptive control in the steering of ships. The reason for applying adaptive controllers to ship steering, rather than using classical controllers is that the conditions under which a ship has to be controlled can change considerably. Examples of such conditions are: the depth of the water, the loading of the ship, the current of the water, the wave-height etc. These conditions influence the dynamics of the ship and can therefore be seen as part of the dynamical system describing the ship. A ship can be modeled as a dynamical system with two inputs, rudder-angle and thrustpower, and two outputs, course-angle and speed [2]. In this model several parameters appear, some of which are constant and known and others such as the ones mentioned are subject to unknown changes. Classical control of the ship would call for manual adjustment of the controller parameters which can be difficult and time and fuel consuming.
In [1] two control problems for ship steering are considered. The first is course-tracking and the second is course-changing. These problems are formulated as optimal control problems; the cost criterion to be minimized is the fuel consumption. Adaptive algorithms that provide automatic adjustment of the controller parameters have been developed, theoretically analyzed, simulated and finally tested on full-scale experiments. Compared with classical controllers the adaptive controller resulted in more efficient rudder control. For the course-tracking problem this led to either a speed increase of 0.3% to 1.5% when the thrustpower was kept constant, or to a reduction of fuel consumption of 1.5% to 3% when the speed was kept constant. Expressed in terms of money this appears to be a significant improvement.

□

1.2 Approaches to adaptive control

Several approaches to adaptive control have been proposed and reported, all with their own features, advantages and disadvantages. At this moment it seems that there is little consensus as to what is the best way of defining and treating adaptive control. This is partly due to the wide variety of different adaptive control problems and partly because of different viewpoints. Before focusing on our viewpoint, we would like to give a brief description of the mainstreams in the field.

Although papers on adaptive control have been published since the fifties, see [24], we can take [5] as a starting point for the modern literature on the subject. There an algorithm for adaptive control of an ARMAX system is proposed. The algorithm is based on certainty-equivalence and minimum-variance control. The authors proved that *if* the parameter estimates converge, not necessarily to the true parameter, then asymptotically the variance of the output is minimized. This is the celebrated self-tuning property. However, they did not give a proof of convergence. A rigorous treatment of the self-tuning regulator and a proof of its convergence was provided by [23]. In [6] the same problem is considered, but analyzed in a different way. Based on the geometrical properties of the algorithm it was proved that the parameter estimates converge to a random multiple of the true system parameters. Asymptotically the minimization of the output variance was achieved also. A disadvantage of minimum-variance control is that only minimum-phase systems can be handled. A survey of adaptive control of stochastic systems is [32]. See also [4].

Adaptive stabilization of deterministic systems is another highly active part of the field. In continuous time both the direct and the indirect approach have been used. In the last few years there has been a fascinating discussion on the a priori assumptions on which the existing algorithms were based. These assumptions were that the system is minimum phase, that an upper bound on its order is known, that its relative degree is known and that the sign of the high frequency gain is known. The discussion led to the famous Morse-conjecture [47] which was disproven in [48], and later also in [62]. The final answer was given in [43,44] and [9], where necessary and sufficient conditions for adaptive stabilization were derived. In [43,44] and [9] it was proved that necessary and sufficient knowledge of the system to be adaptively stabilizable, is the order of a stabilizing compensator. The above-mentioned references are based on the direct method. The direct method has the advantage that the search is done in the controller space, which does not contain singular points. In the indirect approach certain singular points (e.g. non-minimal triples) have to be avoided, which causes extra difficulties. On the other hand it is not quite clear how to use the direct method for more sensitive control problems than just stabilization. In [28] the indirect method is used. Adaptive stabilization of discrete time systems using various design methods are treated in [3,21,31,42,50,53,55], see also Chapter III. Some of these references will be commented upon in Chapter III.

In Model Reference Adaptive Control the control objective is the design of a compensator such that the closed-loop system behaves like a prescribed system

for a given reference input signal. See [12] and [58] for a unified treatment.
The term Dual Control was introduced in [16]. In adaptive control it refers to the fact that the input of the system has two tasks. The first one is that of controlling the system, the second one is that of learning, that is, reducing the uncertainty one has about the system parameters. If the initial uncertainty is modeled as an a priori probability distribution on the parameter space, then in principle one can write down the corresponding dynamic programming equation, but to our knowledge closed-form solutions have not been reported. Approximate solutions have been proposed by [60]. See also [25,32].
A very interesting problem, at least at a theoretical level, is that of the adaptive control of a finite-state Markov chain. It provides a way of studying the main issues of adaptive control in an elementary setting. For instance, the closed-loop identification problem can very well be illustrated using this class of systems. In fact the first explicit account of the closed-loop identification problem was described in [7]. Implicitly it was already mentioned in [5]. See also [37]. A series of interesting papers on this adaptive control problem has been published by Kumar and co-workers. [33,35,36]. Applications to general input/output systems of the ideas developed there can be found in [34].
A fundamental problem in system theory is that of robustness. It was reported in [57] that existing adaptive control algorithms were not robust with respect to unmodeled dynamics. This was the starting point for research in the area of robust adaptive control. See [19] and the references given there.

It follows from the previous description that several approaches to the problem are possible. This monograph considers the problem of adaptively controlling deterministic linear finite-dimensional time-invariant discrete-time systems of known order. It is divided into two main parts. In Chapter II a mathematical framework is developed. This chapter consists of four sections. In the first section a definition of adaptive control is given and two subsets of the parameter space are introduced. These sets play a central role in the remainder of the monograph, they enable us to study some of the potential possibilities of an adaptive control problem without referring to a particular algorithm. It will turn out that the question whether or not the closed-loop identification problem causes serious difficulties for a given adaptive control problem, can be formulated in terms of these sets. In the second and third sections these sets and their relations are investigated for the pole assignment and LQ problem respectively. In the last section an (only partially successful) attempt is made to classify all control problems for which the closed-loop identification problem does not cause extra difficulties.
Chapter III is devoted to algorithms based on the considerations of Chapter II. It contains four sections. In the first section a general method is described on how to modify *any* indirect adaptive control algorithm so as to assure that the singular points (e.g. non-reachable pairs) in the parameter space are avoided. In the second section an adaptive pole-assignment algorithm for n-th order systems is proposed. The third section treats adaptive LQ control for a first order system. Finally the last section is a contribution to the continuous time

adaptive control literature. There an indirect algorithm for adaptive exponential stabilization is proposed.

Chapter II

Mathematical Framework

1. INTRODUCTION

1.1 Adaptive controllers

The purpose of this section is to develop a mathematical framework for the remainder of the monograph. First we will give a definition of an adaptive controller for a specific class of systems. We will then define certain principles on which an algorithm might be based. This will finally lead to the definitions of a recursive, a neutral, a sensitive and a neutral certainty equivalent adaptive control algorithm. This type of algorithm will form the main object of study in this monograph.

In part 2 of this section we will associate two subsets of the parameter space with the forementioned class of algorithms. It will turn out that part of the potential quality of an algorithm can be studied in terms of these sets.

The definitions in this section will be motivated and illustrated by simple examples. We would like to emphasize that our definitions do not reflect the ultimate definition of adaptive control. They should rather be viewed as a framework for a reasonable large class of algorithms whose properties we want to study.

This monograph is concerned with linear finite-dimensional deterministic systems only. In setting up a mathematical framework we shall therefore focus on this restricted class of systems. Much of what follows can easily be extended to more general classes of systems, but we do not want to impress or bother the reader with more abstract definitions and structures than are needed and used! On the other hand we believe that before concentrating on more specific problems, a certain level of generality can be of help in revealing and studying fundamental properties.

In the previous chapter we have seen that several approaches to the problem of adaptive control are possible. We will concentrate on the adaptive stabilization and self-tuning approach.

Loosely speaking an adaptive control problem can be seen as the problem of controlling an unknown but fixed plant. The adjective adaptive does then not refer to a controller which adapts itself accordingly to the changed characteristics of the plant, but should be understood in the following sense. The

controller is first confronted with a system of which it knows very little. Nevertheless it has to control it. In the beginning the control cannot be accurate, but after some time the behavior of the system will yield its characteristics more and more and the control can therefore become more accurate. Hence the controller is adapted according to the knowledge gained about the system. An important feature of an adaptive controller is that control and learning have to be done simultaneously.
A first verbal definition of an adaptive controller could be given by pointing out the difference between a controller and an adaptive controller. To this end we will introduce the notion of universal controller.

A *controller* is a device that is able to control one single system.

A *universal controller* is a device that is able to control a class of systems.

This description does not yet explain the word adaptive. In our opinion the adjective adaptive should reflect the phenomenon of the adaptation of the controller in reaction to increased knowledge of the system.

EXAMPLE 1.1.1 Let the control objective be stabilization. Consider the class of first-order linear time-invariant systems in discrete time.

$$y(k+1) = ay(k)+bu(k), \quad b\neq 0 \tag{1.1}$$

Every controller of the form $u(k)=fy(k)$ with f such that $|a+bf|<1$ will stabilize the system. On the other hand the fixed controller

$$u(k) = \bar{f}y(k) \tag{1.2}$$

will stabilize every system (a,b) for which $|a+b\bar{f}|<1$. Hence the controller $\bar{f}$ is able to control a class of systems, and should therefore be called universal for the class of systems which satisfy $|a+bf|<1$. However $\bar{f}$ is not universal for the class of all systems (a,b).
We conclude that universality depends on the class of systems. Universality not only depends on the class of systems but also on the control objective. Let us change the control objective from stabilization into a stronger one, namely exponential stabilization with a prescribed rate of stability. I.e. let there be given $\alpha\in\mathbb{R}$, with $|\alpha|<1$, and consider the problem to find a controller such that $y(k+1)=\alpha y(k)$. For given (a,b) the following feedback will do:

$$u(k) = fy(k), \quad f = \frac{\alpha-a}{b}. \tag{1.3}$$

Now there is exactly one controller for each system. But again, the same controller can be used for different systems. More precisely: If (a,b) satisfies:

$$a+bf = \alpha, \tag{1.4}$$

then f is the controller for (a,b). Hence f is universal for every (a,b) satisfying (1.4), but again f is not universal for the class of all first-order systems.

□

In the first example, the type of universality is usually called robustness. If we

equip the class of first order systems with the Euclidean topology, then for each controller there is an open subset of systems which can be stabilized with that controller. Robustness is not the topic of this monograph, but we will see that it is very difficult to make a formal distinction between robust and adaptive controllers.
Intuitively one would like to call a controller adaptive if it is able to control a reasonable large class of systems, for instance the class of all linear time-invariant first-order systems. Robust controllers are then locally universal and adaptive controllers are globally universal. One of the weak points in such a description is the term "reasonably large". For if the class of systems is sufficiently small, the difference between global and local may totally disappear.
Another attempt to come to a good definition of an adaptive controller is the following. Suppose that we have a class of systems and that for each system of that class we know a controller. We then have a family of controllers parametrized by the elements of the class of systems. The class of controllers could then be seen as an overall controller with adjustable parameters. If we now include some mechanism to adjust the parameters of the controller, then this could be called an adaptive controller. This description comes very close to the controller structures which are usually called adaptive.

EXAMPLE 1.1.2 Consider the following class of systems, parametrized by $(a,b)\in\mathbf{R}\times\mathbf{R}^+$.

$$\dot{y} = ay+bu, \quad y(0)\in\mathbf{R} \tag{1.5}$$

and suppose we want to stabilize (1.5) without knowing (a,b). Consider the following scheme:

$$\dot{k} = y^2, \quad k(0) = 0. \tag{1.6.a}$$

$$u = -ky \tag{1.6.b}$$

We claim that (1.5) together with (1.6) is stable in the sense that,

$$\lim_{t\to\infty} y(t) = 0, \quad \lim_{t\to\infty} k(t) = k_\infty. \tag{1.7}$$

PROOF From (1.6.a) it follows that $k(t)$ is non-decreasing. Hence there are two possibilities. Either k tends to infinity or it reaches a finite limit. Suppose k tends to infinity. For all t_0 we have:

$$y(t) = y(t_0)\, e^{\int_{t_0}^{t} (a - bk(s))ds} \tag{1.8}$$

Choose t_0 such that $a - bk(t_0) < -1$, this is possible since k tends to infinity and $b > 0$. Then from (1.5) it follows that:

$$|y(t)| \leqslant |y(t_0)|\, e^{-\int_{t_0}^{t} ds} \tag{1.9.a}$$

$$= |y(t_0)| e^{-(t-t_0)} \tag{1.9.b}$$

We conclude that $y \in \mathcal{L}_2(0,\infty)$, together with (1.6.a) this implies that k is bounded, which is a contradiction.
Suppose now that k is bounded, then by (1.6.a) it follows that $y \in \mathcal{L}_2(0,\infty)$, and hence by (1.6.b) also $u \in \mathcal{L}_2(0,\infty)$. By (1.5) we conclude that $\dot{y} \in \mathcal{L}_2(0,\infty)$. Finally y, $\dot{y} \in \mathcal{L}_2(0,\infty)$ implies:

$$\lim_{t\to\infty} y(t) = 0 \tag{1.10}$$

This finishes the proof.

REMARK The main reason that this type of adaptive controller stabilizes the system is that the sign of the "b"-parameter is known. This means that one knows in which direction k should change. The case where the sign of b is not known inspired S. Morse to conjecture that then the system could not be stabilized adaptively [47]. This conjecture was first disproved in [48], and later also in [62].

One can consider (1.6) as a fixed nonlinear dynamic compensator which is universal for every system $(a,b) \in \mathbf{R} \times \mathbf{R}^+$. As such it is not clear why one should call this controller adaptive. Another way of looking at (1.6) is the following. The class of feedback controllers for (1.5) can be parametrized by $k \in \mathbf{R}$. The problem is now to find a $k \in \mathbf{R}$ such that $a - bk < 0$. Now, (1.6.a) is a mechanism which adjusts the parameter k in (1.6.b) in such a way that asymptotically one ends up with a fixed stability linear controller, namely k_∞. Indeed, it is not difficult to prove that if $y(0) \neq 0$, then there must hold:

$$a - bk_\infty < 0. \tag{1.11}$$

Note that the gain adjustment mechanism ($\dot{k} = -y^2$) is driven by the output of the system.
Hence either one considers (1.6) as a "complicated" (= nonlinear) fixed controller or one considers it as a combination of a parametrized family of simple (= linear) controllers and an adjustment procedure driven by the observed data. The objective of the adjustment procedure is then to ultimately select a controller which stabilizes the system of which the parameter values are unknown. In our opinion the term adaptive is best reflected by the second viewpoint.

□

From the previous discussion it is clear that a general and natural definition of an adaptive controller is not easy to give. Therefore we will now restrict our attention to the class of systems which will be the main object of study in this monograph, namely the class of linear time-invariant systems of fixed order. For the rest of this section we will mainly work in discrete time.

We will formulate our definitions in terms of systems in input/output form rather than in input/state/output form. The latter would lead to complicated and awkward definitions, mainly due to the fact that observers have then to be included. We will now describe the class of systems and the class of controllers under consideration.

DEFINITION 1.1.3 (class of systems)

$$\sum(n) := \{(A,B)\in\mathbb{R}[z]^2 \mid A(z) = z^n - a_0 z^{n-1} - .. - a_{n-1}, \quad B(z) = b_0 z^{n-1} + .. + b_{n-1}, \quad \text{g.c.d.}(A,B) \equiv 1\} \tag{1.12}$$

□

INTERPRETATION: with $(A,B)\in\mathbb{R}[z]^2$

$$A(z) = z^n - a_0 z^{n-1} - .. - a_{n-1} \tag{1.13}$$

$$B(z) = b_0 z^{n-1} + .. + b_{n-1} \tag{1.14}$$

we associate the input/output system:

$$y(k+1) = a_0 y(k) + .. + a_{n-1} y(k-n+1) + b_0 u(k) + .. + b_{n-1} u(k-n+1) \tag{1.15}$$

DEFINITION 1.1.4 (class of controllers)

$$\Sigma_c(r) := \{(M,N)\in\mathbb{R}[z]^2 \mid M(z) = z^r - m_1 z^{r-1} - .. - m_{r-1}, \quad N(z) = n_0 z^r + .. + n_{r-1}, \quad \text{g.c.d.}(M,N) \equiv 1\} \tag{1.16}$$

DEFINITION 1.1.5 A control objective on $\Sigma(n)$ is determined by a set of specifications on the behavior of each element of $\Sigma(n)$. A solution of order r of the control problem is a (possibly multivalued) function:

$$F: \sum(n) \to \Sigma_c(r) \tag{1.17}$$

□

INTERPRETATION By a solution of order r of a control objective on $\sum(n)$ we mean that there exists a (multivalued) function F which assigns to a pair $(A,B)\in\Sigma(n)$, at least one pair $(M,N)\in\Sigma_c(r)$ with the property that each pair $(M,N)\in F(A,B)$ is such that if these two systems are connected in feedback:

$$y(k+1) = a_0 y(k) + .. + a_{n-1} y(k-n+1) + b_0 u(k) + .. + b_{n-1} u(k-n+1) \tag{1.18.a}$$

$$u(k) = m_1 u(k-1) + .. + m_{r-1} u(k-r+1) + \tag{1.18.b}$$

$$n_0 y(k)+..+n_{r-1}y(k-r+1),$$

the resulting behavior of the controlled system is as specified by the control objective. Note that it is not necessary to give a precise definition of the notion of control objective since this is implicitly done by the map F.
Examples of control objectives:
(i) Stability
(ii) Pole Assignment
(iii) Linear-Quadratic Control

We will now give a definition of an adaptive controller for the class of systems $\Sigma(n)$.

DEFINITION 1.1.6 An adaptive controller of order r on $\sum(n)$ is a pair (θ,f), where:

$$\theta\colon (\mathbb{R}^2)^{\mathbb{N}} \to \sum(n) \quad f\colon \sum(n)\times\mathbb{R}^{2n+1} \to \sum_c(r) \tag{1.19}$$

□

INTERPRETATION An adaptive controller consists of an identification part (the function θ), which assigns to every data sequence (think of input/output sequence) a system (model) and a control part, given by the function f. The controller that is assigned to the estimate may also depend on the most recent $n+1$ outputs and n inputs. Of course this is a special choice; we could also allow f to depend on more inputs and outputs, but since we will not use that, we do not want to include this possibility in the definition. An example of this type of dependence is obtained when the controller also depends on the prediction error:
Let $(\hat{A}_k,\hat{B}_k) := \theta(y(0),..,y(k),y(k+1),u(0),..,u(k))$. Define:

$$(\hat{M}_k,\hat{N}_k) := f(\hat{A}_k,\hat{B}_k,y(k+1) - \hat{y}(k+1)) \tag{1.20}$$

where

$$\hat{y}(k+1) := \hat{a}_0(k)y(k)+..+\hat{a}_{n-1}(k)y(k-n+1) + \hat{b}_0(k)u(k)+..+\hat{b}_{n-1}(k)u(k-n+1) \tag{1.21}$$

(the coefficients of $\hat{A}_k$ and $\hat{B}_k$ are denoted by $\hat{a}_i(k)$ and $\hat{b}_i(k)$ respectively).

DEFINITION 1.1.7 (certainty equivalence) The adaptive controller (θ,f) is said to be based on *certainty equivalence* for the control objective F, solved by $\Sigma_c(r)$, if for all $(A,B)\in\sum(n)$, and for all $x\in\mathbb{R}^{2n+1}$:

$$f(A,B,x)\in F(A,B). \tag{1.22}$$

(if F is a map then (1.22) implies $f(A,B,x) = F(A,B)$).

□

INTERPRETATION Certainty equivalence means that the applied controller is calculated as if the estimation θ of the unknown system were indeed the true system. That means that the adaptive controller does not actively choose the inputs so as to learn more about the system. In that sense the learning is passive.

DEFINITION 1.1.8 (recursiveness) An adaptive controller (θ,f) is said to be recursive of order (l_1,l_2) if there exists

$$\tilde{\theta}\colon \sum(n)^{l_1}\times\mathbf{R}^{l_2}\to\sum(n) \tag{1.23}$$

such that for all $m=(m_0,m_{-1},m_{-2},..)\in(\mathbf{R}^2)^{\mathbf{N}}$

$$\theta(m)=\tilde{\theta}(\theta(m_{-1},m_{-2},..),..,\theta(m_{-l_1+1},m_{-l_1},m_{-l_1-1},..),m_0,m_{-1},..,m_{-l_2+1}) \tag{1.24}$$

(for m_i one can think of an input/output sequence).

□

INTERPRETATION A recursive adaptive controller is an adaptive controller for which the estimation part is recursive. The estimation at time k can be calculated on the basis of l_1 past estimates and l_2 past observations.

DEFINITION 1.1.9

(i) (neutrality) A recursive adaptive controller $(\tilde{\theta},f)$ of order $(1,2n+1)$ is said to be *neutral* if for all $(A,B)\in\sum(n)$, and for all $2n+1$-tuples

$$[y(k+1),y(k),u(k),..,y(k-n+1),u(k-n+1)] \tag{1.25}$$

with the property that:

$$y(k+1)=a_0y(k)+..+a_{n-1}y(k-n+1)+ \tag{1.26}$$
$$b_0u(k)+..+b_{n-1}u(k-n+1)$$

(the coefficients of (A,B) are denoted by a_i and b_i respectively). the following holds:

$$\tilde{\theta}((A,B),y(k+1),..,y(k-n+1),u(k),..,u(k-n+1))=(A,B). \tag{1.27.a}$$

(ii) (sensitivity) The recursive adaptive controller $(\tilde{\theta},f)$ of order $(1,2n+1)$ is said to be *sensitive* if for all $(A,B)\in\Sigma(n)$, and for all $2n+1$-tuples (25) with the property that (1.26) does *not* hold, the following holds:

$$\tilde{\theta}((A,B),y(k+1),..,y(k-n+1),u(k),..,u(k-n+1))\neq(A,B). \tag{1.27.b}$$

□

INTERPRETATION A recursive adaptive controller is called neutral if the following property holds: if the current estimate is compatible with the observed data (i.e. the current estimate is not falsified by the observed data), then the next estimate is equal to the current one. That means that the adaptive

controller does not actively choose the inputs so as to learn more about the system. In that sense the learning is passive. The adaptive controller is called sensitive if the estimate is changed if it is not compatible with the observed data.

DEFINITION 1.1.10 (neutral certainty equivalence) An adaptive controller (θ,f) is called *neutral certainty equivalent* for F if for all $(A,B)\in\Sigma(n)$ and for all $2n+1$-tuples

$$[y(k+1),y(k),u(k),..,y(k-n+1),u(k-n+1)] \tag{1.28}$$

such that:

$$y(k+1) = a_0y(k)+..+a_{n-1}y(k-n+1)+b_0u(k)+..+b_{n-1}u(k-n+1) \tag{1.29}$$

the following holds:

$$f(A,B,y(k+1),y(k),u(k),..,y(k-n+1),u(k-n+1))\in F(A,B) \tag{1.30}$$

□

INTERPRETATION Neutral certainty equivalence is weaker than certainty equivalence. Loosely speaking it means that one is restricted to use the certainty equivalence principle only if the observed data do not falsify the estimate (A,B). This type of adaptive controller can be very useful, since it allows one to deviate from certainty equivalence as long as the prediction error is nonzero. On the other hand *if* the prediction is zero, then the estimate *could* be the true system and hence we should apply the controls according to certainty equivalence.
Recursiveness, certainty equivalence and neutrality are widely used principles in the adaptive control literature. The algorithms we will present in Chapter III will be derived from these principles.

REMARK 1.1.11 Descriptions of adaptive controllers for continuous-time systems may be found in [9]. See also [22,27,28,29] for a general treatment of identification algorithms for continuous-time systems. Most of the algorithms in the literature are based on the principles which we have just developed.
Sometimes the map (f,θ) is not explicitly factorized but considered as a map that assigns a controller to observed data. Such algorithms are called *direct*. If the factored form is used we speak about *indirect* algorithms.
As may be concluded from the previous discussion it is difficult, if not impossible, to give a satisfactory and mathematical definition of adaptive control on a general level. For a specific class of systems a definition has been given however. For this class we can now distinguish between adaptive and robust (universal) controllers, by saying that a robust controller is a local controller and an adaptive controller is a global controller. Here local is with respect to some natural topology on the $\sum$.
There is another important difference between robust and adaptive controllers. Thus far we have only given definitions without specifying what we would like

to do with an adaptive controller. Of course a desirable property of an adaptive controller is that, at least asymptotically, exactly the right control signals are applied. This is a property that robust controllers do not have in general. For instance if the control objective is pole-assignment, then if the controller is calculated on the basis of a model and the true system is a slight perturbation of that model, the poles are not assigned exactly but only approximately. A good adaptive controller would adjust the model until the poles are placed exactly.

1.2. Identification in closed-loop: the sets G and H

In the second part of this section we want to study some of the potential possibilities and fundamental limitations of adaptive controllers as defined in the first part.

Let (f,θ) be an adaptive controller for $\Sigma(n)$ and the control objective F. The map $\theta:(\mathbb{R}^2)^{\mathbb{N}} \to \Sigma(n)$ represents the identification part of the controller. The term identification, however, does not have any meaning as long as we do not specify the arguments of θ and how they are generated.

ASSUMPTION For the rest of this section we will assume that there is given a nonzero vector $\phi(0)\in\mathbb{R}^{2n-1}$ and a fixed but unknown pair $(A_0,B_0)\in\Sigma(n)$. Furthermore it is assumed that each time we refer to a data sequence $\{y(k+1),y(k),u(k),..,y(k-n+1),u(k-n+1)\}$ it satisfies:

$$y(k+1) = a_0^0 y(k)+..+a_{n-1}^0 y(k-n+1)+b_0^0 u(k)+..+b_{n-1}^0 u(k-n+1) \quad (2.1)$$

where the a_i^0's and the b_j^0's are the coefficients of A_0 and B_0 respectively. Finally we will assume that the initial state of the system (A_0,B_0) is :

$$(y(0),..,y(-n+1),u(-1),..,u(-n+1)) = \phi(0) \quad (2.2)$$

□

In part one we have defined the notion of adaptive controller; we now want to define an adaptive control *scheme*. Loosely speaking this will be an adaptive controller (f,θ) together with a rule which tells us how to apply (f,θ) to real data coming from some unknown system.

DEFINITION 1.2.1 An adaptive control algorithm of order r for $\sum(n)$ is an adaptive controller (θ,f) together with the scheme

$$(\hat{A}_k,\hat{B}_k) = \theta(y(k),y(k-1),u(k-1),y(k-2),u(k-2),..) \quad (2.3.a)$$

$$(\hat{M}_k,\hat{N}_k) = f(\hat{A}_k,\hat{B}_k,y(k),y(k-1),u(k-1),..,y(k-n),u(k-n)) \quad (2.3.b)$$

$$u(k) = \hat{m}_1(k)u(k-1)+..+\hat{m}_{r-1}u(k-r+1)+ \quad (2.3.c)$$
$$\hat{n}_0(k)y(k)+..+\hat{n}_{r-1}(k)y(k-r+1)$$

where the data $(u(k),y(k))$ satisfy the equations of the unknown system (A_0,B_0):

$$y(k+1) = a_0y(k)+..+a_{n-1}y(k-n+1)+b_0u(k)..+b_{n-1}u(k-n+1) \quad (2.3.d)$$

□

REMARK 1.2.2 For the definition of an adaptive control algorithm it is not necessary that the unknown system belongs to the class $\sum(n)$. For the evaluation of the behavior of the resulting controlled system, however, one has to impose something on the unknown system. In this monograph we shall mainly study the behavior of the controlled system subject to the assumption that it belongs to $\sum(n)$, where n is known. This is what some people nowadays refer to as the ideal case.

□

Let us now assume that the data $\{(u(k),y(k)\}$ is indeed produced by the system $(A_0,B_0)\in\Sigma(n)$. A first natural question is: how does the sequence $\{(\hat{A}_k,\hat{B}_k)\}$ behave? If we consider $(\hat{A}_k,\hat{B}_k)$ as an estimate for (A_0,B_0), then in particular we could like to know whether or not:

$$\lim_{k\to\infty} (\hat{A}_k,\hat{B}_k) = (A_0,B_0) \quad (2.4)$$

Or otherwise stated we would like to know whether or not the true system is identified. Let us first consider the question of identification. It is well-known that for the identification of a system it has to be sufficiently excited by the input. For instance, it is clear that if the input of the system is chosen to be zero, then no information can be gained about the B-polynomial of the system. Several papers appeared which gave conditions on the input signal to ensure that from the resulting input/output behavior the system can be identified completely.

EXAMPLE 1.2.3 Consider the first-order system

$$y(k+1) = ay(k)+bu(k) \quad (2.5)$$

(a,b) unknown, $b\neq 0$.
An estimation method for (a,b) which will be used and commented upon in Section III.2 is the projection algorithm. In recursive form it reads as follows:

$$\hat{a}(k+1) = \hat{a}(k)+\frac{y(k)}{y^2(k)+u^2(k)}(y(k+1)-\hat{a}(k)y(k)-\hat{b}(k)u(k)) \quad (2.6.a)$$

$$\hat{b}(k+1) = \hat{b}(k)+\frac{u(k)}{y^2(k)+u^2(k)}(y(k+1)-\hat{a}(k)y(k)-\hat{b}(k)u(k)) \quad (2.6.b)$$

Now take as control law:

$$u_k=(-1)^k y(k), \quad (2.6.c)$$

then (2.6.a,2.6.b) can be written as:

$$\hat{a}(k+1) = \hat{a}(k)+\frac{1}{2}((a_0-\hat{a}(k))+(b_0-\hat{b}(k))(-1)^k) \quad (2.6.d)$$

$$\hat{b}(k+1) = \hat{b}(k)+\frac{1}{2}((a_0-\hat{a}(k))(-1)^k+(b_0-\hat{b}(k))) \tag{2.6.e}$$

Define:

$$V_k := \|(a_0,b_0) - (\hat{a}(k),\hat{b}(k))\|^2 \tag{2.7}$$

Then from (2.6.c,2.6.d,2.6.e) it follows that:

$$V_{k+1}-V_k \leqslant \frac{-1}{2}((a_0-\hat{a}_k)+(b_0-\hat{b}_k)(-1)^k)^2 \tag{2.8}$$

The first conclusion is that $\|(a_0,b_0) - (\hat{a}(k),\hat{b}(k))\|$ converges. Also one can easily prove, see Lemma III.2.2.1, that

$$\lim_{k\to\infty}(\hat{a}_{k+1}-\hat{a}_k)^2+(\hat{b}_{k+1}-\hat{b}_k)^2 = 0 \tag{2.9}$$

Choose a subsequence $\{s_k\}$ of $2\mathbb{N}$ (the even natural numbers) such that $(\hat{a}_{s_k},b_{s_k})$ converges, say

$$\lim_{k\to\infty}(\hat{a}_{s_k},\hat{b}_{s_k})=(\bar{a},\bar{b}) \tag{2.10}$$

then by (2.9) also:

$$\lim_{k\to\infty}(\hat{a}_{s_k+1},\hat{b}_{s_k+1}) = (\bar{a},\bar{b}) \tag{2.11}$$

It follows from (2.10) and (2.11) and the fact that s_k is even that $(\bar{a},\bar{b})$ is an invariant point of (2.6) for both k even and odd. This gives:

$$\bar{a} = \bar{a} + \frac{1}{2}((a_0-\bar{a}) + (b_0-\bar{b})) \tag{2.12}$$

and:

$$\bar{a} = \bar{a} + \frac{1}{2}((a_0-\bar{a}) - (b_0-\bar{b})) \tag{2.13}$$

(2.12) and (2.13) yield $\bar{a}=a_0$, $\bar{b}=b_0$. This together with (2.8) gives:

$$\lim_{k\to\infty}(\hat{a}_k,\hat{b}_k) = (a_0,b_0). \tag{2.14}$$

□

(2.6.c) is a sufficient condition for the sequence of estimates to converge to the true system parameters. Two remarks are in order. It should be emphasized that (2.6.c) is just an example of a sufficient condition for identification and serves as an illustration that simple conditions on $u(k)$ can be given. On the other hand, since it is clear that $u(k)\equiv 0$ can never identify the b-parameter, one should at least have some conditions on the "richness" of the input signal. We now want to study the richness of the input signals if they are generated by an adaptive control algorithm. We will restrict our attention to recursive, neutral, sensitive, and neutral certainty equivalent algorithms. In particular

algorithms of the form:

$$(\hat{A}_{k+1},\hat{B}_{k+1}) = \tilde{\theta}(\hat{A}_k,\hat{B}_k,y(k+1),y(k),u(k),..,y(k-n+1),u(k-n+1)) \quad (2.15.a)$$

$$\hat{y}(k+1) = \hat{a}_0(k)y(k)+..+\hat{a}_{n-1}(k)y(k-n+1)+ \quad (2.15.b)$$
$$\hat{b}_0(k)u(k)+..+\hat{b}_{n-1}(k)u(k-n+1)$$

$$(\hat{M}_k,\hat{N}_k) = f(\hat{A}_k,\hat{B}_k,y(k)-\hat{y}(k)) \quad (2.15.c)$$

$$u(k) = \hat{m}_1(k)u(k-1)+..+\hat{m}_{r-1}u(k-r+1)+ \quad (2.15.d)$$
$$\hat{n}_0(k)y(k)+..+\hat{n}_{r-1}(k)y(k-r+1)$$

and $\tilde{\theta}$ has the property

$$y(k+1) = \hat{y}(k+1) \Rightarrow (\hat{A}_{k+1},\hat{B}_{k+1}) = (\hat{A}_k,\hat{B}_k) \quad \text{(neutrality)} \quad (2.16)$$

and f satisfies:
for all $(A,B)\in\Sigma(n)$ and for all $y\in\mathbb{R}$:

$$f(A,B,0)\in F(A,B) \quad \text{(neutral certainty equivalence)} \quad (2.17)$$

Note that by abuse of notation, the third argument of f is a real number rather than a $2n+1$-vector as in the original definition of f.

REMARK 1.2.4 Why focus on this type of algorithm?

(i) Recursiveness. For computational reasons.

(ii) Neutrality. If the predicted output ($\hat{y}(k+1)$) equals the observed output, then there is no other obvious choice for the next estimate than the current one.

(iii) Sensitivity. If the prediction error is non-zero then we know for sure that the current estimate is wrong. It is reasonable to require that the algorithm then changes the estimate.

(iv) Neutral certainty equivalence. If the predicted output equals the observed output, then it is always possible that the estimate on which the prediction was based is in fact the true parameter (A_0,B_0). If that is the case it will be kept constant for ever, due to neutrality. Hence in that case it is reasonable to require that a desired control ($\in F(A_0,B_0)$) is applied. Since (A_0,B_0) is not known, the principle follows.

□

Note that the convergence proof in Example 1.2.3 was based on the observation that the only invariant point of the estimation scheme was (a_0,b_0), the boundedness of the sequence of estimates and the vanishing difference between successive estimates.

On this level of generality the question of identification can only be studied in terms of the invariant points of the estimation part of the algorithm, in the sense that every invariant point is a potential limit of the sequence of estimates. From the neutrality and sensitivity assumptions it follows that:

$$(\hat{A}_{k+1},\hat{B}_{k+1}) = (\hat{A}_k,\hat{B}_k) \quad \text{iff} \quad y(k+1) = \hat{y}(k+1) \tag{2.18}$$

Hence invariant points are those $(A,B)\in\Sigma(n)$ for which for all k: $y(k) = \hat{y}(k)$.

DEFINITION 1.2.5

$$G := \{(A,B)\in\Sigma(n) \mid \text{for all } k: y(k) = \hat{y}(k)\} \tag{2.19}$$

where $\hat{y}(k)$ and $u(k)$ are defined by the algorithm (2.15). G is the set of invariant points of the algorithm (2.15) applied to the system (A_0,B_0) with initial state $\phi(0)$.

□

REMARK Note that although $u(k)$,$y(k)$ and $\hat{y}(k)$ depend on the particular algorithm, G will be the same for all algorithms that satisfy (2.15).

□

The question of identification can now be studied by analyzing G. If $G = \{(A_0,B_0)\}$, then boundedness of $\{(\hat{A}_k,\hat{B}_k)\}$ and $\lim_{k\to\infty}(A_{k+1}-A_k,\hat{B}_{k+1}-\hat{B}_k) = 0$ is sufficient for identification of (A_0,B_0). However, since the input of the system is not generated arbitrarily, but on the basis of the observed behavior of the system, it cannot be expected that G will consist of nothing more than (A_0,B_0).
We will illustrate this with an example.

EXAMPLE 1.2.6 Consider the system

$$y(k+1) = a_0y(k)+b_0u(k), \quad b_0\neq 0, \quad y(0)\neq 0 \tag{2.20}$$

desired control law:

$$u(k) = F(a_0,b_0)y(k) \tag{2.21}$$

Let (a,b) be an invariant point of any algorithm which is neutral and neutral certainty equivalent. With invariant we mean that if we start in (a,b) we stay there (for instance (a_0,b_0) itself is invariant).
Starting at $k=0$, we will apply

$$u(0) = F(a,b)y(0) \tag{2.22}$$

which gives:

$$y(1) = (a_0+b_0F(a,b))y(0) \tag{2.23}$$

$$\hat{y}(1) = (a+bF(a,b))y(0) \tag{2.24}$$

Since (a,b) is invariant we conclude that $y(1)=\hat{y}(1)$. In general we will have:

$$y(k+1) = (a_0+b_0F(a,b))y(k) \tag{2.25}$$

$$\hat{y}(k+1) = (a+bF(a,b))y(k) \tag{2.26}$$

Now, if $y(k)=0$ for all k, we can learn very little of (2.20). Assume that there exists $\bar{k}$ such that $y(\bar{k})\neq 0$. Then after dividing by $y(\bar{k})$ we conclude:

$$a_0+b_0F(a,b) = a+bF(a,b) \tag{2.27}$$

On the other hand one may easily check that if (a,b) satisfies (2.27), then it is invariant. Hence:

$$G = \{(a,b) \mid a_0 + b_0F(a,b) = a + bF(a,b),\ b\neq 0\} \tag{2.28}$$

□

In general G will consist of an infinite number of pairs (a,b). The phenomenon that G is larger than just $\{(A_0,b_0)\}$ is due to the fact that identification takes place in closed-loop: Information is obtained only about the closed-loop behavior of the system. It is very likely that there are many parameter values that give rise to the same closed-loop behavior.
However, identification of (A_0,B_0) is not the primary goal of adaptive control. The main goal is just control, and identification of (A_0,B_0) may not be needed. Otherwise stated we will be happy to have an estimate of the system that induces the same sequence of inputs as required by the control objective. Hence what we want is to arrive at an estimate $(A,B)\in\Sigma(n)$ such that:

$$f(A,B,0)\in F(A_0,B_0) \tag{2.29}$$

Let us define the set $\bar{H}$ as the subset of $\Sigma(n)$ of pairs (A,B) with that property:

$$\bar{H} := \{(A,B)\in\Sigma(n) \mid f(A,B,0)\in F(A,B)\} \tag{2.30}$$

The set $\bar{H}$ may still not be what we are looking for. For if $(A,B)\in\bar{H}$, then (A,B) will certainly give rise to the desired controls, but it is not necessarily true that if some (A,B) produces the desired inputs, it then belongs to $\bar{H}$. Namely, it is very well possible that not all the modes of the system are excited and in that case only the action of $f(A,B,0)$ on the excited modes is relevant. Hence we define a slightly more appropriate set:

DEFINITION 1.2.7

$$H := \{(A,B)\in\Sigma(n) \mid \text{ for all } k: u(k) = u^0(k)\} \tag{2.31}$$

where:

$$(M,N) = f(A,B) \tag{2.32.a}$$

$$u(k) = m_0u(k-1)+..+m_{r-1}u(k-r+1) + n_0y(k)+..+n_{r-1}y(k-r+1), \tag{2.32.b}$$

and there exists

$$(M_0,N_0) \in F(A_0,B_0), \tag{2.32.c}$$

and:

$$u^0(k) = m_1^0 u(k-1)+..+m_{r-1}^0 u(k-r+1) + n_0^0 y(k)+..+n_{r-1}^0 y(k-r+1) \quad (2.32.d)$$

□

EXAMPLE 1.2.6 (continued) For the first order case there is no difference between $\bar{H}$ and H:

$$H = \{(a,b) \mid f(a,b) = f(a_0,b_0)\ b \neq 0\} \quad (2.33)$$

□

Now we come to the most crucial point of this chapter. We have just relaxed the requirement of identification and replaced it by the weaker requirement of producing the right controls. Hence instead of (2.4), we want to know whether or not:

$$\lim_{k\to\infty} (\hat{A}_k,\hat{B}_k) \in H \quad (2.34)$$

However since any limit is necessarily invariant we will at least have:

$$\lim_{k\to\infty} (\hat{A}_k,\hat{B}_k) \in G \quad (2.35)$$

Now, (2.34) will automatically be satisfied if:

$$G \subset H \quad (2.36)$$

DEFINITION 1.2.8

(i) The control objective F on $\Sigma(n)$ is said to have the *potential weak self-tuning* property if $G \subset H$.

(ii) It has the *potential self-tuning* property if $G \subset \bar{H}$.

□

COMMENT 1.2.9 It is important to know whether or not a control objective has the potential weak self-tuning property. For, if it has the property, then every limit of an algorithm based on neutrality and neutral certainty equivalence will generate the right controls.

If F does not have the property (i) or (ii), then an algorithm of the type just mentioned may not be convenient, because there will always be invariant points (=possible limits) which do not generate the right controls.

□

In the next sections we will investigate this problem for two different control objectives. Also we will try to classify all control laws F which have property (i).

REMARK 1.2.10 The problem just encountered is known as the closed-loop

identification problem in adaptive control. It has first been studied in [7] for the adaptive control of finite state Markov chains. The sets G and H were introduced in a slightly different way in [49].

□

2. G AND H FOR POLE ASSIGNMENT: CONFLUX OF IDENTIFICATION AND CONTROL

In this section we will study the sets G and H as defined in the previous section for the case where the control objective is to obtain a prescribed configuration of the closed-loop poles of the system. The material in this section is based on Section 4 of [55].

Given the system:

$$y(k+1) = \theta_0\phi(k) + b_0^0 u(k), \tag{1}$$

where:

$$\theta_0 = (a_0^0,..,a_{n-1}^0,b_1^0,..,b_{n-1}^0), \tag{2}$$

where we assume that the associated A and B polynomials have no common factors, and

$$\phi(k) = (y(k),..,y(k-n+1),u(k-1),..,u(k-n+1))^T, \tag{3}$$

and a monic polynomial

$$\pi \in \mathbb{R}[z] \tag{4}$$

of degree n, find a controller:

$$u(k) = m_1 u(k-1)+..+m_{n-1}u(k-n+1)+n_0 y(k)+..+n_{n-1}y(k-n+1), \tag{5}$$

such that the resulting closed-loop polynomial of the system (1,5) is exactly: $z^{n-1}\pi(z)$. The factor z^{n-1} reflects the need of a dead-beat observer.

The main result of this section is that for this control objective indeed:

$$G \subset H \tag{6}$$

This means that potentially adaptive pole assignment can be based on a neutral and neutral certainty equivalent algorithm. Potentially because the only thing we can conclude from (6) is that the invariant points of such an algorithm have the property that they correspond to desired behavior.

First we will reformulate (1,..,5) in input/state/output form. Then we will state (6) as a theorem (Theorem 2.2). Finally we will give the proof of the theorem, divided into several steps.

Let $(A_0,b_0,c_0) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times 1} \times \mathbb{R}^{1\times n}$ be a minimal realization of (1). I.e.

$$c_0(zI-A_0)^{-1}b_0 = \frac{b_0^0 z^{n-1}+..+b_{n-1}^0}{z^n - a_0^0 z^{n-1} - .. - a_{n-1}^0} \tag{7}$$

Then, for every initial condition $\phi(0)$ of (1), there exists a $x(0) \in \mathbb{R}^n$ such that:

$$x(k+1) = A_0 x(k) + b_0 u(k), \quad x(0) \tag{8}$$

$$y(k) = c_0 x(k), \tag{9}$$

for any input sequence $\{u(k)\}$. We will use the standard observable form:

$$A_0 := \begin{bmatrix} a_0^0 & 1 & \cdots & 0 \\ \vdots & 0 & & \vdots \\ \vdots & \vdots & & 0 \\ \vdots & \vdots & & 1 \\ a_{n-1}^0 & 0 & \cdots & 0 \end{bmatrix} \quad b_0 := \begin{bmatrix} b_0^0 \\ \vdots \\ \vdots \\ \vdots \\ b_{n-1}^0 \end{bmatrix} \quad c_0 := [1\ 0\ ..0] \tag{10}$$

and the dead-beat observer:

$$x(k) = M_0 \phi(k)\,, \tag{11}$$

where $M_0 \in \mathbb{R}^{n \times 2n-1}$ is defined by:

$$M_0 := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & a_1^0 & a_2^0 & & a_{n-1}^0 & b_1^0 & b_2^0 & & b_{n-1}^0 \\ \vdots & a_2^0 & a_3^0 & \cdots & 0 & b_2^0 & b_3^0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & a_{n-1}^0 & & \vdots & \vdots & b_{n-1}^0 & & \vdots \\ 0 & a_{n-1}^0 & 0 & \cdots & 0 & b_{n-1}^0 & 0 & \cdots & 0 \end{bmatrix} \tag{12}$$

We will now describe a neutral and neutral certainty equivalent algorithm in state-space terms. Consider the controlled system (1,5), define:

$$x(k) = M_0 \phi(k) \tag{13}$$

$$f(A,b) = -[0,..,0,1][b \vdots Ab \vdots ... \vdots A^{n-1}b]^{-1} \pi(A) \tag{14}$$

PROPERTY 2.1 The sequence $\{(u(k),y(k))\}$ as defined by (1,5) satisfies:

$$u(k) = f(A_0,b_0)x(k) \tag{15}$$

$$x(k+1) = A_0 x(k) + b_0 u(k), \quad ,x(0) \tag{16}$$

$$y(k) = c_0 x(k) \tag{17}$$

□

The formula (14) is known as Ackermann's formula (see [26]). Property 2.1 follows from realization theory and the fact that for all reachable pairs (A,b) the characteristic polynomial of $(A + bf(A,b))$ is exactly equal to π. There is a one-to-one correspondence between the i/o and i/s/o description if the initial conditions are taken into account. Property 2.1 gives the controller (5) in terms of state-feedback. We can now rewrite the control part of an adaptive pole assignment algorithm in state-space form:
Assume available the k-th estimate $(\hat{a}_0(k),..,\hat{a}_{n-1}(k),\hat{b}_0(k),..,\hat{b}_{n-1}(k))$ of $(a_0^0,..,a_{n-1}^0,b_0^0,..,b_{n-1}^0)$. Define the matrices $\hat{A}_k$, $\hat{b}_k$, $\hat{M}_k$, by replacing the entries of (10) and (12) by their k-th estimates. Assume that $(\hat{A}_k,\hat{b}_k)$ is controllable, and define:

$$f(\hat{A}_k,\hat{b}_k) = -[0,..,0,1][\hat{b}_k \vdots \hat{A}_k\hat{b}_k \vdots ... \vdots \hat{A}_k^{n-1}\hat{b}_k]^{-1}\pi(\hat{A}_k) \tag{18}$$

$$z(k) = \hat{M}_k\phi(k) \tag{19}$$

and finally:

$$u(k) = f(\hat{A}_k,\hat{b}_k)z(k), \tag{20}$$

where f is defined by (14).
We will now determine the set G. Define:

$$x(k+1) = A_0x(k) + b_0u(k), \quad x(0) = M_0\phi(0) \tag{21}$$

$$y(k) = cx(k) \tag{22}$$

$$z(k+1) = Az(k) + bu(k), \qquad z(0) = M\phi(0) \tag{23}$$

$$\hat{y}(k) = cz(k) \tag{24}$$

$$u(k) = f(A,b)z(k), \tag{25}$$

where M is obtained from (12) by dropping the superscripts. Then by definition of G (see II.1.2.5), $(A,b)\in G$ if and only if for all k:

$$y(k) = \hat{y}(k) \tag{26}$$

Hence G is given by:

$$G = \{(A,b)\in E_{ob} \mid \text{for all } k: y(k) = \hat{y}(k)\} \tag{27}$$

where $y(k)$ and $\hat{y}(k)$ are defined by (22,24).

Now H will be the set of those starting values (A,b), such that at every time instant the right input is applied:

$$H = \{(A,b)\in E_{ob} \mid \text{for all } k: f(A,b)z(k) = f(A_0,b_0)x(k)\} \tag{28}$$

where $z(k)$ and $x(k)$ are defined by (21,23). Recall that we wanted to investigate the relative location of G and H. The following theorem gives the answer:

THEOREM 2.2 The sets G and H as defined by (27) and (28) satisfy:

$$G \subset H \tag{29}$$

The implication of Theorem 2.2 is that if we start any neutral, neutral certainty equivalent algorithm in a point belonging to G, then the applied inputs are exactly as the ones we would have applied if we had known the system parameters.
The proof of Theorem 2.2 will be divided into several steps, that are mentioned below. The proofs of the intermediate steps will be postponed to the appendix to this section.

THEOREM 2.3 Let $(A_0,b_0),(A,b)\in E_{re}$ and $\mathcal{V}$ a linear subspace of $\mathbf{R}^{n\times n}$ such that:

(i) For all $v\in\mathcal{V}$: $(A_0+b_0f(A,b))v\in\mathcal{V}$ (30)

(ii) For all $v\in\mathcal{V}$: $(A_0+b_0f(A,b))v = (A+bf(A,b))v$ (31)

Then:

for all $v\in\mathcal{V}$: $f(A,b)v=f(A_0,b_0)v$. (32)

PROOF See the appendix to this section.

THEOREM 2.4 Let $\{(u(k),y(k))\}_{k\in\mathbf{N}}$ be a sequence in $\mathbf{R}^2$ and suppose there exist (A_1,b_1,c_1), (A_2,b_2,c_2), minimal triples of order n, and sequences $\{x(k)^{(1)},x(k)^{(2)}\}$ in $\mathbf{R}^n$, such that for all k:

$$x(k+1)^{(1)} = A_1x(k)^{(1)}+b_1u(k) \tag{33}$$

$$x(k+1)^{(2)} = A_2x(k)^{(2)}+b_2u(k) \tag{34}$$

$$y(k) = c_1x(k)^{(1)} \tag{35}$$

$$y(k) = c_2x(k)^{(2)} \tag{36}$$

Define $\mathcal{X}_i = \text{span}\,\{x(k)^{(i)}\}_{k\in\mathbf{N}}$, and $d_i = \dim(\mathcal{X}_i)$, $i=1,2$.

(i) if $d_1 < n$, then there exists a non-singular matrix S, such that: $Sx(k)^{(1)} = x(k)^{(2)}$.

(ii) $d_1 = d_2$.

(iii) if there exists g_1 such that: $u(k) = g_1x(k)^{(1)}$ and $d_1 = n$, then there exists a non-singular matrix S, such that: $Sx(k)^{(1)} = x(k)^{(2)}$.

PROOF See the appendix to this section.

PROOF OF THEOREM 2.2

Let $(A,b)\in G$ and let $\{x(k),z(k),u(k),y(k),\hat{y}(k)\}$ be defined by (21,..,25), then for all k:

$$y(k) = \hat{y}(k) \tag{37}$$

By Theorem 2.4.iii there exists a non-singular matrix S such that for all k:

$$Sz(k) = x(k) \tag{38}$$

This yields two recursions for $x(k)$:

$$x(k+1) = A_0x(k) + b_0f(A,b)z(k) \tag{39.a}$$

$$= (A_0 + b_0f(A,b))S^{-1}x(k) \tag{39.b}$$

$$= (A_0 + b_0f(SAS^{-1},Sb))x(k) \tag{39.c}$$

and:

$$x(k+1) = Sz(k+1) \tag{40.a}$$

$$= S(A + bf(A,b))z(k) \tag{40.b}$$

$$= (SAS^{-1} + Sbf(SAS^{-1},Sb))x(k) \tag{40.c}$$

Define $(\bar{A},\bar{b}) = (SAS^{-1},Sb)$ and $V = \text{span}\{x(k)\}$, then it follows from the two recursions (39,40) that:

$$(A_0 + b_0 f(\bar{A},\bar{b})) \mid_V = (\bar{A} + \bar{b} f(\bar{A},\bar{b})) \mid_V \tag{41}$$

Also it is not difficult to see that:

$$(A_0 + b_0 f(\bar{A},\bar{b}))V \subset V \tag{42}$$

Hence from Theorem 2.3 it follows that:

$$f(\bar{A},\bar{b}) \mid_V = f(A_0,b_0) \mid_V \tag{43}$$

In particular:

$$f(\bar{A},\bar{b})x(k) = f(A_0,b_0)x(k) \tag{44}$$

and hence:

$$f(A,b)z(k) = f(A_0,b_0)x(k) \tag{45}$$

which means that $(A,b) \in H$.

□

Appendix

Proof of Theorem 2.3 Factorize π as: $\pi(z) = \prod_{i=1}^{n}(z-\lambda_i)$. Define $\Lambda := \{\lambda_1,..,\lambda_n\}$. Suppose that $\Lambda \subset \mathbb{R}$ and that $\lambda_i \neq \lambda_j$ for all $i \neq j$. Let $\mathcal{V}$ be one-dimensional. Then by (30,31) $\mathcal{V}$ is generated by an eigenvector v of $(A+bf(A,b))$ corresponding to let us say $\lambda := \lambda_i$. Hence by (30,31): $(A_0+b_0f(A,b))v = \lambda v$. Suppose (A_0,b_0) is in standard controllable form. Then v is a multiple of $[1,\lambda,..,\lambda^{n-1}]^T$, say $v = [1,\lambda,..,\lambda^{n-1}]^T$. The spectrum of $A_0+b_0f(A_0,b_0)$ is by definition of f equal to Λ. Hence λ is an eigenvalue of $(A_0+b_0f(A_0,b_0))$, and there exists $\tilde{v}$ such that $(A_0+b_0f(A_0,b_0))\tilde{v} = \lambda\tilde{v}$. From the standard controllable form it is easy to see that the only candidates for an eigenvector with eigenvalue λ are multiples of v, hence $\tilde{v} = \mu v$, for some $\mu \neq 0$. Hence $(A_0+b_0f(A_0,b_0))v = (A_0+b_0f(A,b))v$. Since $b_0 \neq 0$, we conclude that $f(A,b)v = f(A_0,b_0)v$.
If $\dim\mathcal{V} > 1$, then $\mathcal{V}$ has a basis of eigenvectors and the above reasoning gives the result. For general Λ the proof goes along the same lines, but then one has to study several different cases. We skip the details.

□

For the proof of Theorem 2.4 we will use the following:
Lemma 2.5 Let (A,b) be reachable, and $x(0) \in \mathbb{R}^n$. Let $\{u(k)\}$ be a sequence of

real numbers. Define:

$$x(k+1) = Ax(k)+bu(k) \quad k = 0,1,2,.. \tag{46}$$

Define $\mathfrak{X}:=$ span $\{x(k)\}_{k\in\mathbf{N}}$, and $d:=\dim(\mathfrak{X})$. If $d<n$, then there exists a $g\in\mathbf{R}^{1\times n}$ such that for all k:

$$u(k) = gx(k) \tag{47}$$

PROOF Suppose (A,b) is in standard controllable form, i.e.

$$A = \begin{bmatrix} 0 & 1 & : & : & 0 \\ : & 0 & : & : & : \\ : & : & : & : & 0 \\ 0 & 0 & : & : & 1 \\ a_1 & a_2 & : & : & a_n \end{bmatrix} \qquad b = \begin{bmatrix} 0 \\ : \\ 0 \\ 1 \end{bmatrix} \tag{48}$$

Define $a\in\mathbf{R}^{1\times n}$ by: $a:=(a_1,..,a_n)$. Define:

$$\tilde{A}:=A-ba \tag{49}$$

$$\tilde{u}(k):=ax(k)+u(k) \tag{50}$$

then:

$$x(k+1) = \tilde{A}x(k)+b\tilde{u}(k) \tag{51}$$

Suppose $x(0)=[x_1(0),..,x_n(0)]^T$, define $H\in\mathbf{R}^{n\times\mathbf{N}}$ by:

$$H := [x(0),x(1),x(2),x(3),...] \tag{52}$$

then:

$$H := \begin{bmatrix} x_1(0) & x_2(0) & x_3(0) & . & . & . & x_n(0) & \tilde{u}(0) & . & . & . \\ : & : & : & & & & \tilde{u}(0) & : & & & \\ : & x_n(0) & \tilde{u}(0) & & & & : & : & & & \\ x_n(0) & \tilde{u}(0) & \tilde{u}(1) & . & . & . & \tilde{u}(n-2) & \tilde{u}(n-1) & . & . & . \end{bmatrix} \tag{53}$$

Since $d<n$, rank$(H)<n$. Now H is a truncated Hankel matrix, hence its rank does not increase if we add the last row, shifted to the left, as the $n+1$-th row. This shifted row is:

$$[\tilde{u}(0),\tilde{u}(1),\tilde{u}(2),\tilde{u}(3),...] \tag{54}$$

Since the rank of the increased matrix is equal to the original one, the last row is a linear combination of the first n rows. In other words, there exist $\tilde{g}_1,..,\tilde{g}_n\in\mathbf{R}$ such that:

$$r_{n+1} = \tilde{g}_1 r_1 + .. + \tilde{g}_n r_n \tag{55}$$

where r_i denotes the i-th row. Define $\tilde{g}\in\mathbf{R}^{1\times n}$ by: $\tilde{g} := [\tilde{g}_1,..,\tilde{g}_n]$. Then for

all k:

$$\tilde{u}(k) = \tilde{g}x(k) \tag{56}$$

Define g by: $g := \tilde{g} - a$. Finally:

$$u(k) = \tilde{u}(k) - ax(k) \tag{57}$$

$$= \tilde{g}x(k) - ax(k) \tag{58}$$

$$= gx(k) \tag{59}$$

□

We will now prove Theorem 2.4:

PROOF OF THEOREM 2.4:
(i) Suppose $d_1 < n$.

$$y(k+i) = c_1[A_1^i x(k)^{(1)} + \sum_{j=0}^{i-1} A_1^j b_1 u(k+i-j-1)] \tag{60}$$

Define:

$$W := \begin{bmatrix} c_1 \\ c_1 A_1 \\ \vdots \\ c_1 A_1^{n-1} \end{bmatrix} \tag{61}$$

then:

$$Wx(k)^{(1)} = \begin{bmatrix} y(k) \\ y(k+1) - c_1 b_1 u(k) \\ \vdots \\ y(k+n-1) - c_1 A_1^{n-2} b_1 u(k) - .. - c_1 b_1 u(k+n-2) \end{bmatrix} \tag{62}$$

From which we conclude that:

$$\begin{bmatrix} W & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k)^{(1)} \\ u(k) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ u(k+n-2) \end{bmatrix} \tag{63}$$

$$= \begin{bmatrix} 1 & & & 0 & . & 0 \\ 0 & . & & -c_1 b_1 & . & \vdots \\ \vdots & & . & \vdots & . & \vdots \\ \vdots & & 1 & -c_1 A_1^{n-2} b_1 & . & -c_1 b_1 \\ \vdots & & & 1 & . & 0 \\ \vdots & & & \vdots & . & \vdots \\ 0 & & & 0 & . & 1 \end{bmatrix} \begin{bmatrix} y(k) \\ \vdots \\ \vdots \\ y(k+n-1) \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix}$$

From which we derive:

$$\begin{bmatrix} x_1^{(1)}(k) \\ \vdots \\ x_n^{(1)}(k) \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} = T_1 \begin{bmatrix} y(k) \\ \vdots \\ y(k+n-1) \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} \tag{64}$$

With T_1 non-singular. In the same way one derives that:

$$\begin{bmatrix} x_1^{(2)}(k) \\ \vdots \\ x_n^{(2)}(k) \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} = T_2 \begin{bmatrix} y(k) \\ \vdots \\ y(k+n-1) \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} \tag{65}$$

Hence:

$$\begin{bmatrix} x(k)^{(1)} \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} = R \begin{bmatrix} x(k)^{(2)} \\ u(k) \\ \vdots \\ u(k+n-2) \end{bmatrix} \tag{66}$$

where $R = T_1 T_2^{-1}$. Now since $u(k+i) = b_2^{\#}(x(k+i+1)^{(2)} - A_2 x(k+i)^{(2)})$, there exist matrices $M_1^{(2)},..,M_n^{(2)} \in \mathbb{R}^{n \times n}$, such that for all k:

$$x(k)^{(1)} = M_1^{(2)} x(k)^{(2)} + .. + M_n^{(2)} x(k+n-1)^{(2)} \tag{67}$$

and similarly:

$$x(k)^{(2)} = M_1^{(1)} x(k)^{(1)} + .. + M_n^{(1)} x_{k+n-1}^{(1)} \tag{68}$$

Since by assumption $d_1 < n$, we conclude from Lemma 2.5 that there exists g_1 such that $u(k) = g_1 x(k)^{(1)}$, hence $x(k+1)^{(1)} = (A_1 + b_1 g_1) x(k)^{(1)}$. Together with (68) this gives that there exists a matrix N_1 such that for all k:

$$x(k)^{(2)} = N_1 x(k)^{(1)} \tag{69}$$

Denote by $\mathcal{X}_2$ the linear span of $x(k)^{(2)}{}_{k \in \mathbb{N}}$, and by d_2 its dimension. From (69) it follows that: $d_2 \leqslant d_1 < n$, hence by Lemma 2.5 there exists g_2 such that for all k: $u(k) = g_2 x(k)^{(2)}$. As above we conclude that there exists a matrix N_2 such that for all k:

$$x(k)^{(1)} = N_2 x(k)^{(2)} \tag{70}$$

Finally (69) together with (70) gives the statement.
(ii) This follows immediately form part (i).

(iii) Suppose $u(k) = g_1 x(k)^{(1)}$, then just as in the proof of part (i) (68), we conclude that:

$$x(k)^{(2)} = N_1 x(k)^{(1)} \tag{71}$$

Since $d_1 = d_2 = n$, it follows that N_1 is non-singular.

□

3. G AND H FOR LQ CONTROL: CONFLICT BETWEEN IDENTIFICATION AND CONTROL

In this section we will study the sets G and H for the case where the control objective is described by a quadratic cost criterion. It will turn out that the desirable property:

$$G \subset H \tag{1}$$

does not hold for this control objective. In fact we will show that $G \cap H$ is a negligible subset of G. As in the previous section we formulate this somewhat disappointing result in state space terms. The results in this section are refinements of those obtained in [49,51], and can also be found in [54]. Theorem 3.10.i was also proven in [41] for the first order case.

Given the following system:

$$x(k+1) = Ax(k) + bu(k) \tag{2}$$

$$y(k) = cx(k), \tag{3}$$

where $(A,b,c) \in \mathbf{R}^{n \times n} \times \mathbf{R}^{n \times 1} \times \mathbf{R}^{1 \times n}$ is a minimal triple.
Control objective: find a causal controller such that the following expression is minimized:

$$J = \sum_{k=0}^{\infty} (y(k)^2 + r\, u(k)^2) \qquad r > 0 \tag{4}$$

The solution of this problem is well known (see [38]) and is given by:

$$u(k) = f(A,b,c)x(k), \tag{5}$$

where:

$$f(A,b,c) = -(b^T K b + r)^{-1} b^T K A, \tag{6}$$

and K is the unique symmetric positive definite solution of the Algebraic Riccati equation:

$$K - A^T K A + A^T K b (b^T K b + r)^{-1} b^T K A - c^T c = 0, \tag{7}$$

Moreover, the optimal value of J is given by:

$$x(0)^T K x(0), \tag{8}$$

where $x(0)$ is the initial state of the system. Note that the control law not only depends on (A,b) but also on c.
As in the pole assignment case we will consider the standard observable form:

$$A_0 := \begin{bmatrix} a_0^0 & 1 & \cdots & 0 \\ \vdots & 0 & & \vdots \\ \vdots & \vdots & & 0 \\ \vdots & \vdots & & 1 \\ a_{n-1}^0 & 0 & \cdots & 0 \end{bmatrix} \qquad b_0 := \begin{bmatrix} b_0^0 \\ \vdots \\ \vdots \\ \vdots \\ b_{n-1}^0 \end{bmatrix} \qquad c_0 := [1\ 0\ ..0] \tag{9}$$

Recall the definition of E_{ob}:

$$E_{ob} := \{(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid (A,b) \text{ reachable}, \quad (c_0,A) \text{ in standard observable form}\} \tag{10}$$

and define:

$$E_{ns} := \{(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid (A,b,c_0) \text{ minimal}, A \text{ non-singular}\} \tag{11}$$

REMARK We restrict ourselves to non-singular A-matrices for technical reasons only.

Define:

$$P := \{K\in\mathbf{R}^{n\times n} \mid K=K^T > 0\} \tag{12}$$

Throughout this section let (A_0,b_0) be fixed. Also let there be given some fixed non-zero non-minimal initial state $\phi(0)\in\mathbf{R}^{2n-1}$. For $(A,b)\in E_{ob}$, define the sequences $\{x(k)\}$, $\{z(k)\}$, $\{u(k)\}$, $\{y(k)\}$ and $\{\hat{y}(k)\}$ as follows:
Firstly:

$$x(0) := M_0\phi(0) \qquad z(0) := M\phi(0) \tag{13}$$

where the matrices M_0 and M are derived as in Section II.2 (12). from (A_0,b_0) and (A,b) respectively. Assume that $x(0)$ is non-zero.
Secondly:

$$x(k+1) = A_0x(k) + b_0u(k) \qquad z(k+1) = Az(k) + bu(k) \tag{14}$$

$$y(k) = c_0x(k) \qquad \hat{y}(k) = c_0z(k) \tag{15}$$

where:

$$u(k) := f(A,b,c_0)z(k) \tag{16}$$

The sets G and H, depending on $\phi(0)$, are then:

$$G := \{(A,b)\in E_{ob} \mid \text{for all } k: \hat{y}(k) = y(k)\} \tag{17}$$

$$H := \{(A,b)\in E_{ob} \mid \text{for all } k: f(A,b,c_0)z(k) = f(A_0,b_0,c_0)x(k)\} \tag{18}$$

We will first state the main result of this section:

CLAIM 3.1 $G\cap H$ is a negligible subset of G.

□

Claim 3.1 is of course not a mathematical statement. We will have to do some work before we can formalize the contents of the claim. Intuitively it means that within the set of invariant points of an adaptive algorithm only a negligible part consists of points that correspond to the desired (optimal) control law. This is in contrast to the pole assignment problem where *every* invariant point

corresponds to the desired control law. In this sense adaptive LQ control is more difficult. In pole assignment the only concern is convergence of the parameter estimates, every limit point will be invariant and will hence produce the right controls. In LQ control we have to avoid that the estimates converge to sub-optimal invariant points. That means that we have to develop an algorithm for which those invariant points can never be attractive!
It is difficult to get a direct grip on the sets G and $G \cap H$, therefore we will introduce two other sets, G_0 and H_0, which are easier to analyze and which are closely related to G and H. In order to relate G and H with G_0 and H_0 we will also define a subset $\tilde{G}$ of G and a subset $\tilde{G}_0$ of $\tilde{G}$.

DEFINITION 3.2

$$G_0 := \{(A,b) \in E_{ns} \mid A_0 + b_0 f(A,b,c_0) = A + bf(A,b,c_0)\} \tag{19}$$

$$H_0 := \{(A,b) \in E_{ns} \mid f(A,b,c_0) = f(A_0,b_0,c_0)\} \tag{20}$$

□

DEFINITION 3.3 For every $(A,b) \in E_{ob}$, define:

$$\mathcal{V}(A,b) := \text{span}\,\{x(k)\}_{k \in \mathbf{N}} \tag{21}$$

where $\{x(k)\}$ and $\{z(k)\}$ are defined by (13,14,15,16).

□

DEFINITION 3.4

$$\tilde{G}_0 := \{(A,b) \in G_0 \mid (A_0 + b_0 f(A,b,c_0), x(0)) \text{ is reachable}\} \tag{22}$$

$$\tilde{G} := \{(A,b) \in G \mid \mathcal{V}(A,b) = \mathbf{R}^n\} \tag{23}$$

□

THEOREM 3.5 $\tilde{G}$ and $\tilde{G}_0$ are C^ω diffeomorphic.

PROOF See the appendix to this section.

THEOREM 3.6
(i) $\tilde{G}_0$ is open and dense in G_0.
(ii) $\tilde{G}$ is open and dense in G.

PROOF See the appendix to this section.

THEOREM 3.7 G_0 is an embedded analytic manifold of dimension n. (a definition of embedded manifold can be found in the appendix).

PROOF See the appendix to this section.

LEMMA 3.8 For all $(A,b)\in G\cap H$: $\mathcal{V}(A,b) = \mathcal{V}(A_0,b_0)$.

PROOF This is immediate from the fact that $(A,b)\in H$ implies: for all k: $f(A,b,c_0)z(k) = f(A_0,b_0,c_0)x(k)$.

□

DEFINITION 3.9 $\mathcal{V}_0 := \mathcal{V}(A_0,b_0)$, the linear span of the the optimal state trajectory.

□

THEOREM 3.10
(i) If $\dim(\mathcal{V}_0) = n$, then $G\cap H = \{(A_0,b_0)\}$.
(ii) If $\dim(\mathcal{V}_0) < n$, then $G\cap H$ is contained in $G\setminus\tilde{G}$.

PROOF See the appendix to this section.

COMMENT Let us now discuss some of the consequences of Theorem 3.10. First of all it is the mathematical formalization of Claim 3.1. For suppose $\phi(0)$ is such that $\dim(V_0) = n$. Then from Theorem 3.10.i we know that $G\cap H = \{A_0,b_0)\}$, a singleton. Now G contains an open and dense subset that is diffeomorphic to an open and dense subset of an n −dimensional manifold (by Theorems 3.5, 3.6 and 3.7). In that sense $G\cap H$, being a singleton, is a negligible subset of G. In the other case, where $\phi(0)$ is such that $\dim(V_0)<n$, $G\cap H$ is contained in $G\setminus\tilde{G}$. In other words $G\cap H$ is contained in the boundary of a set that is diffeomorphic to an open and dense subset of an n-dimensional manifold. Since the boundary of an n-dimensional manifold has a strictly smaller dimension again $G\cap H$ is a negligible subset of G.
Now suppose that we want to use a neutral and neutral certainty equivalent adaptive algorithm for LQ control. Then almost every invariant point of the algorithm will result in sub-optimal behavior. This means that almost every invariant point must not be stable, i.e. must not be a possible limit of the algorithm. This seems to be very difficult, if not impossible.

□

EXAMPLE 3.11
In Figure 3.1, we have depicted the sets G and H for a first order system. The parameter values were: $(a_0,b_0) = (1,1)$, $r = 2$. The first part of the picture shows the branches of G and H in the right half plane, the second part shows the left half plane branches of G and H. The picture illustrates that $G\cap H = \{a_0,b_0)\}$, as was already predicted by Theorem 3.10.i.

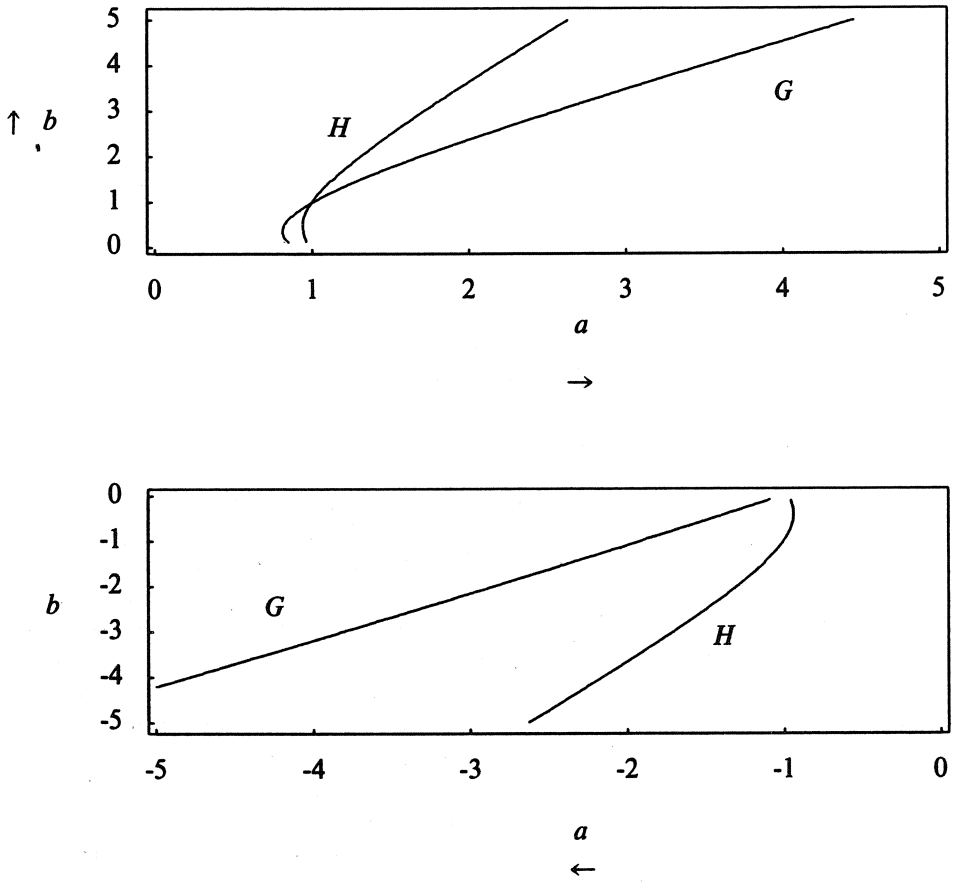


FIGURE 3.1 G and H for a first order system.

□

APPENDIX

Let us first recall some preliminaries.

DEFINITION 3.12 (see [59]). Let $X \subseteq \mathbf{R}^n$. X is an embedded m-dimensional C^k-manifold, if $\forall \tilde{x} \in X$, $\exists U \subseteq \mathbf{R}^n$, open, with $\tilde{x} \in U$ and a C^k-function $L: U \to \mathbf{R}^{n-m}$ such that:

(i) $L(\tilde{x}) = 0$

(ii) $L^{-1}(\{0\}) = X \cap U$

(iii) The derivative of L with respect to x, evaluated in $\tilde{x}$, has full rank.

□

LEMMA 3.13 Let $M_i, N_i \in \mathbf{R}^{p_i \times q_i}$, $i = 1,2$. Define $[(M_1, M_2), (N_1, N_2)]$ by:

$[(M_1,M_2),(N_1,N_2)] := Tr(M_1N_1^T) + Tr(M_2N_2^T)$. This defines an inner-product on $\mathbb{R}^{p_1\times q_1}\times\mathbb{R}^{p_2\times q_2}$. ($Tr$ denotes the trace of a matrix)

□

LEMMA 3.14 Let $(X,[.,.]_X)$ and $(Y,[.,.]_Y)$ be finite-dimensional inner-product spaces, and $F:X\to Y$ a linear map.

(i) There exists one and only one linear map $F^*:Y\to X$ such that for all $x\in X$ and for all $y\in Y$ $[Fx,y]_Y=[x,F^*y]_X$. F^* is called the adjoint operator of F with respect to $[.,.]_X$ and $[.,.]_Y$.

(ii) F is surjective iff F^* is injective.

PROOF See [18].

□

LEMMA 3.15 Let $M,N\in\mathbb{R}^{p\times p}$, let $\Lambda : \mathbb{R}^{p\times p}\to\mathbb{R}^{p\times p}$ be defined by: $\Lambda(X)=X-M^TXN$, then:

$$\mathrm{Spec}(\Lambda)=1-\mathrm{Spec}(M)\times\mathrm{Spec}(N) = \{\ 1-\lambda\mu \mid \lambda\in\mathrm{Spec}(M),\ \mu\in\mathrm{Spec}(N)\ \}$$

("Spec" denotes spectrum)

PROOF See [39].

□

LEMMA 3.16 For every minimal triple (A,b,c) one has: $Ker\ (A+bf(A,b,c)) = Ker\ A$

PROOF Suppose $x_0\in Ker\ (A+bf(A,b,c))$, then $x_k=0$ and $u_k=0$, for all $k\geqslant 1$. Hence:

$$\begin{aligned}
x_0^TKx_0 &= x_0^Tc^Tcx_0+u_0^Tru_0 && \text{by (4) and (8)}\\
&= x_0^T(c^Tc+f(A,b,c)^Trf(A,b,c))x_0 && \text{by (5)}\\
&= x_0^T(K - A^TK(A+bf(A,b,c))+ f(A,b,c)^Trf(A,b,c))x_0 && \text{by (6) and (7)}\\
&= x_0^TKx_0 + x_0^Tf(A,b,c)^Trf(A,b,c)x_0
\end{aligned}$$

This implies that: $x_0^Tf(A,b,c)^Trf(A,b,c)x_0 = 0$ and thus that $f(A,b,c)x_0=0$. Together with $(A+bf(A,b,c))x_0 = 0$ this gives $Ax_0 = 0$.
Suppose on the other hand that $Ax_0 = 0$, then also $f(A,b,c)x_0 = 0$ (by (6)) and thus $(A+bf(A,b,c))x_0 = 0$.

□

COROLLARY 3.17 For all $(A,b,c)\in E_{ns}$, $A + bf(A,b,c)$ is non-singular.

PROOF This follows from Lemma 3.16 and from the fact that by definition of E_{ns}, $(A,b,c)\in E_{ns}$, implies that A is non-singular.

□

LEMMA 3.18 For all $(\tilde{A},\tilde{b},c_0)\in G_0$ there exists an $\tilde{\epsilon} > 0$ such that $\forall \bar{f}$ with: $\|\bar{f}-f(\tilde{A},\tilde{b},c_0)\| < \tilde{\epsilon}$, there exists $(\bar{A},\bar{b},c_0)\in E$ such that:
(i) $f(\bar{A},\bar{b},c_0) = \bar{f}$
(ii) $A_0 + b_0 f(\bar{A},\bar{b},c_0) = \bar{A} + \bar{b}f(\bar{A},\bar{b},c_0)$
(iii) $(\bar{A},\bar{b})$ depends continuously on $\bar{f}$.

PROOF Choose $(\tilde{A},\tilde{b},c_0)\in G_0$. We will prove that the map f subject to the constraint that $(A,b,c_0)\in G_0$, is locally surjective. To this end it is enough to prove that, locally, (A,b) can be written as a continuous function of $\bar{f}$. Define:

$$L:\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}\times\mathbf{R}^{\frac{1}{2}n(n+1)}\times\mathbf{R}^{1\times n}\to\mathbf{R}^{n\times n}\times\mathbf{R}^{1\times n}\mathbf{R}^{\frac{1}{2}n(n+1)}$$

by:

$$L(A,b,K,\bar{f}) := (L_1(A,b,K,\bar{f}),L_2(A,b,K,\bar{f}),L_3(A,b,K,\bar{f})) \tag{24}$$

where:

$$L_1(A,b,K,\bar{f}) := A_0 + b_0\bar{f} - A - b\bar{f} \tag{25}$$

$$L_2(A,b,K,\bar{f}) := b^TKb\bar{f} + r\bar{f} + b^TKA \tag{26}$$

$$L_3(A,b,K,\bar{f}) := K - A^TKA + A^TKb(b^TKb + r)^{-1}b^TKA - c_0^Tc_0 \tag{27}$$

By definition of L it follows that: $L(\tilde{A},\tilde{b},\tilde{K},\tilde{f}) = (0,0,0)$, where $\tilde{K}$ is the positive definite solution of the algebraic Riccati equation and $\tilde{f} = f(\tilde{A},\tilde{b},c_0)$. We will now calculate the derivative of L with respect to (A,b,K) evaluated in $(\tilde{A},\tilde{K},\tilde{b},\tilde{f})$:

$$\Lambda_1(\Delta A,\Delta b,\Delta K) = -\Delta A - \Delta b\tilde{f} \tag{28}$$

$$\Lambda_2(\Delta A,\Delta b,\Delta K) = \Delta b^T\tilde{K}\tilde{b}\tilde{f} + \tilde{b}^T\Delta K\tilde{b}\tilde{f} + \tilde{b}^T\tilde{K}\Delta b\tilde{f} + \Delta b^T\tilde{K}\tilde{A} + \tilde{b}^T\Delta K\tilde{A} + \tilde{b}^T\tilde{K}\Delta A \tag{29}$$

$$\begin{aligned}\Lambda_3(\Delta A,\Delta b,\Delta K) = {} & \Delta K - (\tilde{A} + \tilde{b}\tilde{f})^T\Delta K(\tilde{A} + \tilde{b}\tilde{f}) - \\ & \Delta A^T\tilde{K}(\tilde{A} + \tilde{b}\tilde{f}) - (\tilde{A} + \tilde{b}\tilde{f})^T\tilde{K}\Delta A - \\ & \tilde{A}^T\tilde{K}\Delta b\tilde{f} - \tilde{f}^T\Delta b^T\tilde{K}\tilde{A} - \tilde{f}^T(\Delta^T\tilde{K}\tilde{b} + \tilde{b}^T\tilde{K}\Delta b)\tilde{f}\end{aligned} \tag{30}$$

To show that Λ has full rank it is sufficient to show that it is injective:
Put:

$$E_1\colon \Lambda_1(\Delta A,\Delta b,\Delta K) = 0 \tag{31}$$

$$E_2\colon \Lambda_2(\Delta A,\Delta b,\Delta K) = 0 \tag{32}$$

$$E_3\colon \Lambda_3(\Delta A,\Delta b,\Delta K) = 0 \tag{33}$$

$E_3 + (\tilde{A} + \tilde{b}\tilde{f})^T\tilde{K}E_1 + E_1^T\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})$ gives:

$$\Delta K - (\tilde{A} + \tilde{b}\tilde{f})^T\Delta K(\tilde{A} + \tilde{b}\tilde{f}) = 0 \tag{34}$$

By Lemma 3.15 and the strict stability of $(\tilde{A} + \tilde{b}\tilde{f})$ it follows that: $\Delta K = 0$. Substituting this in E_2 gives:

$$E_2' \; \Delta b^T \tilde{K} \tilde{b} \tilde{f} + \tilde{b}^T \tilde{K} \Delta b \tilde{f} + \Delta b^T \tilde{K} \tilde{A} + \tilde{b}^T \tilde{K} \Delta A = 0 \tag{35}$$

$E_2' - b^T \tilde{K} E_1$ gives:

$$\Delta b^T \tilde{K} (\tilde{A} + \tilde{b}\tilde{f}) = 0 \tag{36}$$

By Corollary 3.17 $(\tilde{A} + \tilde{b}\tilde{f})$ is non-singular, and hence $\Delta b = 0$. Finally substituting this in E_1, gives $\Delta A = 0$.
Now, the implicit function theorem yields the existence of an open neighborhood of $\tilde{f}$ and a C^ω function defined on that open set to (A,b,K). This completes the proof.

□

PROOF OF THEOREM 3.5 Define:

$\phi : E_{ns} \to E_{ob}$

by:

$$\phi(A,b) = (SAS^{-1}, Sb) \tag{37}$$

where $S \in Gl(n)$ is the unique non-singular matrix which transforms (A,c_0) into standard observable form. Since S depends C^ω on A, it follows that ϕ is C^ω.
Now, let $(A,b) \in \tilde{G}_0$. Define $x(0) := M_0 \phi(0)$, and $x(k)$ by:

$$x(k+1) = (A_0 + b_0 f(A,b,c_0))x(k) \tag{38}$$

Define:

$$z(k) := Sx(k) \tag{39}$$

then:

$$z(k+1) = Sx(k+1) = S(A_0 + b_0 f(A,b,c_0))x(k) \tag{40}$$

$$= S(A + bf(A,b,c_0))x(k) \tag{41}$$

$$= S(A + bf(A,b,c_0))S^{-1} z(k) \tag{42}$$

$$= (SAS^{-1} + Sbf(SAS^{-1}, Sb, c_0))z(k) \tag{43}$$

From the standard observable form and the recursion for $z(k)$ it follows that $z(k) = M\phi(k)$, where M is derived from (SAS^{-1}, Sb) as in Section II.2 (12). In particular it follows that $z(0) = M\phi(0)$. Finally, $y(k) = c_0 x(k) = c_0 S^{-1} z(k) = c_0 z(k) = \hat{y}(k)$. We conclude that $\phi(A,b) \in G$. Moreover since by definition of G_0, span $\{x(k)\} = \mathbb{R}^n$, it follows that $\phi(A,b) \in \tilde{G}$.
Define $\psi : \tilde{G} \to \tilde{G}_0$ as follows: Choose $(A,b) \in \tilde{G}$. By Theorem II.2.4 there exists $T \in Gl(n)$ such that for all k: $x(k) = Tz(k)$, and since $V(A,b) = \mathbb{R}^n$, this T is unique. Moreover from the proof of Theorem II.2.4 it follows easily that T depends C^ω on (A,b). Define:

$$\psi(A,b) := (TAT^{-1},Tb) \tag{44}$$

Since $y(k) = \hat{y}(k)$, it follows from $x(k) = Tz(k)$ and $V(A,b) = \mathbf{R}^n$ that $c_0T^{-1} = c_0$. Now:

$$x(k+1) = A_0x(k) + b_0f(A,b,c_0)z(k) \tag{45}$$

$$= (A_0 + b_0f(TAT^{-1},Tb,c_0T^{-1}))x(k) \tag{46}$$

and also:

$$x(k+1) = Tz(k+1) = T(A + bf(A,b,c_0))z(k) \tag{47}$$

$$= T(A + bf(A,b,c_0))T^{-1}x(k) \tag{48}$$

$$= (TAT^{-1} + Tbf(TAT^{-1},Tb,c_0))x(k) \tag{49}$$

Since $V(A,b) = \mathbf{R}^n$, it follows that:

$$A_0 + b_0f(TAT^{-1},Tb,c_0) = TAT^{-1} + Tbf(TAT^{-1},Tb,c_0) \tag{50}$$

hence $\psi(A,b)\in\tilde{G}_0$.
Finally, from the uniqueness of the matrices S and T one can easily check that:

$$\psi.\phi = id_{\tilde{G}} \tag{51}$$

$$\phi.\psi = id_{\tilde{G}_0} \tag{52}$$

This finishes the proof.

□

PROOF OF THEOREM 3.6
(i) Choose $(\tilde{A},\tilde{b})\in G_0$ and suppose that $(A_0 + b_0f(\tilde{A},\tilde{b}c_0),x(0))$ is non-reachable. Choose an open neighborhood W of $(\tilde{A},\tilde{b})$ in G_0. By Lemma 3.18 there exists an open neighborhood V of $f(\tilde{A},\tilde{b},c_0)$, such that for every $\bar{f}\in V$, the unique pair $(\bar{A},\bar{b})\in G_0$ with $f(\bar{A},\bar{b},c_0) = \bar{f}$, has the property that $(\bar{A},\bar{b})\in W$. Choose $\bar{f}\in V$ such that $(A_0 + b_0\bar{f},x(0))$ is reachable and it follows that $\tilde{G}_0$ is dense in G_0. Since $\tilde{G}$ is the complement of the zero-set of a continuous function it follows that $\tilde{G}_0$ is also open in G_0.
(ii) Choose $(A,b)\in G$ and suppose that $V(A,b)\neq\mathbf{R}^n$. Choose an open neighborhood W of (A,b) in G. From the proof of Theorem 3.5 it follows that there exists a non-singular matrix S such that $(SAS^{-1},Sb)\in G_0$. The function ϕ as defined by (37) is continuous and hence there exists an open neighborhood V of (SAS^{-1},Sb) in G_0) such that $\phi(V)\subset W$. By part (i) we know that $V\cap\tilde{G}_0\neq\emptyset$. It is not difficult to check that this implies that $W\cap\tilde{G}\neq\emptyset$, which shows that $\tilde{G}$ is dense in G. Also, since $\tilde{G}$ is the complement of the zero-set of a collection of polynomials, $\tilde{G}$ is open in G.

□

LEMMA 3.19 There exists a C^ω-function $K:E\to P$ such that $K(A,b,c)$ satisfies

(7) , for all $(A,b,c)\in E$.

PROOF A proof for the continuous time case can be found in [11], we give the proof for the discrete time case for the sake of completeness.
The implicit function theorem will be used to get the result. Define $L : E\times P\rightarrow \mathbf{R}^{\frac{1}{2}n(n+1)}$ by:

$$L(A,b,c,K) := K-A^TKA+A^TKb(b^TKb+r)^{-1}b^TKA-c^Tc \tag{53}$$

Since $r>0$ L is C^ω. Note that $\forall\ (\tilde{A},\tilde{b},\tilde{c})\in E$ $L(\tilde{A},\tilde{b},\tilde{c},\tilde{K})=0$ where $\tilde{K}$ is the solution within P of (7). We will now calculate the derivative of L with respect to K, evaluated in such a triple $(\tilde{A},\tilde{b},\tilde{c},\tilde{K})$. This will be a linear map $\Lambda : \mathbf{R}^{\frac{1}{2}n(n+1)}\rightarrow \mathbf{R}^{\frac{1}{2}n(n+1)}$ of which the action on $\Delta K\in \mathbf{R}^{\frac{1}{2}n(n+1)}$ can be found by the following calculation: (We will use the private notation $=^l$ to denote equality as far as linear terms in the "Δ variable(s)" are concerned).

$$\Lambda(\Delta K) =^l L(\tilde{A},\tilde{b},\tilde{c},\tilde{K}+\Delta K) \tag{54}$$

$$=^l \tilde{K}+\Delta K-\tilde{A}^T(\tilde{K}+\Delta K)\tilde{A} \tag{55}$$
$$+\tilde{A}^T(\tilde{K}+\Delta K)\tilde{b}(\tilde{b}^T(\tilde{K}+\Delta K)\tilde{b}+r)^{-1}\tilde{b}^T(\tilde{K}+\Delta K)\tilde{A}-c^Tc$$

$$=^l \Delta K-\tilde{A}^T\Delta K\tilde{A}+\tilde{A}^T(\tilde{K}+\Delta K)\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1} \tag{56}$$
$$(\sum_{j=0}^{\infty}(-1)^j[\tilde{b}^T\Delta K\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1}]^j)\tilde{b}^T(\tilde{K}+\Delta K)\tilde{A}$$

$$=^l \Delta K-\tilde{A}^T\Delta K\tilde{A}+\tilde{A}^T\Delta K\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1}\tilde{b}^T\tilde{K}\tilde{A} \tag{57}$$
$$+\tilde{A}^T\tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1}\tilde{b}^T\Delta K\tilde{A}$$
$$-\tilde{A}^T\tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1}\tilde{b}^T\Delta K\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b}+r)^{-1}\tilde{b}^T\tilde{K}\tilde{A}$$

$$=^l \Delta K-\{(\tilde{A}+\tilde{b}f(\tilde{A},\tilde{b},\tilde{c}))^T\Delta K(\tilde{A}+\tilde{b}f(\tilde{A},\tilde{b},\tilde{c}))\} \tag{58}$$

Since $\tilde{A}+\tilde{b}f(\tilde{A},\tilde{b},\tilde{c})$ is strictly stable, (see [38]) it follows by Lemma 3.15 that $0\notin$Spec(Λ), hence Λ is non-singular.
Now the implicit function theorem yields the existence of the function K in a neighborhood of $(\tilde{A},\tilde{b},\tilde{c})$. Since $(\tilde{A},\tilde{b},\tilde{c})$ was arbitrary and the solution of (7) is unique (within P), K is well defined on E.

□

COROLLARY 3.20 f is a C^ω- function on E.

PROOF This is immediate from the facts that f is a C^ω-function of (A,b,c,K) and Lemma 3.19.

□

PROOF OF THEOREM 3.7

By Theorem 3.6 $\tilde{G}_0$ is nonempty. Define $\tilde{G}_0' \subset \mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}\times\mathbf{R}^{n(n+1)/2}$ by:

$$\tilde{G}_0' := \{(A,b,K) \mid (A,b)\in\tilde{G}_0\ ,\ K = K(A,b)\in P\ \}$$

Define $L: E_{ns}\times P \to \mathbf{R}^{n(n+1)/2}\times\mathbf{R}^{n\times n}$ by:

$$L(A,b,K) = (L_1(A,b,K), L_2(A,b,K)), \tag{59}$$

where:

$$L_1(A,b,K) = K - A^TKA + A^TKb(b^TKb+r)^{-1}b^TKA - c_0^Tc_0 \tag{60}$$

$$L_2(A,b,K) = (A - b(b^TKb+r)^{-1}b^TKA) - (A_0 - b_0(b^TKb+r)^{-1}b^TKA) \tag{61}$$

Note that $(\tilde{A},\tilde{b},\tilde{K})\in\tilde{G}_0'$ if and only if $L(\tilde{A},\tilde{b},\tilde{K}) = (0,0)$, and that L is C^ω. Fix a triple $(\tilde{A},\tilde{b},\tilde{K})\in\tilde{G}_0'$, and let $\tilde{f} = f(\tilde{A},\tilde{b},c_0)$. We will show that the derivative of L with respect to (A,b,K), evaluated in $(\tilde{A},\tilde{b},\tilde{K})$ has full rank. The derivative of L evaluated in $(\tilde{A},\tilde{b},\tilde{K})$ is a linear map Λ given by:

$$\begin{aligned}\Lambda_1(\Delta A,\Delta b,\Delta K) = {} & \Delta K - (\tilde{A} + \tilde{b}\tilde{f})^T\Delta K(\tilde{A} + \tilde{b}\tilde{f}) - \Delta A^T\tilde{K}(\tilde{A} + \tilde{b}\tilde{f}) \\ & - (\tilde{A} + \tilde{b}\tilde{f})^T\tilde{K}\Delta A - \tilde{A}^T\tilde{K}\Delta b\tilde{f} - \tilde{f}^T\Delta b^T\tilde{K}\tilde{A} \\ & - \tilde{f}^T(\Delta b^T\tilde{K}\tilde{b} + \tilde{b}^T\tilde{K}\Delta b)\tilde{f}\end{aligned} \tag{62}$$

$$\begin{aligned}\Lambda_2(\Delta A,\Delta b,\Delta K) = {} & \Delta A - \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}(\Delta b^T\tilde{K}\tilde{A} + \tilde{b}^T\Delta K\tilde{A} \\ & + \tilde{b}^T\tilde{K}\Delta A) - \Delta b(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T\tilde{K}\tilde{A} \\ & - \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}(\Delta b^T\tilde{K}\tilde{b} + \tilde{b}^T\Delta K\tilde{b} + \tilde{b}^T\tilde{K}\Delta b)\tilde{f} \\ & + b_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}(\Delta b^T\tilde{K}\tilde{A} + \tilde{b}^T\Delta K\tilde{A} + \tilde{b}^T\tilde{K}\Delta A) \\ & + b_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}(\Delta b^T\tilde{K}\tilde{b} + \tilde{b}^T\Delta K\tilde{b} + \tilde{b}^T\tilde{K}\Delta b)\tilde{f}\end{aligned} \tag{63}$$

Let $(M,N)\in\mathbf{R}^{n(n+1)/2}\times\mathbf{R}^{n\times n}$. We will calculate the inner-product of $\Lambda(\Delta A,\Delta b,\Delta K)$ with (M,N) in order to establish a formula for its adjoint:

$$[\Lambda(\Delta A,\Delta b,\Delta K),(M,N)] = \tag{64}$$

$$Tr(\Lambda_1(\Delta A,\Delta b,\Delta K)M) + Tr(\Lambda_2(\Delta A,\Delta b,\Delta K)N^T)$$

$$\begin{aligned}= {} & Tr(\Delta K[M - (\tilde{A} + \tilde{b}\tilde{f})M(\tilde{A} + \tilde{b}\tilde{f})^T \\ & \quad - \tilde{A}N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T - \tilde{b}\tilde{f}N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T \\ & \quad + \tilde{A}N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T + \tilde{b}\tilde{f}N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T]) \\ & + Tr(\Delta A[-2M(\tilde{A} + \tilde{b}\tilde{f})^T\tilde{K} + N^T \\ & \quad - N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T\tilde{K} + N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T\tilde{K}]) \\ & + Tr(\Delta b[-2\tilde{f}M\tilde{A}^T\tilde{K} - 2\tilde{f}M\tilde{f}^T\tilde{b}^T\tilde{K} - (\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{A}^TK \\ & \quad + \tilde{f}N^T - (\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{f}^T\tilde{b}^T\tilde{K}\end{aligned} \tag{65}$$

$$-\tilde{f}N^T\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^T\tilde{K} + (\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{A}^T\tilde{K}$$
$$+(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{f}^T\tilde{b}K + \tilde{f}N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TK])$$

Hence by Lemma 3.14: $\Lambda^* = (\Lambda_1^*,\Lambda_2^*,\Lambda_3^*,\Lambda_4^*)$, where:

$$\Lambda_1^*(M,N) = M - (\tilde{A} + \tilde{b}\tilde{f})M(\tilde{A} + \tilde{b}\tilde{f})^T \tag{66}$$
$$- \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{A}^T - \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{f}^T\tilde{b}^T$$
$$+ \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{A}^T + \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{f}^T\tilde{b}^T$$

$$\Lambda_2^*(M,N) = -2\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})M + N \tag{67}$$
$$- \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN + \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN$$

$$\Lambda_3^*(M,N) = -2\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})M\tilde{f}^T + N\tilde{f}^T \tag{68}$$
$$- \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{f}^T$$
$$- \tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}$$
$$+ \tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}$$

To show that Λ^* is injective, we put $\Lambda^*(M,N) = 0$. Which gives the following equations:

$$E_1\colon M - (\tilde{A} + \tilde{b}\tilde{f})M(\tilde{A} + \tilde{b}\tilde{f})^T - \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{A}^T - \tag{69}$$
$$\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{f}^T\tilde{b}^T + \tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{A}^T +$$
$$\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN\tilde{f}^T\tilde{b}^T = 0$$

$$E_2\colon -2\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})M + N - \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN \tag{70}$$
$$+ \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}b_0^TN = 0$$

$$E_3\colon -2\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})M\tilde{f}^T + N\tilde{f}^T - \tilde{K}\tilde{b}(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1}\tilde{b}^TN\tilde{f}^T - \tag{71}$$
$$\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} + \tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} = 0$$

$E_3 - E_2\tilde{f}^T$ gives:

$$- \tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} \tag{72}$$
$$+ \tilde{K}(\tilde{A} + \tilde{b}\tilde{f})N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} = 0$$

Substituting (72) in E_1 gives:

$$M - (\tilde{A} + \tilde{b}\tilde{f})M(\tilde{A} + \tilde{b}\tilde{f})^T = 0 \tag{73}$$

By Lemma 3.15 we conclude that $M = 0$. Since $\tilde{K}(\tilde{A} + \tilde{b}\tilde{f})$ is non-singular, (72) implies that: $-N^Tb(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} + N^Tb_0(\tilde{b}^T\tilde{K}\tilde{b} + r)^{-1} = 0$. Substituting this and $M = 0$ in E_2 gives:

$$N = 0 \tag{74}$$

This shows that $\tilde{G}_0'$ is an n-dimensional manifold in $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}\times\mathbf{R}^{n(n+1)/2}$.

Since $\tilde{K}$ depends C^ω on $(\tilde{A},\tilde{b})$, it follows that $\tilde{G}_0$ is an n-dimensional C^ω manifold in $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}\times\mathbf{R}^{1\times n}$. This completes the proof.

□

LEMMA 3.21 Let $(A,b)\in G_0$, denote the solution of (7) by K and let K_0 be the solution of (7) with (A,b,c_0) replaced by (A_0,b_0,c_0). Then $K\geqslant K_0$.

PROOF Let $x_0\in\mathbf{R}^n$, the optimal cost for the system (A,b,c_0), starting in x_0 is $x_0^TKx_0$, the optimal cost for (A_0,b_0) is $x_0^TK_0x_0$. The real cost incurred when the feedback $f(A,b,c_0)$ is applied to the system (A_0,b_0) is equal to the optimal cost of the system (A,b,c_0), since $(A,b)\in G_0$ and hence both the state and input trajectory of $A+bf(A,b,c_0)$ and $A_0+b_0f(A,b,c_0)$ are equal. However, for (A_0,b_0), $f(A,b,c_0)$ can do no better than $f(A_0,b_0,c_0)$. Hence $x_0^TKx_0\geqslant x_0^TK_0x_0$. Since x_0 was arbitrary it follows that $K\geqslant K_0$.

□

COROLLARY 3.22 If $(A,b)\in G_0$ and $f(A,b,c_0)=f(A_0,b_0,c_0)$, then $K=K_0$.

PROOF Since $(A,b)\in G_0$, we have $A+bf(A,b,c_0)=A_0+b_0f(A,b,c_0)$, which by Lemma 3.21 implies that $K\geqslant K_0$. On the other hand, since $f(A,b,c_0)=f(A_0,b_0,c_0)$, we also have $A_0+b_0f(A_0,b_0,c_0)=A+bf(A_0,b_0,c_0)$. We can apply Lemma 3.21 once again, now with (A_0,b_0c_0) and (A,b,c_0) interchanged, showing that $K_0\geqslant K$.

□

PROOF OF THEOREM 3.10

PROOF Choose $(A,b)\in G\cap H$. Define $(\bar{A},\bar{b})\in E_{ns}$ by:

$$(\bar{A},\bar{b}) := \psi(A,b) \tag{75}$$

with ψ defined as in (44).
Then $(\bar{A},\bar{b})\in\tilde{G}_0$, and also since $(A,b)\in H$: $f(\bar{A},\bar{b},c_0)=f(A_0,b_0,c_0)$. Hence by Corollary 3.22: $\bar{K}=K_0$. Now:

$$(\bar{A},\bar{b})\in G_0 \Rightarrow \bar{A}=A_0+(b_0-\bar{b})\bar{f} \tag{76}$$

$$= A_0 + (b_0-\bar{b})f_0 \tag{77}$$

$$f(\bar{A},\bar{b},c_0)=f(A_0,b_0,c_0) \Rightarrow (\bar{b}^TK_0\bar{b}+r)^{-1}\bar{b}^TK_0\bar{A} = -f_0 \tag{78}$$

substituting (77) in (78) gives:

$$\bar{b}^TK_0(\bar{A}_0+(b_0-\bar{b})f_0)=-(\bar{b}^TK_0\bar{b}+r)f_0 \tag{79}$$

which implies:

$$\bar{b}^TK_0(A_0+b_0f_0)=-rf_0 \tag{80}$$

Now, since K_0 and $A_0+b_0f_0$ are non-singular, and $\bar{b}=b_0$ is by

construction a solution of (80), it follows that $\bar{b} = b_0$. Substituting this in (76) gives $\bar{A} = A_0$. $\bar{A} = SAS^{-1}$, for some $S \in Gl(n)$. Hence (A,c_0) and (SAS^{-1},c_0) are in standard observable form. This implies that $S = I$. This completes the proof of Theorem 3.10 (i).
The proof of part (ii) is immediate.

□

4. WHEN IS G CONTAINED IN H?

In section II.2 we have seen that if the control objective is pole assignment, then $G \subset H$. In section II.3 we proved that for LQ-control this is not the case and that in fact $G \cap H$ is a negligible subset of G. The natural question now arises for which control laws the set G is contained in H. In this section we will investigate this question for control problems that are solvable by state feedback. The surprising result is that under mild assumptions we can give a characterization of the control laws for which $G \subset H$. Recall from section II.2 that the crucial properties enabling us to prove that $G \subset H$ were:

(i) For all $(A_1,b_1) \in E_{re}$ and all $(A_2,b_2) \in E_{re}$:

$$A_1 + b_1 f(A_2,b_2) = A_2 + b_2 f(A_2,b_2) \tag{1}$$

$$\Rightarrow$$

$$f(A_1,b_1) = f(A_2,b_2) \tag{2}$$

(ii) $\forall S \in Gl(n)$, $\forall (A,b,) \in E_{re}$:

$$f(SAS^{-1},Sb) = f(A,b)S^{-1} \tag{3}$$

The question we would like to address is the following:
For which functions $f : E_{re} \to \mathbf{R}^{1 \times n}$ do we have properties (i) and (ii)?
To give a complete answer to this question we will make two more assumptions:

(iii) f is continuous on E_{re}.

(iv) There exists $(A_0,b_0) \in E_{re}$ such that:

$$\text{Spec}(A_0 + b_0 f(A_0,b_0)) \cap \text{Spec}(A_0) = \varnothing \tag{4}$$

COMMENT

(i) This is the crucial property of f. It is the algebraic translation of the assumption that $G \subset H$.

(ii) Assumption (ii) reflects that the control objective is a criterion on the input/output behavior of the system, and does not depend on any input/state/output realization.

(iii) Continuity is a natural assumption.

(iv) Assumption (iv) is a technical assumption. It means that there is at least one system such that the open-loop poles differ completely from the closed-loop poles of the controlled system. A sufficient condition for this property to hold, is that f is a stabilizing control-law.

□

We can now give a complete characterization of all functions f satisfying assumptions (i) through (iv). It will turn out that they are all of the pole-assignment type.

THEOREM 4.1 Let f be such that assumptions (i) through (iv) are satisfied.

There exists a monic polynomial π, of degree n such that for all $(A,b)\in E_{re}$:

$$f(A,b) = -[0..01].[b\vdots..\vdots A^{n-1}b]^{-1}\pi(A) \tag{5}$$

PROOF Fix (A_0,b_0) such that assumption (iv) holds. Let π_0 be the characteristic polynomial of $(A_0 + b_0f(A_0,b_0))$. Define:

$$G_0 := \{(A,b)\in E_{re} \mid A_0 + b_0f(A,b) = A + bf(A,b)\} \tag{6}$$

and:

$$\bar{G}_0 := \{(A,b)\in E_{re} \mid A_0 + b_0f(A_0,b_0) = A + bf(A_0,b_0)\} \tag{7}$$

Then, by (i), $G_0 = \bar{G}_0$. Choose $(A,b)\in G_0$, then, again by (i):

$$A_0 + b_0f(A_0,b_0) = A + bf(A,b) \tag{8}$$

Hence the characteristic polynomial of $(A + bf(A,b))$ is equal to π_0. By the uniqueness of the control law that assigns the poles, it follows that for all $(A,b)\in G_0$:

$$f(A,b) = -[0..01].[b\vdots..\vdots A^{n-1}b]^{-1}\pi_0(A) \tag{9}$$

We will now show that f has the form (9) for all pairs (A,b) that are similar to a pair in G_0. Choose $S\in Gl(n)$, then:

$$f(SAS^{-1},Sb) = f(A,b)S^{-1} \qquad \text{by assumption (ii)} \tag{10}$$

$$= -[0..01].[b\vdots..\vdots A^{n-1}b]^{-1}\pi_0(A)S^{-1} \tag{11}$$

$$= -[0..01].[Sb\vdots..\vdots(SAS^{-1})^{n-1}Sb]^{-1}\pi_0(SAS^{-1}) \tag{12}$$

Hence for every pair that is similar to a pair in G_0, we have established the desired formula. Next we will show that the set of all pairs that can be reached by a similarity transformation on the elements of G_0 is open and dense, and the continuity of f will then yield the result.

Define:

$$\Omega := \{(A,b)\in E_{re} \mid \exists\ S\in Gl(n) \text{ with } (SAS^{-1},Sb)\in G_0\} \tag{13}$$

Since $G_0 = \bar{G}_0$, $(A,b)\in\Omega$ if and only if the following equation has a non-singular solution S:

$$(A_0 + b_0f(A_0,b_0)) - S^{-1}AS = S^{-1}bf(A_0,b_0) \tag{14}$$

(14) is equivalent to:

$$S(A_0 + b_0f(A_0,b_0)) - AS = bf(A_0,b_0) \tag{15}$$

Define:

$$\Omega_1 := \{(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid \mathrm{Spec}(A_0 + b_0f(A_0,b_0))\cap\mathrm{Spec}(A) = \varnothing\} \tag{16}$$

Then Ω_1 is open and dense in $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}$. By Lemma II.3.15 it follows that (15) has a unique solution $S\in\mathbf{R}^{n\times n}$ for every $(A,b)\in\Omega_1$.
Define the linear map $L_A : \mathbf{R}^{n\times n} \to \mathbf{R}^{n\times n}$ by:

$$L_A(S) := S(A_0 + b_0 f(A_0,b_0)) - AS \tag{17}$$

L_A is injective if and only if $\mathrm{Spec}(A_0 + b_0 f(A_0,b_0)) \cap \mathrm{Spec}(A) = \varnothing$.
Define $d_L : \mathbf{R}^{n\times n} \to \mathbf{R}$, by:

$$d_L(A) := \det(L_A) \tag{18}$$

L_A is a linear map depending linearly on A, hence d_L is a polynomial in the coefficients of A. For every $(A,b)\in\Omega_1$, the unique solution of (15) is given by:

$$S(A,b) = \frac{n_L(A,b)}{d_L(A)}, \tag{19}$$

where n_L: $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \to \mathbf{R}^{n\times n}$ is a polynomial in the coefficients of (A,b).
Define $d : \mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \to \mathbf{R}$, by:

$$d(A,b) := \det(n_L(A,b)) \tag{20}$$

and:

$$\Omega_2 := \{(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid d(A,b)\neq 0\} \tag{21}$$

By assumption (iv), $(A_0,b_0)\in\Omega_1$, and hence $S(A_0,b_0) = I$, from which we conclude that $(A_0,b_0)\in\Omega_2$, and hence Ω_2 is nonempty. Since d is polynomial, it follows that Ω_2 is open and dense in $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}$. For a proof of this statement see the remark on genericity in [63]. Finally, it is easy to see that: $\Omega_1\cap\Omega_2 \subset \Omega$, and hence Ω contains a subset that is open and dense in $\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}$. Suppose there exists $(A,b)\in E_{re}\setminus\Omega$. Choose a sequence $(A_k,b_k)\in\Omega$, such that:

$$\lim_{k\to\infty}(A_k,b_k) = (A,b) \tag{22}$$

Then, for all k the characteristic polynomial of $(A_k + b_k f(A_k,b_k))$ is equal to π_0. By the continuity of f, it follows that the characteristic polynomial of $(A + bf(A,b))$ is equal to π_0. Finally take $\pi = \pi_0$. This completes the proof. □

We have now established a characterization of all feedback laws, depending only on (A,b), for which $G\subset H$. We would like to extend this classification result to the more general case where the function f may also depend on the c-vector.
Consider the realizations:

$$x(k+1) = A_0x(k) + b_0u(k) \qquad z(k+1) = Az(k) + bu(k) \tag{23.a}$$

$$y(k) = c_0x(k) \qquad \hat{y}(k) = c_0z(k) \tag{23.b}$$

$$x(0) = M_0\phi(0) \qquad z(0) = M\phi(0) \tag{23.c}$$

$$u(k)=f(A,b,c_0)z(k) \tag{23.d}$$

Where (A_0,c_0) and (A,c_0) are in standard observable form. Recall from the previous sections that:

$$G := \{(A,b)\in E_{ob} \mid \text{for all } k: y(k) = \hat{y}(k)\} \tag{24}$$

$$H := \{(A,b)\in E_{ob} \mid \text{for all } k: f(A,b,c_0)z(k) = f(A_0,b_0,c_0)x(k)\} \tag{25}$$

where we should keep in mind that G and H depend on the initial state $\phi(0)$. The question now becomes:
For which functions f do we have the property that for all $\phi(0)\in\mathbb{R}^{2n-1}$, $G\subset H$?
Again we restrict our attention to control criteria on the input/output behavior of the system. This means that if the control law is stated in state space form, then it should satisfy:

$$f(SAS^{-1},Sb,cS^{-1}) = f(A,b,c)S^{-1} \tag{26}$$

for every non-singular matrix S. A necessary condition for $G\subset H$ is given by the following lemma:

LEMMA 4.2 If for all $\phi(0)\in\mathbb{R}^{2n-1}$, $G\subset H$, then: For all $(A_0,b_0)\in E_{ob}$ and all $(A,b)\in E_{re}$ we have:

$$A_0 + b_0 f(A,b,c_0) = A + bf(A,b,c_0) \Rightarrow f(A_0,b_0,c_0) = f(A,b,c_0) \tag{27}$$

PROOF Choose $\phi(0)\in\mathbb{R}^{2n-1}$ such that

$$(A_0 + b_0 f(A_0,b_0,c_0), M_0\phi(0)) \text{ is reachable} \tag{28}$$

Choose a triple (A,b,c_0) that satisfies:

$$A_0 + b_0 f(A,b,c_0) = A + bf(A,b,c_0) \tag{29}$$

Let S be the unique non-singular matrix such that (SAS^{-1},c_0S^{-1}) is in standard observable form.
Then:

$$c_0S^{-1} = c_0 \tag{30}$$

Define:

$$x(0):=M_0\phi(0) \text{ and } x(k+1) = (A_0 + b_0 f(A,b,c_0))x(k) \tag{31}$$

Define $(\bar{A},\bar{b},c_0) := (SAS^{-1},Sb,c_0S^{-1})$, and $z(k) := Sx(k)$. Then for all k:

$$z(k+1) = \bar{A}z(k) + \bar{b}f(\bar{A},\bar{b},c_0)z(k) \qquad \text{by (26,29,31)} \tag{32}$$

Also:

$$\hat{y}(k) = c_0z(k) = c_0Sx(k) = c_0x(k) = y(k) \tag{33}$$

Hence $(\bar{A},\bar{b})\in G$. Since $G\subset H$, we also have for all k:

$$f(\bar{A},\bar{b},\bar{c})z(k) = f(A_0,b_0,c_0)x(k) \tag{34}$$

Which implies:

$$f(A,b,c_0)x(k) = f(A_0,b_0,c_0)x(k) \tag{35}$$

Since by (28) $\{x(k)\}$ spans $\mathbf{R}^n$, it follows that

$$f(A,b,c_0) = f(A_0,b_0,c_0) \tag{36}$$

□

DEFINITION 4.3 Define for $(A_0,b_0)\in E_{ob}$ the sets:

$$G_0 := \{(A,b)\in E_{re} \mid A_0 + b_0 f(A,b,c_0) = A + bf(A,b,c_0)\} \tag{37}$$

and:

$$\bar{G}_0 := \{(A,b)\in E_{re} \mid A_0 + b_0 f(A_0,b_0,c_0) = A + bf(A_0,b_0,c_0)\} \tag{38}$$

□

LEMMA 4.4 If f is such that $G\subset H$, then $G_0 = \bar{G}_0$.

PROOF Choose $(A,b)\in G_0$, then by Lemma 4.2, $f(A,b,c_0) = f(A_0,b_0,c_0)$ and hence $(A,b)\in\bar{G}_0$.
Choose $(A,b)\in\bar{G}_0$. Let $S\in Gl(n)$ be the unique matrix such that (SAS^{-1},c_0S^{-1}) is in standard observable form. Then $c_0S^{-1} = c_0$. From (38) it follows that:

$$S(A_0 + b_0 f(A_0,b_0,c_0))S^{-1} = S(A + bf(A_0,b_0,c_0))S^{-1} \tag{39}$$

which implies:

$$SA_0S^{-1} + Sb_0 f(SA_0S^{-1},Sb_0,c_0S^{-1}) = \tag{40}$$
$$SAS^{-1} + Sbf(SA_0S^{-1},Sb_0,c_0S^{-1})$$

Since $c_0S^{-1} = c_0$, this gives:

$$SA_0S^{-1} + Sb_0 f(SA_0S^{-1},Sb_0,c_0)) = SAS^{-1} + Sbf(SA_0S^{-1},Sb_0,c_0)) \tag{41}$$

We can now apply Lemma 4.2 and conclude:

$$f(SAS^{-1},Sb,c_0) = f(SA_0S^{-1},Sb_0,c_0) \tag{42}$$

which yields:

$$f(A,b,c_0) = f(A_0,b_0,c_0) \tag{43}$$

Hence $(A,b)\in G_0$.

□

We can now characterize the control laws for which $G\subset H$. However, we have a complete result for the case $n=1$ only.

THEOREM 4.5 (n=1) Let $f:\mathbf{R}\times\mathbf{R}\setminus\{0\}\times\mathbf{R}\setminus\{0\}\to\mathbf{R}$ be such that $G\subset H$. Assume that there exists (a_1,b_1) and (a_2,b_2) such that:

$$f(a_1,b_1,1)\neq f(a_2,b_2,1), \tag{44}$$

then there exists an $\alpha\in\mathbf{R}$, such that for all $(a,b,c)\in\mathbf{R}\times\mathbf{R}\setminus\{0\}\times\mathbf{R}\setminus\{0\}$:

$$f(a,b,c)=\frac{\alpha-a}{b} \tag{45}$$

PROOF Fix $c=1$. Choose $(a_1,b_1),(a_2,b_2)$ such that $f(a_1,b_1,1)\neq f(a_2,b_2,1)$, and define:

$$l_1:=\{(a,b)\mid a+bf(a_1,b_1,1)=a_1+b_1f(a_1,b_1,1)\} \tag{46}$$

$$l_2:=\{(a,b)\mid a+bf(a_2,b_2,1)=a_2+b_2f(a_2,b_2,1)\} \tag{47}$$

From Lemma 4.4 it follows that $(a_i,b_i)\in l_i$ implies that $f(a,b,1)=f(a_i,b_i,1).(i=1,2)$. Since $f(a_1,b_1,1)\neq f(a_2,b_2,1)$, we conclude that $l_1\cap l_2\cap\{b\neq 0\}=\varnothing$, however for the same reason $l_1\cap l_2\neq\varnothing$. Hence there exists $\alpha\in\mathbf{R}$ such that $l_1\cap l_2=\{(\alpha,0)\}$ Now, choose $(\bar{a},\bar{b})$, arbitrarily and define:

$$\bar{l}:=\{(a,b)\mid a+bf(\bar{a},\bar{b},1)=\bar{a}+\bar{b}f(\bar{a},\bar{b},1)\} \tag{48}$$

Without loss of generality we may assume that $f(\bar{a},\bar{b},1)\neq f(a_1,b_1,1)$, hence $\bar{l}\cap l_1\neq\varnothing$. As before we also have $\bar{l}\cap l_1\cap\{b\neq 0\}=\varnothing$. Therefore there exists $\bar{\alpha}\in\mathbf{R}$ such that $\bar{l}\cap l_1=\{(\bar{\alpha},0)\}$. Since $(\alpha,0)\in l_1$, and $l_1\neq\{(a,b)\mid b=0\}$, we have: $\alpha=\bar{\alpha}$. This gives:

$$\alpha=\bar{a}+\bar{b}f(\bar{a},\bar{b},1) \tag{49}$$

which implies:

$$f(\bar{a},\bar{b},1)=\frac{\alpha-\bar{a}}{\bar{b}} \tag{50}$$

Finally, by assumption (ii) and (50):

$$f(a,b,c)=f(a,bc,1)c=\frac{\alpha-a}{bc}c=\frac{\alpha-a}{b} \tag{51}$$

This finishes the proof.

The general result, $n>1$, has not yet been found. Lemmata 4.2 and 4.4 give the mathematical translation of the requirement that $G\subset H$. However, the idea on which the proof of Theorem 4.5 is based does not apply in the higher order case.

□

Chapter III

Adaptive Control Algorithms

1. AVOIDING SINGULARITIES IN THE PARAMETER SPACE

In almost all control problems, a solution for a particular system can be found provided that certain regularity conditions on the system are satisfied. For instance for the pole-assignment problem the system has to be reachable. If an adaptive control algorithm is based on certainty equivalence, then at each time instant the controller is calculated on the basis of the current estimate. This can of course only be done if this estimate satisfies the regularity conditions. In many situations this means that the estimates should not belong to a certain (small) subset of the parameter space, namely the set of those parameter values for which these conditions are not satisfied. The usual algorithms cannot assure that estimates will always be regular unless extra knowledge of the true system is assumed. Typically this extra knowledge will be that the true system belongs to a known convex subset of the parameter space that does not contain any singular points. In this section we will present a modification method for algorithms that work well under the assumption that estimates are always regular. The modification is such that this assumption can be relaxed and the asymptotic properties of the algorithm remain unchanged. Other approaches to this problem are given in [10,31,40,42,46]. In [40] a modification procedure is proposed which requires a lower bound on the stabilizability of the system. [10] uses excitation signals until the prediction error comes below some prescribed threshold. Hence tracking and zero-regulation can be achieved only approximately. In [42] the parameter estimation scheme is modified to keep the estimates away from the set of non-reachable pairs. In [31] the identification is done simultaneously in the parameter space and the controller space. By using an extra (non-linear) feedback driven by the prediction error, these two schemes are brought into agreement with each other. The extra feedback can be interpreted as an asymptotically vanishing excitation signal. A drawback of the last two methods is that modification ([42]) and extra feedback ([31]) does not stop in finite time, and hence the algorithms deviate from the usual simple schemes during the complete time period. [46] gives a modification procedure for stochastic systems. The method which we want to propose here is based on a modification procedure developed in [55]. We will start with an example.

EXAMPLE 1.1 Consider the system:

$$y(k+1) = a_0 y(k) + b_0 u(k), \quad b_0 \neq 0. \tag{1}$$

which we want to control adaptively, according to the control objective:

$$y(k+1) = \alpha y(k) \quad \alpha \in \mathbf{R} \ (|\alpha| < 1). \tag{2}$$

the corresponding control law is of the form

$$f(a,b) = \frac{\alpha - a}{b}. \tag{3.a}$$

$$u(k) = f(a,b) y(k). \tag{3.b}$$

Suppose we want to use an algorithm based on certainty equivalence and neutrality, say:

$$\hat{\theta}_k = (\hat{a}_k, \hat{b}_k) \tag{4.a}$$

$$\hat{\theta}_{k+1} = \tilde{\theta}(\hat{\theta}_k, u(k), y(k), y(k+1)) \tag{4.b}$$

$$u(k+1) = \frac{\alpha - \hat{a}_k}{\hat{b}_k} y(k+1), \tag{4.c}$$

then it is clear that to be able to calculate $u(k)$, we must not have $\hat{b}_k = 0$. In many cases $\tilde{\theta}$ is a rational function of its arguments. Since the coefficients will depend on the unknown system parameters, it cannot be guaranteed a priori that $\hat{b}_k$ stays non-zero.

On the other hand from the rationality of $\tilde{\theta}$ it also follows that the set of initial values $\hat{\theta}_0$ for which $\hat{b}_k = 0$ for some k has Lebesgue measure zero, see [45].

However $\hat{b}_k \neq 0$ for all finite k might not be enough for the analysis of the algorithm. One could also want to have that $\hat{b}_k$ is kept bounded away from zero. For instance one may want that $\lim_{k \to \infty} \hat{b}_k \neq 0$. In general one cannot conclude that the set of initial values $\hat{\theta}(0)$ for which $\hat{b}_k$ is bounded away from zero has Lebesgue measure zero. If this property is needed, the usual algorithms have to be modified.

Recall the projection algorithm for estimating (a_0, b_0):

$$\hat{a}_{k+1} = \hat{a}_k + \frac{y(k)}{y(k)^2 + u(k)^2} (y(k+1) - \hat{y}(k+1)) \tag{5.a}$$

$$\hat{b}_{k+1} = \hat{b}_k + \frac{u(k)}{y(k)^2 + u(k)^2} (y(k+1) - \hat{y}(k+1)) \tag{5.b}$$

$$u(k) = \frac{\alpha - \hat{a}_k}{\hat{b}_k} y(k) \quad \text{(according to certainty equivalence)} \tag{5.c}$$

We will now describe a modification of (5) so as to ensure that for all k:

$$|\hat{b}_k| \geq \epsilon > 0, \tag{6}$$

for some $\epsilon > 0$.

A fundamental property of (5) is, that regardless of the sequences of inputs:

$\{\|(a_0,b_0)-(\hat{a}_k,\hat{b}_k)\|\}_{k\in\mathbb{N}}$ is a non-increasing sequence. (7)

and:

$$\lim_{k\to\infty} \|(\hat{a}_{k+1},\hat{b}_{k+1}) - (\hat{a}_k,\hat{b}_k)\| = 0 \tag{8}$$

which implies:

$$\lim_{k\to\infty} \frac{y(k+1) - \hat{y}(k+1)}{y(k)^2 + u(k)^2} = 0 \tag{9}$$

(see Lemma's III.2.2.2 and III.2.2.3 for a proof of (7) and (8)). We also know that if the input is rich enough, that $\lim_{k\to\infty} (\hat{a}_k,\hat{b}_k) = (a_0,b_0)$. Now, since by assumption $b_0\neq 0$, this means that if the input is rich enough, then within finite time $\hat{b}_k$ will be bounded away from zero. The problem is that we do not know when, since we do not know b_0. We propose the following modification of (5). Choose any sequence:

$$\epsilon_k \downarrow 0 \tag{10}$$

Start the algorithm in any initial value $(\hat{a}(0),\hat{b}(0))$, $\hat{b}(0)\neq 0$, and calculate $u(k)$ as:

$$u(k) = \frac{\alpha-\hat{a}_k}{\hat{b}_k} y(k) \tag{11.a}$$

until $|\hat{b}(k)|\leq\epsilon_1$. Call this time instant τ_1.
Take $u(\tau_1)=y(\tau_1)$, $u(\tau_1+1)=-y(\tau_1+1)$ (the estimates $(\hat{a}(\tau_1+1),\hat{b}(\tau_1+1))$ are calculated according to (5.a,5.b))
Then take

$$u(k)=\frac{\alpha-\hat{a}_k}{\hat{b}_k} y(k), \tag{11.b}$$

until $|\hat{b}_k|\leq\epsilon_2$. Call this time instant τ_2.
Take $u(\tau_2)=y(\tau_2)$, $u(\tau_2+1)=-y(\tau_2+1)$
More general, let τ_k be the first time instant j after $\tau_{k-1}+1$ such that $|\hat{b}_j|\leq\epsilon_k$. For $j\in[\tau_{k-1}+2,\tau_k-1]$ take:

$$u(j) = \frac{\alpha-\hat{a}_k}{\hat{b}_k} y(j) \tag{11.c}$$

for $i=0,1$, take:

$$u(\tau_k+i) = (-1)^i y(\tau_k+i) \tag{11.d}$$

Now, in any case $u(k)$ is well-defined for every k, moreover we have the following:

LEMMA 1.2 $\{\tau_k \mid k \in \mathbb{N},\ \tau_k < \infty\}$ is finite.

PROOF Assume the contrary, then $\forall k\ |\hat{b}(\tau_k)| \leqslant \epsilon_k$. Since $\epsilon_k \downarrow 0$ this means

$$\lim_{k\to\infty} \hat{b}(\tau_k) = 0. \tag{12}$$

And hence by (8) also

$$\lim_{k\to\infty} \hat{b}(\tau_k + i) = 0 \quad \forall i. \tag{13}$$

Take a convergent subsequence $\{(\hat{a}_{s_k}, \hat{b}_{s_k})\}$ of $\{(\hat{a}_{\tau_k}, \hat{b}_{\tau_k})\}$. Say $\lim_{k\to\infty} (\hat{a}_{s_k}, \hat{b}_{s_k}) = (\bar{a}, \bar{b})$. Then also: $\lim_{k\to\infty} (\hat{a}_{1+s_k}, \hat{b}_{1+s_k}) = (\bar{a}, \bar{b})$, and $\lim_{k\to\infty} (\hat{a}_{2+s_k}, \hat{b}_{2+s_k}) = (\bar{a}, \bar{b})$.
Hence by (5.a and 11.d), see also the proof of Example II.1.2.3:

$$\bar{a} = \bar{a} = \frac{1}{2}(a_0 - \bar{a}) + (b_0 - \bar{b})) \tag{14}$$

and:

$$\bar{a} = \bar{a} = \frac{1}{2}(a_0 - \bar{a}) - (b_0 - \bar{b})) \tag{15}$$

Hence:

$$a_0 + b_0 - \bar{a} - \bar{b} = 0 \tag{16}$$

and:

$$a_0 - b_0 - \bar{a} + \bar{b} = 0 \tag{17}$$

This yields:

$$\lim_{k\to\infty} \hat{a}(s_k) = \lim_{k\to\infty} \hat{a}(s_k + 1) = a_0. \tag{18}$$

and:

$$\lim_{k\to\infty} \hat{b}(s_k) = \lim_{k\to\infty} \hat{b}(s_k + 1) = b_0 \neq 0, \tag{19}$$

which is already a contradiction, but we even have

$$\lim_{k\to\infty} (\hat{a}_k, \hat{b}_k) = (a_0, b_0), \tag{20}$$

by (7). The statement follows.

□

The idea behind this algorithm is that every time that $\hat{b}_k$ comes too close to zero, we apply a piece of a sufficiently exciting input sequence. If we have to do this infinitely many times, the complete exciting sequence is applied and the system is identified. Now if we adapt our strip around zero which $\hat{b}(k)$ must not enter, in a decreasing way, we know that $\hat{b}(k)$ will never enter it after some finite time instant. This time instant depends on, among other things, the true

system, hence it is unknown. All we know is that it is finite and that is enough, because that means that the asymptotic behavior of the algorithm is not influenced by the modification procedure. So, if the original algorithm worked well under the assumption that $\hat{b}_k$ stays bounded away from zero, the modified algorithm also works without that assumption.
Although Example 1.1 may seem to be a little tricky, a fairly general method can be derived from it. We will now describe this method. First we will give the assumptions on which the method is based, then we will state the theorem that describes the main property of the method, we will then comment upon the assumptions and the theorem and finally we will prove the theorem.
The parameter space is $\mathbf{R}^n \times \mathbf{R}^n$, equipped with the Euclidean norm. Let the algorithm be recursive and based on certainty equivalence:

$$\hat{\theta}_{k+1} = \tilde{\theta}(\hat{\theta}_k, y(k+1),..,y(k-n+1),u(k),..,u(k-n+1)) \tag{21.a}$$

$$(\hat{M}_k,\hat{N}_k) = f(\hat{\theta}_k) \tag{21.b}$$

$$u(k) = \hat{m}_1(k)u(k-1)+..+\hat{m}_{n-1}(k)u(k-n+1)+ \tag{21.c}$$
$$\hat{n}_0(k)y(k)+..+\hat{n}_{n-1}(k)y(k-n+1)$$

The notation is as in II.1. Let Ω be the set of regular parameters:

$$\Omega = \{\theta \in \mathbf{R}^n \times \mathbf{R}^n \mid f(\theta) \text{ is well-defined } \} \tag{22}$$

Then (21.b) is well-defined only if for all k:

$$\hat{\theta}_k \in \Omega \tag{23}$$

Let the true, to be controlled system be represented by $\theta_0 = (A_0,B_0) \in \mathbf{R}^n \times \mathbf{R}^n$. This means that the observed data satisfies:

$$y(k+1) = a_0^0 y(k)+..+a_{n-1}^0 y(k-n+1)+b_0^0 u(k)+..+b_{n-1}^0 u(k-n+1) \tag{24}$$

ASSUMPTIONS:

A1 $\theta_0 = (A_0,B_0) \in \Omega$.

A2 Ω is open and dense.

A3 There exists a sequence of controllers $(M^*(k),N^*(k)) \in \Sigma_c(n)$, such that if for all k:
$u(k) = m_1^*(k)u(k-1)+..+m_{n-1}^*(k)u(k-n+1)+$
$n_0^*(k)y(k)+..+n_{n-1}^*(k)y(k-n+1)$,
then:
$\lim_{k\to\infty} \hat{\theta}_k = \theta_0$
uniformly in the initial state $(y(0),..,y(-n+1),u(-1),..,u(-n+1))$.

A4 $\tilde{\theta}$ is such that for every initial state $(y(0),..,y(-n+1),u(-1),..,u(-n+1))$, and every initial estimate $\hat{\theta}_0$, there exists a sequence of non-negative real numbers $\{\delta_k\}_{k\in\mathbf{N}}$, such that for every input sequence $\{u_k\}_{k\in\mathbf{N}}$, the following holds:
$\|\hat{\theta}_{k+1} - \theta_0\| \leq \|\hat{\theta}_k - \theta_0\| + \delta_k$
and:

$$\sum_{k=0}^{\infty} \delta_k < \infty$$

MODIFICATION Choose a sequence $\{\epsilon_k\}_{k\in\mathbb{N}}$, such that:

$$\epsilon_k \downarrow 0 \ (k\to\infty) \tag{25.a}$$

Define:

$$d_k = \inf_{\theta\in\delta(\Omega)} \|\hat{\theta}_k - \theta\| \tag{25.b}$$

($\delta(\Omega)$ = boundary (Ω)). Define the input sequence as follows:

$$u(k) = \hat{m}_1(k)u(k-1)+..+\hat{m}_{n-1}(k)u(k-n+1) \tag{25.c}$$
$$+\hat{n}_0(k)y(k)+..+\hat{n}_{n-1}(k)y(k-n+1)$$

for $\tau_j+j+1\leqslant k\leqslant\tau_{j+1}-1$, $j=0,1,2,3,..,$

$$u(k) = m_1^*(i)u(k-1)+..+m_{n-1}^*(i)u(k-n+1) \tag{25.d}$$
$$+n_0^*(i)y(k)+..+n_{n-1}^*(i)y(k-n+1)$$

for $k=\tau_j+i$, $j=0,1,2,3,..,$ $i=0,1,2,..,j$.
where the sequence $\{\tau_j\}_{k\in\mathbb{N}}$ is defined by:

$$\tau_0 = \min\{i\geqslant 0 \mid d_i\leqslant\epsilon_0\} \tag{25.e}$$

$$\tau_{j+1} = \min\{i\geqslant\tau_j+j+1 \mid d_i\leqslant\epsilon_{j+1}\} \tag{25.f}$$

The minimum is defined to be infinity if the set over which the minimization takes place is empty.

THEOREM 1.3 Under the assumptions A1,..,A4 and with the modification (25) the following holds: For all initial states $(y(0),..,y(-n+1),u(-1),..,u(-n+1))$ and for all initial estimates $\tilde{\theta}_0$, there exist an $\epsilon>0$ and a $K\in\mathbb{N}$, such that for all $k\geqslant K$:

(i) $\tau_k = \infty$
(ii) $d_k\geqslant\epsilon$
(iii) $\hat{\theta}_k\in\Omega$

Before we prove this theorem let us first comment upon it and the assumptions under which it holds.

A1 $(A_0,B_0)\in\Omega$. This means that it is assumed that the true system belongs to Ω. If Ω is for instance the set of all (A,B) such that the corresponding polynomials have no common roots, then A1 is a standard assumption in the context of adaptive control. It is hard to imagine that one could do without A1, because if the assumption would not hold for the true system, then it could not be controlled even if we knew (A_0,B_0), let alone controlled adaptively.

A2 The requirement that Ω is open, together with A1 guarantee that (A_0,B_0)

lies in the interior of Ω. A4 will then yield that once the estimates are close enough to (A_0,B_0), they will never leave Ω. The requirement that Ω is dense is not strictly needed, but is fulfilled in most potential applications of our method.

A3 A3 means that it is assumed that there exists a sequence of feedbacks that is sufficiently exciting with respect to the algorithm (5).
Uniform convergence of θ_k with respect to the initial conditions should not be too restrictive. For instance, if $\tilde{\theta}$ satisfies

(i) $\tilde{\theta}(\theta,\lambda y(0),\lambda y(1),..,\lambda y(n-1),\lambda u(0),..,\lambda u(n-1))$
$= \theta(\theta,y(0),..,y(n-1),u(0),..,u(n-1))$
for all $\theta\in\mathbf{R}^n\times\mathbf{R}^n$, $y(i),u(i)\in\mathbf{R}$, $\lambda\in\mathbf{R}$.

(ii) $\tilde{\theta}$ is continuous in $(y(0),..,u(n-1))$,

then convergence of $\hat{\theta}_k$ implies uniform convergence. Uniform convergence is needed since the exciting sequence is started at different time instants and hence for different initial conditions.

A4 Is a standard property of many algorithms in the literature for the case that $\delta_k\equiv 0$. Our assumption is slightly weaker and is useful if other than dead-beat observers are used. In the continuous time case, where dead-beat observers do not exist, a similar property holds with $\delta_t>0$. The geometrical interpretation of A4 is that at each iteration the estimate comes closer to θ_0 modulo some summable disturbance. In particular this implies boundedness of the sequence of estimates.

$$\|\hat{\theta}(k)-\theta_0\| \leq \|\hat{\theta}(0)-\theta_0\|+\sum_{j=0}^{k-1}\delta_j \tag{26.a}$$

$$\leq \|\hat{\theta}(0)-\theta_0\|+\sum_{j=0}^{\infty}\delta_j < \infty \tag{26.b}$$

and, more general, for all k and all $l\geq 1$:

$$\|\theta(k+l)-\theta_0\| \leq \|\hat{\theta}(k)-\theta_0\|+\sum_{j=k}^{l-1}\delta_k \tag{27}$$

THE MODIFICATION The input sequence is constructed as follows. Assume $\tau_0,..,\tau_j$ have already been defined. At time τ_j we start to apply the sufficiently exciting input sequence

$$\begin{aligned} u(j+i) = {} & m_1^*(i)u(j+i-1)+..+m_{n-1}^*(i)u(j+i-n+1) \\ & +n_0^*(i)y(j+i)+..+n_{n-1}^*(i)y(j+i-n+1) \end{aligned} \tag{28}$$

we do this for j time steps. After that, inputs are calculated according to the certainty equivalence principle until the distance of the estimate to the boundary of Ω becomes smaller than ϵ_{j+1}. This time instant is called τ_{j+1} and again we start to apply the exciting signals, but now for $j+1$ steps. The sequence $\{\tau_j\}$ can be seen as a sequence of stopping times (terminology borrowed from the theory of stochastic processes).

PROOF OF THEOREM 1.3 (i) Assume the contrary. I.e. $\forall k$: $\tau_k < \infty$. Then

$$\lim_{k\to\infty} d(\tau_k) = 0. \tag{29}$$

On the other hand, for every $\epsilon > 0$ there exists $k(\epsilon)$, such that for every initial state $(\bar{y}(0),..,\bar{y}(n-1),\bar{u}(0),..,\bar{u}(n-1))$ and $\hat{\hat{\theta}}_0$ such that

$$\|\hat{\hat{\theta}}_0 - \theta_0\| \leq \|\hat{\theta}_0 - \theta_0\| + \textstyle\sum_{k=0}^{\infty} \delta_k, \tag{30}$$

the following holds: Start the algorithm in $\hat{\hat{\theta}}_0$, with the system in state $(\bar{y}(0),..,\bar{y}(-n+1),\bar{u}(-1),..,\bar{u}(-n+1))$:

$$\hat{\hat{\theta}}_{k+1} = \tilde{\theta}(\hat{\hat{\theta}}_k, y(k+1), y(k), u(k), .., y(k-n+1), u(k-n+1)) \tag{31.a}$$

$$u(k) = m_1^*(k)u(k-1) + .. + m_{n-1}^*(k)u(k-n+1) \tag{31.b}$$
$$n_0^*(k)y(k) + .. + n_{n-1}^* y(k-n+1)$$

then, by A3, for all $k \geq k(\epsilon)$:

$$\|\hat{\hat{\theta}}_k - \theta_0\| \leq \epsilon \tag{32}$$

Denote by ϵ_0, the distance between θ_0 and the boundary of Ω. By A1 and A2 $\epsilon_0 > 0$. Since for all k:

$$\|\hat{\theta}_k - \theta_0\| \leq \|\hat{\theta}_0 - \theta_0\| + \textstyle\sum_{k=0}^{\infty} \delta_k, \tag{33}$$

we know from (30,31.b,32) that there exists k_1 such that for all $k \geq k_1$:

$$\|\hat{\theta}_{\tau_k + k} - \theta_0\| \leq \frac{1}{4}\epsilon_0 \tag{34}$$

Choose k_2 such that for all $k \geq k_2$:

$$\sum_{j=k}^{\infty} \delta_j < \frac{1}{4}\epsilon_0 \tag{35}$$

then, for all $l_1, l_2 \geq k_2$:

$$\|\hat{\theta}_{l_2} - \theta_0\| \leq \|\hat{\theta}_{l_1} - \theta_0\| + \sum_{j=l_1}^{l_2} \delta_j \tag{36.a}$$

$$\leq \|\hat{\theta}_{l_1} - \theta_0\| + \frac{1}{4}\epsilon_0 \tag{36.b}$$

In particular, if we take:

$$k_3 = \min_{j \geq k_1} \{\tau_j + j \geq k_2\}, \tag{37}$$

then, by (34,36) for all $k \geq k_3$:

$$\|\hat{\theta}_k - \theta_0\| \leq \|\hat{\theta}_{k_3} - \theta_0\| + \frac{1}{4}\epsilon_0 \tag{38.a}$$

$$\leq \frac{1}{4}\epsilon_0 + \frac{1}{4}\epsilon_0 = \frac{1}{2}\epsilon_0 \tag{38.b}$$

This implies that for all $k \geqslant k_3$:

$$d_k \geqslant \frac{1}{2}\epsilon_0 \tag{39}$$

Choose k_4 such that for all $k \geqslant k_4$:

$$\epsilon_k \leqslant \frac{1}{4}\epsilon_0, \tag{40}$$

then, by (25.f,39), we conclude that:

$$\tau_{k_4} = \infty \tag{41}$$

Finally take:

$$K = 1 + \max\{k \mid \tau_k < \infty\} \tag{42}$$

by (41) K is finite.

(ii) Take $\epsilon = \epsilon_K$.

(iii) From (ii) we know that for all $k \geqslant K$, $d_k > \epsilon$. Now, since Ω is open and dense this means that for all $k \geqslant K$: $\hat{\theta}_k \in \Omega$.

□

The importance of this theorem lies in the fact that it can be used in adaptive control algorithms in which the complement of an open set must not be touched.
Suppose we have an adaptive control algorithm which satisfies A1, A2, A3 and A4. Suppose that the algorithm behaves well asymptotically under the extra assumption:
A4' $\forall k: d(\hat{\theta}_k, \delta(\Omega)) \geqslant \epsilon'$ for some open set $\Omega \subset \mathbf{R}^n \times \mathbf{R}^n$ and $\epsilon' > 0$.
Then Theorem 3 can be used to modify the algorithm so as to ensure that A4' is satisfied without changing the asymptotic behavior of the algorithm. The modification is described by (25). Theorem 3 then says that after some finite time instance the modification is not used anymore.
The conclusion is that assumption A4' may be replaced by A4 without changing the asymptotic behavior of the algorithm.
We have described a general method of how to avoid certain open subsets of the parameter space. The idea on which it is based was first described in [55]. The modification there differs in the sense that the time intervals during which the exciting inputs are used, are constant in length. For specific classes of system and algorithms other modifications based on the same idea may be used, as can be concluded from the first-order example.

2. ADAPTIVE POLE-ASSIGNMENT

2.1 Introduction

In this section we will present an algorithm for adaptive pole-assignment of single-input/single-output linear time-invariant systems of which only the order is assumed to be known. This problem has received considerable attention in the literature, see for instance [3,13,14,15,17,21,31,42,56]. In [3,14,15,17,21] the problem is studied for systems in input/output form with the only assumption that the order of the system is known. Algorithms based on parameter estimation and the certainty equivalence principle are proposed. In all of these papers stability results are derived under additional assumptions. The main reason that these assumptions have to be made seems to be that during the estimation procedure (unstable) pole-zero cancellation can occur. This can be avoided by assuming extra knowledge of the true system, which reduces the results essentially to local ones. Another way of avoiding that parameter estimates eventually have common factors is to use sufficiently exciting signals to assure convergence of the estimates to the true parameter value. It should be clear that additional injected signals can influence the performance of the system negatively, moreover it is always difficult to guarantee internal excitation by means of conditions on an external signal, since external excitation may be annihilated by unpredictable signals in the feedback loop.
In [31,42] algorithms are presented that overcome this difficulty. The reader is referred to Section III.1 for a discussion of these papers.
The algorithm presented here is based on the ones that were developed in [50,53,55].
We will use both the input/output description as well as the input/state/output representation of the systems at hand. The state space description is more convenient for parts of the analysis and to state the results in a clear fashion, whereas the input/output description seems to be the right tool for the estimation part of the algorithm.
The main result is that the asymptotic closed-loop behavior of the adaptive controlled system equals the behavior we would have obtained knowing the true system parameters. The proof of this result is independent of the desired pole locations. Hence even in the somewhat unrealistic situation where one wants to place the closed-loop poles in the unstable region our algorithm is applicable. This may look purely academic, but it shows that the adaptation of the controller parameters does not depend on stability properties of the system. The reason that we are able to derive such a result is that we consider the unknown parameters in the state space description as linear maps of which we want to know the action on certain subspaces. The variables on which these maps act can then be normalized without losing any information.
The algorithm is based on neutrality and certainty equivalence. In II.2 we proved that for pole assignment the set of invariant points G of any neutral certainty equivalent algorithm is contained in the set of all parameter values that give rise to the desired controls, H. The algorithm presented here relies on

this property. To avoid that parameter estimates eventually become non-reachable a modification of the type described in the previous section is used. The rest of this section is subdivided into four subsections. The first contains the problem statement and a general theorem. The second is devoted to the case where the state of the system is directly available. That part serves mainly as a preparation of the general case which is treated in the third part. Finally, in the fourth subsection we will give simulation results.

2.2. Problem statement

Consider the following time-invariant finite dimensional linear system:

$$x(k+1) = Ax(k) + bu(k), \quad \text{with initial state } x(0), \tag{2.1.a}$$

$$y(k) = cx(k) \tag{2.1.b}$$

$(A,b)\in E_{ob}$, where

$$E_{ob} := \{(A,b)\in \mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid (A,b,c) \text{ minimal} \quad (c_0,A) \text{ in standard observable form } \} \tag{2.2}$$

Also define:

$$E_{re} := \{(A,b)\in \mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid (A,b) \text{ reachable } \} \tag{2.3}$$

Let $\Lambda := \{\lambda_1,..,\lambda_n\}\subset\mathbf{C}$ be such that $\lambda\in\Lambda \ \bar{\lambda}\in\Lambda$. Let the control objective be the assignment of the closed-loop poles to the configuration Λ. Define $\pi\in\mathbf{R}[X]$ by: $\pi(X)=\prod_{i=1}^{n}(X-\lambda_i)$.

Define $f : E_{re} \to \mathbf{R}^{1\times n}$ by:

$$f(A,b) := -[0..01]\,[b\vdots Ab\vdots ...\vdots A^{n-1}b]^{-1}\pi(A) \tag{2.4}$$

Then the characteristic polynomial of $A+bf(A,b)$ is exactly π, and moreover, since the system is single-input, $f(A,b)$ is the only feedback law with that property. (see [26]).

Suppose now that the true value (A_0,b_0,c_0) of the system parameters is unknown. Then the control objective has to be replaced by a weaker one. As a modified version of the original control objective we choose the following:

> Generate a sequence of inputs such that asymptotically the applied inputs equal the inputs that would have been calculated on the basis of the true system parameters.

The following theorem relates the requirements described above to the resulting closed-loop behavior of the system.

THEOREM 2.2.1 Let $(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1}$, not necessarily reachable, and let $f\in\mathbf{R}^{1\times n}$. Let the sequence $\{u(k)\}_{k\in\mathbf{N}}$ and $x(0)\in\mathbf{R}^n$ be given. Define $x(k)$ by:

$$x(k+1) = Ax(k)+bu(k) \tag{2.5}$$

Assume that for all k: $x(k) \neq 0$ and suppose:

$$\lim_{k\to\infty} \left| \frac{u(k)-fx(k)}{\|x(k)\|} \right| = 0 \tag{2.6}$$

Then there exists a sequence of matrices $\{\Delta_k\}_{k\in\mathbb{N}}$ such that for all k:

$$i)\ x(k+1) = (A+bf+\Delta_k)x(k) \tag{2.7}$$

$$ii)\ \lim_{k\to\infty} \Delta_k = 0 \tag{2.8}$$

PROOF Define:

$$\Delta_k := \frac{b(u(k)-fx(k))x(k)^T}{x(k)^T x(k)} \tag{2.9}$$

then one can easily check that (2.7) holds. (2.8) follows from (2.6).

□

COMMENT Theorem 2.2.1 tells us that if the input of a linear system is asymptotically given by state feedback, then asymptotically the system will behave as if this feedback was used. This result holds whether f is stabilizing or not. An important feature in the assumption of the theorem is the normalization. This decouples the result from the norm of the state trajectory, and emphasizes that everything depends only on the directions of the states.
Now consider Theorem 2.2.1 and replace (A,b) by (A_0,b_0) and f by $f(A_0,b_0)$. This illustrates what kind of result we get if we are able to produce a sequence of inputs which satisfies (2.6). In sections 3 and 4 we will present algorithms which produce such sequences.

2.3. An algorithm for the observed state case

We will now propose an algorithm for the case that the state of the system (2.1) is observed. This is of course not a very realistic situation, but on a conceptual level it provides a good preparation for the non-observed state case. The algorithm is a modification of the one described in [50,53]. There an essential assumption was made on the controllability of the limit points of the sequence of parameter estimates. By introducing an alternative procedure when the parameter estimates are close to non-reachable, this assumption can be relaxed. A drawback is that the analysis of the algorithm becomes more complicated. However, the superficial reader can take it for granted that parameter estimates and their limit points are reachable, without losing appreciation of what is going on.

ALGORITHM 3.3.2

We will introduce the algorithm inductively. Choose any sequence $\{\epsilon_k\}$ and any sequence $\{C_k\}$ such that:

$$\epsilon_k \downarrow 0 \text{ and } C_k \uparrow \infty \tag{3.1}$$

INITIALIZATION $(\hat{A}_0,\hat{b}_0)$: arbitrarily, $h_0 = 0, j_0 = 0, x(0)$: given.

RECURSION

$$\delta_k = d(\hat{A}_k,\hat{b}_k) \tag{3.2}$$

($d(A,b)$ denotes the determinant of the controllability matrix of (A,b)).

if $h_k = 0$

then:

{ if $\delta_k \geq \epsilon_{j_k}$

then:

$$\{\ u(k) = f(\hat{A}_k,\hat{b}_k)x(k) \tag{3.2.a.1}$$

$$h_{k+1} = h_k \tag{3.2.b.1}$$

$$j_{k+1} = j_k\ \} \tag{3.2.d.1}$$

else (if $\delta_k < \epsilon_{j_k}$)

$$\{\ u(k) = C_{j_k}\|x(k)\| \tag{3.2.a.2}$$

$$h_{k+1} = n \tag{3.2.b.2}$$

$$\tau_{j_k} = k \tag{3.2.c.2}$$

$$j_{k+1} = j_k+1\ \}\} \tag{3.2.d.2}$$

else (if $h_k > 0$)

$$\{\ u(k) = 0 \tag{3.2.a.3}$$

$$h_{k+1} = h_k - 1 \tag{3.2.b.3}$$

$$j_{k+1} = j_k\ \} \tag{3.2.d.3}$$

$$\{\ x(k+1) = A_0x(k)+b_0u(k) \tag{3.2.e}$$

$$\hat{x}(k+1) = \hat{A}_kx(k)+\hat{b}_ku(k) \tag{3.2.f}$$

$$\hat{A}_{k+1} = \hat{A}_k + (\|u(k)\|^2+\|x(k)\|^2)^{-1}(x(k+1)-\hat{x}(k+1))x(k)^T \tag{3.2.g}$$

$$\hat{b}_{k+1} = \hat{b}_k + (\|u(k)\|^2+\|x(k)\|^2)^{-1}(x(k+1)-\hat{x}(k+1))u(k)\ \} \tag{3.2.h}$$

□

REMARK The division in (3.2.g) and (3.2.h) can of course only be done if $x(k)$ or $u(k)$ is non-zero. Therefore, if $x(k_0) = 0$ for some k_0, we do not change the estimates anymore and we take $u(k) = 0$ for all $k \geq k_0$. For the analysis of the algorithm we will assume that $x(k) \neq 0$ for all k. This assumption is also needed if we want to apply Theorem 2.2.1.

COMMENT Let us first explain how the $(k+1)-th$ estimate, $(\hat{A}_{k+1},\hat{b}_{k+1})$ of (A_0,b_0) is calculated from $(\hat{A}_k,\hat{b}_k,u(k))$. Suppose $(\hat{A}_k,\hat{b}_k)$ has been calculated and that $u(k)$ has been applied to the true system, this gives:

$$x(k+1)=A_0x(k)+b_0u(k) \tag{3.3}$$

Define

$$\hat{G}_{k+1}:=\{(A,b)\in\mathbf{R}^{n\times n}\times\mathbf{R}^{n\times 1} \mid Ax(k)+bu(k)=x(k+1)\} \tag{3.4}$$

then $\hat{G}_{k+1}$ is exactly the set of those parameters that are able to explain the transition from $x(k)$ to $x(k+1)$, given $u(k)$. Since a fortiori $(A_0,b_0)\in\hat{G}_{k+1}$, it is natural to choose $(\hat{A}_{k+1},\hat{b}_{k+1})$ somewhere in $\hat{G}_{k+1}$. Since $\hat{G}_{k+1}$ is linear affine, we can take the orthogonal projection of $(\hat{A}_k,\hat{b}_k)$ on $\hat{G}_{k+1}$. One may check that the recursions for $(\hat{A}_k,\hat{b}_k)$ as defined by (3.2.g, 3.2.h) are indeed based on this geometrical consideration. As a first consequence we have that $(\hat{A}_{k+1},\hat{b}_{k+1})$ will be closer to (A_0,b_0) than to $(\hat{A}_k,\hat{b}_k)$ and hence the sequence of estimates is bounded. The idea of orthogonal projection is not new, it was already used in [20] and [6].

The algorithm is obviously recursive. Also, since $(\hat{A}_{k+1},\hat{b}_{k+1})=(\hat{A}_k,\hat{b}_k)$ if and only if $x(k+1)=\hat{x}(k+1)$, neutrality and sensitivity are guaranteed.

Also it is based on certainty equivalence except in the case that the special inputs are applied. In Lemma 2.3.3 we formally prove that these special inputs are applied only a finite number of times, but first we will explain intuitively how $u(k)$ is calculated. Of course one would prefer to take $u(k)=f(\hat{A}_k,\hat{b}_k)x(k)$, for all k, however it is always possible that $(\hat{A}_k,\hat{b}_k)$ is non-reachable, which makes it impossible to calculate $f(\hat{A}_k,\hat{b}_k)$. If we assume extra knowledge of the system, for instance if we assume that (A_0,b_0) belongs to a known convex subset of E_{re}, then $(\hat{A}_k,\hat{b}_k)$ will be reachable for all k, but we want to have a global result. Another possibility is to inject sufficiently exciting input signals to force $(\hat{A}_k,\hat{b}_k)$ to converge to (A_0,b_0). In finite time $(\hat{A}_k,\hat{b}_k)$ will then be in a convex neighborhood of (A_0,b_0) contained in E_{re}. But we do not want to add external signals all the time. What we do is the following. δ_k measures how close $(\hat{A}_k,\hat{b}_k)$ is to non-reachable. If δ_k is large, we take $u(k)=f(\hat{A}_k,\hat{b}_k)x(k)$, if δ_k is small (measured by the sequence ϵ_j) we start an alternative procedure. First we take a large input, large compared to the norm of $x(k)$. Then we apply n times the zero-input, and the distance checking procedure starts again. We denote by τ_j the time instant at which the alternative procedure starts for the $j-th$ time. τ_1 is the first time that an estimate $(\hat{A}_k,\hat{b}_k)$ is closer to the boundary of the set of reachable pairs than ϵ_1. τ_{j+1} is the first time after τ_j+n that an estimate is closer to the boundary of the set of reachable pairs than ϵ_{j+1}. At time τ_j, the input is taken to be $C_j\|x(\tau_j)\|$, after which we apply the zero input for n time steps and the distance checking procedure starts again. The sequence τ_j constructed in this fashion can be interpreted as a sequence of stopping times (terminology borrowed from the theory of stochastic processes).

As soon as $k=\infty$ (which can take quite a long time!), it is easy to see that:

$$\tau_0=\min\{j \mid d(\hat{A}_j,\hat{b}_j)\leqslant\epsilon_0\} \tag{3.5}$$

$$\tau_k=\min\{j\geqslant\tau_{k-1}+n+1 \mid d(\hat{A}_j,\hat{b}_j)\leqslant\epsilon_k\} \tag{3.6}$$

The minimum is understood to be infinity if the set over which the minimization takes place is empty.
Suppose now that there is an infinite number of finite stopping times. Then essentially what happens is the following. Due to the growing inputs at time τ_k, the dynamics of the system (i.e. the matrix A_0), will be dominated by the input. As a consequence b_0 will be identified asymptotically. Moreover, the states $x(\tau_k+1)$ will converge to the subspace spanned by b_0. Finally, since we apply zero-inputs and because of the controllability of (A_0,b_0), the states $x(\tau_k+1),\ldots,x(\tau_k+n)$ will asymptotically span the whole state space. This means that asymptotically we will measure the action of A_0 on the whole state space and hence A_0 will be identified too. In other words $(\hat{A}_k,\hat{b}_k)$ will converge to (A_0,b_0). But since (A_0,b_0) is reachable, it has a positive distance to the boundary of E_{re}. However, the assumption that the set of finite stopping times is infinite and the fact that ϵ_j tends to zero, imply that the limit points of $(\hat{A}_{\tau_k},\hat{b}_{\tau_k})$ are non-reachable. This is a contradiction and hence the number of finite stopping times is finite. This procedure of avoiding that estimates come too close to the boundary of the set of reachable pairs is, of course, not exclusively applicable to pole assignment. It can be used for every adaptive control problem where pole-zero cancellation of the estimates can occur. It proves that a search through the (A,b)-space can be done as long as one is willing to accept temporary alternative inputs. Moreover, the alternative procedure is started and switched off automatically, which is completely in the spirit of adaptive control. The inputs $u(\tau_k),\ldots,u(\tau_{\max})$ can be interpreted as an initial excitation signal, not for identification purposes but only to avoid pole-zero cancellation of the estimates. The sequences ϵ_j, C_j can be seen as design parameters.
We know now that after some time instant we will always use $u(k)=f(\hat{A}_k,\hat{b}_k)x(k)$. The original motivation of this control policy combined with the projection on $\hat{G}_{k+1}$ lies in the theory described in II.2. For suppose that the sequence of estimates converges to (A,b) say. Then (A,b) is an invariant point of the algorithm and hence it should satisfy:

$$A+bf(A,b)\mid_{\mathfrak{X}} = A_0+b_0f(A,b)\mid_{\mathfrak{X}} \tag{3.7}$$

where $\mathfrak{X}$ is the invariant subspace spanned by the asymptotic state trajectory. From Theorem II.2.3 we can then conclude that $f(A,b)\mid_{\mathfrak{X}} = f(A_0,b_0)\mid_{\mathfrak{X}}$, and in particular that asymptotically the applied input equals the desired input.

□

In the next three lemmata we will derive some basic properties of the algorithm including the finiteness of the set of finite stopping times.

LEMMA 2.3.1 $\|(\hat{A}_k,\hat{b}_k)-(A_0,b_0)\|$ is a decreasing sequence, hence it converges to some real constant $R \geqslant 0$.

PROOF This is a direct consequence of the orthogonal projection feature, which assures that $\|(\hat{A}_k,\hat{b}_k)-(A_0,b_0)\| \geqslant \|(\hat{A}_{k+1},\hat{b}_{k+1})-(A_0,b_0)\|$.

□

Although Lemma 2.3.1 is very simple, not to say trivial, it is an important feature of our algorithm. A direct consequence of 3.1 is that $(\hat{A}_k,\hat{b}_k)$ converges to a sphere with centre (A_0,b_0) and radius R. If $R=0$ then $(A_k,b_k)\to(A_0,b_0)$ and we are done. In the sequel we will therefore assume that $R>0$.

LEMMA 2.3.2 $\lim_{k\to\infty} \|[(\hat{A}_{k+1},\hat{b}_{k+1})-(\hat{A}_k,\hat{b}_k)]\| = 0$

PROOF Suppose the claim is not true. Then there exists $\epsilon>0$ and a sequence $\{s_k\}$, such that for all k: $\|[(\hat{A}_{s_k+1},\hat{b}_{s_k+1})-(\hat{A}_{s_k},\hat{b}_{s_k})]\| \geqslant \epsilon$.
Now denote $\|(\hat{A}_k,\hat{b}_k)-(A_0,b_0)\|$ by r_k. Choose $\delta>0$ and let k_0 be such that $R\leqslant r_{s_k}\leqslant R+\delta$ for all $k \geqslant k_0$. Using Pythagoras' theorem we see that for all $k \geqslant k_0$:

$$r_{s_k}-r_{1+s_k} \geqslant r_{s_k}-(r_{s_k}^2-\epsilon^2)^{1/2} \tag{3.8}$$

$$\geqslant R-((R+\delta)^2-\epsilon^2)^{1/2} = C \tag{3.9}$$

for some positive constant C and δ sufficiently small.
Since r_k is non-increasing we have $r_{s_k}-r_{s_{k+1}} \geqslant C$, which yields:

$$r_{s_k}<r_{s_{k_0}}-C(k-k_0)\leqslant R+\delta-C(k-k_0). \tag{3.10}$$

Hence there exists k such that $r_{s_k}<R$, which is a contradiction.

□

LEMMA 2.3.3 $\{\tau_k \mid k\in\mathbb{N},\ \tau_k<\infty\}$ is finite.

PROOF Suppose the contrary. Assume that $(\hat{A}_{\tau_k},\hat{b}_{\tau_k})$ converges, say $\lim_{k\to\infty}(\hat{A}_{\tau_k},b_{\tau_k}) = (A,b)$. (Otherwise take a suitable subsequence). Then, for all k:

$$d(A,b) < \epsilon_k \tag{3.11}$$

Hence (A,b) is non-reachable. Since for all k: $(\hat{A}_{k+1},\hat{b}_{k+1})\in\hat{G}_{k+1}$, we have:

$$x(k+1) = \hat{A}_{k+1}x(k)+\hat{b}_{k+1}u(k) \tag{3.12.a}$$

$$= A_0x(k)+b_0u(k) \tag{3.12.b}$$

In particular:

$$\frac{x(\tau_k+1)}{\|x(\tau_k+1)\|} = \frac{\hat{A}_{\tau_k+1}x(\tau_k)+\hat{b}_{\tau_k+1}C_k\|x(\tau_k)\|}{\|\hat{A}_{\tau_k+1}x(\tau_k)+\hat{b}_{\tau_k+1}C_k\|x(\tau_k)\|\|} \tag{3.13}$$

$$= \frac{A_0x(\tau_k)+b_0C_k\|x(\tau_k)\|}{\|A_0x(\tau_k)+b_0C_k\|x(\tau_k)\|\|} \tag{3.14}$$

Now:

$$\lim_{k\to\infty} \frac{x(\tau_k+1)}{\|x(\tau_k+1)\|} = \lim_{k\to\infty} \frac{A_0x(\tau_k)+b_0C_k\|x(\tau_k)\|}{\|A_0x(\tau_k)+b_0C_k\|x(\tau_k)\|\ \|} \tag{3.15}$$

$$= \lim_{k\to\infty} \frac{A_0\dfrac{x(\tau_k)}{\|x(\tau_k)\|}+b_0C_k}{\|A_0\dfrac{x(\tau_k)}{\|x(\tau_k)\|}+b_0C_k\|} \tag{3.16}$$

$$= \lim_{k\to\infty} \frac{b_0C_k}{\|A_0\dfrac{x(\tau_k)}{\|x(\tau_k)\|}+b_0C_k\|} = \frac{b_0}{\|b_0\|} = \rho b_0 \tag{3.17}$$

for some $\rho\neq 0$. On the other hand, if we take the limit in (3.13), we obtain:

$$\rho b \tag{3.18}$$

Hence:

$$\lim_{k\to\infty} \hat{b}_{\tau_k} = b_0 \tag{3.19}$$

From (3.2.a.3) and (3.19) we can now conclude that:

$$\lim_{k\to\infty} \frac{x(\tau_k+i)}{\|x(\tau_k+i)\|} = \frac{A_0^{i-1}b_0}{\|A_0^{i-1}b_0\|} \qquad i=1,..,n+1 \tag{3.20}$$

Since for all k: $x(k+1) = \hat{A}_{k+1}x(k)+\hat{b}_{k+1}u_k$, we have:

$$x(\tau_k+i+1) = \hat{A}_{\tau_k+i+1}x(\tau_k+i) = A_0x(\tau_k+i) \tag{3.21}$$

$$\hat{A}_{\tau_k+i+1}\frac{x(\tau_k+i)}{\|x(\tau_k+i)\|} = \frac{A_0x(\tau_k+i)}{\|x(\tau_k+i)\|} \tag{3.22}$$

Taking limits on both sides gives:

$$AA_0^{i-1}b_0 = A_0A_0^{i-1}b_0 \qquad i=1,..,n \tag{3.23}$$

Since (A_0,b_0) is reachable, we conclude that $A=A_0$ and hence:

$$\lim_{k\to\infty} (\hat{A}_{\tau_k},\hat{b}_{\tau_k}) = (A_0,b_0) \tag{3.24}$$

And thus

$$(A,b) = (A_0,b_0) \tag{3.25}$$

Since by assumption (A,b) is non-reachable, we have a contradiction and the statement follows.

□

COROLLARY 2.3.4 For all k sufficiently large, $(\hat{A}_k,\hat{b}_k)$ is reachable and moreover

all the limit points of $\{\hat{A}_k,\hat{b}_k\}$ are reachable.

ANALYSIS OF THE ALGORITHM

The properties of the algorithm will be derived in several steps. First we will state our main result.

THEOREM 2.3.5 Consider the (controlled) system (2.1,3.2), there exists a sequence of matrices $\{\Delta_k\}_{k\in\mathbf{N}}$, such that:

$$i)\ x(k+1) = (A_0+b_0f(\hat{A}_k,\hat{b}_k))x(k) \tag{3.26}$$

$$= (A_0+b_0f(A_0,b_0)+\Delta_k)x(k) \tag{3.27}$$

$$ii)\ \lim_{k\to\infty}\Delta_k = 0 \tag{3.28}$$

COMMENT. Theorem 2.3.5 tells us that asymptotically the action of the closed-loop matrix is identical to that of the optimal closed-loop matrix. It should be noticed that we do not claim that the real closed-loop matrix converges to the optimal one, but only as far as the action on the real state trajectory is concerned. This weaker form of convergence is not surprising, if we realize the fact that the estimation procedure only receives information about the action of the real closed-loop matrix on the state trajectory. We propose the term 'weak self-tuning' for this kind of behavior. Self-tuning would have implied that $\lim_{k\to\infty} A_0+b_0f(\hat{A}_k,\hat{b}_k) = A_0+B_0f(A_0,b_0)$, which we do not claim. Note that the above result is valid whether or not Λ is contained in the unit disk. This shows that the adaptation part of the algorithm does not depend on the stability properties of the closed-loop system. The reason that the result holds even for the unstable case, is that the estimation part of the algorithm depends on the direction of $x(k)$ (i.e. $\frac{x(k)}{\|x(k)\|}$), rather than $x(k)$ itself. The normalization plays an important role in the proof of Theorem 2.3.5. Of course for stability of the closed-loop system it is needed that Λ is contained in the unit disk.

□

LEMMA 2.3.6 $\lim_{k\to\infty}[(A_0+b_0f(\hat{A}_k,\hat{b}_k))-(\hat{A}_k+\hat{b}_kf(\hat{A}_k,\hat{b}_k))]\frac{x(k)}{\|x(k)\|} = 0$

PROOF By Lemma 2.3.3 there exists k_0 such that $u_k = f(\hat{A}_k,\hat{b}_k)x(k)$ for all $k > k_0$. Hence by definition of $\hat{G}_{k+1}$, we have for all $k > k_0$:

$$[(A_0+b_0f(\hat{A}_k,\hat{b}_k))-(\hat{A}_{k+1}+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k))]\frac{x(k)}{\|x(k)\|} = 0 \tag{3.29}$$

Using Lemma 2.3.2, Corollary 2.3.4, and taking limits at both sides of (3.29) gives the statement.

□

THEOREM 2.3.7

(i) $\lim_{k\to\infty} \|(f(\hat{A}_k,\hat{b}_k)-f(A_0,b_0))\frac{x(k)}{\|x(k)\|}\| = 0.$

(ii) $\lim_{k\to\infty} [(A_0+b_0f(\hat{A}_k,\hat{b}_k))-(A_0+b_0f(A_0,b_0))]\frac{x(k)}{\|x(k)\|} = 0.$

PROOF (i) Suppose the claim is not true. Then there exist $\epsilon>0$ and a subsequence $\{s_k\}$, such that for all k:

$$\|(f(\hat{A}_{s_k},\hat{b}_{s_k})-f(A_0,b_0))\frac{x(s_k)}{\|x(s_k)\|}\| \geq \epsilon \tag{3.30}$$

Assume that $\{s_k\}$ was already such that:

$$\lim_{k\to\infty} (\hat{A}_{s_k},\hat{b}_{s_k}) = (A,b) \tag{3.31}$$

$$\lim_{k\to\infty} \frac{x(s_k)}{\|x(s_k)\|} = x^* \tag{3.32}$$

for some $(A,b)\in E_{re}$ and $x^*\in\mathbb{R}^n$. Then for all l:

$$\lim_{k\to\infty} (\hat{A}_{l+s_k},\hat{b}_{l+s_k}) = (A,b) \tag{3.33}$$

Define $\bar{x}_k$ and $\bar{z}_k$ by:

$$\bar{x}_k = (A+bf(A,b))^k x^*, \qquad k=0,1,2,.. \tag{3.34}$$

$$\bar{z}_k = (A_0+b_0f(A,b))^k x^*, \qquad k=0,1,2,.. \tag{3.35}$$

Then by Lemma 2.3.6 and (3.33), we have for all k:

$$\bar{x}_k = \bar{z}_k \tag{3.36}$$

Hence by Theorem II.2.3 it follows that in particular:

$$f(A,b)x^* = f(A_0,b_0)x^* \tag{3.37}$$

This contradicts (3.30), and the statement follows.
(ii) This follows from Lemma 2.3.6 and (i).

□

PROOF of Theorem 2.3.5:
This is now a direct application of Theorem 2.3.7 and Theorem 2.2.1.

□

REMARK In [50,53] Theorem 2.3.5 was proven in a slightly different way. There the notion of excitation space was introduced. This is the space spanned by the limit points of $\{\frac{x(k)}{\|x(k)\|}\}_{k\in\mathbb{N}}$. Denote this space by $\mathfrak{X}$. From Theorem 2.3.7 we can then derive that:

$$\lim_{k\to\infty} [f(\hat{A}_k,\hat{b}_k)-f(A_0,b_0)]x = 0 \tag{3.38}$$

for all $x \in \mathcal{X}$, which implies that only on the asymptotic active part of the state space the action of $f(\hat{A}_k, \hat{b}_k)$ is as desired. This illustrates the term weak self-tuning. Also it can be proven that $\mathcal{X}$ is invariant under $A_0 + b_0 f(A_0, b_0)$.

2.4. An algorithm for the unobserved state case

We will now propose an adaptive pole assignment algorithm for the class of single-input/single-output discrete time systems of (known) order n. The algorithm is based on ideas developed in the previous section. There it was assumed that the state of the system was observed. This assumption is now relaxed, and hence the algorithm should also contain an observer. Indeed, the algorithm consists of an estimation part including an adaptive observer and a control part. However, one can also view the estimation part as an adaptive partial realization procedure, since no attempt is made to identify the system parameters completely, and the true state trajectory (whatever that may be, since there is no such thing as a true state trajectory) is not reconstructed either. What we really end up with is not a complete realization of the unknown system, but an input/state/output description that is suitable for one input/output sequence, namely the asymptotic one. This section runs very much parallel to section 2 and the reader is referred to that section for some of the details and discussions.

We will first give three different descriptions of the system. Secondly we will introduce the algorithm. Then we will prove some basic properties. Next we will formulate the main theorem of this section: the characterization of the asymptotic closed-loop behavior of the controlled system. Finally, we will give the analysis of the algorithm, ultimately leading to the proof of the main theorem.

THE SYSTEM:

The true system is supposed to be linear, time-invariant, single-input, single-output and of known order n. Hence it has an input/output description of the form:

$$y(k+1) = a_0^0 y(k) + .. + a_{n-1}^0 y(k-n+1) + b_0^0 u(k) + .. + b_{n-1}^0 u(k-n+1) \quad (4.1)$$

Since we want to work in i/s/o form, we realize (4.1) as follows: Define $(A_0, b_0, c_0) \in \mathbf{R}^{n \times n} \times \mathbf{R}^{n \times 1} \times \mathbf{R}^{1 \times n}$, by:

$$A_0 := \begin{bmatrix} a_0^0 & 1 & \cdots & 0 \\ \vdots & 0 & & \vdots \\ \vdots & \vdots & & 0 \\ \vdots & \vdots & & 1 \\ a_{n-1}^0 & 0 & \cdots & 0 \end{bmatrix} \qquad b_0 := \begin{bmatrix} b_0^0 \\ \vdots \\ \vdots \\ \vdots \\ b_{n-1}^0 \end{bmatrix} \qquad c_0 := [1\ 0\ ..0] \quad (4.2)$$

And define for every k, $x(k) \in \mathbf{R}^n$, by:

$$x_1(k) = y(k) \quad (4.3.a)$$

$$x_i(k) = \sum_{j=0}^{n-i} a^0_{i+j-1} y(k-1-j) + \sum_{j=0}^{n-i} b^0_{i+j-1} u(k-1-j) \qquad i = 2,..,n \quad (4.3.b)$$

Then for all k:

$$x(k+1) = A_0 x(k) + b_0 u(k) \quad (4.4.a)$$

$$y(k) = c_0 x(k) \quad (4.4.b)$$

Although there are of course many other realizations of 4.1, we will refer to 4.4 as the true realization, and to the sequence $\{x(k)\}_{k\in\mathbb{N}}$ as the true state trajectory.

We will also need the following non-minimal realization of 4.1. Define $F_0 \in \mathbb{R}^{(2n-1)\times(2n-1)}$ and $g_0 \in \mathbb{R}^{(2n-1)\times 1}$ by:

$$F_0 := \begin{bmatrix} a^0_0 & \cdot & a^0_{n-2} & a^0_{n-1} & b^0_1 & \cdot & b^0_{n-2} & b^0_{n-1} \\ 1 & & 0 & 0 & 0 & & 0 & 0 \\ 0 & & \vdots & \vdots & \vdots & & \vdots & \vdots \\ \vdots & & 0 & \vdots & \vdots & & \vdots & \vdots \\ \vdots & & 1 & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & & 0 & \vdots & 1 & & \vdots & \vdots \\ \vdots & & \vdots & \vdots & 0 & & \vdots & \vdots \\ \vdots & & \vdots & \vdots & \vdots & & 0 & \vdots \\ 0 & \cdot & 0 & 0 & 0 & \cdot & 1 & 0 \end{bmatrix} \qquad g_0 := \begin{bmatrix} b^0_0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix} \quad (4.5)$$

Define:

$$\phi(k) := [y(k), y(k-1), .., y(k-n+1), u(k-1), .., u(k-n+1)]^T \quad (4.6)$$

then for all k:

$$\phi(k+1) = F_0 \phi(k) + g_0 u(k) \quad (4.7)$$

Moreover, since (A_0, b_0) is reachable, so is (F_0, g_0). Finally define $M \in \mathbb{R}^{n\times(2n-1)}$ by:

$$M := \begin{bmatrix} 1 & 0 & 0 & . & . & 0 & 0 & 0 & . & . & 0 \\ 0 & a_1 & a_2 & & & a_{n-1} & b_1 & b_2 & & & b_{n-1} \\ \vdots & a_2 & a_3 & & . & 0 & b_2 & b_3 & & . & 0 \\ \vdots & \vdots & \vdots & . & & \vdots & \vdots & \vdots & . & & \vdots \\ \vdots & \vdots & a_{n-1} & & & \vdots & \vdots & b_{n-1} & & & \vdots \\ 0 & a_{n-1} & 0 & . & . & 0 & b_{n-1} & 0 & . & . & 0 \end{bmatrix} \quad (4.8)$$

then for all k we have by (4.3) that:

$$x(k) = M\phi(k) \quad (4.9)$$

THE ALGORITHM 2.4.11
Choose any sequence $\{\epsilon_k\}$ and any sequence $\{C_k\}$ such that:

$$\epsilon_k \downarrow 0 \text{ and } C_k \uparrow \infty \tag{4.10}$$

INITIALIZATION $(\hat{A}_0, \hat{b}_0)$: arbitrarily, $h_0 = 0$, $j_0 = 0$, $\phi(0)$: given.

RECURSION

$$\hat{A}_k := \begin{bmatrix} \hat{a}_0(k) & 1 & \cdots & 0 \\ \vdots & 0 & & \vdots \\ \vdots & \vdots & & 0 \\ \vdots & \vdots & & 1 \\ \hat{a}_{n-1}(k) & 0 & \cdots & 0 \end{bmatrix} \qquad \hat{b}_k := \begin{bmatrix} \hat{b}_0(k) \\ \vdots \\ \vdots \\ \vdots \\ \hat{b}_{n-1}(k) \end{bmatrix} \tag{4.11.A}$$

$$\hat{\theta}(k) := [\hat{a}_0(k),..,\hat{a}_{n-1}(k),\hat{b}_0(k),..,\hat{b}_{n-1}(k)] \tag{4.11.B}$$

$$\hat{M}_k := \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \hat{a}_1(k) & & \hat{a}_{n-1}(k) & \hat{b}_1(k) & & \hat{b}_{n-1}(k) \\ \vdots & \hat{a}_2(k) & \cdots & 0 & \hat{b}_2(k) & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & \hat{a}_{n-1}(k) & \cdots & 0 & \hat{b}_{n-1}(k) & \cdots & 0 \end{bmatrix} \tag{4.11.C}$$

$$\delta_k = d(\hat{A}_k, \hat{b}_k) \tag{4.11.D}$$

($d(A,b)$ denotes the determinant of the controllability matrix of (A,b)).

if $h_k = 0$

then:

{ if $\delta_k \geqslant \epsilon_{j_k}$

then:

$$\{\ u(k) = f(\hat{A}_k, \hat{b}_k)\hat{z}(k) \tag{4.11.a.1}$$

$$h_{k+1} = h_k \tag{4.11.b.1}$$

$$j_{k+1} = j_k\ \} \tag{4.11.d.1}$$

else (if $\delta_k < \epsilon_{j_k}$)

$$\{\ u(k) = C_{j_k} \|\phi(k)\| \tag{4.11.a.2}$$

$$h_{k+1} = 2n-1 \tag{4.11.b.2}$$

$$\tau_{j_k} = k \tag{4.11.c.2}$$

$$j_{k+1} = j_k + 1 \;\}\} \tag{4.11.d.2}$$

else (if $h_k > 0$)

$$\{\, u(k) = 0 \tag{4.11.a.3}$$

$$h_{k+1} = h_k - 1 \tag{4.11.b.3}$$

$$j_{k+1} = j_k \;\} \tag{4.11.d.3}$$

$$\{\, y(k+1) = a_0^0 y(k) + .. + a_{n-1}^0 y(k-n+1) + b_0^0 u(k) + .. + b_{n-1}^0 u(k-n+1) \tag{4.11.e}$$

$$\hat{y}(k+1) = \hat{\theta}(k)\,\phi(k) \tag{4.11.f}$$

$$\lambda(k) = (\|\phi(k)\|^2)^{-1}[y(k+1) - \hat{y}(k+1)] \tag{4.11.g}$$

$$\hat{a}_i(k+1) = \hat{a}_i(k) + \lambda(k) y(k-i) \qquad i = 0,..,n-1 \tag{4.11.h}$$

$$\hat{b}_i(k+1) = \hat{b}_i(k) + \lambda(k) u(k-i) \qquad i = 0,..,n-1 \tag{4.11.i}$$

$$\hat{x}(k) = \hat{M}_{k+1}\phi(k) \tag{4.11.j}$$

$$\hat{z}(k+1) = \hat{M}_{k+1}\phi(k+1) \;\} \tag{4.11.k}$$

□

REMARK The division in (4.11.g) can of course be done only if $\phi(k)$ or $u(k)$ is non-zero. Therefore if $\phi(k_0) = 0$ for some k_0, we do not change the estimates anymore and we take $u(k) = 0$ for all $k \geqslant k_0$. For the analysis of the algorithm we will assume that $\phi(k) \neq 0$ for all k. This assumption is also needed if we want to apply Theorem 2.2.1.

□

COMMENT The interpretation of the algorithm is more or less the same as for the observed state case, the main difference being that now also a state trajectory has to be invented too. Define:

$$\hat{G}_{k+1} := \{(a_0,..,a_{n-1},b_0,..,b_{n-1}) \,|\, y(k+1) = \sum_{i=0}^{n-1} (a_i y(k-i) + b_i u(k-i))\;\} \tag{4.12}$$

Then $\hat{G}_{k+1}$ is linear affine and $(a_0^0,..,a_{n-1}^0,b_0^0,..,b_{n-1}^0) \in \hat{G}_{k+1}$. Define $(\hat{a}_0(k+1),..,\hat{a}_{n-1}(k+1),\hat{b}_0(k+1),..,\hat{b}_{n-1}(k+1))$ as the orthogonal projection of $(\hat{a}_0(k),..,\hat{a}_{n-1}(k),\hat{b}_0(k),..,\hat{b}_{n-1}(k))$ on G_{k+1}.

□

THE OBSERVER

The definition of $\hat{x}(k)$ and $\hat{z}(k+1)$ is motivated by the following analysis.
At time $k+1$ we compute $(\hat{a}_0(k+1),..,\hat{a}_{n-1}(k+1),\hat{b}_0(k+1),..,\hat{b}_{n-1}(k+1))$ on

the basis of the observed data $(u(k-n+1),y(k-n+1),..,u(k),y(k),y(k+1))$. Suppose we want to have an i/s/o description of this finite i/o sequence:

$$\bar{x}(j+1) = \hat{A}(k+1)\bar{x}(j)+\hat{b}(k+1)u(j) \qquad j=k-n+1,..,k \tag{4.13.a}$$

$$y(j) = c_0\bar{x}(j) \tag{4.13.b}$$

Let $\bar{x}$ be the unique solution of the equations:

$$c_0\hat{A}_{k+1}^{j-1}\bar{x} + c_0\sum_{l=0}^{j-2}\hat{A}_{k+1}^{l}\hat{b}_{k+1}u(k-n+j-1-l) = y(k-n+j) \tag{4.14}$$

$$j=1,..,n$$

Then, if $(\hat{A}_{k+1},\hat{b}_{k+1})$ was the true parameter, $\bar{x}$ would have been the estimate of $x(k-n+1)$ based on the dead-beat observer for (A_0,b_0). Since we use $(\hat{A}_{k+1},\hat{b}_{k+1})$ instead of (A_0,b_0), the observer part of the algorithm can be interpreted as a certainty equivalence dead-beat observer. Now one may check that:

$$\hat{x}(k) = \hat{A}_{k+1}^{n-1}\bar{x} + \sum_{l=0}^{n-2}\hat{A}_{k+1}^{l}\hat{b}_{k+1}u(k-n+j-1-l) \tag{4.15}$$

$$\hat{z}(k+1) = \hat{A}_{k+1}\hat{x}(k)+\hat{b}_{k+1}u(k) \tag{4.16}$$

where $\hat{x}(k)$ and $\hat{z}(k+1)$ are defined by (4.11.j) and (4.11.k).

□

The next three lemmata give some essential properties of the algorithm.

LEMMA 2.4.1 $\|(\hat{A}_k,\hat{b}_k)-(A_0,b_0)\|$ is a decreasing sequence, hence it converges to some real constant $R \geqslant 0$.

PROOF See the proof of Lemma 2.3.1.

□

LEMMA 2.4.2 $\lim_{k\to\infty} \|[(\hat{A}_{k+1},\hat{b}_{k+1})-(\hat{A}_k,\hat{b}_k)]\| = 0$

PROOF See the proof of Lemma 2.3.2.

□

LEMMA 2.4.3 $\{\tau_k \mid k\in\mathbf{N},\ \tau_k<\infty\}$ is finite.

PROOF Suppose the contrary. Assume that $(\hat{A}_{\tau_k},\hat{b}_{\tau_k})$ converges, say $\lim_{k\to\infty}(\hat{A}_{\tau_k},b_{\tau_k}) = (A,b)$. (Otherwise take a suitable subsequence). Then (A,b) is non-reachable. Define $(\hat{F}_k,\hat{g}_k)\in\mathbf{R}^{(2n-1)\times(2n-1)}\times\mathbf{R}^{(2n-1)\times 1}$ by replacing a_i^0 by $\hat{a}_i(k)$ and b_i^0 by $\hat{b}_i(k)$ in (4.2). Now the proof is completely analogous to the proof of Lemma 2.3.3.

□

COROLLARY 2.4.4 For all k sufficiently large, $(\hat{A}_k,\hat{b}_k)$ is reachable and moreover all the limit points of $\{\hat{A}_k,\hat{b}_k\}$ are reachable.

□

The analysis of (4.11) is as follows: first we will prove that the sequences $\{\hat{z}(k+1)\}$ and $\{(\hat{A}_k,\hat{b}_k)\}$ provide asymptotically a realization of the controlled system. Then we will apply Theorem II.2.4 to the limiting behavior of the system to connect the true state trajectory with the constructed one.

ANALYSIS OF THE ALGORITHM

THEOREM 2.4.5 Consider the (controlled) system (2.1,4.11). Assume that there exists $\lambda \in \Lambda$ such that $\lambda \neq 0$. Then there exists a sequence of matrices $\{\Delta_k\}_{k\in\mathbb{N}}$, such that for k sufficiently large:

$$\text{(i) } x(k+1) = A_0x(k)+b_0f(\hat{A}_k,\hat{b}_k)\hat{z}(k) \tag{4.17}$$

$$= (A_0+b_0f(A_0,b_0)+\Delta_k)x(k)) \tag{4.18}$$

$$\text{(ii) } \lim_{k\to\infty} \Delta_k = 0 \tag{4.19}$$

REMARK Just as in Theorem 2.3.5 we do not claim that (A_0,b_0) is identified, nor is $f(A_0,b_0)$ identified. Even the state trajectory is not reconstructed. The constructed state trajectory $\hat{z}(k)$ will in general not be equal to $x(k)$, nor will $f(\hat{A}_k,\hat{b}_k)$ be close to $f(A_0,b_0)$. In the limit, both $f(\hat{A}_k,\hat{b}_k)$ and $\hat{z}(k)$ may be wrong, but the resulting input sequence $u(k) = f(A_k,b_k)\hat{z}(k)$ will be as desired, and that is what really matters. Again this could be seen as a weak form of self-tuning. As in section 5, the above result is valid whether or not Λ is contained in the unit disk. This shows that the adaptation part of the algorithm does not depend on the stability properties of the closed-loop system. The reason that the result holds even for the unstable case, is that the estimation part of the algorithm depends on the direction of $\phi(k)$ (i.e. $\frac{\phi(k)}{\|\phi(k)\|}$), rather than $\phi(k)$ itself. This normalization plays an important role in the proof of Theorem 2.4.5. Of course, for stability of the closed-loop system it is needed that Λ is contained in the unit disk.

□

Define:

$$d_k = \|\phi(k)\| = [\sum_{j=0}^{n-1} y^2(k-j)+\sum_{j=1}^{n-1} u^2(k-j)]^{\frac{1}{2}} \tag{4.22}$$

LEMMA 2.4.6

$$\lim_{k\to\infty} \|\frac{1}{d_k}[\hat{z}(k+1)-(\hat{A}_k+\hat{b}_k f(\hat{A}_k,\hat{b}_k))\hat{z}(k)]\| = 0 \tag{4.23}$$

PROOF From Lemma 2.4.3 and (4.11.a.1) we deduce that for k sufficiently large:

$$\hat{z}(k+1) = \hat{A}_{k+1}\hat{x}(k)+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k)\hat{z}(k) \tag{4.24}$$

Hence:

$$\hat{z}(k+1) = (\hat{A}_{k+1}\hat{M}_{k+1}+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k)\hat{M}_k)\phi(k) \tag{4.25}$$

Now, since $\|(\hat{A}_k,\hat{b}_k)-(\hat{A}_{k+1},\hat{b}_{k+1})\| \to 0$, the continuity of f on E, the controllability of the limit points of $(\hat{A}_k,\hat{b}_k)$, we conclude:

$$\lim_{k\to\infty} \|\frac{1}{d_k}[\hat{z}(k+1)-(\hat{A}_{k+1}+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k))\hat{z}(k)\| \tag{4.26}$$

$$= \lim_{k\to\infty} \|(\hat{A}_{k+1}\hat{M}_{k+1}+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k)\hat{M}_k)\frac{\phi(k)}{\|\phi(k)\|} \tag{4.27}$$

$$- (\hat{A}_{k+1}\hat{M}_k+\hat{b}_{k+1}f(\hat{A}_k,\hat{b}_k)\hat{M}_k\frac{\phi(k)}{\|\phi(k)\|}\| = 0$$

□

THEOREM 2.4.7

$$\lim_{k\to\infty} \|\frac{f(A_0,b_0)x(k)-f(\hat{A}_k,\hat{b}_k)\hat{z}(k)}{d_k}\| = 0 \tag{4.28}$$

PROOF Suppose the claim is not true. Then there exist $\epsilon>0$ and a sequence $\{t_k\}$ such that for all k:

$$\|\frac{f(A_0,b_0)x(k)-f(\hat{A}_k,\hat{b}_k)\hat{z}(k)}{d_k}\| \geqslant \epsilon \tag{4.29}$$

Let x^* be a limit point of $\frac{x(t_k)}{d_{t_k}}$. Say $\lim_{k\to\infty} \frac{x(t_k^{(1)})}{d_{t_k^{(1)}}} = x^*$, for some subsequence $\{t_k^{(1)}\}$ of $\{t_k\}$. Let $\{t_k^{(2)}\}$ be a subsequence of $\{t_k^{(1)}\}$, such that:

$$\lim_{k\to\infty} (\hat{A}_{t_k^{(2)}},\hat{b}_{t_k^{(2)}}) = (A,b) \tag{4.30}$$

and:

$$\lim_{k\to\infty} \frac{z(t_k^{(2)})}{d_{t_k^{(2)}}} = z^* \tag{4.31}$$

Then for all l:

$$\lim_{k\to\infty} (\hat{A}_{l+t_k^{(2)}},\hat{b}_{l+t_k^{(2)}}) = (A,b) \tag{4.32}$$

Define sequences $\{x(k)^*\}$ and $\{z(k)^*\}$ as follows:

$$x(0)^* = x^* \qquad z(0)^* := z^* \tag{4.33}$$

$$z(k)^* = (A+bf(A,b))^k z^* \tag{4.34}$$

$$x(k+1)^* = A_0 x(k)^* + b_0 f(A,b) z(k)^* \tag{4.35}$$

Then, by Lemma 2.4.6, the fact that $c\hat{z}(k) = cx(k)$, and (4.32), for all k:

$$cx(k)^* = cz(k)^* \tag{4.36}$$

hence by Theorem II.2.4 there exists a non-singular matrix S such that for all k:

$$Sz(k)^* = x(k)^* \tag{4.37}$$

Hence from the proof of Theorem II.2.2 we conclude:

$$f(A,b)z^* = f(A_0,b_0)x^* \tag{4.38}$$

which contradicts (4.29).

□

REMARK Theorem 2.4.7 tells us that asymptotically the applied input equals the desired input if we normalize the state trajectory with the norm of $\phi(k)$. For the derivation of the same result, but this time normalized with the norm of $x(k)$ (in order to be able to apply Theorem 2.2.1), we will study, as an intermediate step, the behavior of the non-minimal realization (4.7) of (4.1). We have for all k:

$$\phi(k+1) = F_0\phi(k) + g_0 u(k) \tag{4.39}$$

Define $h_0 \in \mathbb{R}^{1\times(2n-1)}$ by:

$$h_0 := f(A_0,b_0)M \tag{4.40}$$

Then the desired input is:

$$u(k) = f(A_0,b_0)x(k) = f(A_0,b_0)M\phi(k) = h_0\phi(k) \tag{4.41}$$

Hence the desired closed-loop representation of (4.39) is given by:

$$\phi(k+1) = (F_0 + g_0 h_0)\phi(k) \tag{4.42}$$

However, the applied control is:

$$u(k) = f(\hat{A}_k,\hat{b}_k)\hat{z}(k) = f(\hat{A}_k,\hat{b}_k)\hat{M}_k\phi(k) \tag{4.43}$$

□

The following theorem characterizes the asymptotic closed-loop behavior of the realization (4.11,4.39).

THEOREM 2.4.8 There exists a sequence of matrices $\{\Delta_k'\}_{k\in\mathbb{N}}$ such that:

$$\text{(i)}\ \phi(k+1) = (F_0 + g_0 f(\hat{A}_k, \hat{b}_k)\hat{M}_k)\phi(k) \tag{4.44}$$

$$= (F_0 + g_0 h_0 + \Delta_k')\phi(k) \tag{4.45}$$

$$\text{(ii)}\ \lim_{k\to\infty} \Delta_k' = 0 \tag{4.46}$$

PROOF The proof follows immediately from Theorem 2.4.7 and Theorem 2.2.1 □

LEMMA 2.4.9 Let $\{M_k\}_{k\in\mathbb{N}}$ be a sequence of matrices in $\mathbb{R}^{n\times n}$, such that $\lim_{k\to\infty} M_k = M \neq 0$. Choose $x(0)\in\mathbb{R}^{n\times n}$ and define:

$$x(k+1) = M_k x(k) \tag{4.47}$$

Assume that for all k $x(k) \neq 0$. Denote by $\mathcal{X}$ the linear span of the limit points of $\frac{x(k)}{\|x(k)\|}$. Then:

$$\text{(i)}\ M\mathcal{X} \subset \mathcal{X} \tag{4.48}$$

$$\text{(ii) for all } x\in\mathcal{X}, x\neq 0 : Mx \neq 0 \tag{4.49}$$

PROOF i) Suppose x^* is a limit point of $\{\frac{x(k)}{\|x(k)\|}\}$. Say $\lim_{k\to\infty} \frac{x(s_k)}{\|x(s_k)\|} = x^*$, for some subsequence $\{s_k\}$. Then:

$$Mx^* = \lim_{k\to\infty} \frac{1}{\|x(s_k)\|} M_{s_k} x(s_k) = \lim_{k\to\infty} \frac{1}{\|x(s_k)\|} x(1+s_k) \tag{4.50}$$

$$= \lim_{k\to\infty} \frac{\|x(1+s_k)\|}{\|x(s_k)\|} \frac{x(1+s_k)}{\|x(1+s_k)\|} = \lim_{k\to\infty} \frac{\|M_{s_k} x(s_k)\|}{\|x(s_k)\|} \frac{x(1+s_k)}{\|x(1+s_k)\|} \tag{4.51}$$

$$= \|M_{s_k} \frac{x(s_k)}{\|x(s_k)\|}\| \frac{x(1+s_k)}{\|x(1+s_k)\|} = \|Mx^*\| \lim_{k\to\infty} \frac{x(1+s_k)}{\|x(1+s_k)\|} \tag{4.52}$$

Hence $Mx^* \in X$. By linearity the result follows.
ii) After a change of basis, M and M_k can be decomposed as:

$$M = \begin{bmatrix} M_{11} & 0 \\ M_{21} & 0 \end{bmatrix}, \quad M_k = \begin{bmatrix} M_{11}(k) & M_{12}(k) \\ M_{21}(k) & M_{22}(k) \end{bmatrix} \tag{4.53}$$

such that M_{11} is non-singular. Then:

$$\lim_{k\to\infty} \frac{x_1(k+1)}{\|x(k)\|} = \lim_{k\to\infty} \frac{(M_{11}(k)x_1(k) + M_{12}(k)x_2(k))}{\|x(k)\|} \neq 0 \tag{4.54}$$

This yields:

$$\frac{x(k)}{\|x(k)\|} \to \text{coker}(M) \tag{4.55}$$

By linearity the statement follows.

□

COROLLARY 2.4.10 Let x^* be a limit point of $\frac{x(k)}{d_k}$, then: $x^* \neq 0$.

PROOF One may check that $\Lambda \subset \text{Spec}(G_0)$. Since by assumption at least one of the $\lambda's$ is nonzero, we conclude by Lemma 2.4.9 that there exists $\mu > 0$ such that for i:

$$\frac{\phi(k+i)}{d_k} \geqslant \mu^i \tag{4.56}$$

Now suppose $x^* = 0$. Say $\frac{x(t_k)}{d_{t_k}} \to 0$, then:

$$\frac{\|x(t_k+1)\|}{d_{t_k}} = \frac{\|A_0 x(k) + b_0 f(\hat{A}_{t_k}, \hat{b}_{t_k})\hat{z}(k)\|}{d_{t_k}} \tag{4.57}$$

$$= \frac{\|(A_0 + b_0 f(A_0,b_0))x(t_k) + b_0(f(\hat{A}_{t_k}, \hat{b}_{t_k})\hat{z}(t_k) - f(A_0,b_0)x(t_k))\|}{d_{t_k}} \to 0 \tag{4.58}$$

In the same way we obtain:

$$\frac{\|x(t_k+l)\|}{d_{t_k}} \to 0 \tag{4.59}$$

This implies:

$$\frac{(y(t_k), y(t_k+1), .., y(t_k+n-1), u(t_k), u(t_k+1), .., u(t_k+n-1))}{d_{t_k}} \to 0 \tag{4.60}$$

$$\frac{\phi(t_k+n-1)}{d_{t_k}} \to 0 \tag{4.61}$$

which contradicts (4.56).

□

COROLLARY 2.4.11

$$\lim_{k\to\infty} \| \frac{(u(k) - f(A_0,b_0)x(k))}{\|x(k)\|} \| = 0 \tag{4.62}$$

PROOF Since by the previous corollary $\frac{d_k}{\|x(k)\|} \leqslant \delta$, for some $\delta > 0$, we have:

$$\lim_{k\to\infty} \| \frac{(u(k) - f(A_0,b_0)x(k))}{\|x(k)\|} \| \tag{4.63}$$

$$= \lim_{k\to\infty} \| \frac{(u(k) - f(A_0,b_0)x(k))}{d_k} \frac{d_k}{\|x(k)\|} \| = 0$$

□

PROOF OF THEOREM 2.4.5

The proof of Theorem 2.4.5 is now just an application of Theorem 2.2.1 and Corollary 2.4.11.

□

2.5. Simulations

The algorithms presented in sections 2 & 3 have been simulated extensively. As could be expected, convergence gets slower as the order of the system increases. The asymptotic behavior of the controlled system was characterized in terms of the action of the asymptotic closed-loop matrix on the state trajectory. The weak self-tuning property, however, can be better illustrated by a comparison between the applied input and the desired input. In many cases the assumption (2.6) in Theorem 2.2.1 implies that:

$$\lim_{k\to\infty} \frac{u(k)}{fx(k)} = 1 \tag{5.1}$$

We will now give the graphs of the output of a second order unstable non-minimum phase system and of $\frac{u(k)^a}{u(k)^d}$ (where superscripts a and d stand for "applied" and "desired" respectively). The applied algorithm is the one introduced in section 6 (non-observed state case). The true system has the realization:

$$x(k+1) = \begin{bmatrix} -1 & 1 \\ 6 & 0 \end{bmatrix} x(k) + \begin{bmatrix} 1 \\ 2 \end{bmatrix} u(k) \tag{5.2.a}$$

$$y(k) = [1\ 0]x(k) \tag{5.2.b}$$

The system was initially guessed as:

$$\hat{A}_0 = \begin{bmatrix} -3 & 1 \\ 7 & 0 \end{bmatrix} \quad \hat{b}_0 = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \tag{5.3}$$

The desired closed-loop characteristic polynomial was chosen to be:

$$X^2 - 1.7X + 0.72 \tag{5.4}$$

If we look at figures 5.1 and 5.2, we see that initially the system behaves badly. Then after a certain learning period, the quotient of applied and desired input is close to 1 and the system begins to stabilize. One may check that $y(k)$ tends to zero exponentially fast, the exponent tending to the slowest desired pole:

0.9. At iteration 32 there is a peak in the graph of $\frac{u(k)^a}{u(k)^d}$. The explanation is that the state was suddenly too far away from the subspace it was converging to. Outside this subspace the control law was still far away from the desired one. After 150 iterations $\frac{u(k)^a}{u(k)^d}$ is very close to 1, even though the true system has not been identified at all. The simulations show that the Euclidean distance between the true system parameters and $(\hat{A}_{150},\hat{b}_{150})$ was almost 2.24 (initially 2.65). After 150 iterations the closed-loop poles were 0.89998 and 0.81490. The larger deviation of the second pole from the desired one can be explained by the fact that the state trajectory converges to the invariant subspace belonging to the other pole. Hence information about the invariant subspace belonging to the second pole gets poor very quickly. This illustrates the weak self-tuning feature: only the poles which are excited asymptotically are placed properly.

□

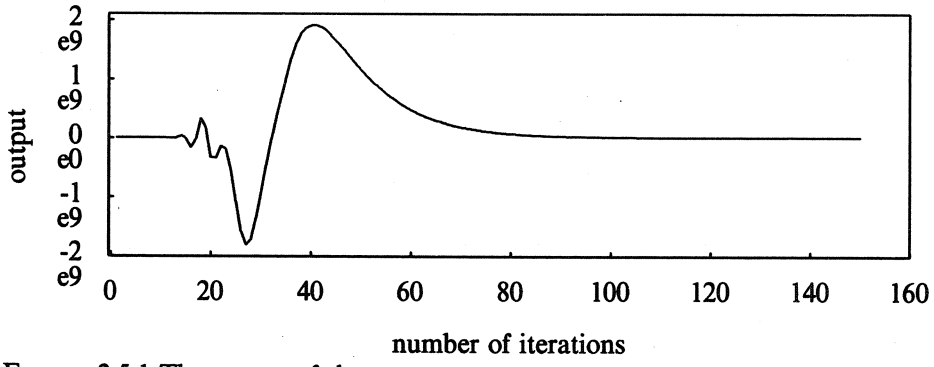


FIGURE 2.5.1 The output of the system

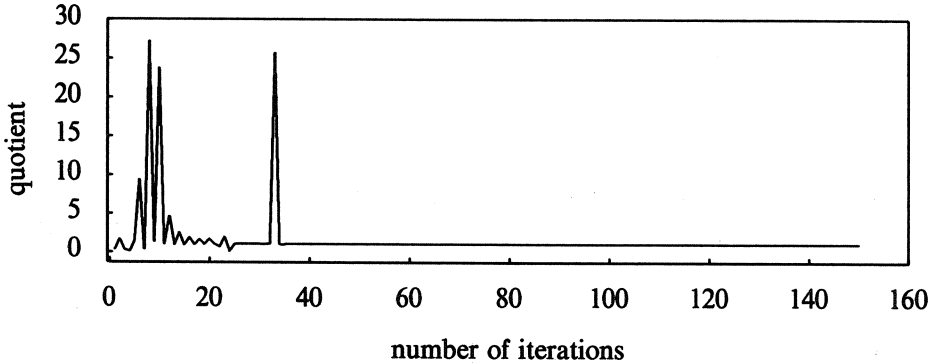


FIGURE 2.5.2 The quotient of applied and desired input

3. An Adaptive LQ Controller for a First Order System

In the previous section we have presented an algorithm for adaptive pole assignment. It appeared that the algorithm could be based on neutrality and certainty equivalence and that identification of the system parameters was not necessary. In Chapter II we proved that if the control objective was the minimization of a quadratic cost functional, then there is a serious conflict between identification and control. In terms of the sets G and H this meant that $G \cap H$ is a negligible subset of G, the set of invariant points of an algorithm.

In this section we will present an algorithm for adaptive LQ-control which is based on dual control rather than on certainty equivalence only. By this we mean the following. The input will consist of two parts. A pure control part, based on certainty equivalence, and an excitation part. The excitation part of the input will be chosen in such a way that the set G will shrink to a singleton, namely the true system parameter. The excitation part could also be called an active learning part. The excitation part will contain a design parameter. This parameter reflects the trade-off between identification on the one hand and optimal control on the other hand. The smaller the parameter is the better the optimal costs can be approached, but also the slower is the convergence of the parameter estimates. Another important feature of the excitation part of the input is that it is proportional to the output of the system and hence regulation (output going to zero) is not in contradiction with the presence of an excitation signal as would have been the case if the excitation consisted of an additive signal. We will consider the first order case only, the general case will be treated elsewhere.

Consider the system:

$$y(k+1) = a_0 y(k) + b_0 u(k), \quad b_0 \neq 0, \, y(0) \in \mathbf{R} \tag{1}$$

Cost functional:

$$J = \sum_{k=0}^{\infty} y^2(k) + r u^2(k), \quad r > 0 \tag{2}$$

Define $f : \mathbf{R} \times \mathbf{R} \setminus \{0\} \to \mathbf{R}$, by:

$$f(a,b) := \frac{-abp(a,b)}{(b^2 p(a,b) + r)} \tag{3}$$

where $p(a,b)$ is the positive solution of:

$$p - a^2 p + \frac{a^2 b^2 p^2}{b^2 p + r} - 1 = 0 \tag{4}$$

then the optimal control is:

$$u(k) = f(a_0, b_0) y(k) \tag{5}$$

and the optimal value of J is given by:

$$J^* = p(a_0,b_0)y^2(0) \tag{6}$$

Formula's (3,4,5,6) are the first order versions of II.3 (5,6,7,8). If $(\hat{a}_k,\hat{b}_k)$ is an estimate of (a_0,b_0) at time k, then a certainty equivalence adaptive control scheme would take:

$$u(k) = f(\hat{a}_k,\hat{b}_k)y(k) \tag{7}$$

We know however that (7) would result in an undesirably large set of invariant points if in addition the estimation part of the algorithm is neutral. Therefore we will slightly perturb (7) so as to ensure that (a_0,b_0) will be the only invariant point of the algorithm. Consider the following perturbation of (7):

$$u(k) = f(\hat{a}_k,\hat{b}_k)y(k) + (-1)^k\alpha_k y(k) = [f(\hat{a}_k,\hat{b}_k) + (-1)^k\alpha_k]y(k) \tag{8}$$

where $\{\alpha_k\}$ is a sequence of positive real numbers. The second term of (8) is the excitation part of the controls. Now assume that the input of the system (1) is generated by (8) and that $(\hat{a}_k,\hat{b}_k)$ is produced by a neutral estimation scheme, then we have:

LEMMA 3.1 If there exist positive constants δ and M such that for all k: $\delta \leqslant \alpha_k \leqslant M$, then, if for all k $y(k) \neq 0$:

$$G = \{(a_0,b_0)\} \tag{9}$$

PROOF Recall from II.1.2 (2.19) that:

$$G = \{(a,b) \mid \text{for all } k\colon y(k) = \hat{y}(k)\} \tag{10}$$

Hence $(a,b)\in G$ implies that for all k:

$$y(k+1) = (a_0 + b_0 f(a,b) + b_0(-1)^k\alpha_k)y(k) \tag{11.a}$$

$$= (a + bf(a,b) + b(-1)^k\alpha_k)y(k) = \hat{y}(k+1) \tag{11.b}$$

Since $\{\alpha_k\}$ is bounded there exists a subsequence $\{s_k\}$ of $2\mathbb{N}$, such that:

$$\lim_{k\to\infty} \alpha_{s_k} \text{ exists, say } \alpha \tag{12.a}$$

$$\lim_{k\to\infty} \alpha_{1+s_k} \text{ exists, say } \alpha' \tag{12.b}$$

Hence it follows from (11) that:

$$a_0 + b_0 f(a,b) + b_0\alpha = a + bf(a,b) + b\alpha \tag{13.a}$$

$$a_0 + b_0 f(a,b) - b_0\alpha' = a + bf(a,b) - b\alpha' \tag{13.b}$$

Subtracting (13.b) from (13.a) gives:

$$(\alpha + \alpha')b_0 = (\alpha + \alpha')b \tag{14}$$

Since by assumption α and α' are both positive, we conclude:

$$b = b_0 \tag{15}$$

Substituting (15) in (13.a) gives:

$$a = a_0 \tag{16}$$

The statement follows.

□

COMMENT From Lemma 3.1 we conclude that if we use (8), then for a suitable choice of the sequence $\{\alpha_k\}$, boundedness $\{(\hat{a}_k,\hat{b}_k)\}$ already implies convergence to (a_0,b_0). On the other hand it is intuitively clear that the existence of $\{\alpha_k\}$ will cause the behavior of the system to deviate from the desired (optimal) behavior. We will first study the behavior of the controlled system as dependent on the choice of $\{\alpha_k\}$. We will do this for the case the initial guess of the system parameters is (a_0,b_0) itself and $\{\alpha_k\}$ is constant.

LEMMA 3.2 Consider the closed-loop system:

$$y(k+1) = a_0 y(k) + b_0 u(k) \tag{17.a}$$

$$u(k) = f(a_0,b_0)y(k) + \alpha(-1)^k y(k) \tag{17.b}$$

for some fixed $\alpha > 0$. Then:

(i) For α sufficiently small: $\lim_{k\to\infty} y(k) = 0$.

(ii) $\lim_{\alpha\downarrow 0} [J^* - \sum_{k=0}^{\infty} y^2(k) + ru^2(k)] = 0$. (where J^* is defined by (6)).

PROOF (i) Obvious. Take $0 < \alpha < 1 - | a_0 + b_0 f(a_0,b_0) |$. Since $| (a_0 + b_0 f(a_0,b_0) | < 1$, the statement follows.

(ii) First observe that:

$$J^* = \sum_{k=0}^{\infty} y^2(k) + rf(a_0,b_0)^2 y^2(k) \tag{18.a}$$

$$= (1 + rf(a_0,b_0)^2)y^2(0) \sum_{k=0}^{\infty} (a_0 + b_0 f(a_0,b_0))^{2k} \tag{18.b}$$

$$= (1 + rf(a_0,b_0)^2)y^2(0) \frac{1}{1-(a_0 + b_0 f(a_0,b_0))^2} \tag{18.c}$$

Second:

$$J^* \leq \sum_{k=0}^{\infty} y^2(k) + u^2(k) \tag{19.a}$$

$$= \sum_{k=0}^{\infty} y^2(k) + r(f(a_0,b_0) + (-1)^k \alpha)^2 y^2(k) \tag{19.b}$$

Since (19.b) is absolute summable for α sufficiently small, we may interchange the order of summation and taking the limit, and the statement follows.

□

COMMENT From Lemma 3.2 we conclude that we can approach the optimal costs arbitrarily closely by choosing α sufficiently small.
We will now combine the conclusions of Lemma's 3.1 and 3.2 to develop an adaptive algorithm.

ALGORITHM

$$\hat{a}_{k+1} = \hat{a}_k + \frac{y(k)}{u^2(k) + y^2(k)}(y(k+1) - \hat{y}(k+1)) \tag{20.a}$$

$$\hat{b}_{k+1} = \hat{b}_k + \frac{u(k)}{u^2(k) + y^2(k)}(y(k+1) - \hat{y}(k+1)) \tag{20.b}$$

$$\beta_k = (1 - |\hat{a}_k + \hat{b}_k f(\hat{a}_k,\hat{b}_k)|) \tag{20.c}$$

$$u(k) = (f(\hat{a}_k,\hat{b}_k) + (-1)^2\alpha\beta_k)y(k) \tag{20.d}$$

$$\hat{y}(k+1) = \hat{a}_k y(k) + \hat{b}_k u(k) \tag{20.e}$$

LEMMA 3.3 The sequence $\{(\hat{a}_k,\hat{b}_k)\}$ as defined by (20) has the following two properties.
(i) $\{\|(a_0,b_0) - (\hat{a}_k,\hat{b}_k)\|\}$ is a non-increasing sequence.
(ii) $\lim_{k\to\infty} \|(\hat{a}_{k+1},\hat{b}_{k+1}) - (\hat{a}_k,\hat{b}_k)\| = 0.$

PROOF See Lemma's III.2.1 and III.2.2.

□

To simplify the discussion we will make an extra assumption. This assumption will be relaxed later on.

ASSUMPTION Assume that there exists a positive constant γ such that for all k:

$$|\hat{b}_k| \geqslant \gamma \tag{21}$$

□

THEOREM 3.4
(i) for all $\alpha > 0$: $\lim_{k\to\infty}(\hat{a}_k,\hat{b}_k) = (a_0,b_0)$.
(ii) $\lim_{k\to\infty} \beta_k = (1 - |a_0 + b_0 f(a_0,b_0)|$.
(iii) for all $0 < \alpha < 1$: $\lim_{k\to\infty} y(k) = 0$.
(iv) $\lim_{\alpha\downarrow 0}[\lim_{N\to\infty} \frac{1}{y^2(2N)}(\sum_{k=2N}^{\infty} y^2(k) + ru^2(k) - p(a_0,b_0)y^2(2N))] = 0.$
(v) $\lim_{\alpha\downarrow 0}[\lim_{N\to\infty} \frac{1}{y^2(2N+1)}(\sum_{k=2N+1}^{\infty} y^2(k) + ru^2(k) - p(a_0,b_0)y^2(2N+1))] = 0.$

PROOF (i) Since by assumption $|\hat{b}_k| \geqslant \gamma$, and since $\{(\hat{a}_k,\hat{b}_k)\}$ is bounded,

there exists a compact set $C \subset \mathbf{R} \times \mathbf{R} \setminus \{0\}$ such that for all k $(\hat{a}_k, \hat{b}_k) \in C$. Since β_k is a continuous function of $(\hat{a}_k, \hat{b}_k) \in C$, and since by the strict stability of $\hat{a}_k + \hat{b}_k f(\hat{a}_k, \hat{b}_k)$, $\beta_k > 0$, we conclude that there exist positive constants δ and M such that for all k:

$$\delta \leqslant \alpha\beta_k \leqslant M \tag{22}$$

Now Lemma 3.1 yields the result.
This follows from (i)
(iii) This follows from (i) and the fact that $0 < \alpha < 1$.
(iv) Define:

$$J_N^\alpha := \sum_{k=N}^{\infty} y^2(k) + ru^2(k), \tag{23}$$

Now:

$$\frac{1}{y^2(2N)} \left(\sum_{k=2N}^{\infty} y^2(k) + ru^2(k) \right)$$

$$= \frac{1}{y^2(2N)} \sum_{k=2N}^{\infty} y^2(k)(1 + r(f(\hat{a}_k, \hat{b}_k) + (-1)^k \alpha\beta_k)^2) \tag{24}$$

$$= \frac{1}{y^2(2N)} \sum_{k=2N}^{\infty} [y^2(2N)(1 + r(f(\hat{a}_k, \hat{b}_k) + (-1)^k \alpha\beta_k)^2) \tag{25}$$

$$\left(\prod_{j=2N}^{k-1} (a_0 + b_0(f(\hat{a}_j, \hat{b}_j) + (-1)^j \alpha\beta_j))^2 \right)]$$

$$= \sum_{l=0}^{\infty} [(1 + rf(\hat{a}_{l+2N}, \hat{b}_{l+2N}) + (-1)^{l+2N} \alpha\beta_{l+2N})^2) \tag{26}$$

$$\left(\prod_{m=0}^{l-1} (a_0 + b_0(f(\hat{a}_{m+2N}, \hat{b}_{m+2N}) + (-1)^{m+2N} \alpha\beta_{m+2N}))^2 \right)]$$

$$= \sum_{l=0}^{\infty} [(1 + rf(\hat{a}_{l+2N}, \hat{b}_{l+2N}) + (-1)^l \alpha\beta_{l+2N})^2) \tag{27}$$

$$\left(\prod_{m=0}^{l-1} (a_0 + b_0(f(\hat{a}_{m+2N}, \hat{b}_{m+2N}) + (-1)^m \alpha\beta_{m+2N}))^2 \right)]$$

Since (27) is summable for all N sufficiently large and α sufficiently small, we may interchange the summation and taking the limit. By (i) this yields:

$$\lim_{N \to \infty} \frac{J_{2N}^\alpha}{y^2(2N)} = \sum_{l=0}^{\infty} [(1 + r(f(a_0, b_0) + (-1)^l \alpha\beta^*)^2) \tag{28}$$

$$\left(\prod_{m=0}^{l-1} (a_0 + b_0 f(a_0, b_0) + (-1)^m \alpha\beta^*) \right)^2)],$$

where $\beta^* = \lim_{k \to \infty} \beta_k$. Again since (28) is absolute summable for α sufficiently small, it follows that:

$$\lim_{\alpha \downarrow 0} \lim_{n \to \infty} \frac{J_{2N}^{\alpha}}{y^2(2N)} = \sum_{l=0}^{\infty} (1 + rf(a_0,b_0)^2)(a_0 + b_0 f(a_0,b_0))^{2l} \tag{29}$$

The statement now follows from (6) and (18.b).
(v) Completely analogous to (iv).

□

COMMENT

(i) Regardless of the value of α, as long it is positive, (a_0,b_0) is identified. This is of course due to the excitation term in the control.

(iii) For α sufficiently small, the closed-loop system is globally asymptotically stable. The reason that this property holds despite the excitation term in the input is that the excitation is in closed loop. Otherwise stated, the excitation is proportional to the output, which implies that it damps out as the output gets smaller. This is in contrast to the case where an open-loop signal is injected to the system, as has often been proposed in the literature. In that case regulation of the output can never be achieved, since the excitation signal will always be active. Another important advantage of closed-loop excitation is that it can never be dominated by the other signals in the loop. For if the signals in the loop are large then also the excitation signal will be large.

(iv,v) Asymptotically the optimal costs can be approached arbitrarily well, by taking α sufficiently small. It should however be noted that the convergence speed of $(\hat{a}_k,\hat{b}_k)$ will decrease as α gets smaller. Hence there is a trade-off between identification and control.

□

MODIFICATION OF THE ALGORITHM

The assumption that $|\hat{b}_k| \geq \gamma$ can be relaxed by using the modification described in the pole assignment algorithm or by any other method based on III.1. We skip the details, since they are completely the same as in the pole assignment algorithm.

OPTIMAL CHOICE OF α

In Theorem 3.4.ii we claimed that the controlled system will be asymptotically stable provided that the design parameter α lies between zero and one. The question arises which value of α is in some sense optimal. First we look at this problem from the identification point of view.
Recall from Section III.2 that the estimation algorithm (20) was based on the orthogonal projection of $(\hat{a}_k,\hat{b}_k)$ on $\hat{G}_{k+1}$, where:

$$\hat{G}_{k+1} = \{(a,b) \mid ay(k) + bu(k) = y(k+1)\} \tag{30}$$
$$= \{(a,b) \mid a + b(f(\hat{a}_k,\hat{b}_k) + \alpha\beta_k(-1)^k) = a_0 + b_0(f(\hat{a}_k,\hat{b}_k + \alpha\alpha_k(-1)^k)\}$$

Now, one observes immediately that if $\hat{G}_k$ and $\hat{G}_{k+1}$ are orthogonal, then:

$$(\hat{a}_{k+1},\hat{b}_{k+1}) = (a_0,b_0) \tag{31}$$

Hence as far as fast identification is concerned it would be good if $\hat{G}_k$ and $\hat{G}_{k+1}$ are as orthogonal as possible within the given structure.

PROPOSAL 1 From now on we will allow α to depend on k, therefore instead of α we write α_k. Take:

$$\alpha_k = \frac{(a + f(\hat{a}_k,\hat{b}_k)^2)^{1/2}}{1 - |\hat{a}_k + \hat{b}_k f(\hat{a}_k,\hat{b}_k)|} \tag{32}$$

□

If we take this value for α_k, then we have the following property:
PROPERTY If $(\hat{a}_{k-1},\hat{b}_{k-1}) = (\hat{a}_k,\hat{b}_k)$, then $\hat{G}_k$ and $\hat{G}_{k+1}$ are orthogonal.

PROOF One can easily check that the product of the slopes of $\hat{G}_k$ and $\hat{G}_{k+1}$ is -1, which implies that they are orthogonal.

□

The consequence of accepting Proposal 1 is the following. If for some k $(\hat{a}_{k-1},\hat{b}_{k-1})$ and $(\hat{a}_k,\hat{b}_k)$ are very close, then $\hat{G}_k$ and $\hat{G}_{k+1}$ are almost orthogonal, and hence $(\hat{a}_{k+1},\hat{b}_{k+1})$ will be very close to (a_0,b_0). Hence if the estimates hesitate to converge then they are forced to speed up. On the other hand if $(\hat{a}_{k-1},\hat{b}_{k-1})$ are not close, then there is already a certain speed of convergence.
The conclusion is that the choice (32) for α_k is good for identification purposes. However, since α_k will then be larger than one, it is clear that the resulting closed-loop system will not be asymptotically stable. That means that although (32) may be good for identification, it is bad for control. We will now propose a modification of (32).

PROPOSAL 2 Choose $\lambda \in (0,1)$ and let $g:\mathbb{R} \to \mathbb{R}^+ \cup \{0\}$ be a continuous function such that:

(i) $g(0) = 0$

(ii) $\forall x \in \mathbb{R}: g(x) = g(-x)$

(iii) $\lim_{x\to\infty} g(x) = 1$

Take:

$$\alpha_k = \lambda + (1 - \lambda)\, g\left(\frac{y(k) - \hat{y}(k)}{|y(k)|}\right) \frac{(1 + f(\hat{a}_k,\hat{b}_k)^2)^{1/2}}{1 - |\hat{a}_k + \hat{b}_k f(\hat{a}_k,\hat{b}_k)|} \tag{33}$$

□

The idea behind Proposal 2 is the following. Since $\forall k\ \alpha_k \geqslant \lambda > 0$, we still have that $\lim_{k\to\infty} (\hat{a}_k,\hat{b}_k) = (a_0,b_0)$. This means that as long as $(\hat{a}_k,\hat{b}_k) \neq (a_0,b_0)$, $\frac{y(k) - \hat{y}(k)}{|y(k)|}$ will be non-zero, and then the influence of the term involving g is still present. If $g(\frac{y(k) - \hat{y}(k)}{|y(k)|})$ is close to one, then the value of α_k will be close to that of Proposal 1. As a consequence the initial behavior of the estimation part will be good. Then, as soon as $(\hat{a}_k,\hat{b}_k)$ comes close to (a_0,b_0), the normalized prediction error will be small and α_k will be close to λ. Hence Proposal 2 can be viewed as a modification of the initial behavior of the estimation scheme without influencing the asymptotic properties of the controlled system.

Simulations

We have simulated the algorithm for different systems and for different values of r. Also we experimented with different values of λ in proposal 2. Since $G \cap H = \{(a_0,b_0)\}$, almost every point in G corresponds to sub-optimal behavior of the system. For $(a,b) \in G$ we can easily determine the real cost incurred if we apply

$$u(k) = f(a,b)y(k), \tag{34}$$

to the true system (a_0,b_0), namely:

$$\frac{1}{y(0)^2} \sum_{k=0}^{\infty} y(k)^2 + ru(k)^2 \tag{35}$$

$$= \frac{1}{y(0)^2} \sum_{k=0}^{\infty} (1 + rf(a,b)^2)y(k)^2 \tag{36}$$

$$= \sum_{k=0}^{\infty} (1 + rf(a,b)^2)\,(a_0 + b_0 f(a,b))^{2k} \tag{37}$$

$$= \sum_{k=0}^{\infty} (1 + rf(a,b)^2)\,(a + bf(a,b))^{2k} \quad \text{since } (a,b) \in G \tag{38}$$

$$= p(a,b) \tag{39}$$

Where $p(a,b)$ is the positive solution of (4). The last equality follows from (6) and (18.b), with (a_0,b_0) replaced by (a,b). From the fact that we apply (34) whereas we should have applied $u(k) = f(a_0,b_0)y(k)$, we conclude that for all $(a,b) \in G$:

$$p(a,b) \geqslant p(a_0,b_0) \tag{40}$$

(see also Lemma II.3.21). Now the larger the difference between $p(a,b)$ and $p(a_0,b_0)$, the more sub-optimal is (a,b). However, for small values of r the control law $f(a,b)$ will be approximately equal to $\frac{-a}{b}$. This implies that for small values of r, G is approximately equal to H (see II.2). Hence the sub-

optimality of the limit of the sequence of estimates can be illustrated best for a large value of r. Let us now discuss a particular simulation:
True system:

$$a_0 = 1, \quad b_0 = 2 \tag{41}$$

Cost criterion:

$$r = 100 \tag{42}$$

Initial guess of the (a_0,b_0):

$$\hat{a}_0 = 6 \quad \hat{b}_0 = 9 \tag{43}$$

The number of iterations was 100. Figure 3.1 shows the trajectories of the estimates $(\hat{a}_k,\hat{b}_k)$.

(i) Figure 3.1.a shows the trajectory for the algorithm (20) with $\alpha = 0$. After 100 iterations we found:

$$\hat{a}_{100} = 4.08 \tag{44.a}$$

$$\hat{b}_{100} = 10.00 \tag{44.b}$$

$$p(\hat{a}_{100},\hat{b}_{100}) = 16.7 \text{ compare with } p(a_0,b_0) = 5.52 \tag{44.c}$$

(ii) Figure 3.1.b shows the trajectory for the algorithm (20) with $\alpha = 0.2$. After 100 iterations we found:

$$\hat{a}_{100} = 1.67 \tag{45.a}$$

$$\hat{b}_{100} = 10.00 \tag{45.b}$$

$$p(\hat{a}_{100},\hat{b}_{100}) = 7.03 \text{ compare with } p(a_0,b_0) = 5.52 \tag{45.c}$$

(iii) Figure 3.1.c shows the trajectory for the algorithm (20) with Proposal 2, $\lambda = 0.2$ and $g(x) = 1 - e^{-1000*x^2}$. After 100 iterations we found:

$$\hat{a}_{100} = 1.00 \tag{46.a}$$

$$\hat{b}_{100} = 2.00 \tag{46.b}$$

$$p(\hat{a}_{100},\hat{b}_{100}) = 5.52 \text{ compare with } p(a_0,b_0) = 5.52 \tag{46.c}$$

It is obvious which of the three simulations gave the best convergence results. However, although in (46) the true parameter was identified, there were additional costs involved due to the excitation term. That is the price one has to pay if one wants to do adaptive LQ control.

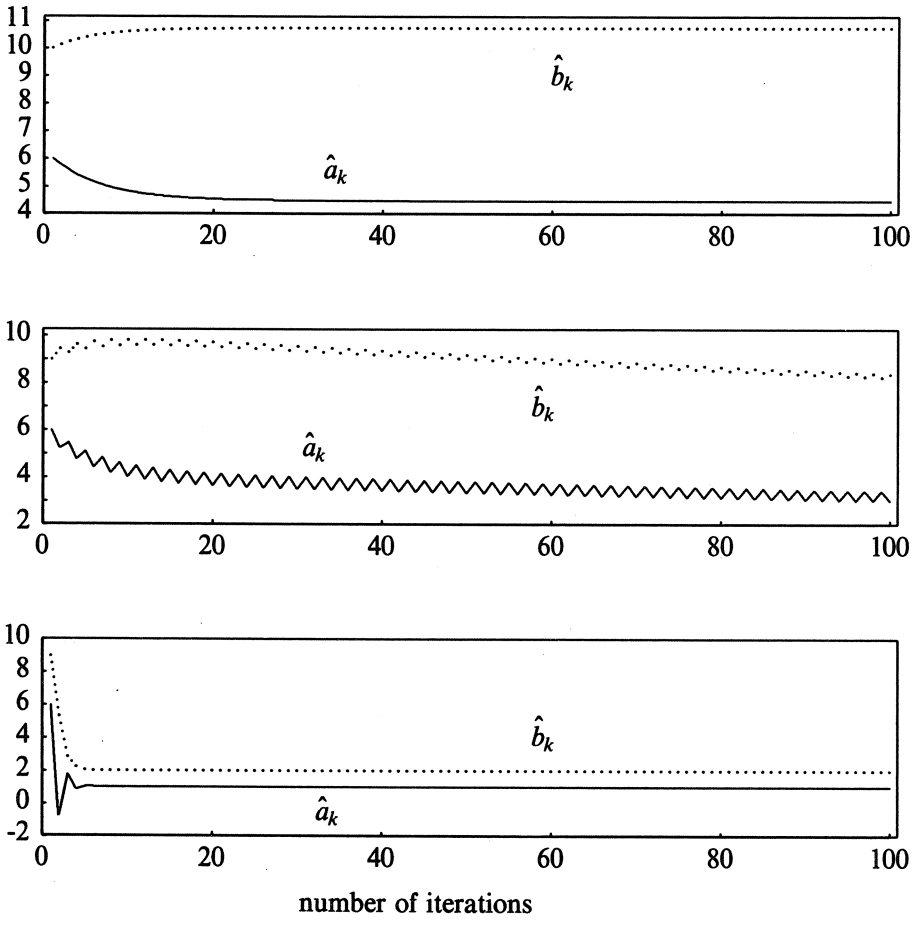


FIGURE 3.1. TRAJECTORY OF ESTIMATES

4. AN ADAPTIVE POLE ASSIGNMENT ALGORITHM FOR A FIRST ORDER CONTINUOUS-TIME SYSTEM

4.1. Introduction

The bulk of this monograph is concerned with discrete-time systems. However, many of the results presented in section II.b carry over to the continuous-time case. The algorithm for pole assignment for discrete-time systems relies heavily on the fact that $G \subset H$. We did not set up a formalism for adaptive control of continuous-time systems, hence strictly speaking G and H are not even defined in that case. Nevertheless, if we replace the difference equations in discrete-time by differential equations, we can again think of invariant points of an algorithm as possible limits. And again we can ask ourselves the question whether or not an invariant point corresponds to the (or a) desired controller.

An important difference between discrete and con-inuous time is that in continuous-time there do not exist dead-beat observers, since differentiation of the signals is undesirable. This causes certain complications in the higher order case. See [27,28,29,30].

In this section we want to present an adaptive pole assignment algorithm for a first order continuous-time system. The motivation for doing this is twofold. The first reason is that it shows that some of the ideas used in the discrete-time case, can be used in the continuous-time case also. This holds in particular for some of the geometrical arguments from Section III.2. And also for the method of keeping estimates away from non-controllable as described in Section III.1. The other source of inspiration is formed by the discussion on adaptive stabilization of continuous-time systems as commmented upon in the introduction of this monograph. See [43,44,47,48,62]. We wanted to design an algorithm that not only stabilizes the system, but does so with a prescribed rate of stability. The material in this section is based on [54].

We will first formulate the problem mathematically, then we will describe the algorithm and its geometrical and asymptotic properties. It will turn out that, like in the discrete-time case, the controlled system behaves asymptotically the same as if the parameters were known, although they are not necessarily identified. Finally, we comment upon the possibility of implementing the algorithm.

4.2. The algorithm

We will first give the system description and the problem statement. Then we will give the algorithm in a form suitable for analysis and relatively easy to understand. This will involve the derivative of the output. In the next subsection we show that the algorithm is implementable without differentiating the output.

Let the true system be given by:

$$\dot{y} = a_0 y + b_0 u, \quad y_0 \in \mathbf{R}, \tag{2.1}$$

where (a_0,b_0) is fixed but unknown. The only assumption we make is that the system is controllable, that is $b_0 \neq 0$. Choose any $\alpha \in \mathbb{R}$, and let the desired closed-loop behavior be:

$$\dot{y} = \alpha y. \tag{2.2}$$

The unique feedback-law that achieves the control objective is:

$$u(t) = f(a_0,b_0)y(t) \tag{2.3}$$

where f is defined by:

$$f(a,b) = \frac{\alpha - a}{b} \tag{2.4}$$

Our algorithm will be based on closed-loop identification of (a_0,b_0) and the certainty equivalence principle. On the basis of the available data an estimation $(a(t),b(t))$ is made of (a_0,b_0), and then we take $u(t) = f(a(t),b(t))y(t)$. This can of course only be done when $b(t) \neq 0$. Therefore our algorithm consists of two parts. Roughly speaking it works as follows. At every time instant t a check is made whether $b(t)$ is not too close to zero. If not, we calculate the input according to certainty equivalence. In the other case we apply a special input for a certain time period to bias $b(t)$ away from zero. The crucial observation is that within finite time $b(t)$ will be bounded away from zero and hence within finite time we will use certainty equivalence forever.

THE ALGORITHM

Choose sequences $\{\epsilon_k\}_{k\in\mathbb{N}}$ and $\{C_k\}_{k\in\mathbb{N}}$ such that:

$$\epsilon_k \downarrow 0 \text{ and } C_k \uparrow \infty \tag{2.5}$$

Choose $(a(0),b(0))$ arbitrarily and $j = 0$, $z(0) = 0$. Define the functions $a(t)$, $b(t)$, $z(t)$, $u(t)$ and the sequence τ_k by the following conditional differential equations:

$$\dot{a} = \frac{y}{y^2+u^2}(\dot{y} - ay - bu) \tag{2.6.a}$$

$$\dot{b} = \frac{u}{y^2+u^2}(\dot{y} - ay - bu) \tag{2.6.b}$$

$$\dot{z} = -I_{\{z>0\}} \tag{2.6.c}$$

if $z(t) > 0$ then:

$$u(t) = C_j y(t) \tag{2.6.1}$$

else:

if $|b(t)| \geq \epsilon_j$ then:

$$u(t) = f(a(t),b(t))y(t) \tag{2.6.2.a}$$

else (if $|b(t)| < \epsilon_j$)

$$j := j+1 \tag{2.6.2.b}$$

$$z(t) := j \tag{2.6.2.c}$$

$$\tau_j := t \tag{2.6.2.d}$$

$$u(t) := C_j y(t) \tag{2.6.2.e}$$

□

COMMENT The equations for a and b are the continuous versions of the projection algorithm for discrete-time ([21] and [55]). They can be derived from the discrete-time equations by using infinitesimal arguments. As in [55], the sequence $\{\tau_k\}_{k\in\mathbf{N}}$ can be seen as a sequence of stopping times. As soon as $t = \infty$ it is easy to see that:

$$\tau_1 = \inf\{t \mid |b(t)| \leqslant \epsilon_1\} \tag{2.7.a}$$

$$\tau_{k+1} = \inf\{t \geqslant \tau_k + k \mid |b(t)| \leqslant \epsilon_{k+1}\} \tag{2.7.b}$$

The infimum is understood to be infinity if the set over which it is taken, is empty. Define:

$$I_k := [\tau_k, \tau_k + k) \tag{2.8}$$

Outside I_k, $u(t)$ is calculated according to certainty equivalence. For $t \in I_k$ we take: $u(t) = C_k y(t)$. Before we analyze the algorithm formally, let us try to explain intuitively that $\tau_k = \infty$ for some finite k. Suppose $b_0 > 0$ and $b(0) < 0$. Since $b(t)$ tries to estimate b_0, it can be expected that $b(t)$ has to pass through the set $\{b = 0\}$. If $b(t) = 0$, we cannot calculate $f(a(t),b(t))$. Now as soon as $b(t)$ comes too close to zero (measured by the sequence $\{\epsilon_k\}$), we start to apply special inputs: from time τ_k to time $\tau_k + k$ we take $u(t) = C_k y(t)$. Assume for the moment that this alternative procedure has to be started infinitely often (i.e. for every k: $\tau_k < \infty$). Then due to the growing inputs and the increasing time interval during which they are applied, it can be proven that $b(\tau_k + k)$ will converge to b_0. Then, using the geometrical properties of the trajectory of $(a(t),b(t))$, we prove that since by assumption $|b_0| > 0$, $|b(t)|$ will eventually be bounded from below by some $\epsilon > 0$. This implies that, since ϵ_k tends to zero, the alternative procedure will not be started again as soon as $\epsilon_k < \epsilon$. This contradicts the assumption. Hence after some finite time instant t_0 we will always apply $u(t) = f(a(t),b(t))y(t)$. This method of avoiding zero-division is of course closely related to the method described in III.a. The formal proof of the above reasoning will be given in Lemma 4.2.3.

□

Define:

$$g(a(t),b(t)) = C_k \text{ if } t \in I_k \tag{2.9.a}$$

$$= f(a(t),b(t)) \text{ otherwise} \tag{2.9.b}$$

then: for all t:

$$u(t) = g(a(t),b(t))y(t) \tag{2.10}$$

LEMMA 4.2.1 $(a_0-a(t))^2 + (b_0-b(t))^2$ is non-increasing.

PROOF Define $V(a,b):=(a_0-a)^2 + (b_0-b)^2$, then along trajectories of $(a(t),b(t))$, it follows from (2.1), (2.6.a), (2.6.b) and (2.10) that:

$$\frac{1}{2}\dot{V}(a,b) = \frac{-1}{g^2+1}(a_0-a+(b_0-b)g)^2 \leqslant 0 \tag{2.11}$$

□

From Lemma 4.2.1 we conclude that $(a(t),b(t))$ converges to a circle with radius $R \geqslant 0$ for some R and center (a_0,b_0). For $t \in [\tau_k+k, \tau_{k+1})$, the trajectory of (a(t),b(t)) has the following interesting geometrical property:

LEMMA 4.2.2 For all k and for all $t \in [\tau_k+k, \tau_{k+1})$:

$$(a(t)-\alpha)^2 + b(t)^2 = (a(\tau_k+k))^2 + (b(\tau_k+k))^2 =: r_k^2 \tag{2.12}$$

PROOF Define $W(a,b) := (a-\alpha)^2 + b^2$. Along trajectories of $(a(t),b(t))$ for $t \in [\tau_k+k, \tau_{k+1})$ we have:

$$\frac{1}{2}\dot{W}(a,b) = (a-\alpha)\dot{a} + b\dot{b} \tag{2.13}$$

$$= \left(\frac{a-\alpha}{1+f(a,b)^2} + \frac{bf(a,b)}{1+f(a,b)^2}\right)\left(\frac{\dot{y}}{y} - a - bf(a,b)\right) = 0 \tag{2.14}$$

The statement follows.

□

COMMENT Lemma 4.2.1 and 4.2.2 imply that for $t \in [\tau_k+k, \tau_{k+1})$, $(a(t),b(t))$ moves on a circle S_k with radius r_k and center $(\alpha,0)$, in such a way that the distance between $(a(t),b(t))$ and (a_0,b_0) is decreasing. Define l_0 as the line passing through $(\alpha,0)$ and (a_0,b_0), and denote by $(\bar{a}_k,\bar{b}_k)$ the point of intersection of S_k and l_0, which is closest to (a_0,b_0). Then $(a(t),b(t))$ moves along S_k in the direction of $(\bar{a}_k,\bar{b}_k)$. Since $(a(t),b(t))$ cannot leave this point along S_k without increasing the distance from (a_0,b_0), it follows that if $(a(t),b(t))$ reaches $(\bar{a}_k,\bar{b}_k)$ before τ_{k+1}, it will stay there for ever. In other words it will then converge to $(\bar{a}_k,\bar{b}_k)$ and as a consequence τ_{k+1} will be infinity. In the next lemma we will prove that $(a(t),b(t))$ will indeed converge to $(\bar{a}_k,\bar{b}_k)$, for some k.

LEMMA 4.2.3 $\{\tau_k \mid \tau_k < \infty\}$ is a finite set.

PROOF The proof will rely heavily on the geometrical properties of the trajectory of $(a(t),b(t))$. See Figure 4.1 for a pictorial explanation. Now suppose the claim is not true. From (2.6.b) and (2.6.1) it follows that on I_k we have:

$$\dot{b} = \frac{C_k}{1+C_k^2}(a_0-a+(b_0-b)C_k) \tag{2.15}$$

From Lemma 4.2.1 we know that $\{a(t)\}$ is bounded and hence, since C_k tends to infinity and since also the length of I_k tends to infinity, we conclude that:

$$\lim_{k\to\infty} b(\tau_k+k) = b_0 \tag{2.16}$$

Suppose that:

$$\lim_{k\to\infty} (a_0-a(t))^2+(b_0-b(t))^2 = R^2 > 0 \tag{2.17}$$

then by (2.16):

$$\lim_{k\to\infty} (a_0-a(\tau_k+k))^2 = R^2. \tag{2.18}$$

Hence at least one of the points $a_0\pm R$ is a limit point of $a(\tau_k+k)$. Assume without loss of generality that:

$$\lim_{k\to\infty} a(\tau_k+k) = a_0-R \tag{2.19}$$

For $\tau_k+k \leqslant t < \tau_{k+1}$, $(a(t),b(t))$ moves along S_k , hence by (2.16) and (2.18) we conclude that:

$$\lim_{k\to\infty} r_k = [b_0^2+(\alpha-a_0+R)^2]^{\frac{1}{2}} =: r \tag{2.20}$$

In other words, the sequence of circles S_k converges to a circle S with center $(\alpha,0)$ and radius r. Define $(\bar{a},\bar{b})$ as the point of intersection of S and l_0 which is closest to (a_0,b_0), then:

$$\lim_{k\to\infty} \bar{a}_k = \bar{a} \quad \text{and} \quad \lim_{k\to\infty} \bar{b}_k = \bar{b} \tag{2.21}$$

Straightforward calculations give:

$$\bar{b}^2 = (1+(\frac{a_0-\alpha}{b_0})^2)^{-1}r^2 \tag{2.22}$$

This implies that: $|\bar{b}| > 0$. One may also check that $\text{sign}(\bar{b}) = \text{sign}(b_0)$, and hence it follows that in going from (a_0-R,b_0) to $(\bar{a},\bar{b})$ along S and without increasing the distance from (a_0,b_0), we do not pass through $\{b = 0\}$. Let P_k be the path from $(a(\tau_k+k),b(\tau_k+k))$ to $(\bar{a}_k,\bar{b}_k)$ along S_k, then it follows that there exist $\epsilon > 0$ and k_0 such that for all $k \geqslant k_0$:

$$\inf \{ |b| \mid \exists a \text{ such that: } (a,b)\in P_k \} \geqslant \epsilon \tag{2.23}$$

Now choose $k_1 \geqslant k_0$ such that for all $k \geqslant k_1$: $\epsilon_k < \epsilon$. Then in going along P_{k_1}, $|b(t)|$ will never become smaller than ϵ_{k_1+1}, and hence $(a(t),b(t))$ will

converge to $(\bar{a}_{k_1},\bar{b}_{k_1})$. From (2.7) it follows that $\tau_{k_1+1} = \infty$. This contradicts the assumption, and the statement follows.

□

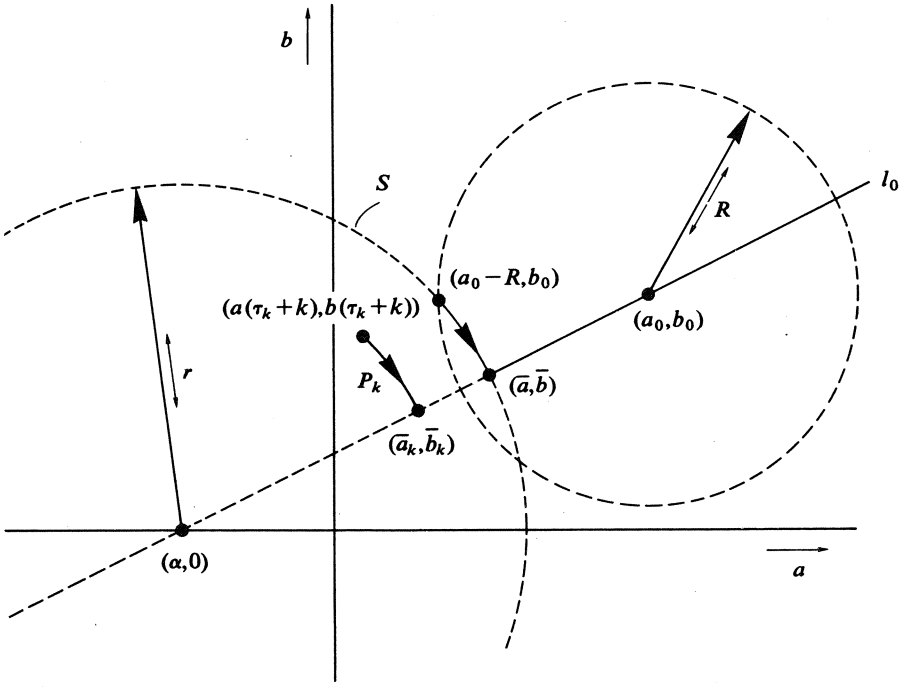


FIGURE 4.1: Behavior of $(a(t),b(t))$.

COROLLARY 4.2.4 There exists t_0 and $\epsilon > 0$, such that for all $t \geq t_0$:

$$(i)\ |\ b(t)\ | \geq \epsilon \tag{2.24}$$

$$(ii)\ u(t) = f(a(t),b(t))y(t) \tag{2.25}$$

$$(iii)\ \text{sign}(b(t)) = \text{sign}(b_0) \tag{2.26}$$

□

We will now study the asymptotic behavior of $(a(t),b(t))$. From now on we will assume that $t \geq t_0$. One of the main results of this section is:

THEOREM 4.2.5 $\lim_{t\to\infty} f(a(t),b(t)) = f(a_0,b_0)$

PROOF From Lemma 4.2.3 we know that: $\lim_{t\to\infty} (a(t),b(t)) = (\bar{a},\bar{b}) \in l_0$. Now $l_0 = \{(a,b)\ |\ a + b\frac{\alpha-a_0}{b_0} = \alpha\}$. One may check that $(a,b)\in l_0$ and $b \neq 0$ implies: $\frac{\alpha-a}{b} = \frac{\alpha-a_0}{b_0}$, which is equivalent to $f(a,b) = f(a_0,b_0)$. Since

$\bar{b} \neq 0$, the statement follows.

□

COMMENT Theorem 4.2.5 states that asymptotically the applied feedback-law equals the desired one. We do not claim that $\lim_{t\to\infty}(a(t),b(t)) = (a_0,b_0)$. The reason that nevertheless 2.5 holds is based on the following observations. Since identification takes place in closed-loop, the following set of parameter-values is certainly invariant under the algorithm:

$$G := \{(a,b) \mid a+bf(a,b) = a_0+b_0f(a,b)\} \tag{2.27}$$

By substituting for f we see that:

$$G = \{(a,b) \mid a+b\frac{\alpha-a_0}{b_0} = \alpha,\ b \neq 0\} \tag{2.28}$$

$$\subset l_0 = \{(a,b) \mid f(a,b) = f(a_0,b_0)\} \cup \{(\alpha,0)\} \tag{2.29}$$

It is not surprising that $(a(t),b(t))$ converges to G, since G is invariant under the algorithm. It is however due to the functional form of f that every point of G corresponds to the desired control-law. In III.1 we proved that there are no other control-laws with that property.

□

We will now characterize the asymptotic closed-loop behavior of the controlled system.

THEOREM 4.2.6 There exists a function $\delta : \mathbf{R} \to \mathbf{R}$, such that:

$$(i)\ \dot{y} = (\alpha+\delta(t))y \tag{2.30}$$

$$(ii)\ \lim_{t\to\infty}\delta(t) = 0 \tag{2.31}$$

PROOF (i) Take $\delta(t) = b_0(f(a(t),b(t))-f(a_0,b_0))$, then: $a_0 + b_0f(a(t),b(t)) = \alpha+\delta(t)$,which gives (2.30).
(ii) This follows from Theorem 4.2.5.

□

REMARK Note that Theorem 4.2.6 holds whether or not $\alpha < 0$. This shows that the adaptation part of the algorithm does not depend on the stability of the closed-loop system. If we assume that $\alpha < 0$, we obtain the following:

COROLLARY 4.2.7 If $\alpha < 0$, then the system is exponentially stable: for all $\epsilon > 0$ there exists t_ϵ such that for all $t \geq t_\epsilon$:

$$|y(t_\epsilon)|\, e^{(\alpha-\epsilon)(t-t_\epsilon)} \leq |y(t)| \leq |y(t_\epsilon)|\, e^{(\alpha+\epsilon)(t-t_\epsilon)} \tag{2.32}$$

□

4.3. Implementation

The form in which the algorithm has been described in the previous sub-section is not directly implementable, since it depends on the derivative of the output. In this sub-section we will show that differentiating the output can be avoided. First we will show this for $t \in [\tau_k + k, \tau_{k+1})$, and then for $t \in I_k$.

For $t \in [\tau_k + k, \tau_{k+1})$, the equation for a can be written as:

$$\dot{a} = \frac{b^2}{b^2 + (\alpha - a)^2}\left(\frac{\dot{y}}{y} - \alpha\right) \tag{3.1}$$

From (2.12) we know that for $t \in [\tau_k + k, \tau_{k+1})$: $b^2 + (\alpha - a)^2 = r_k^2$, hence (3.1) can be written as:

$$\dot{a} = \frac{r_k^2 - (\alpha - a)^2}{r_k^2}\left(\frac{\dot{y}}{y} - \alpha\right) \tag{3.2}$$

Or, equivalently:

$$\frac{r_k^2\,\dot{a}}{r_k^2 - (\alpha - a)^2} = \frac{\dot{y}}{y} - \alpha \tag{3.3}$$

Integrating both sides of (3.3) gives:

$$\frac{1}{2} r_k \log\left(\frac{r_k + a - \alpha}{r_k - a + \alpha}\right) = \log |y| - \alpha t + c_k \tag{3.4}$$

where c_k is determined by $a(\tau_k + k)$, $b(\tau_k + k)$, and $y(\tau_k + k)$. Taking exponentials at both sides of (3.4) and solving for a gives, for $t \in [\tau_k + k, \tau_{k+1})$:

$$a(t) = [1 + (d_k e^{-\alpha t} |y|^{\frac{-2}{r_k}}]^{-1} [\alpha - r_k + (r_k + \alpha)(d_k e^{-\alpha t} |y|)^{\frac{-r_k}{2}}] \tag{3.5}$$

where $d_k = e^{c_k}$, and:

$$c_k = \frac{1}{2} r_k \log\left(\frac{r_k + a(\tau_k + k) - \alpha}{r_k - a(\tau_k + k) + \alpha}\right) - \log |y(\tau_k + k)| - (\tau_k + k)\alpha \tag{3.6}$$

$$r_k = [(a(\tau_k + k) - \alpha)^2 + b(\tau_k + k)^2]^{\frac{1}{2}} \tag{3.7}$$

Finally, from (2.12) we derive:

$$b(t) = \text{sign}(b(\tau_k + k))\,[r_k^2 - (a(t) - \alpha)^2]^{\frac{1}{2}} \tag{3.8}$$

We will now study the equations for $t \in I_k$. For $t \in I_k$, we have:

$$\dot{a} = \frac{1}{1 + C_k^2}\left(\frac{\dot{y}}{y} - a - bC_k\right) \tag{3.9}$$

$$\dot{b} = \frac{C_k}{1 + C_k^2}\left(\frac{\dot{y}}{y} - a - bC_k\right) \tag{3.10}$$

From which we deduce:

$$a(t) = a(\tau_k) + \frac{1}{1+C_k^2}(\log|y| - \int_{\tau_k}^{t} a(s)ds - C_k \int_{\tau_k}^{t} b(s)ds) \tag{3.11}$$

$$b(t) = b(\tau_k) + \frac{C_k}{1+C_k^2}(\log|y| - \int_{\tau_k}^{t} a(s)ds - C_k \int_{\tau_k}^{t} b(s)ds) \tag{3.12}$$

Hence both parts of the algorithm are implementable. The implementable form of the algorithm is now obtained by replacing (2.6.a) by (3.11) and (2.6.b) by (3.12).

EPILOGUE

We have presented a mathematical framework for a class of adaptive controllers. Based on this framework we studied two well-known control problems in terms of the sets G and H. These sets proved to be useful instruments to determine what kind of algorithm was potentially suitable for a particular adaptive control problem. It turned out that for the pole assignment problem an algorithm based on certainty equivalence and neutral certainty equivalence could be developed. The situation for LQ control was more complicated. The analysis in II.3 already showed that algorithms based on certainty equivalence could result in sub-optimal behavior. By adding a closed-loop probing signal this problem was overcome.

A delicate matter is the problem of avoiding certain "dangerous" regions in the parameter space. We have proposed a fairly general method for modifying algorithms so as to assure that these regions are indeed avoided without changing the asymptotic properties of the algorithm. A variant of this method was applied to the pole assignment algorithm and the adaptive LQ controller.

Although the set-up was mainly meant to be used in discrete time, we were able to develop an adaptive control algorithm for a first order continuous time system, based on similar ideas.

Some questions have not completely been answered. Let us list a few:

(i) The multivariable case has not been addressed. For the pole assignment problem single-input is important, since that yields the uniqueness of the control law that assigns the poles. The weak self-tuning property is guaranteed by Theorem II.2.4, which relies on the uniqueness of the control law. In the multi-input case uniqueness of the control law no longer holds and hence we can not hope to obtain a similar result as for the single-input case. However, it should not be too difficult to extend the results for the adaptive pole assignment problem to the multi-output case.

For adaptive LQ-control for higher order systems the problems are of a different nature. Regarding Theorem II.3.10 we have to assure that the sequence of estimates converges to the true parameter. The problem of generalizing the ideas on which the first order algorithm was based is still under consideration.

(ii) The classification problem from II.4 has only been solved partly. The general case where the control law may also depend on the output-vector is still under investigation.

(iii) We have considered the continuous time case for first order systems only. It is our aim to generalize the idea to higher order systems.

The most important question is of course how to apply our ideas in practical situations. It is obvious that direct application of theoretical results like ours is not feasible, since we did not address robustness issues. Maybe our work should be seen as a starting point for further refinements. In our opinion the big challenge is to come up with algorithms which have a hierarchy of properties. They should work perfect in the ideal noise-free and disturbance free case,

and they should be robust with respect to all kinds of disturbances, such as noise, unmodelled dynamics, non-linearities and time changes. This reflects in a certain sense the ideal adaptive controller, but it is not yet clear how to achieve this goal. In our opinion an important part of the adaptive control problem should be the modelling issue. Modelling *pur sang* is a difficult problem, let alone in combination with control. Maybe the combination of modelling and control will provide a good framework for adaptive control in the future.

NOTATIONAL CONVENTIONS

We use the following notation:

$\mathbb{N}$: the natural numbers.

$\mathbb{R}$: the real numbers.

$\mathbb{C}$: the complex numbers.

$Gl(n)$: the general linear group.

$\mathbb{R}^{k\times l}$: $k\times l$ matrices with coefficients in $\mathbb{R}$.

$\mathbb{R}^+$: the positive real numbers, exclusive zero.

$\mathbb{R}[z]$: the ring of polynomials in one indeterminate and coefficients in $\mathbb{R}$.

$\Sigma(n)$ class of single-input/single-output systems of order n.

$\Sigma_c(r)$: class of controllers for $\Sigma(n)$ of order r.

c_0: this notation is exclusively reserved for the n-vector $[1,0,...,0]\in\mathbb{R}^{1\times n}$.

Estimates of parameters are denoted by "hats", like $\hat{a}$ and $\hat{B}$. A variable, parameter or estimate at time k is denoted by subscript k, like a_k, or it is denoted by $a(k)$.

If $A\in\mathbb{R}[z]$ is a polynomial of degree n, then we denote the coefficients of A by lower case characters; $a_n,...,a_0$.

$$E := \{(A,b,c)\in\mathbb{R}^{n\times n}\times\mathbb{R}^{n\times 1}\times\mathbb{R}^{1\times n} \mid (A,b,c) \text{ minimal } \}$$

$$E_{re} := \{(A,b)\in\mathbb{R}^{n\times n}\times\mathbb{R}^{n\times 1} \mid (A,b) \text{ reachable } \}$$

$$E_{ob} := \{(A,b)\in\mathbb{R}^{n\times n}\times\mathbb{R}^{n\times 1} \mid (A,b) \text{ reachable } (c_0,A) \text{ in standard observable form } \}$$

$$E_{ns} := \{(A,b)\in\mathbb{R}^{n\times n}\times\mathbb{R}^{n\times 1} \mid (A,b,c_0) \text{ minimal } A \text{ non-singular } \}$$

$$P := \{K\in\mathbb{R}^{n\times n} \mid K = K^T > 0 \}$$

A □ denotes the end of a remark, proof, theorem etc.

REFERENCES

1. J. VAN AMERONGEN AND A.J. UDINK TEN CATE (1975). Model reference adaptive autopilots for ships. *Automatica 11,* 441-449.
2. J. VAN AMERONGEN (1984). Adaptive steering of ships - A model reference approach. *Automatica 20,* 3-14.
3. B.D.O. ANDERSON AND R.M. JOHNSTONE (1985). Global adaptive pole positioning. *IEEE Trans. Aut. Contr. AC-30,* 11-22.
4. K.J. ASTRÖM (1983). Theory and applications of adaptive control - A survey. *Automatica 19,* 471-486.
5. K.J. ASTRÖM AND B. WITTENMARK (1973). On self tuning regulators. *Automatica 9,* 185-199.
6. A. BECKER, P.R. KUMAR AND C.Z. WEI (1985). Adaptive control with the stochastic approximation algorithm: geometry and convergence. *IEEE Trans. Aut. Contr. AC-30,* 330-338.
7. V. BORKAR AND P. VARAIYA (1979). Adaptive control of Markov chains, I: Finite parameter set. *IEEE Trans. Aut. Contr. AC-24,* 953-957.
8. U. BORISSON AND R. SYDING (1976). Self-tuning control of an ore crusher. *Automatica 12,* 1-7.
9. C.I. BYRNES, U. HELMKE AND A.S. MORSE (1986). Necessary conditions in adaptive Control. *Modelling, Identification and Robust Control.* C.I. Byrnes and A. Lindquist (eds.). Elsevier Science Publishers B.V. (North-Holland), 3-14.
10. R. CRISTI (1986). Adaptive control with finite time persistency of excitation. *IEEE Trans. Aut. Contr. AC-31,* 982-985.
11. D.F. DELCHAMPS (1983). Analytic stabilization and the algebraic Ricatti equation. *Proceedings of the 22nd IEEE Conference on Decision and Control,* San Antonio TX. 1396-1401.
12. B. EGARDT (1979). *Stability of Adaptive Controllers.* Springer-Verlag, Berlin, Heidelberg, New York.
13. H. ELLIOT (1982). Direct pole placement with application to non-minimum phase systems. *IEEE Trans. Aut. Contr. AC-27,* 720,722.
14. H. ELLIOTT, R. CRISTI AND M. DAS (1985). Global stability of adaptive pole placement algorithms. *IEEE Trans. Aut. Control. AC-30,* 348-356.
15. H. ELLIOTT AND W.A. WOLOVICH (1979). Parameter adaptive identification and control. *IEEE Trans. Aut. Control. AC-24,* 592-599.
16. A.A. FEL'DBAUM (1960,1961). Dual control theory. I,II,III,IV. *Aut. & Remote Control,* 21: 874-880, 21: 1033-1039, 22: 1-12, 22: 109-121.
17. A. FEUER (1986). A parametrization for model reference adaptive pole placement. *IEEE Trans. Aut. Control. AC-31,* 782-785.
18. F.R. GANTMACHER (1960). *Matrix Theory I, II.* Chelsea publishing company, New York.
19. G.C. GOODWIN, D.J. HILL AND D.Q. MAYNE (1986). Adaptive robust control. *Proceedings of the 25th IEEE Conference on Decision and Control,* Athens, Greece. 468-473.
20. G.C. GOODWIN AND K.S. SIN (1984). *Adaptive Filtering Prediction and*

Control. Prentice-Hall, Inc., Englewood Cliffs, N.J.

21. G.C. GOODWIN AND K.S. SIN (1981). Adaptive control of nonminimum phase systems. *IEEE Trans. Aut. Contr. AC-26,* 478-483.
22. G.C. GOODWIN AND D.Q. MAYNE (1987). A parameter estimation perspective of continuous time model reference adaptive control. *Automatica 23,* 57-70.
23. G. GOODWIN, P.J. RAMADGE AND P. CAINES (1981). Discrete time stochastic adaptive control. *SIAM J. Contr. and Opt. 19,* 829-853.
24. M. GUPTA (1986). *Adaptive Methods for Control System Design.* IEEE Press, New York.
25. O. HIJAB (1984). Optimal adaptive control and stabilization of families of linear systems. *Systems & Control Letters 4,* 195-198.
26. T. KAILATH (1980). *Linear Systems.* Prentice-Hall, Inc., Englewood Cliffs, N.J.
27. G. KREISSELMEIER (1977). Adaptive observers with exponential rate of convergence. *IEEE Trans. Aut. Contr. AC-22,* 2-8.
28. G. KREISSELMEIER (1980). Adaptive control via adaptive observation and asymptotic feedback matrix synthesis. *IEEE Trans. Aut. Contr. AC-25,* 717-722.
29. G. KREISSELMEIER (1980). Algebraic separation in realizing a linear state feedback control law by means of an adaptive observer. *IEEE Trans. Aut. Contr. AC-25,* 238-243.
30. G. KREISSELMEIER (1985). An approach to stable indirect adaptive control. *Automatica 21,* 425-431.
31. G. KREISSELMEIER AND M.C. SMITH (1986). Stable adaptive regulation of arbitrary *n*th-order plants. *IEEE Trans. Aut. Contr. AC-31,* 299-305.
32. P.R. KUMAR (1985). A survey of some results in stochastic adaptive control. *SIAM J. Contr. and Opt. 23,* 329-380.
33. P.R. KUMAR AND W. LIN (1982). Optimal adaptive controllers for unknown systems. *IEEE Trans. Aut. Contr. AC-27,* 765-774.
34. P.R. KUMAR (1983). Simultaneous identification and adaptive control of unknown systems over finite parameter sets. *IEEE Trans. Aut. Contr. AC-28,* 68-76.
35. P.R. KUMAR AND A. BECKER (1982). A new family of optimal adaptive controllers. *IEEE Trans. Aut. Contr. AC-27,* 137-146.
36. P.R. KUMAR (1982). Adaptive control with a compact parameter set. *SIAM J. Contr. and Opt. 20,* 9-13.
37. P.R. KUMAR AND P. VARAIYA (1986). *Stochastic Systems: Estimation, Identification, and Adaptive Control.* Prentice-Hall, Inc., Englewood Cliffs, N.J.
38. H. KWAKERNAAK AND R. SIVAN (1972). *Linear Optimal Control Systems.* John Wiley & Sons, Inc., New York.
39. P. LANCASTER (1969). *Theory of Matrices.* Academic Press, New York.
40. PH. DE LARMINAT (1984). On the stabilizability condition in indirect adaptive control. *Automatica 20,* 793-795.
41. W. LIN, P.R. KUMAR AND T.I. SEIDMAN. (1985) Will the self-tuning

approach work for general cost criteria? *Systems & Control Letters 6,* 77-86.

42. R. LOZANO-LEAL AND G.C. GOODWIN (1985). A globally convergent adaptive pole placement algorithm without a persistency of excitation requirement. *IEEE Trans. Aut. Contr. AC-30,* 795-798.
43. B. MARTENSSON (1985). The order of any stabilizing regulator is sufficient a priori information for adaptive stabilization. *Systems & Control Letters 6,* 87-91
44. B. MARTENSSON (1986). *Adaptive Stabilization.* Doctoral Dissertation. Department of Automatic Control, Lund Institute of Technology, Sweden.
45. S.P. MEYN AND P.E. CAINES (1985). The zero divisor problem of multivariable adaptive control. *Systems & Control Letters 6,* 235-238.
46. J.B. MOORE (1986). Central tendency pole assignment. *Proceedings of the 25th IEEE Conference on Decision and Control,* Athens, Greece. 100-105.
47. A.S. MORSE (1983). Recent problems in parameter adaptive control. *Outils et Modèles Mathématiques pour l'Automatique, l'Analyse de Systèmes et le Traitement du Signal. Vol. 3.* I.D. Landau (ed.). Editions CNRS., 733-740.
48. R.D. NUSSBAUM (1983). Some remarks on a conjecture in parameter adaptive control. *Systems & Control Letters 3,* 243-246.
49. J.W. POLDERMAN (1985). A note on the structure of two subsets of the parameter space in adaptive control problems. *Systems & Control Letters 7,* 25-34.
50. J.W. POLDERMAN (1986). Adaptive pole assignment by state feedback (short version). *Proceedings of the 25th IEEE Conference on Decision and Control,* Athens, Greece. 106-108.
51. J.W. POLDERMAN (1986). On the necessity of identifying the true system in adaptive LQ control. *Systems & Control Letters 8,* 87-91.
52. J.W. POLDERMAN (1987). *Adaptive Exponential Stabilization of a First Order Continuous-time System.* Report OS-R8705, Centre for Mathematics and Computer Science.
53. J.W. POLDERMAN (1988). Adaptive pole assignment by state feedback. CH. I. BYRNES, A. KURZHANSKI (Eds.). *Proceedings of the IIASA Conference on Modelling and Adaptive Control.* 1986, July 7-15, Sopron Hungary, 318-325.
54. J.W. POLDERMAN (1988). *Adaptive LQ Control: Conflict between Identification and Control.* Memorandom 718, Department of Applied Mathematics, University of Twente. Also to appear in *Linear Algebra and its Applications.*
55. J.W. POLDERMAN (1989). A State Space Approach to the Problem of Adaptive Pole Assignment. *Mathematics of Control, Signals, and Systems 2.* 71-94.
56. L. PRALY (1984). Towards a globally stable direct adaptive control scheme for not necessarily minimum phase systems. *IEEE Trans. Aut.*

Contr. AC-29, 946-949.

57. C.E. ROHRS, L. VALAVANI, M. ATHANS AND G. STEIN (1981). Analytical verification of undesirable properties of direct model reference adaptive control algorithms. *Proceedings of the 20th IEEE Conference on Decision and Control,* San Diego CA. 1272-1284.
58. S.S. SASTRY (1984). Model reference adaptive control- Stability, parameter convergence, and robustness. *IMA Journal of Mathematical Control & Information 1,* 27-66.
59. M. SPIVAK (1965). *Calculus on Manifolds.* Benjamin, Menlo Park CA.
60. E. TSE, Y. BAR-SHALOM AND L. MEIER (1973). Wide-sense adaptive dual control for nonlinear stochastic systems. *IEEE Trans. Aut. Contr. AC-18,* 98-108.
61. J.C. WILLEMS (1986). From time series to linear system, part I. *Automatica 22,* 561-580.
62. J.C. WILLEMS AND C.I. BYRNES (1984). Global adaptive stabilization in the absence of information on the sign of the high frequency gain. *Analysis and Optimization of Systems. Part 1.* Proceedings of the Sixth International Conference on Analysis and Optimization of Systems. A. Bensoussan and J.L. Lions (eds.). Springer Verlag, Berlin Heidelberg New York Toronto, 49-57.
63. W.M. WONHAM (1979). *Linear Multivariable Control: a Geometric Approach.* Springer, New York.

INDEX

Ackermann's formula 32,70
active learning 91
adaptive control algorithm 23
adaptive control, definition of 4
adaptive control, direct 6
adaptive control, features 7
adaptive control, indirect 7
adaptive control, model reference 12
adaptive control, robust 10
adjoint operator 45
certainty equivalence 20
certainty equivalence, neutral 22,26
closed-loop identification 7,23
conflict between identification and control 9
conflux of identification and control 9
conjecture, Morse's 12,18
continuous-time, algorithm 101
control objective 3,19
controller, adaptive 20
dead-beat observer 32
direct adaptive control 22
dual control 13,91
dynamical system 3
excitation 91
exciting, sufficiently 63,66
G, definition of 27
H, definition of 28
identification 7
identification, off-line 5
identification, on-line 7
indirect adaptive control 22
invariant points, of algorithm 26
laws, of a system 3
learning, passive 21,22
linear quadratic control 40
linear quadratic control, adaptive 91
manifold, embedded 44
markov chain, adaptive control of 13
minimum-variance control 12
Morse's conjecture 12,18
neutral certainty equivalence 22,26
neutrality 21,26
observer, adaptive 82
ore crusher 10
pendulum, inverted 3
pole-assignment, adaptive 69
projection algorithm 73
recursiveness 21,26
riccati equation, analyticity of solution 49
self-tuning 12
self-tuning, weak 29
sensitive 21
ship steering 11
spec 45
stability, exponential 107
stabilization, adaptive 12
standard observable form 31
universal controller 16
universality, adaptive 17
universality, global 17
universality, local 17
universality, robustness 16
weak self-tuning 29,77