Controllers for Bilinear and Constrained Linear Systems

Gutman, Per-Olof

1982

Controllers for Bilinear and Constrained Linear Systems

Per-Olof Gutman

Lund 1982

(2)

Printed in Sweden
Studentlitteratur
Lund 1982

LUND INSTITUTE OF TECHNOLOGY
DEPARTMENT OF AUTOMATIC CONTROL
Box 725
S 220 07 Lund 7 Sweden

Document name
DOCTORAL DISSERTATION

Date of issue
March 1982

Document number
CODEN:LUTFD2/(TFRT-1022)/1-133/(1982)

Author(s)
Per-Olof Gutman

Supervisor
Per Hagander

Sponsoring organization

Title and subtitle

Controllers for Bilinear and Constrained Linear Systems

Abstract

This thesis treats feedback control design for certain non-linear systems. In Part 1 it is demonstrated that a constant plus quadratic feedback law stabilizes a bilinear system under certain conditions. A different controller is found for dyadic bilinear systems. In Part 2, sufficient conditions are given for a saturated linear feedback control for a linear system with bounded controls to be stable in an extended region. In Part 3, some optimal control problems for linear systems with linear input- and state constraints are solved as sequences of linear programming problems. The scheme is implemented in an Open-Loop Optimal Feedback fashion, and used to control a real-life process.

Key words

closed loop, optimal control, stability

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

Number of pages
133

Recipient's notes

Security classification

Distribution: The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubbis lund.

To the memory of Jan

ת.נ.צ.ב.ה

and

to Anatoly Shcharansky, Viktor Brailovsky, Naum Meimann, and Alexander Yoffe.

CONTENTS

TDH

CONTROLLERS FOR BILINEAR AND CONSTRAINED LINEAR SYSTEMS

The art of controlling linear systems is well developed - one might even claim almost complete. When it comes to the control of non-linear systems, the knowledge is rather fragmentaric. There are in general no systematic control design methods.

It is safe to say that linear systems control theory is far ahead of industrial practice. With inexpensive microcomputers, however, the practical control engineer may solve his non-linear control problems in various ad-hoc ways, e.g. by introducing non-linearities, variable controller structures, etc. In most cases there is no theoretical analysis to support the viability of the implemented schemes. Hence, in this respect, practice is far ahead of theory.

In industrial processes there is in general a separation between local control, done by linear regulators, and global control, e.g. start-ups, large reference value changes, and shut-downs, which are performed in an on-off fashion by programmable controllers. These two modes of operation sometimes counteract each other. Non-linear control theory might bridge the gap so that local and global control tasks are performed by the same controller.

With these points in view, it is a timely task to extend the knowledge of non-linear control systems.

This thesis treats control design problems for a few non-linear systems that are akin to linear systems. Stability of the closed loop system and optimality in some time-domain sence are the design criteria. The theorems give sufficient conditions for the proposed controllers to be stabilizing. Thus the thesis covers a few of the white spots on the map of control design for non-linear systems.

The thesis consists of the following papers:

1. Stabilizing controllers for bilinear systems. IEEE Trans. Automatic Control, AC-26, 917 - 922.

2. A new design of constrained controllers for linear systems.

3. Application of linear programming for on-line control. Submitted to the 3rd IFAC/IFIP Symposium on software for computer control. Madrid, Spain, Oct. 5 - 8, 1982.

The problem of controlling the <u>bilinear system</u>

$$\dot{x} = Ax + \sum_{i=1}^{m} (B_i x + b_{i0}) u_i \qquad (1)$$

evokes interest because of several reasons: It is a "simple" class of non-linear systems slightly more complex than linear systems. Knowing how to control this class is a logical step in the development of control theory. Many non-linear systems may be approximated by bilinear models, see Sussman (1976). Especially a bilinear approximation may be acceptable in a greater region in the state space than a linear one. Several real life control processes are naturally modelled bilinearly, see Mohler (1979).

In this thesis, it is demonstrated that a constant linear plus quadratic feedback law forces trajectories of (1) into an arbitrarily small neighbourhood of the origin assuming the non-intersection of the varieties $x^TP(B_ix + b_{i0}) = 0$ (for some positive definite symmetric matrix, P) over a region of interest; and a different controller is given when $B_i = b_{i0}c_i^T$ for some i. The proposed control laws are applied to a simulated example of neutron control in a fission reactor.

After the submission of part 1 for publication, a few more contributions to the control of bilinear systems have appeared. Longchamp (1980) constructed a stable feedback controller that exhibits bang-bang structure and sliding motion for single input bilinear systems, and Koditschek (1981) gives necessary and sufficient conditions for a constant linar feedback to globally asymptotically stabilize second order bilinear systems.

The problem of finding a <u>stabilizing controller for a system with linear dynamics and bounded controls</u> is treated in part 2 and 3. When this problem is treated e.g. as a time-optimal control problem, it usually leads to an open loop bang-bang solution. By storing a complicated "switching surface", the solution can be made into a closed loop control law. The method has rarely been used in practice. Several authors have suggested simpler approximately time-optimal closed loop solutions, e.g. Melsa (1967) and Luh (1970). These methods have not become popular, either.

Today, a control engineer would typically develop a linear controller and then saturate it. Other than local stability of the closed loop system can in general not be assured. To convince himself that the controller will work, he has to perform extensive simulations and tests. Our idea in part 2 is to find a design method for a saturated linear feedback control that guarantees an extended domain of stability. It requires computer programs for Lyapunov equation solving, for some matrix algebra, and for simulation. A depth regulator designed in this way was actually tested on board a submarine. Due to the simplicity of the controller, it was not difficult to convince the control engineer in charge.

Part 3 considers problems similar to the one of part 2. The linear plant is assumed to be discrete-time with linear input constraints or linear input and state constraints. In the sixties, several authors noted that certain optimal control problems for such a plant can be reformulated as a series of linear programming problems that yield an open loop solution. By solving the LP-problems in each sampling interval, you get a feedback control. Up to now the computation times on existing computers have been prohibitively long. This study shows that today the method could be practical for processes with sampling intervals of 10 seconds or more. This time will certainly decrease in the future.

The main contribution of part 3 is that it contains the first (reported) real life use of the LP-regulator in a feedback fashion to control a process with bounds both on the inputs and states. Some new ideas on how to design such a regulator are also presented.

ACKNOWLEDGEMEMT

It is not possible to exaggerate the rôle of my advisor Prof. Per Hagander in the creation of this thesis. He has helped me make my sloppy ideas and writing much clearer. Especially I would like to mention his contribution to the biochemical flow example of part 1 and the theorems of part 2. Most of all I appreciate him because he is a great friend who has shown a lot of tolerance and patience. The members of the Department of Automatic Control form a fantastic group in which it is very stimulating and exciting to work. Prof. Karl Johan Åström is largely responsible for this happy state of affairs through his dynamic and enthusiastic leadership. To him, and to all my collegues, I extend the warmest thanks. My wife Hélène has been a wonderful source of moral support. She has shown great understanding when I have been away from home evenings and week-ends. She has delayed her own doctoral studies in order to speed up mine.

I have been supported from various sources. The Swedish Governement provided me with a grant ("utbildningsbidrag") which is rather skimpy. It was supplemented by a teaching assistantship at the Department of Automatic Control, a stipend from Carl Swartz' Memorial Foundation, and my wife's earnings. A part of my course work was done at University of California, Los Angeles. The stay there was made possible by grants from Lund University, the Sweden-America Foundation, the Fulbright Commission, the Fernström Foundation, The Gålö Foundation, and the Foundation Blanceflor-Boncompagni, born Bildt. I thank these institutions for a very rewarding time at UCLA, and for their contribution to my Ph D-degree.

REFERENCES

Koditschek D.E., Narendra K.S. (1981): Stabilizability of Second Order Bilinear Systems. S&IS Report no. 8109, Yale Univ., New Haven, Connecticut. Submitted to IEEE-AC.

Longchamp R. (1980): Stable Feedback Control of Bilinear Systems. IEEE Trans. Aut. Contr., vol. AC-25, no. 2, 302-306.

Luh J.Y.S., Shafran J.S. (1970): An Approximate Minimal Time Closed-Loop Controller for Processes with Bounded Control Amplitudes and Rates. IEEE Trans. Aut. Contr., vol. AC-15, no. 2, 185-194.

Melsa J.L., Schultz D.G. (1967): A Closed Loop, Approximately Time-Optimal Control Method for Linear Systems. IEEE Trans. Aut. Contr., vol. AC-12, no. 1, 94-97.

Mohler R.R., Kolodziej W.J. (1979): Bilinear Systems in Theory and Applications. Proc. 1979 JACC, Denver, Co., June 17-19, 1-6.

Sussman H.J (1976): Semigroup Representation, Bilinear Approximation of Input-Output Maps and Generalized Inputs, in Marchesini, Mitter (eds.): Mathematical Systems Theory, New York:Springer.

PART 1 - STABILIZING CONTROLLERS FOR BILINEAR SYSTEMS

Reprint from IEEE Transactions on Automatic Control, vol. AC-26, No. 4, August 1981, 917 - 922

IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. AC-26, NO. 4, AUGUST 1981

Stabilizing Controllers for Bilinear Systems

PER-OLOF GUTMAN

Abstract—**The problem of finding stabilizing feedback controllers for continuous bilinear systems, where the controls act additively and multiplicatively simultaneously, is treated. The applicability of the "quadratic" control law of Jacobson [5] and others is extended to the case when the *A*-matrix has arbitrary eigenvalues under certain conditions. A class of bilinear systems common among biochemical flow systems is defined: the dyadic bilinear systems. A control scheme, the so-called division controller, for dyadic bilinear systems is suggested. The practicality of the control schemes is demonstrated on the problem of controlling the neutron level in a fission reactor.**

I. Introduction

This paper deals with the problem of finding stabilizing feedback controllers for continuous bilinear systems, where the controls act additively and multiplicatively simultaneously. The importance of the problem is underscored by the existence of several bilinear models of real life control processes; see, e.g., Mohler [8], Mohler and Ruberti [9], Mohler and Kolodziej [10], and España and Landau [3].

The problem has been treated in several reports, among them Jacobson's [5]. The control problem has been solved in essence under certain assumptions by a "quadratic" control law when the *A*-matrix has no eigenvalues in the right-half plane.

This paper contains an extension of the result of Jacobson and others. The case when the *A*-matrix has arbitrary eigenvalues is covered under certain assumptions.

A special class of bilinear systems, the dyadic bilinear systems, is defined. This class is quite common, especially among biological and biochemic flow systems. A design method for stabilizing controller for dyadic systems, the so-called division controller, is suggested.

The paper is organized as follows. Section II contains definitions and the problem statement. Section III shows the extension of the "quadratic" controller of Jacobson and others and two nontrivial examples. Section IV contains the definition of dyadic bilinear systems, conditions for stabilizability of those systems, and the design of the division controller. Section V is devoted to an example of neutron control in a fission reactor, where both the "quadratic" controller and the division controller can be used. A comparison with the bang–bang control found in Mohler [8] is also done. Section VI contains a summary and discussion. Section VII is the Appendix containing the proof of Theorem 3.1.

Manuscript received November 29, 1979; revised February 11, 1981. Paper recommended by M. Vidyasagar, Past Chairman of the Stability, Nonlinear, and Distributed Systems Committee.

The author is with the Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

0018-9286/81/0800-0917$00.75 ©1981 IEEE

II. Definitions and Problem Statement

Definition 2.1: The bilinear system under consideration is

$$\dot{x}=Ax+\sum_{i=1}^{m}(B_i x+b_{i0})u_i \tag{2.1}$$

where $x\in\mathbb{R}^n$, $u=[u_1\ u_2\ \cdots\ u_m]^T\in\mathbb{R}^m$, and A, B_i, b_{i0} for $i=1,2,\cdots,m$ are real constant matrices of appropriate dimensions. Also define

$$B_0=\left[b_{10}\ \vdots\ b_{20}\ \vdots\ \cdots\ \vdots\ b_{m0}\right]. \tag{2.2}$$

Note that the control u acts additively and multiplicatively simultaneously.

Assumption 2.2: Consider the class of inputs that make the unique solution of (2.1) continuous.

Definition 2.3: Given the system (2.1) with the initial condition, $x(t_0)=x_0$. Ω is said to be a (*null*) *stabilizable region* if, for every $x_0\in\Omega$ and for every neighborhood ω of the origin, there exists a locally bounded function $u(t)$, $t\geqslant t_0$, and a finite time interval T, such that the solution of (2.1) with $u(t)$ as input exists and satisfies $x(t)\in\omega$ for all $t>t_0+T$. If $\Omega=\mathbb{R}^n$, then the system is said to be *stabilizable.*

Problem statement 2.4: Find a stabilizing state feedback control.

III. A "Quadratic" Feedback Control Solution

Several authors, among them Moylan and Anderson [11], Jacobson [5], Slemrod [13], Brockett and Wood [1], and Landau [6], have found that a feedback control of the form

$$u_i=-(B_i x+b_{i0})^T Px, \qquad i=1,2,\cdots,m \tag{3.1}$$

where P is some symmetric, positive definite matrix, will stabilize the system (2.1) under various assumptions on the system. One of the crucial assumptions has been that the A-matrix in (2.1) has no eigenvalues in the right-half plane.

The following theorem gives sufficient conditions for the controller (3.1) to stabilize (2.1) when the A-matrix has arbitrary eigenvalues.

Theorem 3.1: Given the system (2.1), if there exists a matrix $P=P^T>0$, such that

$$\begin{pmatrix}(B_1x+b_{10})^T Px\\ (B_2x+b_{20})^T Px\\ \vdots\\ (B_mx+b_{m0})^T Px\end{pmatrix}\neq 0 \tag{3.2}$$

in the set $\{x\,|\,x\neq 0,\ x^T(PA+A^TP)x\geqslant 0\}$, then there exists an $\alpha>0$, such that the control

$$u_i=-\alpha(B_i x+b_{i0})^T Px, \qquad i=1,2,\cdots,m \tag{3.3}$$

will stabilize (2.1).

Proof: See Appendix.

Remark 3.2: A matrix P satisfying Theorem 3.1 might be found by the following procedure. Compute

$$w = \max_{P=P^T>0} \min_{x\in E\cap H_P} \sum_{i=1}^{m} \left[(B_i x + b_{i0})^T P x\right]^2 \tag{3.4}$$

where E is the region of interest and $H_P = \{x \mid x \neq 0, x^T(PA + A^T P)x \geqslant 0\}$, with the stopping criterion $w \geqslant \delta$ for some $\delta > 0$. An algorithm to solve (3.4) numerically can be found in, e.g., Polak and Mayne [12].

Theorem 3.1 is illustrated by the following three nontrivial examples.

Example 3.3: For single-input systems

$$\begin{cases} \dot{x} = Ax + (Bx + b)u, \quad x \in \mathbf{R}^n,\ u \in \mathbf{R} \\ \text{all eigenvalues of } A \text{ lie in the right-half plane} \end{cases} \tag{3.5}$$

it is possible to find a P that satisfies the condition of Theorem 3.1 iff all eigenvalues of B lie either in the left half plane or in the right-half plane, and $b=0$. In this case, however, there exists a constant control that will make the resulting closed-loop system asymptotically stable.

Example 3.4: Given

$$\dot{x} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} x + \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x u_1 + \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} x u_2 \tag{3.6}$$

with $P=I$ the condition of Theorem 3.1 is satisfied and u_1, u_2 can be calculated according to (3.3). Easy calculations show that the system is forced into $\{x \mid x_1^2 + x_2^2 < \epsilon^2\}$ if $\alpha > 1/\epsilon^2$. Note that no constant control will stabilize (3.6).

Example 3.5: Given

$$\dot{x} = Ax + (B_1 x + b_{10})u_1 + (B_2 x + b_{20})u_2 \tag{3.7}$$

with

$$A = \begin{pmatrix} \frac{1}{6} & 1 \\ 0 & \frac{1}{6} \end{pmatrix}, \quad B_1 = \begin{pmatrix} 5 & 2 \\ 2 & 1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 4 & 5 \\ 5 & 4 \end{pmatrix}, \quad b_{10} = \begin{pmatrix} -3 \\ -2 \end{pmatrix},$$

$$b_{20} = \begin{pmatrix} 2 \\ -2 \end{pmatrix}, \quad \text{and} \quad B_0 = \left(b_{10} \mid b_{20}\right).$$

$P=I$ satisfies Theorem 3.1, since

$$\begin{pmatrix} (B_1 x + b_{10})^T x \\ (B_2 x + b_{20})^T x \end{pmatrix} = 0 \qquad \text{only for } x=0 \text{ or } x \cong \begin{pmatrix} -1.6 \\ 4.0 \end{pmatrix}. \tag{3.8}$$

But

$$x^T(PA + A^T P)x\Big|_{x \cong \begin{pmatrix} -1.6 \\ 4.0 \end{pmatrix}} \approx -7 < 0. \tag{3.9}$$

Thus, the control (3.3) will stabilize (3.7). Note that $[A, B_0]$ is a stabilizable pair in the sense of linear system theory. Therefore, when the control

(3.3), with an appropriate α, has forced the system sufficiently near the origin, a switch to a linear controller $u=K^T x$, with some suitable K, will make the closed-loop system asymptotically stable. This follows easily from the Lyapunov–Poincaré theorem. Note also that no constant control will stabilize (3.7).

IV. Dyadic Bilinear Systems: The Division Controller

In this section a common class of bilinear systems will be defined: the dyadic bilinear systems. Necessary and sufficient conditions for stabilizability will be shown, and a method to construct a stabilizing controller will be suggested.

Definition 4.1: A bilinear system (2.1) is called *dyadic of order d* if for $i=i_1,\cdots,i_d$, $d\in\{1,2,\cdots,m\}$

$$B_i x+b_{i0}=b_{i0}\left(c_i^T x+1\right). \tag{4.1}$$

The following example illustrates that dyadic bilinear systems form quite a common class of systems.

Example 4.2: Consider the following (see Fig. 1) flow system which models, for instance, the effect of drugs on the transfer of some dissolved living matter in the human body. The equations governing the flow system are

$$\begin{cases} V_1\dot{x}_1=u_1(x_1-x_2)+x_1u_2 \\ V_2\dot{x}_2=-u_1(x_1-x_2)+c_2x_2 \end{cases} \tag{4.2}$$

with the constraints $x_1, x_2\geqslant 0$; $c_2>0$; and $u_1, u_2\leqslant 0$, where

V_1 = the volume of pool 1 (m^3)

x_1 = the concentration in pool 1 (kg/m^3)

V_2 = the volume of pool 2 (m^3)

x_2 = the concentration in pool 2 (kg/m^3)

c_2 = the growth rate in pool 2 (s^{-1})

u_1 = the transfer rate between pool 1 and pool 2 (s^{-1})

u_2 = the transfer rate out of pool 1 (s^{-1}).

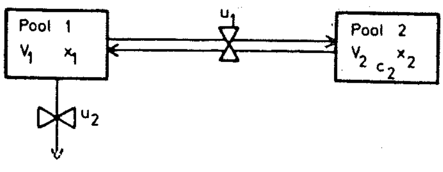

Fig. 1. Model of flow system in the human body.

Rewritten in standard fashion the systems equation became

$$\dot{x}=Ax+B_1xu_1+B_2xu_2 \tag{4.3}$$

where

$$A=\begin{pmatrix} 0 & 0 \\ 0 & \dfrac{c_2}{V_2} \end{pmatrix}, \quad B_1=b_1d_1^T=\begin{pmatrix} \dfrac{1}{V_1} \\ -\dfrac{1}{V_2} \end{pmatrix}(1 \quad -1),$$

$$B_2=b_2d_2^T=\begin{pmatrix} \dfrac{1}{V_1} \\ 0 \end{pmatrix}(1 \quad 0).$$

We note that B_1 and B_2 are dyads.

A desired equilibrium point $x_e=(x_{1e}, x_{2e})^T$ may be chosen, such that $x_{2e}>x_{1e}>0$. Setting the left-hand sides in (4.2) equal to zero gives the corresponding equilibrium input.

$$u_e(u_{1e}, u_{2e})^T=\left(\frac{c_2x_{2e}}{x_{1e}-x_{2e}}, \frac{-c_2x_{2e}}{x_{1e}}\right)^T. \tag{4.4}$$

Transforming (4.3) with x_e as the origin in the state space gives the following bilinear system:

$$\Delta\dot{x}=\tilde{A}\,\Delta x+\left[b_1d_1^T\Delta x+b_1(x_{1e}-x_{2e})\right]\Delta u_1 + \left[b_2d_2^T\Delta x+b_2x_{1e}\right]\Delta u_2 \tag{4.5}$$

where

$$\Delta x=x-x_e, \quad \Delta u=u-u_e,$$

and

$$\tilde{A}=\begin{pmatrix} \dfrac{c_2x_{2e}^2}{V_1(x_{1e}-x_{2e})x_{1e}} & \dfrac{-c_2x_{2e}}{V_1(x_{1e}-x_{2e})} \\ \dfrac{-c_2x_{2e}}{V_2(x_{1e}-x_{2e})} & \dfrac{c_2x_{1e}}{V_2(x_{1e}-x_{2e})} \end{pmatrix}.$$

Define

$$\tilde{b}_1=(x_{1e}-x_{2e})b_1, \quad \tilde{d}_1=\frac{d_1}{(x_{1e}-x_{2e})},$$

$$\tilde{b}_2=x_{1e}b_2, \quad \tilde{d}_2=\frac{d_2}{x_{1e}}. \tag{4.6}$$

Then (4.5) can be rewritten as

$$\Delta\dot{x}=\tilde{A}\,\Delta x+\tilde{b}_1(\tilde{d}_1^T\Delta x+1)\Delta u_1+\tilde{b}_2(\tilde{d}_2^T\Delta x+1)\Delta u_2. \tag{4.7}$$

Consequently (4.5) is dyadic of order 2.

In the sequel bilinear systems dyadic of order 1 will be treated. However, the results can be generalized to systems of higher dyadic order.

Consider the single-input dyadic system

$$\begin{cases} \dot{x}=Ax+(Bx+b)u, & x\in\mathbf{R}^n,\ u\in\mathbf{R} \\ (Bx+b)=b(c^Tx+1). \end{cases} \tag{4.8}$$

Let

$$d(x)=c^Tx+1. \tag{4.9}$$

$d(x)=0$ defines an $(n-1)$-dimensional hyperplane.

Divide the state space into the following sets:

$$\begin{cases} S_+=\{x|d(x)>0\} \\ S_0=\{x|d(x)=0\} \\ S_-=\{x|d(x)<0\}. \end{cases} \tag{4.10}$$

Referring to Assumption 2.2, let $\varphi(x_0, u(\cdot), t)$ be the solution of (4.8) at time t when $x(0)=x_0$ and $u(s)$, $s\in[0, t]$, is the control input. Define

$$\begin{cases} V=\{x_0\in S_+|\exists u_V \text{ such that } \varphi(x_0, u_V, t)\in S_+\ \forall t, \text{ and} \\ \qquad \varphi(x_0, u_V, t)\to 0, t\to\infty\} \\ Y=\{x_0\in S_0|\, e^{At}x_0\in V \text{ for some } t>0\} \\ W=\{x_0|\exists u_Y \text{ such that } \varphi(x_0, u_Y, t)\in Y \text{ for some } t>0\}. \end{cases} \tag{4.11}$$

Theorem 4.3: A necessary and sufficient condition for stabilizing (4.8) is that

$$1)\quad Y \text{ is nonempty, and} \tag{4.12}$$

$$2)\quad V\cup Y\cup W=\mathbf{R}^n. \tag{4.13}$$

Proof: Note that the origin is interior to S_+ and belongs to V.

Sufficiency: If $x_0\in V$, the origin may be reached. If $x_0\in Y$, the trajectory may end in V, from where it may continue to the origin. If $x_0\in W$, the trajectory may end in Y, etc. The sufficiency is proved because $V\cup Y\cup W=\mathbf{R}^n$.

Necessity: (1) Control action ceases in S_0. The only way to pass from $S_0\cup S_-$ to S_+ is by power of the autonomous system $\dot{x}=Ax$. In order to reach the origin at least one of the trajectories of the autonomous system emanating from $S_0\cup S_-$ must end in V. Therefore, Y must be nonempty. (2) By definition a trajectory emanating from the set $C(V\cup Y\cup W)$ will remain in this set. Therefore, it must be empty. □

Example 4.4: Consider the system, dyadic of order 1,

$$\dot{x}=x+xu+u. \tag{4.14}$$

Choose an equilibrium point $x_e\neq-1$. The corresponding equilibrium input is

$$u_e = \frac{-x_e}{1+x_e}.$$

The transformed system equation will be

$$\begin{cases} \Delta\dot{x} = \dfrac{1}{1+x_e}\Delta x + \Delta x\,\Delta u + (x_e+1)\,\Delta u \\ \Delta x = x - x_e, \quad \Delta u = u - u_e. \end{cases} \tag{4.15}$$

It is easily realized that, for $x_e < -1$ and the control $\Delta u = 0$, the conditions of Theorem 4.3 are satisfied and the system is stabilizable to the chosen equilibrium point. With $x_e > -1$, we find that the trajectory of the autonomous system in $S_0 = \{x \mid x = -1\}$ is described by $\Delta\dot{x} = -1$, and leads from $S_+ = \{x \mid x > -1\}$ into $S_- = \{x \mid x < -1\}$. Consequently, the system is not stabilizable to the chosen equilibrium point.

The use of Theorem 4.3 stems from the fact that the system (4.8) might be analyzed as a set of linear systems when suitable controls are applied. Then the control problem is transformed to one of finding controls that satisfy Theorem 4.3. This leads to the following.

Design 4.5 (The Division Controller): Consider the system (4.8), (4.9). Define the sets

$$S_0' = \{x \mid -\epsilon_1 \leqslant d(x) \leqslant \epsilon_2\} \tag{4.16}$$

where $\epsilon_1 \geqslant 0$, $\epsilon_2 \geqslant 0$ are control parameters in the sense that the control designer may let them vary depending on time, the actual state, etc.:

$$\begin{cases} S_+' = S_+ \setminus S_0' \\ S_-' = S_- \setminus S_0'. \end{cases} \tag{4.17}$$

Apply

$$\begin{cases} u_+ = \dfrac{k_+^T x + u_{\text{ref}}(x,t)}{d(x)}, & x \in S_+' \\ u_0 = 0, & x \in S_0' \\ u_- = \dfrac{k_-^T x + u_{\text{ref}}(x,t)}{d(x)}, & x \in S_-' \end{cases} \tag{4.18}$$

where $k_+, k_- \in \mathbf{R}^n$. The closed-loop system now becomes

$$\begin{cases} \dot{x} = (A + bk_+^T)x + bu_{\text{ref}}, & x \in S_+' \\ \dot{x} = Ax, & x \in S_0' \\ \dot{x} = (A + bk_-^T)x + bu_{\text{ref}}, & x \in S_0'. \end{cases} \tag{4.19}$$

Select, if possible, k_+, k_-, u_{ref}, ϵ_1, and ϵ_2, such that the conditions of Theorem 4.3 are satisfied. In particular, it is clear that they can be selected to satisfy Assumption 2.2. □

We see that the problem of controlling (4.8) is reduced to combining suitable state-space trajectory portraits (4.19) into a stable whole. u_{ref} is used to (temporarily) change the equilibrium point in order to change, for instance, the set Y. ϵ_1, ϵ_2 are used as controls (for instance as hystereses) in order to avoid stationary points or limit cycles on or around the boundaries of S_0'.

Several examples of the use of the division controller can be found in Gutman [4]. One important example is displayed in the next section.

V. An Example: Neutron Control in a Fission Reactor

A. *Preliminaries*

As a realistic example of bilinear control, we choose the neutron level control problem in a fission reactor described in Mohler [3, pp. 112–119]. The neutron population is described by a second-order model:

$$\begin{cases} \dot{n}=\dfrac{u-\beta}{l}n+\lambda c \\ \dot{c}=\dfrac{\beta}{l}n-\lambda c \end{cases} \tag{5.1}$$

with the state constraints $n>0$, $c>0$, where n=neutron population, c=precursor population, and u=reactivity, which is the control input.

Typical values of the constants are

$$l=10^{-5}, \quad \beta=0.0065, \quad \lambda=0.4. \tag{5.2}$$

A control constraint might be assumed:

$$|u|\leqslant 10^{-3}. \tag{5.3}$$

The control objective is to stabilize (5.1) to a chosen neutron population equilibrium level n_e. With n_e chosen, the precursor population level c_e follows from (5.1):

$$c_e=\frac{\beta}{l\lambda}n_e.$$

Now, (n_e, c_e) is taken as the origin in the transformed state space. The new state-space variables are chosen dimensionless:

$$\begin{cases} x_1=\dfrac{n-n_e}{n_e}>-1 \\ x_2=\dfrac{c-c_e}{c_e}>-1. \end{cases} \tag{5.4}$$

In the new state space (5.1) becomes

$$\dot{x}=Ax+bd(x)u \tag{5.5}$$

where

$$A=\begin{pmatrix} -\dfrac{\beta}{l} & \dfrac{\beta}{l} \\ \lambda & -\lambda \end{pmatrix}, \quad d(x)=\frac{x_1+1}{l}, \quad b=(1 \quad 0)^T.$$

The eigenvalues of A are 0 and $-(\lambda+\beta/l)$. A has thus an eigenvalue on the imaginary axis, and, according to Section III, there might or might not exist a $P = P^T > 0$, such that the control

$$u = -[d(x) \quad 0]Px \tag{5.6}$$

stabilizes (5.5). The possibility of such a control will be investigated in Section 5.2, where it will be called the quadratic control.

We also note that (5.5) and the state-space constraint (5.4) reveal that the system is dyadic and satisfies the conditions of Theorem 4.4. The dyadicity stems from the change of equilibrium point, cf. Example 4.2. The situation is extremely favorable because the allowable state space [defined by (5.4)] is a subset of S_+. Thus, $S_0 = \{x | d(x) = 0\}$ does not belong to the allowable state space. Moreover, $[A, b]$ is a controllable pair. It can be proved that the control

$$u = \frac{k^T x}{d(x)} \tag{5.7}$$

will make the system asymptotically stable for a suitable choice of k. The division control is displayed in Section V-C.

Section 5.4 contains a brief description of the bang–bang controller by which Mohler [8] solved the control problem.

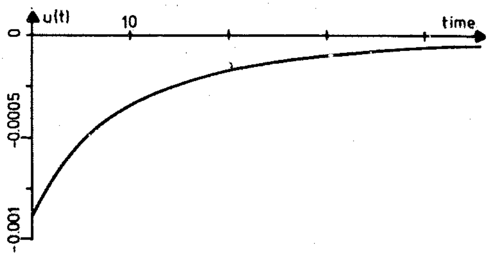

Fig. 2. The control input $u(t)$ for a simulation with the quadratic controller (5.9), (5.10) with the initial condition (2.1).

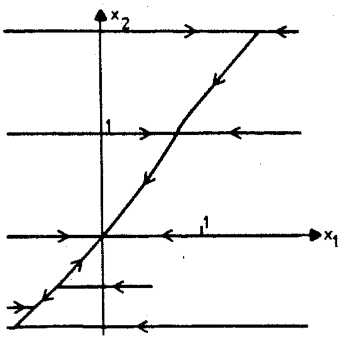

Fig. 3. Phase plane trajectories for simulations with the quadratic controller (5.9), (5.10) subject to the constraint (5.3).

B. *Quadratic Control*

Set

$$P=\begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix}>0, \qquad p_i \in \mathbf{R}, \quad i=1,2,3. \tag{5.8}$$

The quadratic control (5.6) is computed:

$$u=-\frac{x_1+1}{l}(p_1x_1+p_2x_2). \tag{5.9}$$

Let $\Phi=x^TPx$ be a Lyapunov function. Compute $\dot{\Phi}$ as a function of P and select a p_1 and p_2 that stabilize the system. One suitable choice of P is

$$\begin{cases} p_1=40\cdot 10^{-10} \\ p_2=20\cdot 10^{-10} \\ p_3>p_2^2/p_1. \end{cases} \tag{5.10}$$

Simulations were performed (see Figs. 2 and 3). (For details on the simulations, see Gutman [4].) We conclude from the figures that the chosen quadratic control stabilizes the system, although the convergence is rather slow.

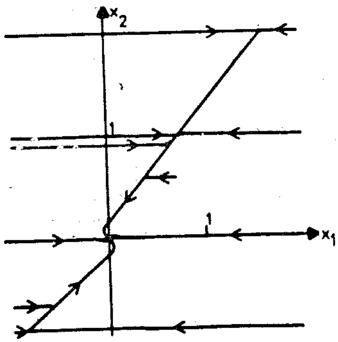

Fig. 4. Phase plane trajectories for simulations with the division controller (5.14) subject to constraint (5.3).

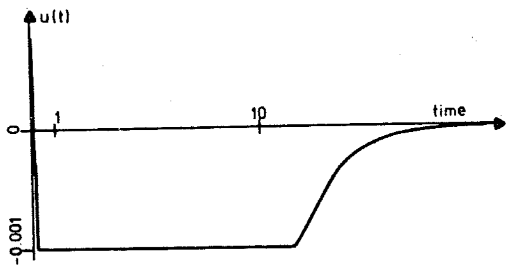

Fig. 5. The control input $u(t)$ for a simulation with the division controller (5.14) subject to constraint (5.3). Initial condition: (2.1).

C. *Division Control*

In the division control (5.7) the vector k^T is chosen as if the linear system

$$\dot{x}=Ax+bu \tag{5.11}$$

is to be controlled by the linear controller

$$u=k^T x. \tag{5.12}$$

The choice can be done, for instance, through the use of linear optimal control with state constraints (see Mårtensson [7]). Here the choice was done to get a stable one-tangent node

$$k^T=(648.4 \quad -650.9). \tag{5.13}$$

Moreover, in the simulation program, a lower bound was set on the denominator in (5.7) in order to avoid numerical overflow. The control input in the simulation was

$$u=\frac{648.4x_1-650.9x_2}{\max(x_1+1,\ \ 0.01)}\cdot 10^{-5}. \tag{5.14}$$

Some simulation results are shown in Figs. 4 and 5.

The division controller seems very attractive. It is easy to design and it provides good local and global control with the same algorithm.

D. *Bang–Bang Control*

Mohler [8] solves the problem of stabilizing the system (5.5) by a time optimal bang–bang controller (see Fig. 6). This control mode is not suitable when the state is near the origin; therefore Mohler [8] suggests a *PI*-controller for local control.

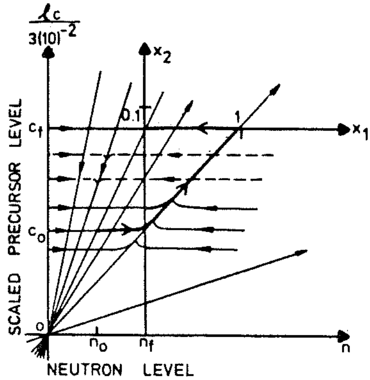

Fig. 6. Adopted from Mohler [8, Fig. 4.1]. Phase plane trajectories for the Mohler [8] bang–bang control. The trajectory from the initial condition (n_0, c_0) to the desired equilibrium point (n_f, c_f) is heavily drawn. (n_f, c_f) corresponds to the origin in the x_1-x_2 space.

E. Conclusions

When comparing the quadratic controller, the division controller, and the bang–bang controller, it seems as if the division controller offers the most advantages. It is very easy to design; the other two are more complicated. Different control objectives can be taken into consideration, including time optimality. The bang–bang control might give an endpoint error, which must be compensated for by another controller, for instance, a *PI*-controller. The division controller provides good local, as well as global control with the same algorithm.

VI. Summary and Discussion

At present there exist no general criteria for necessary and sufficient conditions for stabilizability of bilinear systems. Consequently, there exists no general method to design stabilizing controllers. The literature on the subject contains solved special cases that by no means exhaust all possible bilinear systems.

The contribution of this report is the solution of another few special cases.

1) An extension under certain conditions of the applicability of the "quadratic" controller by Jacobson [5] and others to the case when the A-matrix has arbitrary eigenvalues,

2) a possible design method, the division controller, for the class of dyadic bilinear systems.

The practicality of the above two controllers is demonstrated on the neutron control in a fission reactor.

It is apparent that further work is needed both on general stabilizability theory, and on special cases. It would also be valuable to find more real life processes that can be modeled bilinearly.

Appendix

Proof of Theorem 3.1: The closed-loop system is

$$\dot{x}=f(x) \tag{A1}$$

where

$$f(x)=Ax-\alpha\sum_{i=1}^{m}(B_ix+b_{i0})(B_ix+b_{i0})^TPx$$

with the initial condition $x(0)=x_0$.

Define the Lyapunov function candidate $V=x^TPx$. Then

$$\dot{V}=x^T(PA+A^TP)x-2\alpha\sum_{i=1}^{m}\left[x^TP(B_ix+b_{i0})\right]^2. \tag{A2}$$

The assumption in the theorem implies that, given an $\epsilon>0$, there exists an $\alpha>\text{constant}\cdot 1/\epsilon^2$, such that $\dot{V}<0$ in the set $\{x \mid \|x\|>\epsilon\}$.

Define $\Omega=\{x \mid x^TPx\leqslant c\}$ with c chosen, such that

$$1)\ \dot{V}<0 \text{ on } \partial\Omega, \quad \text{and} \quad 2)\ x_0\in\Omega. \tag{A3}$$

Let $D=\{x \mid |x_i|<d_i,\ i=1,2,\cdots,m\}$ with d_i, $i=1,2,\cdots,m$ chosen such

that $D \supset \Omega$.

It is obvious that $f(x)$ satisfies a Lipschitz condition and is continuous on D. Then, according to Theorem 1.2.3 of [2], there exists a unique solution $\varphi \in C^1$ of (A1) in the finite time interval $[0, \tau]$:

$$\tau = \min_i \frac{d_i}{M}$$

where

$$M = \max_{x \in D} |f(x)|.$$

Now, the following three facts,

$$\begin{cases} V>0 \text{ and } \dot{V}<0 \text{ in the set } \{x \mid \|x\| > \epsilon\}, \\ \text{the definition of } \Omega \text{ (A3), and} \\ \varphi(t) \text{ is continuous,} \end{cases}$$

imply that

$$\varphi(t) \subset \Omega, \qquad 0 \leqslant t \leqslant \tau. \tag{A4}$$

Hence, according to Theorem 1.4.1 of [2], the solution may be continued beyond the time interval $[0, \tau]$.

Repeating the argument above with $\varphi(\tau) \in \Omega$ as the initial condition, it is found that there exists a unique continuation $\varphi \in C^1$ in the time interval $[\tau, 2\tau]$. The argument is now repeated *ad infinitum*, and we conclude that there exists a unique, bounded, and continuous solution of (A1).

Especially it should be noted that V precludes the possibility of a finite escape time.

Since $\dot{V}<0$ in $\{x \mid \|x\| > \epsilon\}$, $x(t)$ converges to an ϵ-neighborhood of the origin and remains there. Thus, the system is stabilized according to Definition 2.2. □

ACKNOWLEDGMENT

I want to thank Dr. D. Hill, Newcastle, Australia, for valuable ideas at the outset of this endeavor. I feel deep gratitude to Prof. P. Hagander, without whose constant advice, assistance, encouragement, good ideas, and endurance nothing would have come out of this effort. And finally, thanks to all my colleagues and friends, especially Dr. P. Molander who have spent numerous hours criticizing and assisting me, and who make this Department of Automatic Control the best possible place at which to work.

References

[1] R. W. Brockett and J. R. Wood, "Electrical networks containing controlled switches," in *Applications of Lie Group Theory to Non Linear Network Problems.* North Hollywood, CA: Western Periodicals, 1974, pp. 7, 8.

[2] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations.* New York: McGraw-Hill, 1955.

[3] M. España and I. D. Landau, "Reduced order bilinear models for distillation columns," *Automatica*, vol. 14, pp. 345–357, 1978.

[4] P. O. Gutman, "Controllers for bilinear systems," Rep. Dep. Automat. Contr., Lund Institute of Technology, Lund, Sweden, 1980.

[5] D. H. Jacobson, *Extensions of Linear-Quadratic Control, Optimization and Matrix Theory.* London: Academic, 1977.

[6] I. D. Landau, "On the optimal regulator problem and stabilization of bilinear systems," Rep. Laboratoire d'Automatique (C.N.R.S.), Grenoble, France, 1979.

[7] K. Mårtensson, *New Approaches to the Numerical Solution of Optimal Control Problems.* Lund, Sweden: Studentlitteratur, 1972.

[8] R. R. Mohler, *Bilinear Control Processes.* New York: Academic, 1973.

[9] R. R. Mohler and A. Ruberti, Eds., *Recent Developments in Variable Structure Systems, Economics, and Biology.* New York: Springer, 1978.

[10] R. R. Mohler and W. J. Kolodziej, "Bilinear systems in theory and applications," in *Proc. 1979 JACC*, Denver, CO, June 17–19, 1979, pp. 1–6.

[11] P. J. Moylan and B. D. O. Anderson, "Non linear regulator theory," *IEEE Trans. Automat. Contr.*, vol. AC-18, pp. 460–465, 1973.

[12] E. Polak and D. Q. Mayne, "On the finite solution of nonlinear inequalities," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 443–445, 1979.

[13] M. Slemrod, "Stabilization of bilinear control systems with applications to nonconservative problems in elasticity," *SIAM J. Contr. Optimiz.*, vol. 16, pp. 131–141, 1978.

PART 2 - A NEW DESIGN OF CONSTRAINED CONTROLLERES FOR LINEAR SYSTEMS

Abstract

A new method is presented to find stabilizing saturated linear state feedback controllers for linear continuous-time and discrete-time systems. A controller of this type was satisfactorily tested on board a submarine as a depth regulator.

1. INTRODUCTION

The problem addressed here has the following background: A plant, modelled by a linear system, works well near a stationary operating point when regulated by PID-, LQ-, or other linear controllers. The control law can be chosen such that the constraints on the control input are not encountered.

However, if the control laws are used for "non-local" control, e.g. for transferring the plant from one stationary point to another, or for large disturbances, then their inherent drawbacks appear. In the words of Frankena and Sivan (1979):

> "Let us assume that the gains of the controller have been adjusted so that at the maximal errors to be encountered, the control vector will be at its maximal allowable level. Because of the linearity of the system, whenever the error is less than the maximal error, the control vector will also be less than its maximal allowable value, and by the same proportion. Thus, particularly for relatively low errors, the control system is operating far from its full capability. A more efficient way of operating is to maintain control vectors which are close to their maximal allowable value until the error has been completely eliminated. Such a controller would evidently be a non-linear controller."

The mathematically most appealing way to overcome the difficulty would be to solve an optimal control problem, for instance the time optimal control problem, see e.g. Athans and Falb (1966) or Fleming and Rishel (1975). This usually leads to a two-point boundary value problem which yields an open loop bang bang solution. By storing a complicated "switching surface", this solution can be made into a closed loop control law.

In most applications, the optimal control solution has not been applied. Notable exceptions are some space applications. One reason seems to be that it is considered complicated and expensive: extensive work off line has to be done to generate the switching surfaces, and their storage requires a lot of expensive memory. Computer hardware technology might change the latter fact.

How is then the problem tackled in industry today? First the control signals are calculated as if no constraints existed, and then they are simply limited. This type of control will be called <u>saturated linear control</u>.

However, it might be seriously deficient. Stability cannot be guaranteed in general, even though the original (unconstrained) linear control yields a stable closed loop system. See e.g. Example 5.1 below.

Sometimes an attempt is made to patch up the saturated linear control. For instance, it is taken into account that the calculated control is not the implemented one, e.g. in the updating of the integral part of a PID-controller, see Åström (1981).

Another way that seems to become popular together with computerized control is dual-mode control. This means for instance that a low gain linear control is used far from the desired working point while a high gain control is used near it. Great care must be taken that limit cycles, or other undesired features, do not occur. See Aström (1971).

For stable systems, a quadratic Lyapunov function can be found, on the basis of which a relay control scheme is based (Aström, 1971). The switching curves are lines, and hence not time optimal, but the closed loop system is speeded up.

Frankena and Sivan (1979) have a solution based on optimal control theory. However, it does not apply to systems with an unstable A-matrix.

It should be noted that unstable systems regulated by a constrained controller, cannot be stabilized for all initial conditions. This paper presents a design technique for saturated linear state feedback control that guarantees stability and allows some tuning to achieve a desired performance.

The idea is the following: the system is first stabilized by a low gain linear state feedback control. A quadratic Lyapunov function is found, on the basis of which another linear state feedback control is computed. The second step is very similar to the above mentioned relay control design. The two controls are added, and saturated.

The paper is organized as follows. Section 2 contains preliminaries, Section 3 presents the design in the continuous time case, and Section 4 in the discrete time case. A few examples are given in Section 5, including the submarine underwater test. A discussion and summary is found in Section 6. Section 7 contains an acknowledgement and Section 8 the references.

2. PRELIMINARIES

The continuous plant which we consider is the following linear time invariant dynamical system

$$\begin{cases} \dot{x} = Ax + Bu \\ x(0) = x_0 \end{cases} \tag{2.1}$$

where $x \in R^n$, $u \in R^m$, and A, B are matrices of appropriate dimensions. Assume that

$$[A,B] \quad \text{is a stabilizable pair,} \tag{2.2}$$

and that the allowed control inputs are constrained

$$g_i \leq u_i \leq h_i, \quad i = 1,2,\ldots,m. \tag{2.3}$$

The control aim is to bring the state from the initial condition x_0 to the vicinity of the origin. We do not define an explicit criterion function, but in view of the discussion in the introduction, it is reasonable to suggest that the transfer should be as speedy as possible while at the same time the inputs, states, and outputs behave to the liking of the design engineer. After all a real-life control system is not judged by how well it satisfies a loss criterion, but by how "reasonable" it performs in the time domain. Compare the discussion in Åström (1976).

Not all points in R^n are initial conditions of interest. Most certainly the control engineer knows to what set the initial conditions may belong. It is also obvious that if the matrix A is unstable, a constrained input (2.3) cannot bring the state to the origin from any point in R^n. This leads to the following:

<u>Definition 2.1</u>: Let D be the set of initial conditions in the state space, from which it is desirable to stabilize the system to the origin.

□

Consider an nxm matrix

$$L \hat{=} [\ell_1 | \ell_2 | \ldots | \ell_m] \tag{2.4}$$

such that

$$A_c \hat{=} A_c(L) \hat{=} (A + BL^T) \tag{2.5}$$

is a stability matrix. Such a matrix does exist due to the assumption (2.2). Associated with each matrix L is the set

$$E \hat{=} E(L) \hat{=} \{z | z \in R^n \text{ and } g_i \le \ell_i^T z \le h_i, \quad i=1,2,\ldots,m\}. \tag{2.6}$$

E(L) is simply the set of initial states at which the stabilizing linear state feedback $L^T x$ does not initially exceed the constraints (2.3).

Another set, associated with each matrix L is

$$F \hat{=} F(L) \hat{=} \bigcap_{t \in [0,\infty)} \left\{ \left[e^{A_c t} \right]^{-1} E \right\}. \tag{2.7}$$

F is a subset of E. Along all trajectories emanating from F, the stabilizing linear state feedback $L^T x$ does not exceed the constraint (2.3). See Fig. 2.1.

As A_c is a stablity matrix the Lyapunov equation

$$PA_c + A_c^T P = - Q \tag{2.8}$$

yields an nxn matrix

$$P(L,Q) \hat{=} P = P^T > 0 \tag{2.9}$$

for every nxn matrix $Q = Q^T > 0$. See Lancaster (1969). It is well known (Åström, 1971) that

$$v(t) = x^T(t) \, Px(t) \tag{2.10}$$

is a Lyapunov function for the system

$$\dot{x} = A_c x. \tag{2.11}$$

A stability region for (2.11) can be defined

$$\Omega \hat{=} \Omega(L,P,c) \hat{=} \{x | x^T Px \leq c\}. \tag{2.12}$$

It is clear (Åström, 1971) that

$$x(0) \in \Omega \Rightarrow x(t) \in \Omega \quad \text{for} \quad t \geq 0. \tag{2.13}$$

From the definitions of Ω and F it follows that

$$\Omega \subset E \Rightarrow \Omega \subseteq F. \tag{2.14}$$

The sets D, E(L) and Ω(L,P,c) are crucially important for the design. It will be shown in Section 3 that a sufficient condition for the design to work is that D is such that for some P, c, and L

$$D \subseteq \Omega \subseteq E. \tag{2.15}$$

The two-dimensional case is depicted in Fig. 2.1.

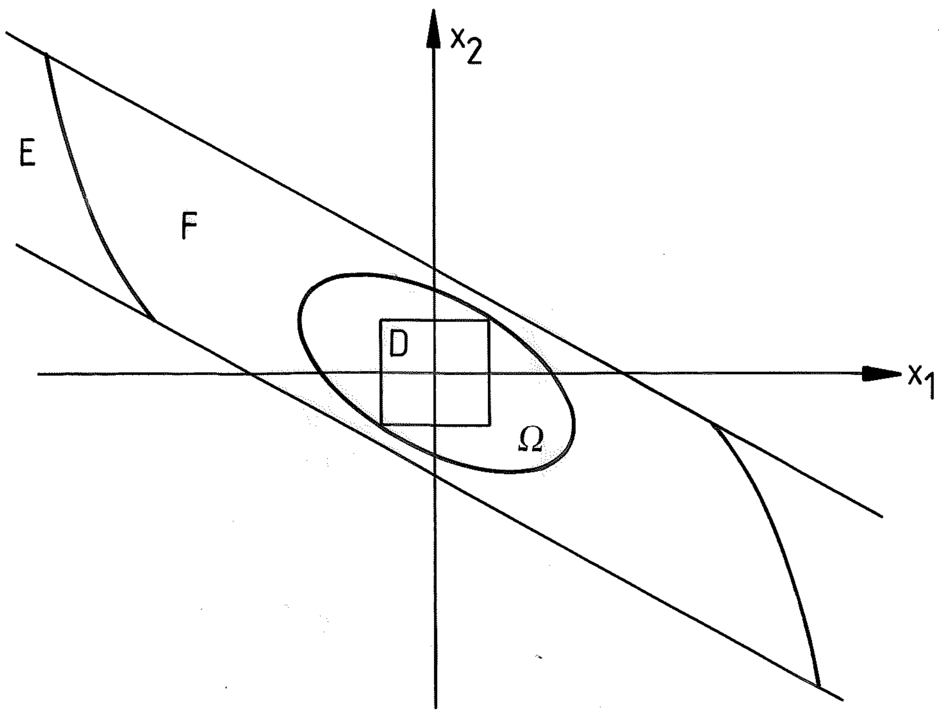

Fig. 2.1 The sets D, Ω, F, and E according to (2.14), (2.15) in the two-dimensional case.

Finally, define the function sat:

<u>Definition 2.2</u>: Let f be an $m\times 1$ vector, and r a scalar. Then

$$\operatorname{sat} f \triangleq \begin{bmatrix} \operatorname{sat}_1 f_1 \\ \operatorname{sat}_2 f_2 \\ \vdots \\ \operatorname{sat}_m f_n \end{bmatrix} \tag{2.16}$$

where

$$\operatorname{sat}_i r = \begin{cases} g_i & \text{if } r \leq g_i \\ r & \text{if } g_i < r < h_i \\ h_i & \text{if } h_i \leq r \end{cases} \tag{2.17}$$

3. THE DESIGN IN THE CONTINUOUS TIME CASE

We proceed to present the design in the continuous time case. As stated above the idea is as follows: a linear state feedback control that does not violate the control constraints is found to stabilize the plant. If this cannot be achieved, the scheme is not applicable. A quadratic Lyapunov function is constructed for the closed loop system. On the basis of the Lyapunov function, another state feedback control is computed. The two controls are added and saturated. Theorem 3.1 contains the stability proof of this scheme, and Algorithm 3.2 suggests how to design the controller.

<u>Theorem 3.1</u>: Let the plant defined by (2.1)-(2.3) be given. Assume that the set D of interest (Definition 2.1) has the property that there exists an nxm matrix L satisfying (2.4),(2.5), an nxn matrix $P = P^T > 0$ satisfying (2.8), and a constant c such that (2.15) is satisfied, i.e.

$$D \subseteq \Omega(L,P,c) \subseteq E(L).$$

Define

$$K \hat{=} \begin{bmatrix} k_1 & & & 0 \\ & \cdot & & \\ & & \cdot & \\ 0 & & & k_m \end{bmatrix}, \quad k_i \geq 0, \quad i=1,2,\ldots,m. \tag{3.1}$$

Then the control

$$u = \text{sat}\,[(L^T - KB^TP)\,x] \tag{3.2}$$

stabilizes the plant for all such K and all initial conditions $x_0 \in D$.

<u>Proof</u>: Initially, it is clear that the control signal in (3.2) is continuous. It follows that the closed loop system has a unique, continuous solution. See Coddington and Levinson (1955).

Consider the unconstrained plant (2.1)-(2.2), with the feedback control

$$u = L^Tx + v \tag{3.3}$$

then v can be chosen such that $x(t) \in \Omega$ if $x_0 \in D \subset \Omega$. For instance, v=0 is such a choice. For v=0, assumption (2.15)

3
5
H

implies that $x(t) \in E(L)$. The remainder of the proof will show how to construct a better v with the constraint (2.3) still enforced.

The closed loop system (2.1),(3.3) is

$$\dot{x} = A_c x + Bv \tag{3.4}$$

with A_c defined in (2.5). Let

$$V = x^T P x \tag{3.5}$$

be a Lyapunov function candidate, with P computed according to (2.8).

Compute

$$\dot{V} = x^T (PA_c + A_c^T P) x + 2\, x^T PBv \tag{3.6}$$

The first term of the right hand side is negative, due to (2.8). In order to ensure that V is a Lyapunov function, we demand that

$$2\, x^T PBv \leq 0. \tag{3.7}$$

Define

$$R = \text{diag}\,(r_1,\ r_2,\ \ldots,\ r_m), \quad r_i \geq 0,\ i=1,2,\ldots,m \tag{3.8}$$

Clearly the choice

$$v = - R B^T Px \qquad (3.9)$$

satisfies (3.7) because

$$2 x^T PBv = - 2 x^T PBRB^T Px \le 0. \qquad (3.10)$$

V is thus a Lyapunov function for the unconstrained system with the control (3.3),(3.9), where it should be noted that the R-matrix may be a function of x.

The next step is to show that the constrained control (3.2) can be written in the form (3.3),(3.9) for $x \in E$ by an appropriate choice of R, i.e. solving

$$L^T x - RB^T Px = \text{sat}\ (L^T x - KB^T Px) \qquad (3.11)$$

for any diagonal $K \ge 0$, and $x \in E$. Consider the i:th row of (3.11)

$$l_i^T x - r_i (B^T P)_i x = \text{sat}\ (l_i^T x - k_i (B^T P)_i x) \qquad (3.12)$$

where $(B^T P)_i$ is the i:th row of $B^T P$. Note that $x \in E$ means that $l_i^T x$ lies inside the interval $[g_i, h_i]$,

$$g_i \le l_i^T x \le h_i . \qquad (3.13)$$

If also $l_i^T x - k_i (B^T P)_i x$ lies inside the interval, then $r_i = k_i \ge 0$. On the other hand, if

$$l_i^T x - k_i (B^T P)_i x > h_i ,$$

then a smaller r_i would decrease the term added to $l_i^T x$, i.e. there is r_i, $0<r_i<k_i$, such that

$$l_i^T x - r_i (B^T P)_i x = h_i ,$$

and similarly at the lower limit g_i. Cf. Figure 3.1.

It is thus shown that $\dot{V} \le 0$, also for the constrained system

$$\dot{x} = Ax + B\,\text{sat}(L^T x - KB^T Px) \tag{3.14}$$

provided $x_0 \in \Omega$, and the stability follows.

□

The main condition of the theorem is

$$D \subseteq \Omega(L,P,c) \subseteq E(L).$$

This is both a condition on L to stabilize the system and a stronger condition to allow for some saturated control. It actually implies

$$D \subseteq F(L) \tag{3.15}$$

or

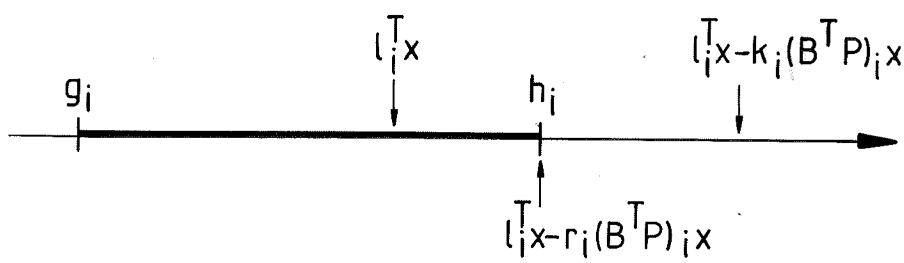

Fig. 3.1 The computation of r_i.

$$g_i \le \ell_i^T e^{A_c t} x \le k_i, \quad t \ge 0, \quad x \in D. \tag{3.16}$$

But $F \supseteq D$ would not have been sufficient to guarantee stability of any extended saturated control.

The design of controller (3.2) involves formulating the set D, and selecting the matrices L, P (or Q), the number c, and K such that the conditions of Theorem 3.1 are satisfied and such that the closed loop system performs desirably. Proceed e.g. as follows:

<u>Algorithm 3.3</u>: Step 1. Determine D. The set D is determined a priori, as restrictively as possible. Note that D is a set of <u>initial</u> states. The trajectory may well pass out of D.

Step 2. Find L. This can be done e.g. by solving an LQ problem, with the state penalty reflecting the design objectives, see Anderson (1971) and Wieslander (1980). The control penalty is increased until the control $L^T x$ satisfies the constraint (2.3) for x in D. This check is easily done since the constraint is linear:

$$g_i \le \max_{x \in D} \ell_i^T x \le h_i, \quad i = 1,2,\ldots,m. \tag{3.17}$$

If D is such that (3.17) cannot be satisfied for any L, then this design is not appropriate.

Obviously, other criteria for linear design (e.g. pole placement) can be used to generate stabilizing matrices L that satisfy (3.14) or (3.15).

By (3.17) we have ensured that $E \supseteq D$. In step 3 we will try to find P and c such that $D \subseteq \Omega(L,P,c) \subseteq E(L)$. When a "lower gain" L-matrix is needed in order to increase the size of E and F relative D so that the construction of Ω in point 3 may go through, a good thing to do is to increase the control penalty in the above LQ-problem and get another L.

Step 3. Find P and c. First a $P = P^T > 0$ is sought such that $PA_c + A_c^T P > 0$. If the LQ-technique was used in point 2, the stationary solution of the Riccati equation may serve as P. Another method is to solve the Lyapunov equation (2.8) with a choice of Q giving a suitable decrease rate of the Lyapunov function V.

Now determine Ω by choosing the parameter c in the definition (2.12) such that $D \subseteq \Omega \subseteq E$:

$$\sup_{x \in D} x^T P x \leq c \leq \min_{x \in \partial E} x^T P x \tag{3.18}$$

where ∂E designates the boundary of E. Using constrained optimization theory (see e.g. Brockett, 1970) it can be shown that

$$\min_{x \in \partial E} x^T P x = \min_i \left\{ \frac{g_i^2}{\ell_i^T P^{-1} \ell_i}, \frac{h_i^2}{\ell_i^T P^{-1} \ell_i} \quad \text{for } i=1,2,\ldots,m \right\} \tag{3.19}$$

Both conditions are then easily checked numerically.

If this fails, choose another P, and if this does not help, select a "lower gain" L in order to enlarge the set E. Finally a cut down of the size of D might be considered.

Step 4. Set up the control u according to eq. (3.2).

□

Algorithm 3.3 is readily carried out when you have computer programs for the LQ-problem, for solving the Lyapunov equation, and for simulation of nonlinear dynamic systems. Preferably the programs should be interactive. For the examples in this paper the interactive programs described in Wieslander (1980) (LQ-design) and Elmqvist (1975) (simulation) have been used.

4. THE DESIGN IN THE DISCRETE TIME CASE

The design in the discrete time case can unfortunately not be made as general as in the continuous time case. Theorem 4.1 will give a stability result in the single input case; a generalization to the multi-input case is given in

46

Proposition 4.3. First, however, come the definitions and preliminaries corresponding to those of Section 2.

The plant is

$$\begin{cases} x(t+1) = \Phi x(t) + \Gamma u(t) \\ x(0) = x_0 \end{cases} \tag{4.1}$$

where $x \in R^n$, $n \in R^m$, $m \leq n$. The matrices Φ, Γ are of appropriate dimensions, and Γ is of full rank. Assume that

$$[\Phi, \Gamma] \text{ is a stabilizable pair} \tag{4.2}$$

$$g_i \leq u_i \leq h_i, \quad i = 1, 2, \ldots, m \tag{4.3}$$

Define

$$L \hat{=} [\ell_1 | \ell_2 | \ldots | \ell_m] \tag{4.4}$$

$$\Phi_c \hat{=} \Phi_c(L) \hat{=} (\Phi + \Gamma L^T) \tag{4.5}$$

where Φ_c is restricted to be a discrete time stability matrix.

Let D be the set of initial conditions of interest as in Def. 2.1, and introduce

$$\begin{cases} E \hat{=} E(L) \hat{=} \left\{ z \in R^n \middle| \; g_i \leq \ell_i^T z \leq h_i, \; i = 1, 2, \ldots, m \right\} & (4.6) \\ F \hat{=} F(L) \hat{=} \bigcap_{t=0}^{\infty} \left\{ \left[\Phi_c^t \right]^{-1} E \right\} & (4.7) \end{cases}$$

Let

$$\Phi_c^T P \Phi_c - P = -Q \tag{4.8}$$

yield a nxn matrix

$$P(L,Q) \hat{=} P = P^T > 0 \tag{4.9}$$

for every $Q = Q^T > 0$. Then

$$V(t) = x^T(t)\, Px(t) \tag{4.10}$$

is a Lyapunov function for

$$x(t+1) = \Phi_c\, x(t). \tag{4.11}$$

Using the notation

$$\Omega \hat{=} \Omega(L,P,c) \hat{=} \{x \in R^n \mid x^T Px \leq c\} \tag{4.12}$$

it follows that

$$x(0) \in \Omega \Rightarrow x(t) \in \Omega, \quad t = 0,1,2,\ldots \tag{4.13}$$

The function sat is defined in Def. 2.2.

The following single input result is analogous to Theorem 3.1.

<u>Theorem 4.1</u>: Let the plant (4.1)-(4.3) be single input, i.e. m=1. Assume that the initial condition set D is such that there exists an nxm matrix L, an nxn matrix $P = P^T > 0$ and a constant c, such that Φ_c defined by (4.5) is a discrete time stability matrix, that

$$\Phi_c^T P \Phi_c - P < 0 \tag{4.14}$$

and that the sets E and Ω, defined in (4.6) and (4.12), satisfy

$$D \subseteq \Omega \subseteq E. \tag{4.15}$$

Then the control

$$u = \text{sat}\ [L^T x + kw] \tag{4.16}$$

where

$$w = -(\Gamma^T P \Gamma)^{-1}\ \Gamma^T P \Phi_c x \tag{4.17}$$

stabilizes the plant for $x_0 \in D$, provided the design parameter k lies in the interval

$$0 \leq k \leq 2. \tag{4.18}$$

<u>Proof</u>: First consider the control

$$u = L^T x + v \tag{4.19}$$

with $v = 0$. Provided $x_0 \in D \subseteq \Omega$ it follows from (4.12) - (4.15) that $x(t) \in \Omega \subseteq E$ and the constraint (4.3) holds for all t.

The remainder of the proof will establish stability for the control (4.16), a more generous choice of v. Introduce (4.19) into (4.1):

$$x(t+1) = \Phi_c \, x(t) + \Gamma \, v(t). \tag{4.20}$$

The function

$$V(t) = x^T(t) \, Px(t) \tag{4.21}$$

is a Lyapunov function for $v = 0$. Now compute the increment in V(t) for the system (4.20):

$$V(t+1) - V(t) = x^T(\Phi_c^{\ T} P\Phi_c - P) \, x + f(v) \tag{4.22}$$

where

$$f(v) = 2x^T \, \Phi_c^T \, P\Gamma \, v + v^T \, \Gamma^T \, P\Gamma \, v. \tag{4.23}$$

For control laws (4.19), such that

$$f(v) \leq 0 \tag{4.24}$$

it follows from (4.14) that V is a Lyapunov function also for (4.20). Introduce w from (4.17) into (4.23) giving

$$f(v) = (v-w)^T \Gamma^T P\Gamma(v-w) - w^T \Gamma^T P\Gamma w. \tag{4.25}$$

Since m = 1 the choice

$$v = r w \tag{4.26}$$

satisfies (4.24) for

$$(1-r)^2 - r^2 \leq 0 \tag{4.27}$$

or equivalently

$$0 \leq r \leq 2. \tag{4.28}$$

Just as in Theorem 3.1 it can be shown that the control (4.16) fulfills

$$u = \text{sat}\ [L^T x + kw] = L^T x + rw \tag{4.29}$$

for $x \in E$ and that

$$0 \leq r \leq k. \tag{4.30}$$

This proves the stability under the control (4.16) - (4.18).

□

In the multi input case, $m > 1$, the saturated linear state feedback

$$\begin{cases} u = \text{sat}\,[L^T x + Kw] \\ w = -(\Gamma^T P \Gamma)^{-1} \Gamma^T P \Phi_c x \\ K = \text{diag}\,(k_1, \ldots, k_m) \\ 0 \leq k_i \leq 2, \quad i = 1, 2, \ldots, m \end{cases} \tag{4.31}$$

would not stabilize the plant in general.

First we note that eqs. (4.16) - (4.25) of the Theorem 4.1 carry through to the multi-input case. Equations (4.19), (4.26) would correspond to

$$\begin{cases} u = L^T x + Rw \\ R = \text{diag}\,(r_1, \ldots, r_m) \end{cases} \tag{4.32}$$

We will then show that (4.24) requires that

$$\begin{cases} R = rI \\ 0 \leq r \leq 2 \end{cases} \tag{4.33}$$

Insert (4.32) into the expression (4.25) for $f(v)$

$$f(Rw) = w^T \{(R-I)\Gamma^T P \Gamma (R-I) - \Gamma^T P \Gamma\}\, w \tag{4.34}$$

and (4.33) follows directly from the following lemma.

<u>Lemma 4.2</u>: Let M be a positive definite symmetric mxm matrix. Let Λ be a diagonal mxm matrix. Then

$$\Lambda M\Lambda - M \leq 0 \quad \text{for all } M \tag{4.35}$$

iff

$$\Lambda = \text{diag}\ (\lambda, \ldots, \lambda), \quad -1 \leq \lambda \leq 1 \tag{4.36}$$

<u>Proof</u>: Sufficiency: Let Λ be given by (4.36). Then

$$\Lambda M\Lambda - M = (\lambda^2-1)\ M \leq 0. \tag{4.37}$$

since $M > 0$ and $(\lambda^2-1) \leq 0$.

Necessity: Let

$$\Lambda = \text{diag}\ (\lambda_1, \lambda_2, \ldots, \lambda_m). \tag{4.38}$$

If $|\lambda_j| > 1$ for some j, it is clear that the choice M=I will make the j:th diagonal element of $\Lambda M\Lambda$-M positive and hence $\Lambda M\Lambda$-M is non-negative semidefinite. Therefore, in the remainder of the proof assume that $|\lambda_j| \leq 1$ for all j. Then it will be shown that (4.35) may be violated if $\lambda_i \neq \lambda_1$ for any i, say i=2. Define

$$U = \begin{bmatrix} 1-\lambda_1^2 & \nu(1-\lambda_1\lambda_2) \\ \nu(1-\lambda_1\lambda_2) & (1-\lambda_2^2) \end{bmatrix}.$$

The choice

$$M = \begin{bmatrix} 1 & \nu & & 0 \\ \nu & 1 & & \\ & & \ddots & \\ 0 & & & 1 \end{bmatrix}, \quad |\nu| < 1 \qquad (4.39)$$

gives that

$$\Lambda M \Lambda - M = \text{diag}\left[U, (1-\lambda_3^2), \ldots, (1-\lambda_m^2)\right]. \qquad (4.40)$$

From

$$\det U = (1-\nu^2)(1-\lambda_1\lambda_2)^2 - (\lambda_1-\lambda_2)^2 \qquad (4.41)$$

it follows that (4.35) is violated for ν^2 close to 1. This completes the necessity part.

□

Hence it is clear that in order to continue the proof in the multivariable case we have to establish an equality between (4.31) and (4.32),(4.33), i.e. between

$$u = \text{sat}\,[L^T x + Kw] \qquad (4.42)$$

and

$$u = L^T x + rw \qquad (4.43)$$

for some $r = r(x) \in [0,2]$, but this can not be done in general. A slightly modified design is therefore suggested.

Proposition 4.3: Let

$$k_i \in [0,2], \quad i = 1,2,\ldots m \tag{4.44}$$

be design parameters. Then the control

$$\begin{cases} u = L^T x + r(x)\ w(x) \\ w(x) = (\Gamma^T P \Gamma)^{-1}\ \Gamma^T\ P\Phi_c\ x \end{cases} \tag{4.45}$$

where

$$\begin{cases} r(x) = \min_i \left\{ s_i \right\}_{i=1}^{m} \\ s_i = \max_z \left[\{z \in [0,k_i]\} \cap \left\{ z \middle| g_i \le \ell_i^T x + z w_i(x) \le h_i \right\} \right] \end{cases} \tag{4.46}$$

will stabilize the plant for $x_0 \in D$.

□

Remark 4.4: Considering (4.25) it is clear that

$$\min_v f(v) = f(-\bar{v}) = -\bar{v}^T \Gamma^T P \Gamma \bar{v} \tag{4.47}$$

since $\Gamma^T P \Gamma > 0$. Thus $k = 1$ in (4.18) yields the fastest decrease in the Lyapunov function (4.21) among the control laws (4.16).

Remark 4.5: For k = 1, the single input control (4.16) can be rewritten, using (4.5):

$$u = \text{sat}\ [-\Gamma^T P\Phi x / \Gamma^T P \Gamma]. \tag{4.48}$$

This control looks like a saturated state feedback dead beat control, see Leden (1975), Åström (1970) and (1973). It is not in general. If it were a dead beat control, then

$$\Phi - \Gamma\cdot(\Gamma^{T}P\Phi/\Gamma^{T}P\Gamma) \qquad (4.49)$$

would be nilpotent of order n. See also Example 5.4.

<u>Algorithm 4.6</u>: Steps 1-3 of Algorithm 3.3 carry over with some obvious changes.

Step 4: Set up the control u according to eq. (4.16). Select a suitable number k, e.g. with the help of simulations of the closed loop system.

□

5. EXAMPLES

We will first consider the double integrator, a single input plant, which has been studied several times, see e.g. Athans (1966) and Kirk (1970). It is described by

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ -1 \le u \le 1 \end{cases} \qquad (5.1)$$

The continuous time controllers will be constructed with

(5.1) as the model. The discrete time controllers will be based on the corresponding discrete time model (which was obtained by sampling the continuous model (5.1) with the sampling period = 1):

$$\begin{cases} x(i+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(i) + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} u(i), \\ -1 \leq u \leq 1 \end{cases} \tag{5.2}$$

The set D is arbitrarily chosen as the following square in the state space:

$$D = \{ x \in R^2 \mid -10 \leq x_i \leq 10, \; i=1,2 \} \tag{5.3}$$

The <u>design objective</u> is to bring x_1 from its initial value to the origin as fast as possible, using a saturated linear control.

Before turning to the actual design according to algorithms 3.3 and 4.6, let us first discuss a few features of the double integrator that are of some interest in our context. It can be shown that in the continuous-time case, every control law of the form

$$u = \text{sat}\,(\ell^T x), \tag{5.4}$$

where the 2x1 matrix ℓ is such that $A + b\ell^T$ is a stability matrix, will stabilize (5.1). This is indeed not so in the discrete time case as demonstrated in example 5.1.

Example 5.1: Given model (5.2). Compute the saturated state feedback dead-beat controller (cf. Aström, 1973),

$$u(i) = \mathrm{sat}\,([-1 \quad -1.5]\, x(i)), \qquad i = 0,1,2,\ldots \tag{5.5}$$

Let the initial condition be

$$x(0) = (10 \;\; 10)^T \in D. \tag{5.6}$$

The phase plane trajectory and the control signal of the closed loop system are shown in Fig. 5.1a. Clearly it is unstable.

We also compute the sets E and F, (4.7,4.8), for the dead-beat controller.

$$E = \{x \mid -1 \leq -x_1 - 1.5\, x_2 \leq 1\}. \tag{5.7}$$

To determine F we have to compute the inverse mappings $(\Phi_c^t)^{-1}$ of

$$\Phi_c = \begin{bmatrix} 0.5 & 0.25 \\ -1 & -0.5 \end{bmatrix} \tag{5.8}$$

We find that

$$F = E \cap \{x \mid -1 \leq x_1 + 0.5\, x_2 \leq 1\}. \tag{5.9}$$

See Fig. 5.1b.

□

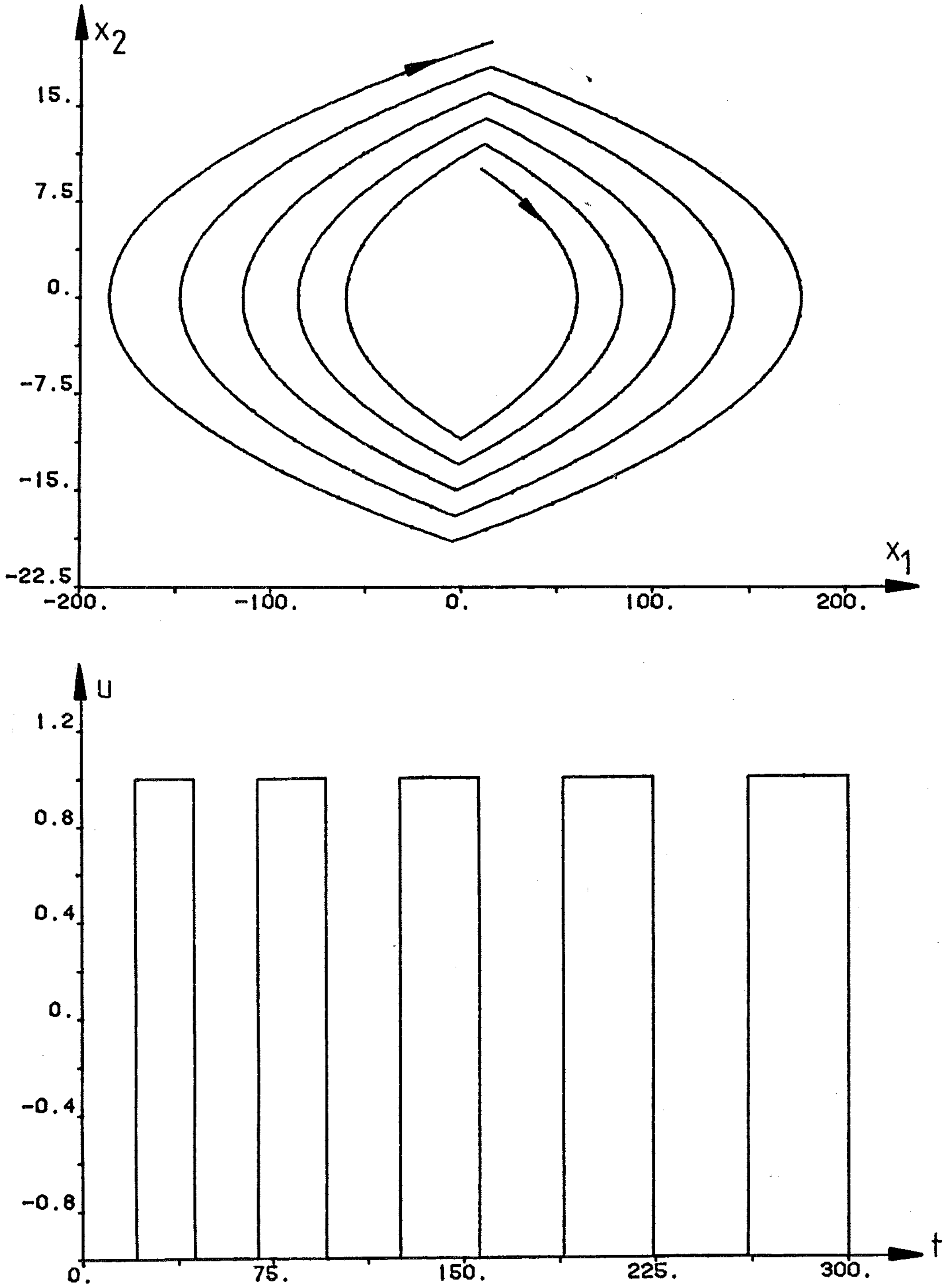

Fig. 5.1a The phase plane trajectory and the control signal of the system (5.1),(5.5). $x(0) = (10\ 10)^T$.

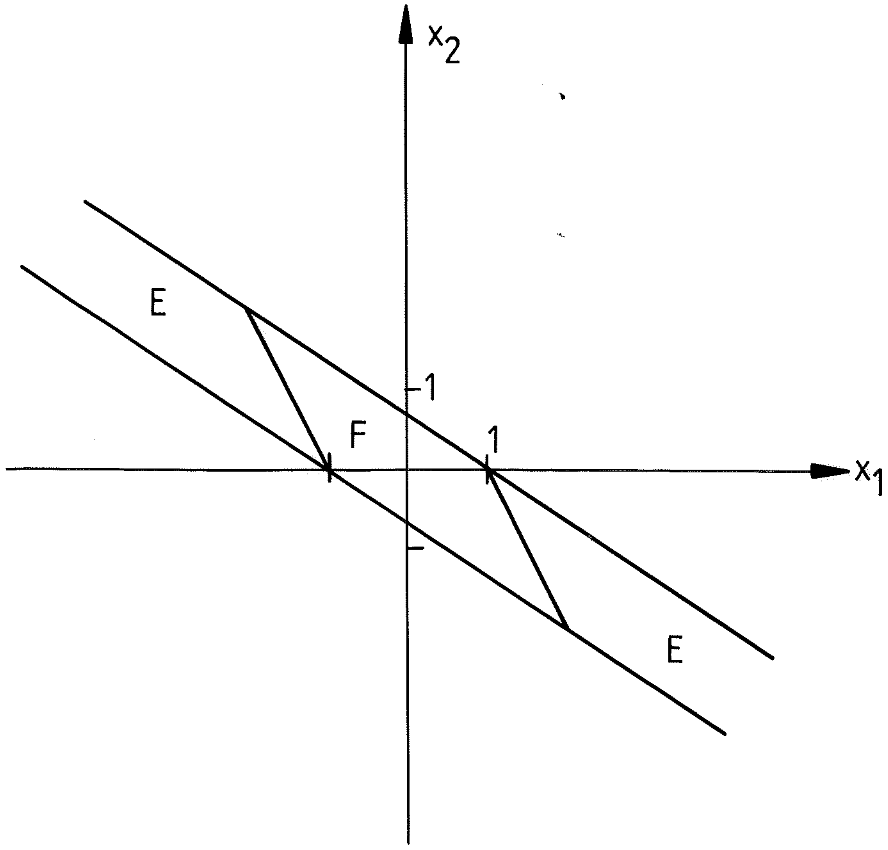

Fig. 5.1b The sets E and F for the model (5.2) with the controller (5.5).

Another interesting feature of the double integrator is that the time-optimal design with constrained input is known. Therefore we can compare our designs with the best possible.

<u>Example 5.2</u>: The time-optimal control of plant (5.1) can be found in e.g. Kirk (1970). The control law is

$$\begin{cases} -1 & \text{if } s > 0 \text{ or } (s = 0 \text{ and } x_2 > 0) \\ 0 & \text{if } x = 0 \\ +1 & \text{if } s < 0 \text{ or } (s = 0 \text{ and } x_2 < 0) \end{cases} \qquad (5.10)$$

where

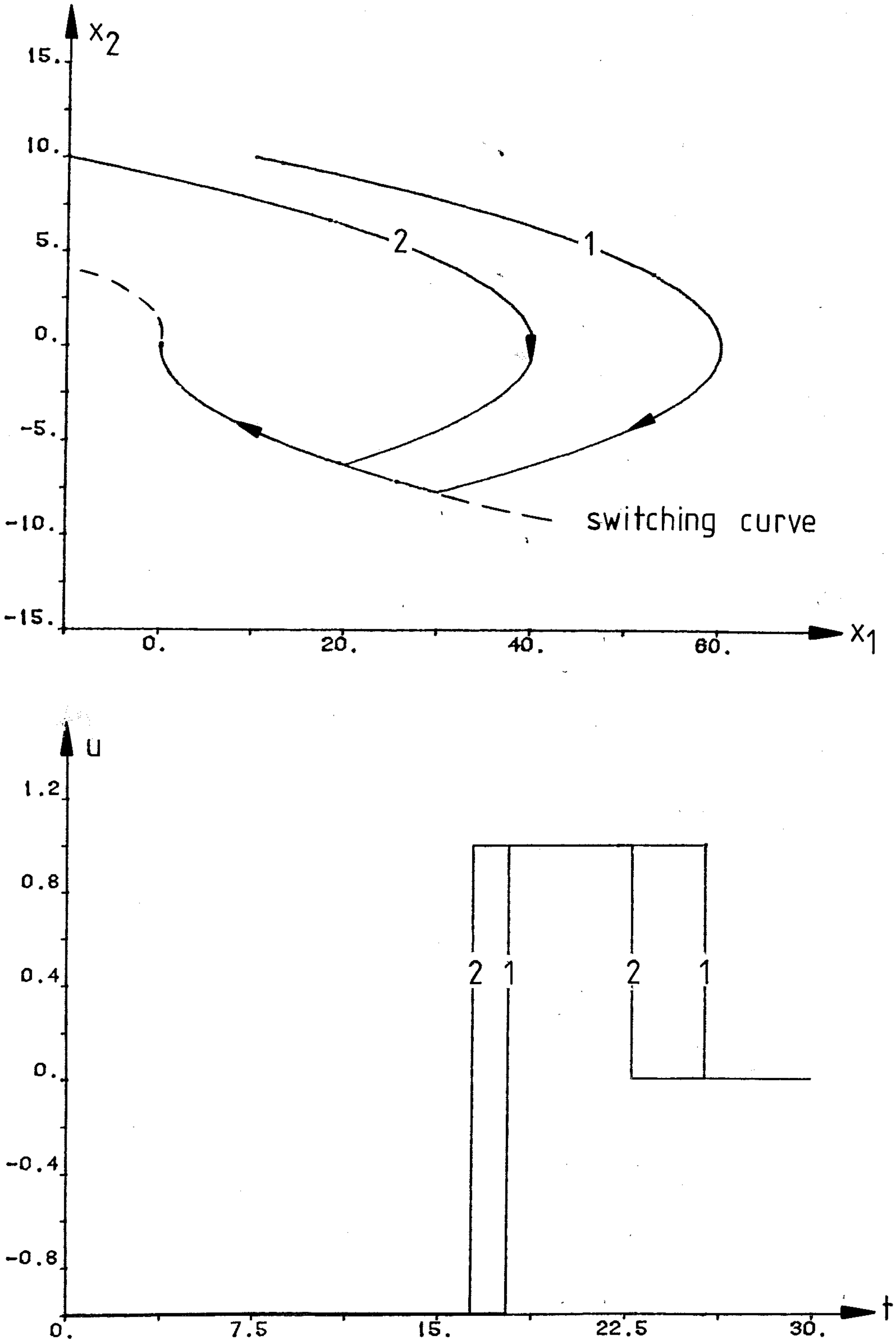

Fig. 5.2 Time-optimal control of the plant (5.1). The control is given by (5.7). Initial conditions: 1. $x(0) = (10\ 10)^T$; 2. $x(0) = (-10\ 10)^T$; a. Phase plane trajectories; b. Control inputs $u(t)$.

$$s \hat{=} x_1 + 0.5\, x_2\, |x_2|.$$

The phase plane trajectory of the closed loop system with the initial condition (5.6), i.e. $x(0) = (10 \quad 10)^T$ is displayed in Fig. 5.2.

It takes approximately 25.5 seconds to reach the origin from $(10 \quad 10)^T$.

□

Let us now turn to the continuous time design presented in Section 3.

<u>Example 5.3</u>: Given the plant (5.1) and the set D (5.3). To find L_1 we use the LQ-design, where the loss is given by

$$\int_0^\infty (x^T Q_1 x + u^T Q_2 u)\, dt. \tag{5.11}$$

The choice

$$Q_1 = \text{diag}\, [10,\ 1] \tag{5.12}$$

reflects the importance we attach to bringing x_1 to the origin. We had to increase Q_2 up to

$$Q_2 = 500\ 000 \tag{5.13}$$

until we got a control law that produced unsaturated control for all t, (3.15),

$$u = - [4.47 \quad 94.6] \cdot 10^{-3} x \tag{5.14}$$

A few control histories for trajectories emanating from the vertices of D, when using the control (5.14) are displayed in Fig. 5.3a. This linear control law will be used as a reference later on. To find a matrix P, (2.8) was solved with

$$\begin{cases} A_c = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix} [4.47 \quad 94.6] \cdot 10^{-3} \\ \\ Q = \text{diag}\ (1,\ 1) \end{cases} \tag{5.15}$$

which gave

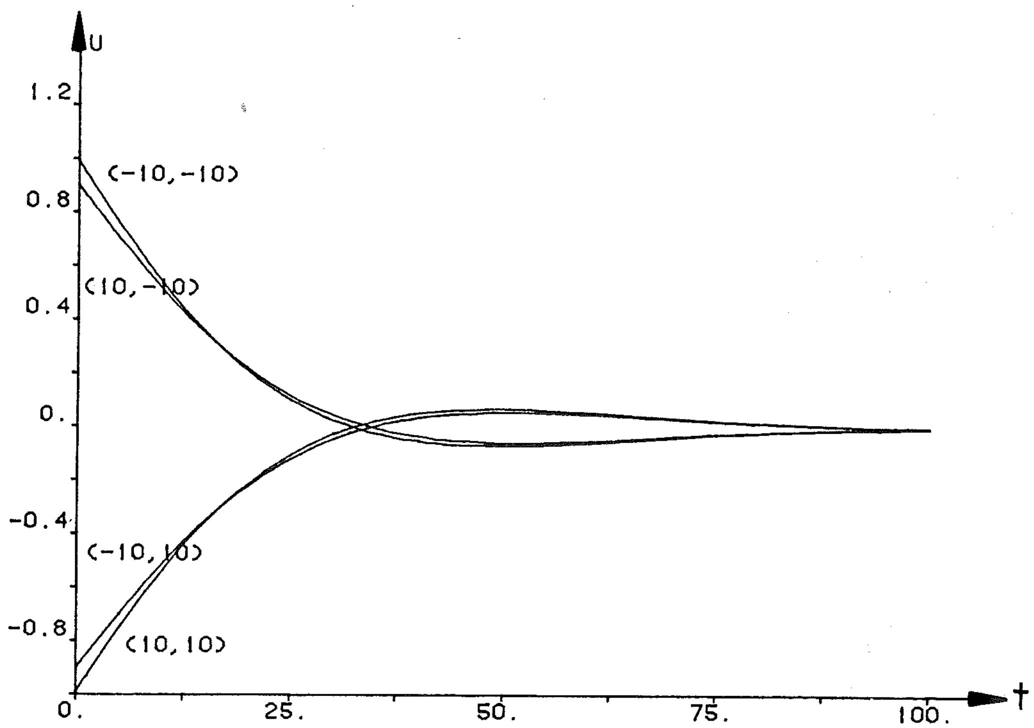

Fig 5.3a The input for the system (5.1) using the linear control law (5.14) for different initial conditions.

$$P = \begin{bmatrix} 15.9 & 111.8 \\ 111.8 & 1187.3 \end{bmatrix}. \qquad (5.16)$$

Then it was found that

$$\sup_{x \in D} x^T P x \leq \min_{x \in \partial E} x^T P x$$

could not be fulfilled. We tried in vain to replace Q in (5.15) in order to get a better P.

We turned back to point 2 to find a lower gain L, so that the Ω defined by

$$c = \sup_{x \in D} x^T P x$$

would lie inside the new E. One way to find a new L was to multiply the old L by a factor less than one, so that the old (discarded) Ω would have lain inside the new E.

We settled for

$$L = -\,0.78 \cdot [4.747 \quad 94.6] \cdot 10^{-3} \qquad (5.17)$$

which gives a linear control satisfying (3.15) and yields an asymptotically stable A_c,

$$A_c = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \end{bmatrix} [4.47 \quad 94.6] \cdot 0.78 \cdot 10^{-3} \qquad (5.18)$$

A new P was generated according to equation (2.8).

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow P = \begin{bmatrix} 173 & 1433 \\ 1433 & 19435 \end{bmatrix}, \qquad (5.19)$$

and the new Ω computed according to (2.12) with

$$c = \sup x^T P x.$$

Now equation (2.15) was satisfied, see Fig. 5.3b. It is interesting to note that the set F is much larger than Ω.

We set up the control

$$u = \text{sat}\ [(L - k[0\ 1]P)x], \text{ where} \tag{5.20}$$

L is given in (5.17), and P in (5.19).

The system was simulated for various k-values, and for various initial conditions. Simulations for the initial condition $(10\ 10)^T$ are shown in Fig. 5.3c.

The k-value ensuring the fastest settling time is

$$k = 0.5 \cdot 10^{-5} \tag{5.21}$$

A simulation of the plant (5.1) with the controller (5.20), (5.21) can be found in Figures 5.3d and 5.3e.

A comparison with the linear controller is also shown. It is clear from these figures and from Fig. 5.2 that while the time optimal control is twice as fast as the control (5.20), (5.21), this is considerably faster than the linear control.

□

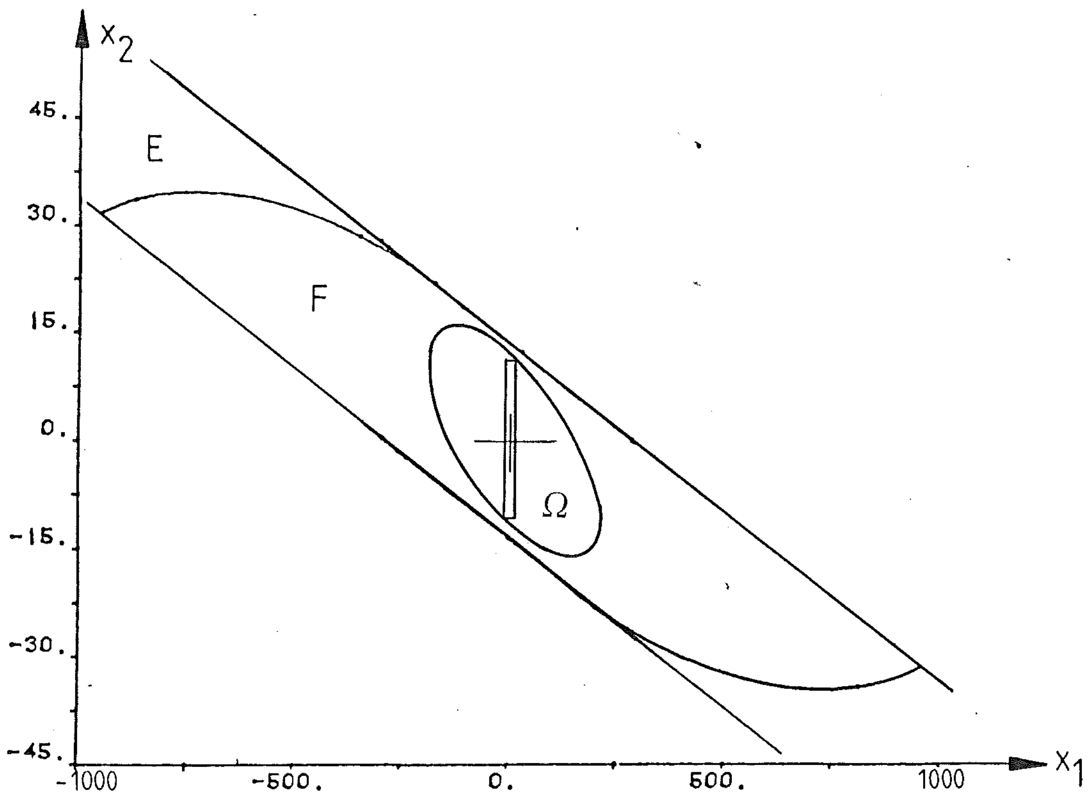

Fig. 5.3b D given in (5.3), Ω in (2.12) and (5.19), F in (2.7) and (5.18), $E = \{x \mid -1 \leq L^T x \leq 1\}$ with L given in (5.14).

We now consider the discrete-time case:

<u>Example 5.4</u>: Given the plant (5.1). We use the model (5.2) to design a discrete time control low according to algorithm 4.6. The set D is given in (5.3). We find L by using the LQ-design, where the loss is given by

$$\sum_{i=0}^{\infty} (x^T Q_1 x + u^T Q_2 u) \qquad (5.22)$$

The choice

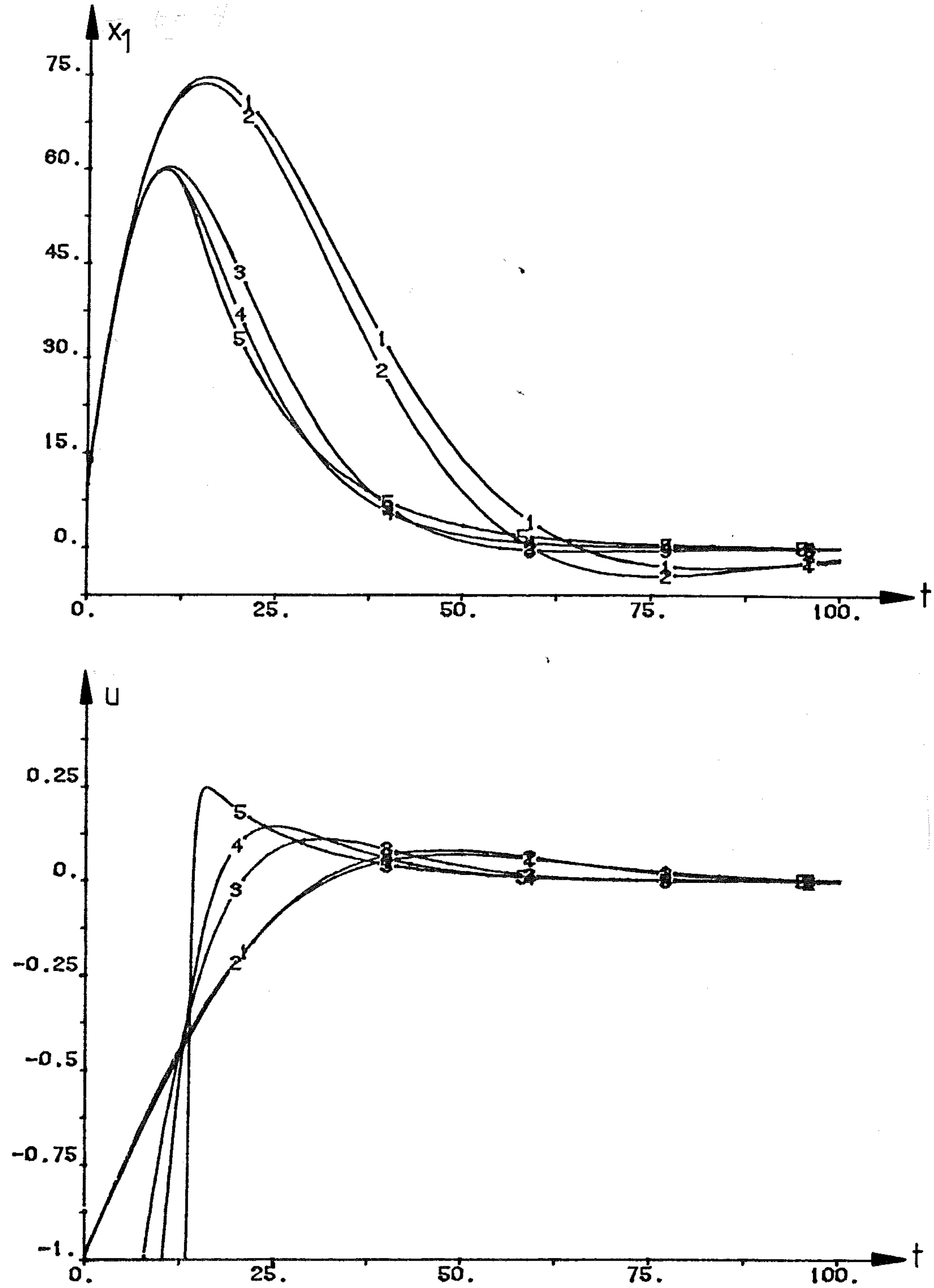

Fig. 5.3c $x_1(t)$ and u(t) for the system (5.1), (5.14), denoted by 1 in the graph, and for the system (5.1), (5.20) with various values for the parameter k. The initial condition: x(0) = $(10 \quad 10)^T$. 1: (5.1) governed by the linear controller (5.14), 2: (5.1), (5.20) with $k = 10^{-6}$, 3: (5.1), (5.20) with $k = 0.5 \cdot 10^{-5}$, 4: (5.1), (5.20) with $k = 10^{-5}$, 5: (5.1), (5.20) with $k = 10^{-4}$.

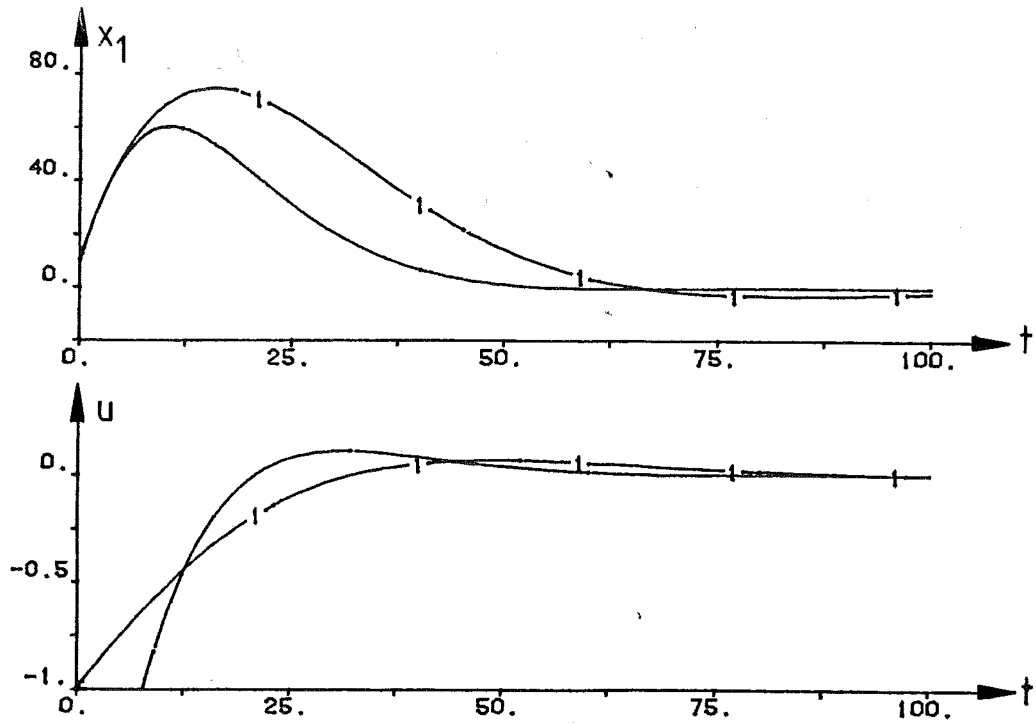

Fig. 5.3d $x_1(t)$ and $u(t)$ for the linear system (5.1), (5.14) (denoted by 1 in the figure), and the system (5.1), (5.20), (5.21), i.e. with $k = 0.5 \cdot 10^{-5}$. Initial condition: $x(0) = (10 \quad 10)^T$.

$$Q_1 = \text{diag}\ (10,\ 1) \tag{5.23}$$

reflects one design objective. We had to increase Q_2 up to

$$Q_2 = 500\ 000 \tag{5.24}$$

until we got a control low that stayed inside the control bounds along all trajectories emanating from D, i.e. made $D \subseteq F$.

$$u(t) = L^T x(i) = -[0.0043 \quad 0.0924]x(i),\ i \leq t < i+1 \tag{5.25}$$

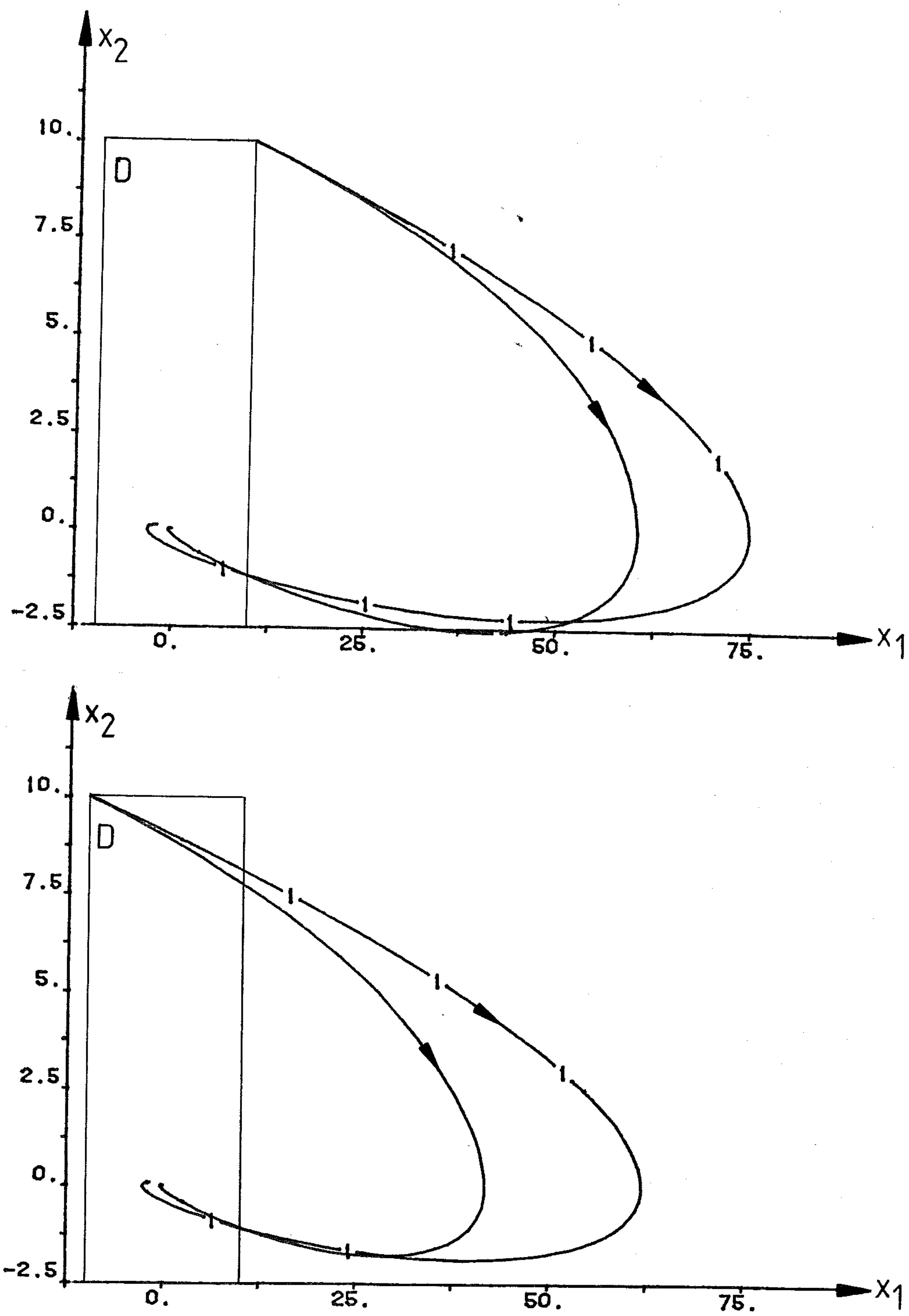

Fig. 5.3e Phase plane trajectories for the linear system (5.1), (5.14), (denoted by 1 in the figures), and the system (5.1), (5.20), (5.21), i.e. with $k = 0.5 \cdot 10^{-5}$, for the initial conditions $(10\ \ 10)^T$ and $(-10\ \ 10)^T$. The set D is also displayed.

The control histories were checked by simulation, see Fig. 5.4b, c.

The corresponding solution P of the stationary Ricatti equation

$$P = (\Phi+\Gamma L^T)^T P(\Phi+\Gamma L^T) + LQ_2L^T + Q_1 \tag{5.26}$$

was

$$P = \begin{bmatrix} 217 & 2238 \\ 2238 & 47340 \end{bmatrix} \tag{5.27}$$

Since

$$Q = LQ_2L^T + Q_1 = \begin{bmatrix} 19 & 199 \\ 199 & 4270 \end{bmatrix} \tag{5.28}$$

is positive definite (cf. equation (4.8)), $V = x^T Px$ is a Lyapunov function for (5.2), (5.25). Now Ω is computed according to (4.12):

$$\Omega = \{x \mid x^T Px \le x_0^T Px_0\} \tag{5.29}$$

with $x_0 = (10 \quad 10)^T$. From Fig. 5.4a it is evident that $D \subseteq \Omega \subseteq E$ with E defined in (4.6) with the $L^T x$ of (5.25). Also, in Fig. 5.4a, the set F (4.7) is displayed.

Finally we set up the control (cf. Theorem 4.1)

$$u(t)=\text{sat}[(L^T-k\cdot\Gamma^T P\Phi_c/\Gamma^T P\Gamma)x(i)], \quad i\le t<i+1 \tag{5.30}$$

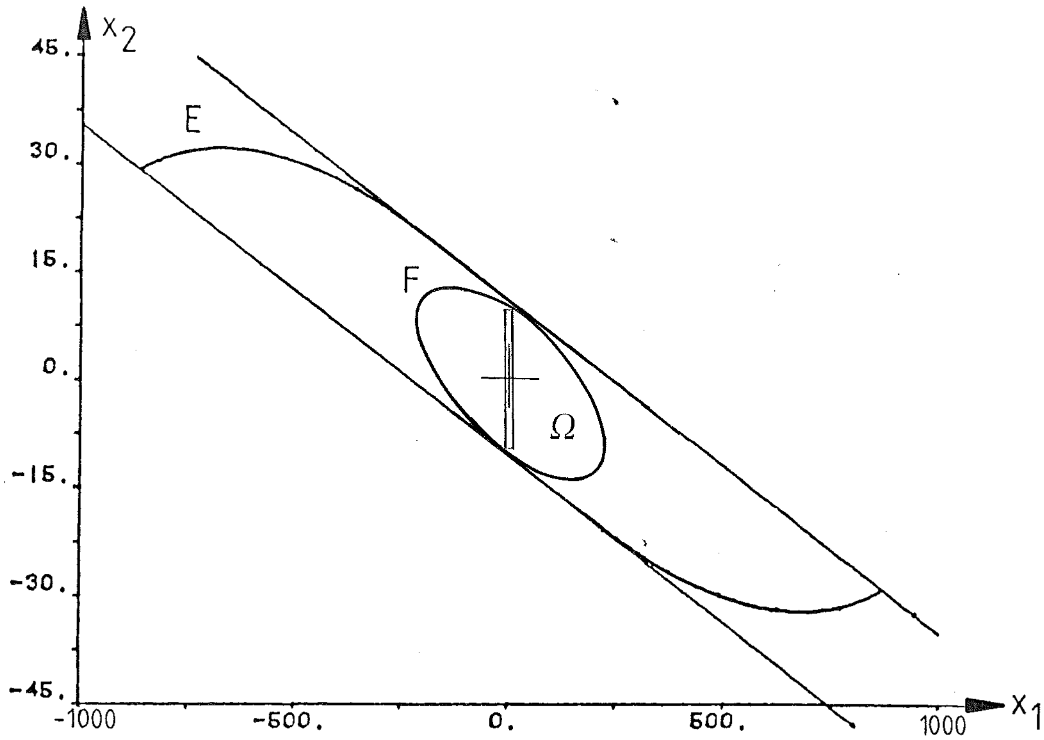

Fig. 5.4a D, given in (5.3); Ω, in (5.29); F, in (4.7), (4.5), (5.2) and (5.25); $E = \{x \mid -1 \le L^T x \le 1\}$ with L given in (5.25).

with

$L^T = -[0.0043 \quad 0.0924]$

$k \in [0,2]$, a design parameter

$\Gamma^T = [0.5 \quad 1]$

P, see (5.27)

$$\Phi_c = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} + \Gamma L^T$$

The system (5.1), (5.30) was simulated for various values of k. See Fig. 5.4b. We chose somewhat arbitrarily

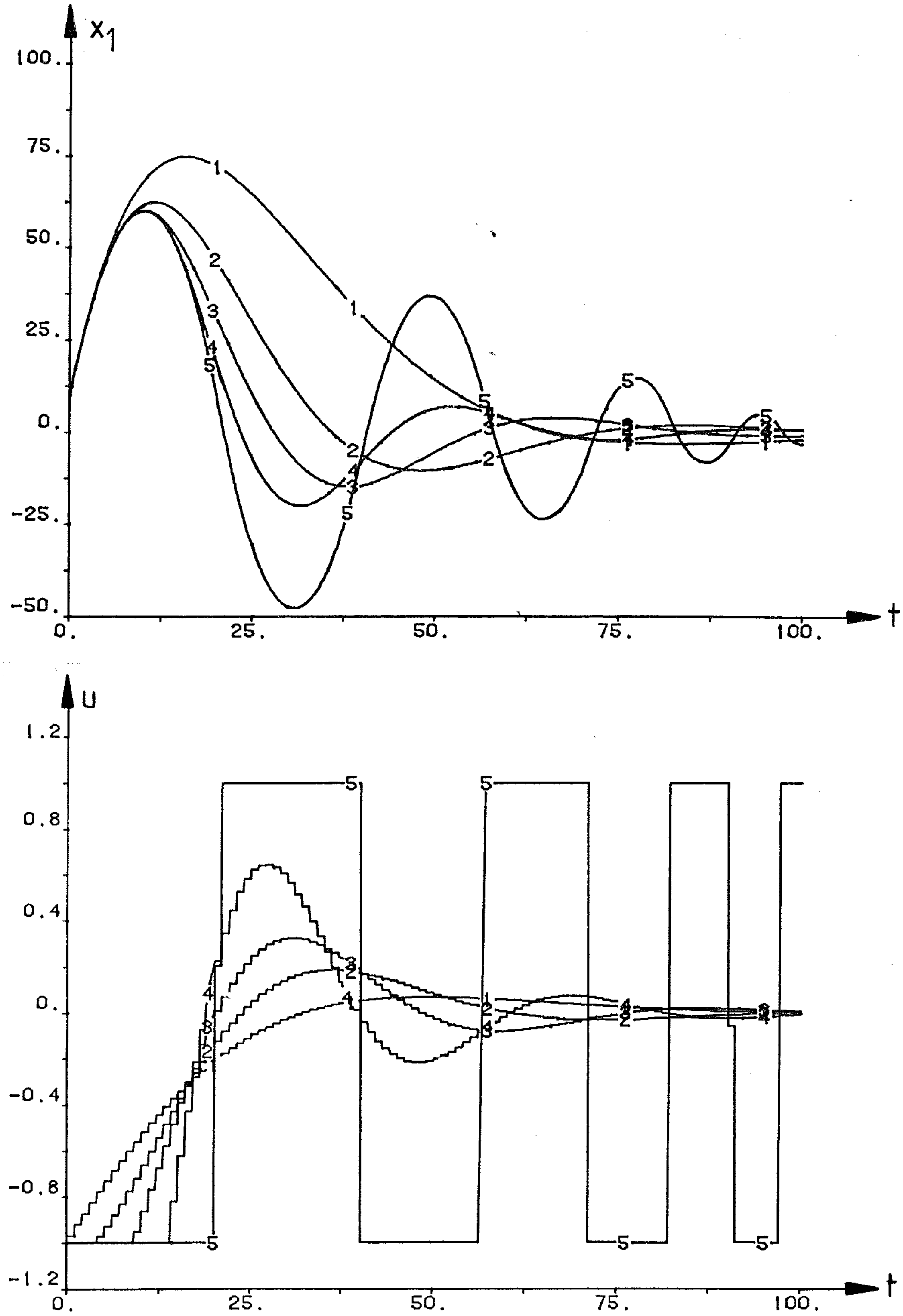

Fig. 5.4b $x_1(t)$ and $u(t)$ for the system (5.1), (5.30) using various k-values. Initial condition $x(0) = (10 \quad 10)^T$. 1: $k = 0$ (i.e. the linear system (5.1), (5.25)), 2: $k = 0.005$, 3: $k = 0.01$, 4: $k = 0.02$, 5: $k = 1$.

$$k = 0.01 \tag{5.31}$$

to get a short solution time without too much of an overshoot. The resulting system (5.1), (5.30), (5.31), is simulated in Figures 5.4c and d, where a comparison with the plant controlled by the linear controller (5.1), (5.25) is made. We note that the new design yields a somewhat faster system, although it is a far cry from the time optimal performance.

Referring to Remark 4.5, we compute

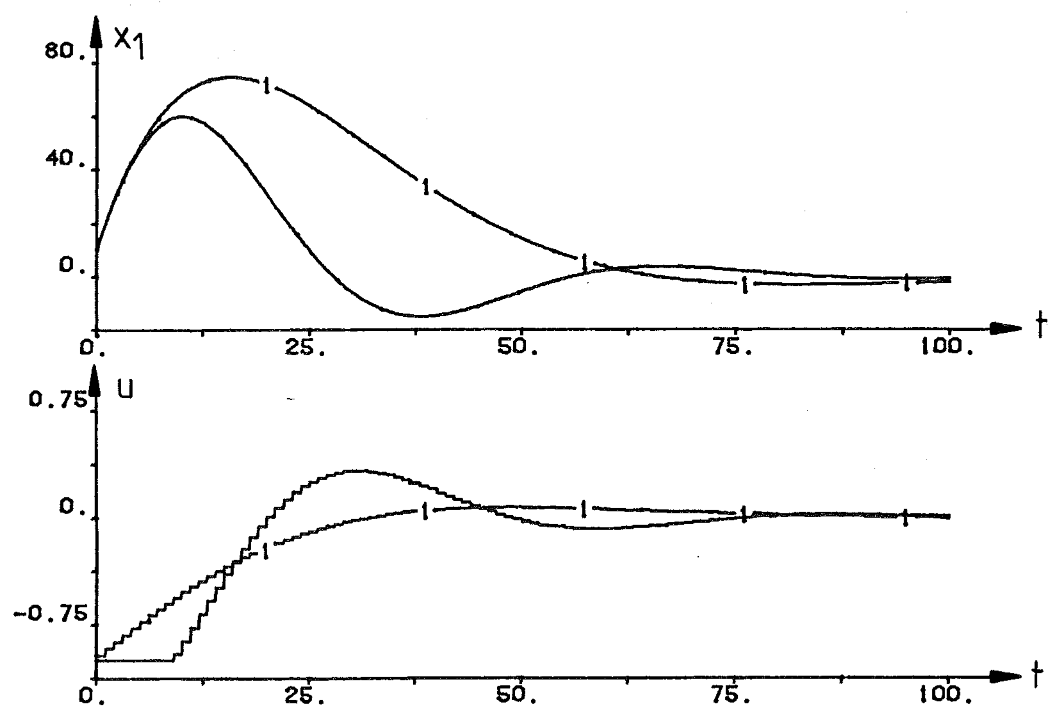

Fig. 5.4c x(t) and u(t) for the linear system (5.1), (5.25), denoted by 1, and the system (5.1), (5.30), (5.31) i.e. with k = 0.01. Initial condition: $x(0) = (10 \quad 10)^T$.

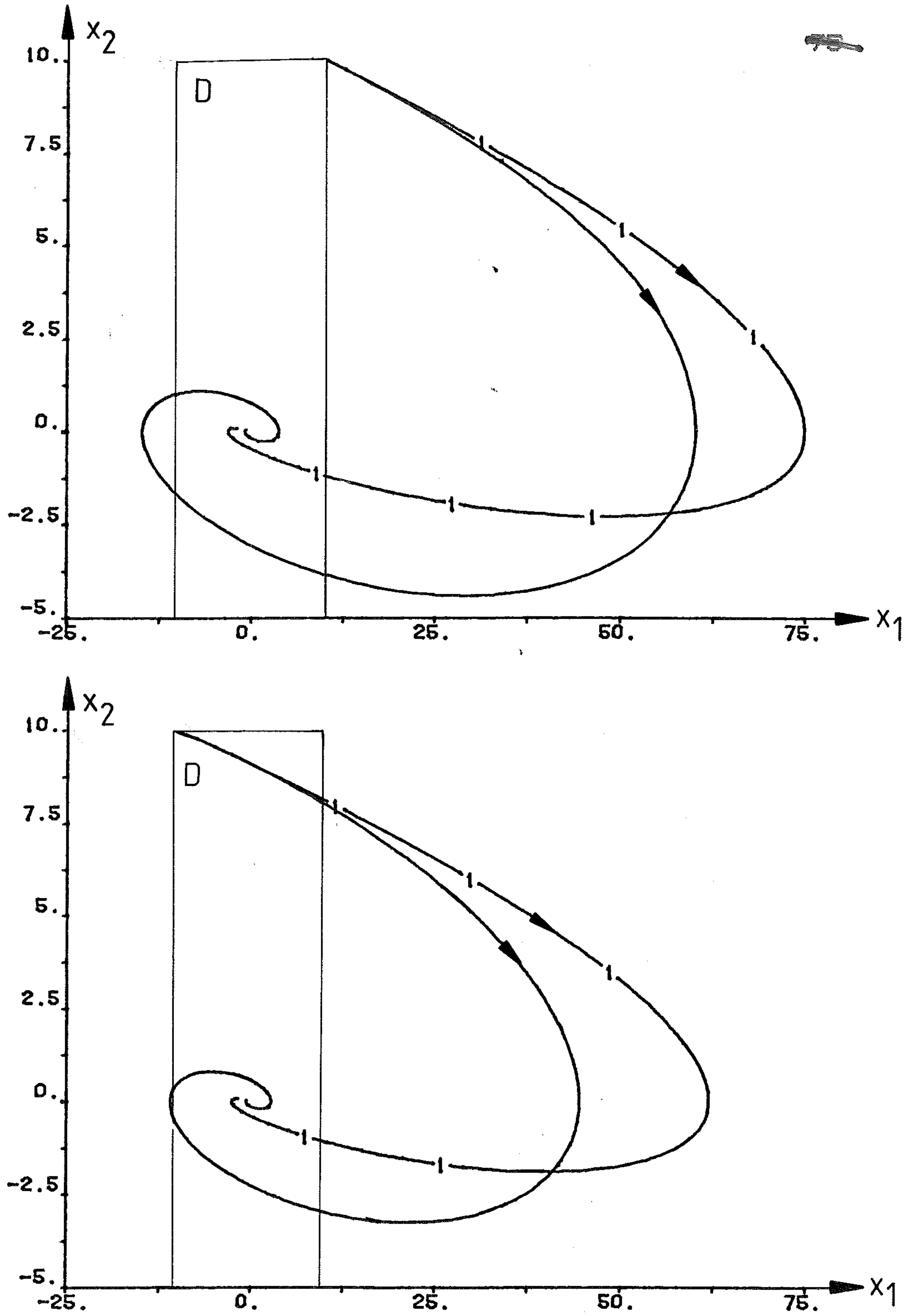

Fig. 5.4d Phase plane trajectories for the linear system (5.1), (5.25), denoted by 1, and the system (5.1), (5.30), (5.31), i.e. with k = 0.01. Initial conditions $(10 \quad 10)^T$ and $(-10 \quad 10)^T$. The set D is displayed.

$$\Phi_{cc} \hat{=} \Phi - \Gamma \cdot (\Gamma^T P \Phi / \Gamma^T P \Gamma) = \begin{bmatrix} 0.976 & 0.488 \\ -0.047 & 0.024 \end{bmatrix} \tag{5.32}$$

and find that

$$\Phi_{cc} \neq \text{diag } (0, 0) \tag{5.33}$$

and hence the control (5.30) with k = 1 is not a saturated dead beat control.

□

It is evident from Figures 5.3 and 5.4 that the improvement of our controller over the linear one is not as dramatic in the discrete time case as in the continous time case. We used different Q-matrices in the two cases.

Example 5.5: The following linearized model for the depth control of a submarine was obtained from Kockumation AB, Malmö, Sweden:

$$\begin{cases} \dot{x} = A\,x + b\,u \\ A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -0.005 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \\ 0.005 \end{bmatrix} \end{cases} \tag{5.34}$$

with the control constraint

$$|u| \leq 0.005, \tag{5.35}$$

and x_1 designating depth.

The set of initial conditions of interest was given as

$$D=\{x|\ |x_1|\le 10,\ |x_2|\le 0.05,\ |x_3|\le 0.005\} \tag{5.36}$$

We choose our initial ℓ somewhat arbitrarily:

$$\ell^T = -\ (8.65\cdot 10^{-6}\quad 6.27\cdot 10^{-3}\quad 0.82) \tag{5.37}$$

with

$$\max_{x\in D}\ \ell^T x = 0.0045. \tag{5.38}$$

The eigenvalues of the closed loop system then become:

$$\lambda(A_c) = \lambda(A + b\ell^T) = \{-0.0039,\ -0.0026\pm i\cdot 0.0021\} \tag{5.39}$$

Proceeding according to Algorithm 3.3 we find that

$$Q = \text{diag}\ (1,\ 10^5,\ 1.5\cdot 10^{10}) \tag{5.40}$$

yields a solution F such that

$$\begin{cases} \max_{x\in D} & x^T P x = 4.55\cdot 10^7 \\ \min_{x\in \partial E} & x^T P x = 4.81\cdot 10^7 \end{cases} \tag{5.41}$$

and

$$m^T \hat{=} b^T P = (5.78\cdot 10^4\quad 3.59\cdot 10^7\quad 8.06\cdot 10^9) \tag{5.42}$$

Hence a control

$$u = \text{sat}\ [\ell^T x - k\ m^T x] \tag{5.43}$$

stabilizes (5.34). We found

$$k = 2.1 \cdot 10^{-9} \tag{5.44}$$

suitable. This yields the control

$$\begin{cases} u = \text{sat}\ (\hat{\ell}^T x) \\ \hat{\ell}^T = -(1.28\cdot 10^{4} \quad 8.16\cdot 10^{-2} \quad 17.74) \end{cases} \tag{5.45}$$

The closed loop system poles become

$$\lambda(A + b\hat{\ell}^T) \approx \{-0.089, -0.0022 \pm i\cdot 0.00015\}$$

Simulations of controller (5.45) showed that it might be adequate. Actually an insignificantly different controller was tested on a real submarine:

$$\begin{aligned} u &= \text{sat}\ (\bar{\ell}^T x) \\ \bar{\ell}^T &= -\ (1.39\cdot 10^{-4} \quad 8.68\cdot 10^{-2} \quad 17.36) \end{aligned} \tag{5.46}$$

Simulations, comparing (5.45), (5.34) with (5.46), (5.34) show that the controllers hardly differ at all.

A simulation of controller (5.46) and the linear model (5.34) with the initial condition (0, 0, −0.004) is shown in Fig. 5.5a. The actual underwater test with the same initial condition is shown in Fig. 5.5b. The crew is reported to be satisfied with the controller.

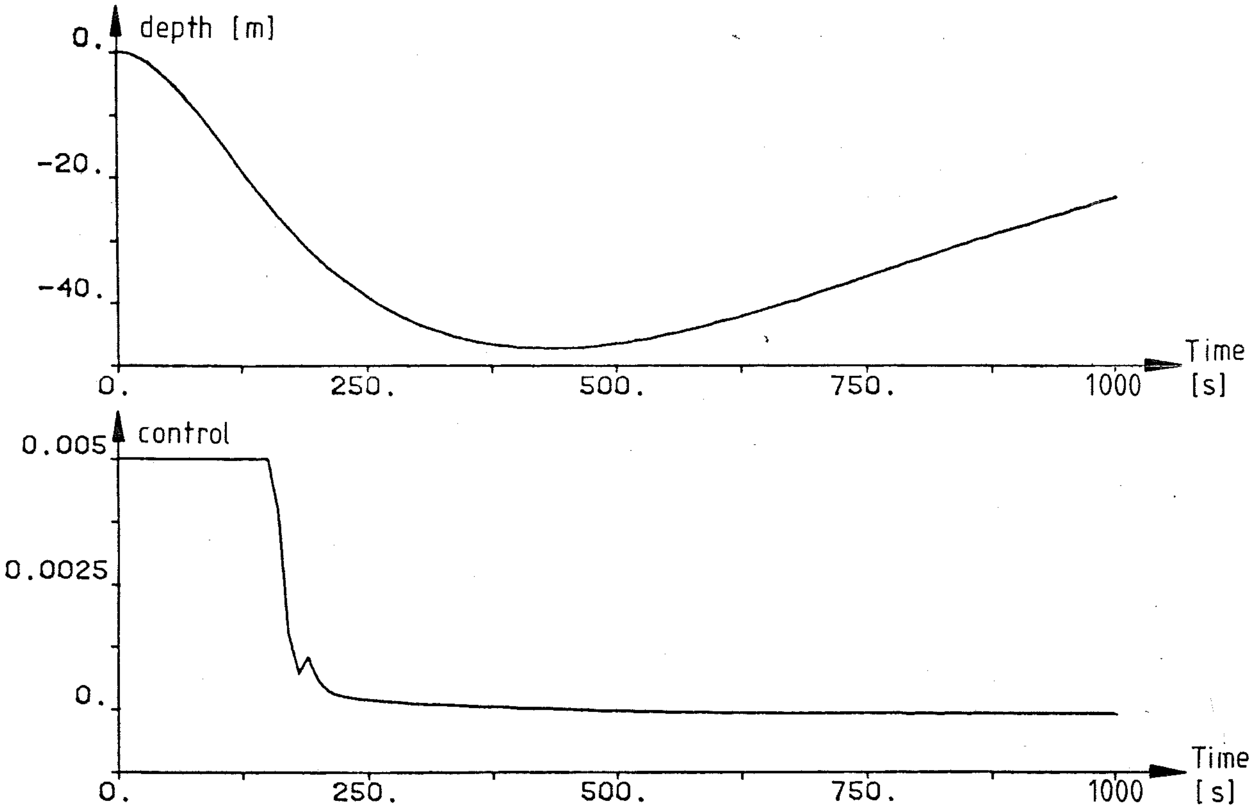

Fig. 5.5a Simulation of controller (5.46) with linear system (5.34). Initial condition $x(0) = (0 \quad 0 \quad -0.004)^T$.

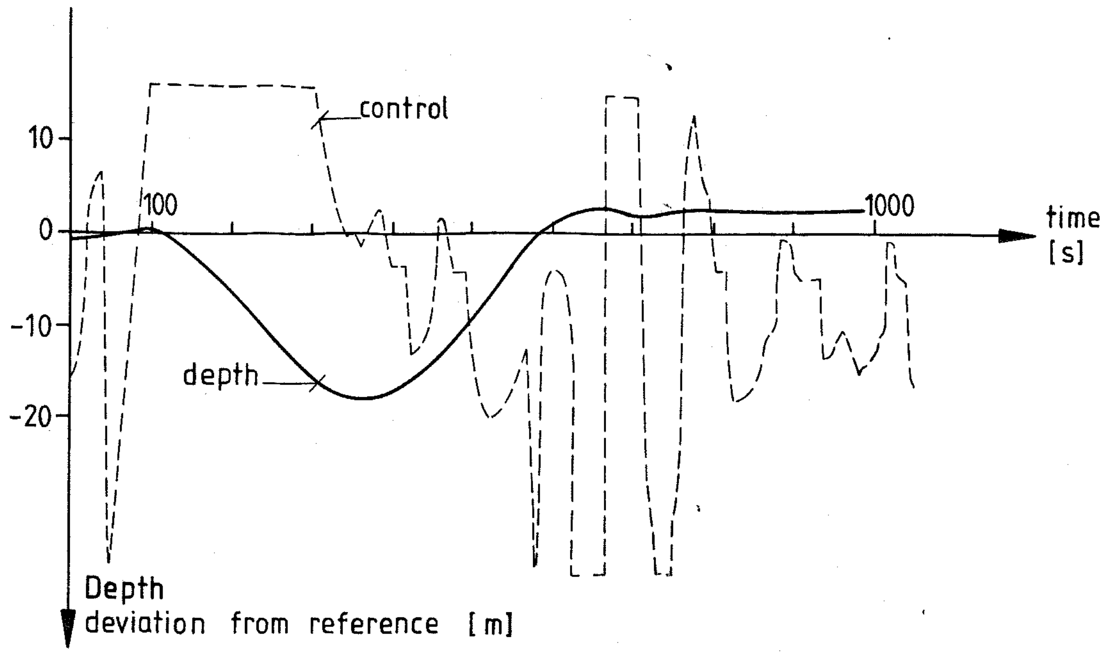

Fig. 5.5b Underwater test of the submarine using controller (5.48). Times greater than 600 should be disregarded since the submarine touched the surface. Initial condition $x(0) = (0\ 0\ -0.004)^T$.

A comparison between figures 5.5a and 5.5b suggests that the linear model is inadequate. Although Theorem 3.1 implies stability for the closed loop system (5.46), (5.34), stability cannot be guaranteed for the real life nonlinear system.

□

Some different ways to work with the algorithms have been demonstrated in the examples. A better understanding of the design method is certainly needed, i.e. how the choice of Q and the gain k affects the result.

6. SUMMARY AND DISCUSSION

In this paper a new method is presented to find stabilizing <u>saturated linear controllers</u> for linear continuous time and discrete time systems with control constraints. The basic theorems and design algorithms are found in Section 3 and 4.

The underlying design criterion is the perfomance in the time domain, which has to be checked by simulations.

The design involves:

1. deciding upon a set D of initial conditions in the state space from which stabilization is desired.

2. finding a low-gain stabilizing control, and the set E, in which this control initially does not exceed the control constraints,

3. solving a Lyapunov equation in order to find a Lyapunov function for the closed loop system, and a stabily region Ω,

4. checking the crucial condition $D \subseteq \Omega \subseteq E$,

5. tuning a diagonal "gain" matrix.

The design scheme might seem involved but with suitable interactive computer software, including e.g. linear-quadratic conrol synthesis, some optimization routines, and simulation facilities, the task is easily managable. Software packages that include facilities useful for our design although they are not taylor-made, are Synpac, Wieslander (1980), and Simnon, Elmqvist (1975) of The Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Stability is the only property of the new controllers that is investigated analytically in this report. More thorough knowledge is of course required - it is left as a topic for further research.

The new controllers are applied to the single input double integrator plant in Section 5. It is found that although they cannot match the time-optimal controller in speed, they are superior to linear controllers designed with the intention not to violate the constraints.

The main advantage of the new controllers are their extremely easy implementation. It is our conviction that for those plants where simulation studies show that they perform satisfactorily they will become popular.

7. ACKNOWLEDGEMENT

Professor Karl Johan Aström suggested this topic and kicked off my work in the right direction. Professor Per Hagander kept the work on track by his encouragement and valuable comments, improvements of the manuscript, advice and criticism. Sven Erik Mattsson and Tore Hägglund backed me up: our fierce discussion were very stimulating. Jan Sternby and Bertil Lundgren of Kockumation AB, Malmö provided me with Example 5.5 and Bertil risked his life by conducting the underwater testing. Eva Dagnegård and Agneta Tuszynski typed the manuscript with a beauty not far from their own, while Britt-Marie Carlsson's curves are delightful as usual. To all these, and to all the other members of the Department of Automatic Control I express my happy thanks.

8. REFERENCES

Anderson, B.D.O and Moore, J.B.(1971): Linear Optimal Control. Englewood Cliffs, N.J.: Prentice-Hall.

Aström, K.J. (1973): Reglerteknik - Samplade System. Lund, Sweden: TLTH/VBV.

Aström, K.J. (1970): Introduction to Stockastic Control Theory, New York: Academic.

Aström, K.J. (1976): Reglerteori. Stockholm: Almqvist & Wiksell.

Aström, K.J. (1971): Reglerteknik - Olinjära System. Lund, Sweden: TLTH/VBV. (In Swedish).

Aström, K.J. (1981): Reglerteknik, Lund, Sweden: Sigmatryck. (In Swedish).

Athans, M and Falb, P.L. (1966): Optimal Control, an Introduction to the Theory and its Applications. New York: McGraw-Hill.

Brockett, R.W. (1970): Finite dimension linear systems. New York: Wiley.

Coddington, E.A. and Levinson, N. (1955): Theory of Ordinary Differential Equations. New York: Mc Graw Hill.

Elmqvist, H. (1975) Simnon: An Interactive Simulation Program for Nonlinear Systems. Lund, Sweden: Dept. Aut. Control, Report 7507.

Fleming, W.H. and Rishel, R.W. (1975): Deterministic and Stockastic Optimal Control. New York: Springer.

Frankena, J.F. and Sivan, R. (1979): A nonlinear optimal control law for linear systems. Int. J. Control, vol 30, no. 1, 159-178.

Kirk, D.E. (1970): Optimal Control Theory - an Introduction. Englewood Cliffs, N.J.: Prentice Hall.

Leden, B. (1975): Identification and Dead-Beat Control of a Heat Diffusion Process. Lund, Sweden: Dept. Autom. Control, Report 7508.

Lancaster, P. (1969): Theory of Matrices. New York: Academic.

Wieslander, J. (1980): Synpac Commands Users Guide. Lund, Sweden: Dept. Autom. Control, CODEN: LUTFD2/(TFRT-3159)/1-130/(1980).

PART 3 - APPLICATION OF LINEAR PROGRAMMING FOR ON-LINE CONTROL

Abstract:

For linear dynamical systems with linear constraints the regulator problem can be formulated as a linear programming problem. In present industrial practice, control and state constraints are usually considered in ad-hoc ways, e.g. with max-min selectors. An LP-regulator, operating in the Open Loop Optimal Feedback (OLOF) fashion, is used on-line to control a laboratory process with both input and state constraints. Several examples are simulated.

The method suits both single- and multiple-input systems with or without time delays. It seems quite robust and especially suitable for large reference value changes and large disturbances. It is feasible for processes with sampling periods of 10 seconds or more (depending on model order, number of inputs, and the speed of the computer), i.e. for many industrial applications involving flow, heat, chemical processes, and climate control. As small computers become increasingly faster, the LP-OLOF regulator will get more attractive.

The regulator was built around a standard LP-program on a VAX 11/780 computer. The laboratory process controlled consisted of two water tanks. The constrained input acted on the pump to the upper tank, from where the water flowed freely into the lower tank. The level of the lower tank was to assume a reference value in minimum time. The state constraints prevented overflow in the tanks. The experiments show that the LP-OLOF regulator behaves excellently although its internal model of the tanks does not take into account the non-linear dynamics.

1. INTRODUCTION

Already Zadeh (1962) and Propoi (1963) noticed that the minimum time problem for linear discrete systems with constraints on the controls can be formulated as a series of linear programming problems. Propoi (1963) proposed that the LP-problems be solved in each sampling interval, thus getting a feedback control. This method is called Open Loop Optimal Feedback (OLOF).

In spite of the simplicity of the LP-OLOF, most control engineers seem to be discouraged by the excessive computations needed in each sampling interval. It was feared that the computations would be so time consuming that the LP-OLOF could only be applied to processes with very long sampling intervals. This need not be the case using a modern computer. We show that sampling periods of a few seconds and more are possible, depending on model order, number and type of constraints, and the computer performance. These results are very encouraging, since many industrial processes would allow sampling periods of these orders of magnitude.

We present here, as far as we know, the first simulations of how the LP-OLOF regulator would behave "on line" for technical applications. It is also the first time the LP-OLOF with state constraints is reported to control a real-life process.

The LP-algorithm we use is not tailormade but "taken from the shelf". We wanted to investigate if it is viable to construct an LP-OLOF using a standard time-proven LP-algorithm. LP-algorithms have now matured to a science of its own. Expertise is needed to avoid numerical problems. Of course we pay a price: some shortcuts can be taken in a tailor-made version. However, many special LP-routines reported in the literature do not seem easy to extend to include state constraints.

The paper is organized as follows: In chapter 2 the regulator problems are defined, and some approaches to solve them are given. In chapter 3 the linear programming method is presented, including a literature survey. Chapter 4 treats the constrained input case, and chapter 5 deals with the case when both input and state constraints are present. Both these chapters include simulations. In chapter 6, the real-life experiment is described. The observed properties of the LP-regulator are listed in chapter 7. The discussion and conclusions follow in chapters 8 and 9. Chapter 10 contains the acknowledgement, and chapter 11 the references.

2. PRELIMINARIES

Assume that a plant is linear with constant control constraints,

$$\dot{x} = Ax + Bu \qquad (2.1a)$$

$$\alpha_u(j) \le u_j(t) \le \beta_u(j), \quad j = 1,2,\ldots,m; \text{ all } t \ge t_0 \qquad (2.1b)$$

or with constant control and state constraints,

$$\dot{x} = Ax + Bu \tag{2.2a}$$

$$\alpha_u(j) \le u_j(t) \le \beta_u(j), \quad j = 1,2,\ldots,m; \text{ all } t \ge t_0 \tag{2.2b}$$

$$\alpha_x(i) \le x_i(t) \le \beta_x(i), \quad i = 1,2,\ldots,n; \text{ all } t \ge t_0 . \tag{2.2c}$$

Define $\alpha_u = [\alpha_u(1), \alpha_u(2), \ldots, \alpha_u(m)]^T$, and similarily for β_u, α_x, and β_x.

More general linear constraints are not considered here. It is straightforward to extend the proposed methods to such cases.

Our objective is to design a regulator that takes the state (or output) from its initial value to a prescribed stationary point, which will define the origin in the state space. The regulator must stabilize the system, and it is assumed that the initial state is such that it is possible to stabilize the systems in spite of the imposed constraints on controls and states. Thus we assume stabilizability.

The regulators could be constructed with various optimization criteria in mind, e.g. minumum time, i.e. it should take the shortest possible time to reach the desired state, or minimum control effort, etc. In this study we will mainly use the minimum time criterion.

It may be assumed that the whole vector x(t) is measured, or only the output y(t) = Cx(t), or that the state is

6
S
H

reconstructed without error, etc. Here it is assumed that the whole state is available for measurement at discrete sampling instants. When not otherwise stated it is assumed that the sampling interval = 1.

In the next section we will give an exposé over some of the methods proposed to design controllers for (2.1) and (2.2).

2.1 Some possible solutions

A straightforward idea is to design a linear controller that never hits the constraints and causes the states never to hit their constraints. This could be done e.g. through the linear-quadratic design method, or via model following, i.e. the set point of the closed loop system is changed in such a modest way (e.g. a ramp reference input) that the constraints are never violated. Both these methods use simulation to check that the constraints are not violated. Although these methods in general are far from time optimal, at least closed loop stability can be guaranteed in a neighbourhood of the reference point.

The common method in practice to deal with (2.1) is to design a linear controller (e.g. state feedback, output feedback, or PID) disregarding the constraints and then saturate it. The region of stability may be very small. For open loop stable plants Åström (1971) proposes a stabilizing relay control scheme, and Shapiro (1972) suggests a method to determine the matrix L so that u(t) = sat[Lx(t)] is stabilizing. Under certain conditions there is a method to determine a stabilizing u(t) = sat[Lx(t)] also for open loop unstable plants, see Gutman (1982).

To find a regulator for (2.2), Glattfelder (1974) and others have proposed the method of max-min selectors. In this method a whole set of controllers is used. When a state constraint is violated, the control is switched to a controller whose aim is to remove the state from the boundary. Although there exists no proof of stability (it often happens that the states oscillate from one boundary to another) the method has become quite popular for instance in the power industry.

2.2 The Open Loop Optimal Feedback idea

An elegant way to design regulators is to apply optimal control theory. The constraints are taken care of in a natural way, while you are optimizing a criterion of your own choice. The disadvantage is that in general you get an open loop control solution from one initial state, not the desired feedback control. There are two ways to create a feedback controller from the open loop optimal control. One is to solve the optimal control problem for a great number of initial conditions, and then to set up a look-up table, which can be used in a feedback fashion. The second way is to solve the optimal control problem once every sampling instant, i.e. the Open Loop Optimal Feedback scheme (OLOF-scheme).

The OLOF idea was proposed in a stochastic setting by Dreyfus (1964), and is described in Wittenmark (1975). It has been considered in the context of adaptive control, see e.g. Bar-Shalom (1976).

In each sampling interval you compute an optimal control sequence. Then you use only the control pertaining to the current sampling interval. This is illustrated in Fig. 2:1.

If the computation time is shorter than the sampling period, you can simply use u(t) as soon as it is computed. Possibly, you will have to include a time delay in the model of your sampled system, which is not done in this report. If the computation time is longer than the sampling period, you cannot use the regulator.

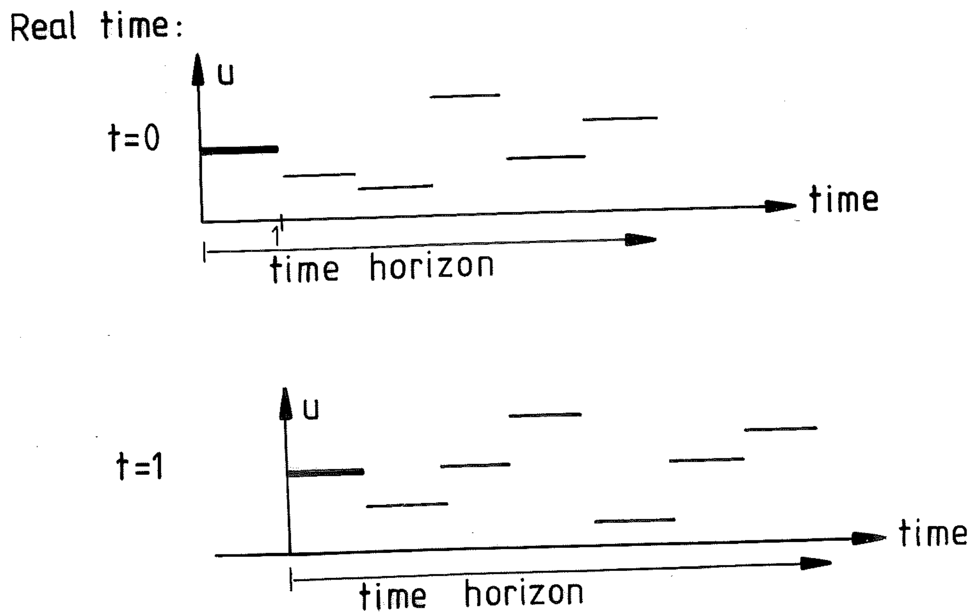

Fig. 2:1. The Open Loop Optimal Feedback idea.

2.3 Solving the optimal control problem

When solving optimal control problems, some optimization methods allow a free end-time. When using e.g. linear programming, it must be specified.

Let the time horizon τ, be the end-time of a fixed end-time optimal control problem. It is possible to solve a free end-time optimal control problem with the linear programming method, by solving a sequence of fixed end-time problems, iterating over τ.

We now turn to the optimization criterion. Various linear criteria could be used in the LP-method. Recall that the reference value is chosen as the origin. As time optimality has been in most researchers' minds, we will present three methods to achieve it, at least suboptimally.

Method 1: Initialize τ. Minimize $|x(\tau)|$. Iterate over τ until $|x(\tau)| < \varepsilon$, where ε is a predetermined test quantity.

□

Method 1 was proposed by e.g. Canon (1970) and Bashein (1971).

When solving the free end-time problem as a sequence of fixed end-time problems, it is essential that the computation continues until "optimal τ" is reached. That a shorter final τ might be harmful is obvious. A longer final τ can also be harmful in the OLOF-setting, since the control may be nonunique, and the one obtained for the current sampling interval might then counteract the control objective.

There is one possible disadvantage with Method 1; the minimization of $|x(\tau_{opt})|$ might give non-unique control sequences u(t). It is at present not clear whether this is harmful for an OLOF-regulator. In any case, many authors have been disturbed by it, and therefore proposed a different optimization to get a unique solution:

<u>Method 2</u>: Find the minimal τ for which $x(\tau) = 0$. Then minimize

$$\max_{j,t} |u_j(t)|.$$

Iterate over τ until (2.1b) is satisfied. This is called the minimum-time minimum-amplitude solution. It was proposed by e.g Cadzow (1974) and Rasmy (1975).

□

<u>Method 3</u>: Find the minimal τ for which $x(\tau) = 0$. Then minimize

$$\sum_{j,t} |u_j(t)|.$$

Iterate of τ until (2.1b) is satisfied. This is called the minimum-time minimum-effort solution. It was proposed by e.g. Kolev (1978).

□

If you solve the minimum-amplitude problem in the above scheme, and if in (2.1b) $\beta_u(j) = -\alpha_u(j) = \alpha$, all j, then you will end up with the same solution as in Method 1 (disregarding non-uniqueness). This is usually not the case for the minimum-effort problem.

We will use method 1 in the most simple way, i.e. we solve one LP-problem for each τ.

3. THE LINEAR PROGRAMMING METHOD

In this chapter it is shown how the regulator problem for plants (2.1) and (2.2) could be formulated to fit into the linear programming framework. A literature survey is also included.

3.1 Some problem formulations

With linear programming, problems of the following type are solved:

Find z such that $f^T z$ is minimized, (3.1a)

under the conditions

$Az = b$, (3.1b)

$0 \le z_i \le v_i$, $i = 1, 2, \ldots, N.$ (3.1c) where f and z are N-vectors, b is an M-vector, and A an M*N matrix. Some LP-routines allow inequality constraints in (3.1b), and some have other upper and lower limits in (3.1c).

When solving an optimal control problem for plant (2.1) with the LP method, we use

$$x(\tau) = \Phi^{\tau} x(0) + \sum_{s=0}^{\tau} \Phi^{\tau-1-s} \Gamma u(s) \qquad (3.2a)$$

$$\alpha_u(j) \le u_j(t) \le \beta_u(j), \qquad j = 1,2,\ldots,m;\ \text{all } t \qquad (3.2b)$$

where τ is the time horizon, and $x(0)$ is the initial state. The LP-routine then determines the optimal $u(t)$, $t = 0, \ldots, \tau-1$, and the corresponding $x(\tau)$. In view of (3.1c), some LP-routines need a dummy constraint on $x(\tau)$:

$$\alpha_x(i) \le x_i(\tau) \le \beta_x(i), \qquad i = 1,2,\ldots,n \qquad (3.2c)$$

where $\alpha_x(i) < 0$, and $\beta_x(i) > 0$. We study this problem in chapter 4.

For the state constrained system (2.2) we directly use the following sampled model of (2.2):

$$x(t+1) = \Phi x(t) + \Gamma u(t) \qquad t = 0,1,\ldots,\tau-1 \qquad (3.3a)$$

$$\alpha_u(j) \le u_j(t) \le \beta_u(j) \qquad j = 1,2,\ldots,m;\ \text{all } t \qquad (3.3b)$$

$$\alpha_x(i) \le x_i(t) \le \beta_x(i) \qquad i = 1,2,\ldots,n;\ \text{all } t \qquad (3.3c)$$

Here the LP-routine solves for the optimal $u(t)$, $t = 0, \ldots, \tau-1$, and the corresponding $x(t)$, $t = 0, \ldots, \tau$.

Note that the solution will not guarantee that (3.3c) is satisfied between the sampling instants. Note also that we can use (3.3) directly, and we do not have to solve for x(t) explicitly in u(s) as in (3.2a).

We treat the LP-regulator problem for plants with both state- and control constraints in chapter 5.

3.2 Literature survey

Many authors have concentrated on algorithm development for finding the time optimal solution for (3.2), along the lines presented above in section 2.3. Among them we note Torng (1964), Weischedel (1970), Canon (1970), Bashein (1971), Cadzow (1974), Rasmy (1975), Kolev (1978), Abdelmalek (1978) and Tracht (1980). Most of these authors have considered the linear programming method. Their main preoccupation has been to increase computing speed and decrease the memory storage. This has usually been achieved by utilizing (partial) results of the solution for τ to speed up (or make redundant) the computation for $\tau+1$. As an example Rasmy (1975) needed appr. 0.1 seconds on a CDC-6400 computer for the solution for a 4th order, single input system with $\tau = 20$.

Sakawa (1977) has developed a simplex method for the multicriteria linear optimal control problem, and Kim (1981) a method for systems with delay.

Rasmy (1977) touched upon the on-line application, although he considered OLOF impossible since he beleived the computations would take too long. Instead he used a longer variable sampling interval but allowed the control to be piecewise constant during the sampling interval.

Nieman (1973a) used the time optimal LP-solution as program control for a pilot plant evaporator. In Nieman (1973b) he used the off-line solution as reference for the evaporator in closed loop.

Knudsen (1975) seems to be the only one to have implemented LP-OLOF for a plant with control constraints. He used his own algorithm: A 4th order continuous time model for a pilot plant evaporator was discretized using a Gauss quadrature formula; $x(\tau):=0$; an initial τ was guessed, and increased until a feasible solution for the LP-problem was found. Non-uniqueness was disregarded. The sampling time was 30 seconds and the computation time on an IBM 1800 computer was 12 - 16 seconds including filtering and data logging.

Recently Baba (1980) has studied how the computing time affects the choice of sampling period for LP-OLOF.

Many authors have pointed out the desirability to include state constraints but the only early reference found touching upon the regulator problem for (2.2) is Lack (1967). Linear programming was used to find the off-line trajectory that maximized the distance to a danger region in the state space whose boundary was linear. The final state was given.

Chang (1981) contains ideas similar to those presented here. But he only considered a suboptimal regulator looking one sampling interval ahead, and used the solution in feedback fashion in simulations. This is a regulator, that for some plants is not even stabilizing.

In economics and related fields linear programming is used for decision and planning, e.g. Propoi (1981). One could say that this constitutes LP-OLOF with very long sampling periods.

4. THE CONSTRAINED INPUT CASE

The Institute of Applied Mathematics, Stockholm, kindly supplied us with PRIMAL, a linear programming subroutine package written in FORTRAN, see Holmberg (1981). PRIMAL solves the problem defined in (3.1), with the exception that $z_1 \hat{=}$ loss. The base inverse matrix is stored in product form, and reinversion is done when the multiplication becomes too time consuming, see Orchard-Hays (1968). The M*N matrix A is stored columnwise, without zero entries. The computation time is approximately proportional to N·M·ln M. The routine contains a number of test quantities that have to be properly chosen.

Using the subroutine PRIMAL, we want to solve the optimal control problem for (2.1), in the formulation of (3.2) with the criterion

$$\text{minimize loss} \hat{=} \min |x(\tau)|_c \hat{=} \min \sum_{i=1}^{n} c_i |x_i(\tau)| \qquad (4.1)$$

where the weight vector $c = [c_1, c_2, \ldots, c_n]$.

In order to compute the loss two dummy variables are introduced, see e.g. Canon (1970):

$$\begin{aligned} w_i - y_i &= x_i(\tau), \qquad i = 1,2,\ldots,n \\ w_i &\geq 0 \\ y_i &\geq 0 \end{aligned} \qquad (4.2)$$

Since the LP-solution guarantees that $\min(w_i, y_i) = 0$, (4.2) implies that

$$w_i + y_i = |x_i(\tau)|, \qquad i = 1,2,\ldots n \qquad (4.3)$$

Hence we can rewrite (3.2), (4.1), (4.2), (4.3) into the PRIMAL-form:

$$A = \begin{pmatrix} -1 & c & c & 0 & \cdots & 0 \\ 0 & I & -I & \Gamma & \cdots & -\Phi^{\tau-1}\Gamma \end{pmatrix}$$

$$z = \left(\text{loss} \quad w^T \quad y^T \quad [u(\tau-1)-\alpha_u]^T \cdots [u(0)-\alpha_u]^T \right)^T$$

$$b = \begin{pmatrix} 0 \\ \Phi^{\tau} x(0) + \left[\Phi^{\tau-1}\Gamma + \ldots + \Gamma\right]\alpha_u \end{pmatrix} \qquad (4.4)$$

$$v = \begin{pmatrix} \text{maxloss} \\ \beta_x \\ -\alpha_x \\ \beta_u - \alpha_u \\ \vdots \\ \beta_u - \alpha_u \end{pmatrix}$$

The structure of (4.4) is very favourable. A and v are easily expanded for increasing τ, and truncated for decreasing τ. The new x(0) obtained at each sampling instant only influences the last n elements of vector b.

The LP-routine was incorporated as a subroutine in a FORTRAN-program LPREG. A skeleton flow chart of LPREG is given in Fig. 4:1.

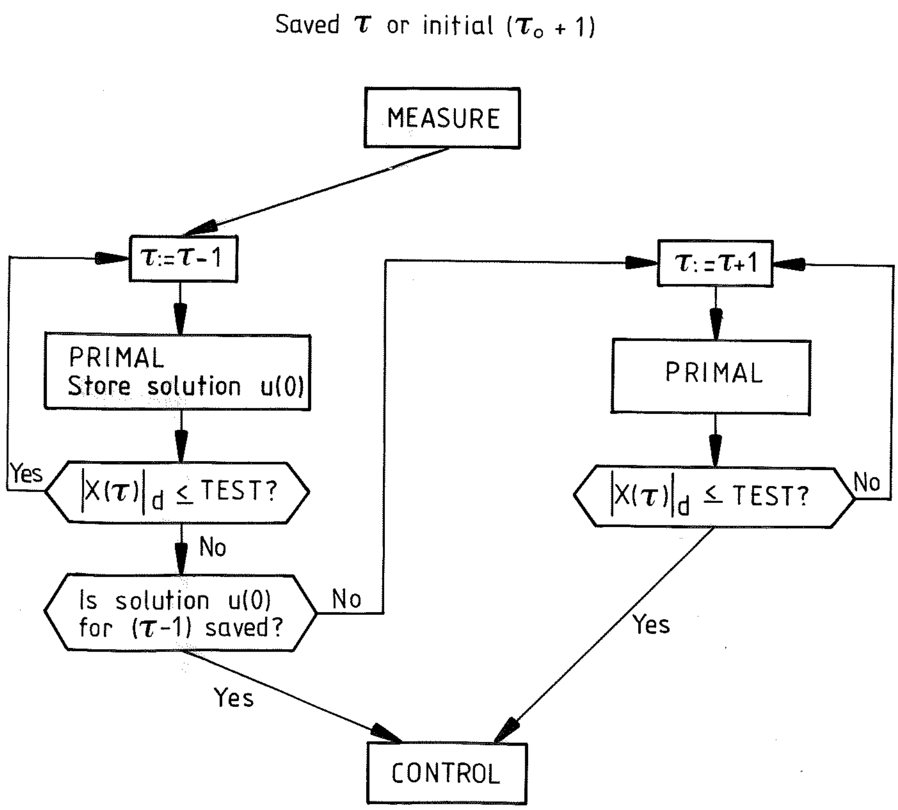

Fig. 4:1. The skeleton flow chart of main program LPREG.

The user defines the initial value τ_0 of τ. The optimal τ is saved from one sampling instant to the next. Observe that the saved τ is decreased by one unit immediately. This corresponds to the anticipation that the optimal time horizon will decrease by one for each sampling interval.

Information about what variables are base variables in the LP-solution, is saved from one τ to the next during the same sampling interval, and from one sampling instant to the next. Sometimes that old information suggests optimal base variables, and the solution for the new τ is thus speeded up considerably. This typically occurs in "steady state", i.e. when the previous solution almost correctly anticipates the correct time horizon and trajectory.

The following test is used to determine if we reach a sufficiently small neighbourhood of the origin:

$$|x(\tau)|_d \hat{=} \sum_{i=1}^{n} d_i |x_i(\tau)| \leq \varepsilon. \qquad (4.5)$$

Let $d = [d_1, d_2, \ldots, d_n]$. We want the minimal τ for which (4.5) is fulfilled. Therefore, τ is decreased until (4.5) is <u>not</u> satisfied, or increased until (4.5) <u>is</u> satisfied. Hence, at least two LP-problems will be solved in each sampling interval. In "steady state", two LP-problems are solved. In theory "steady state" will be reached after one sampling interval.

We will now turn to two simulated on-line examples. In both of them the plant is defined as a continuous system in the simulation language SIMNON, Elmqvist (1975), and the program LPREG is included as a discrete FORTRAN subsystem. The program is run on a VAX 11/780 computer.

Example 4.1: The plant is the standard double integrator

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad (4.6)$$

with the control constraint

$$-1 \leq u(t) \leq 1 \qquad (4.7)$$

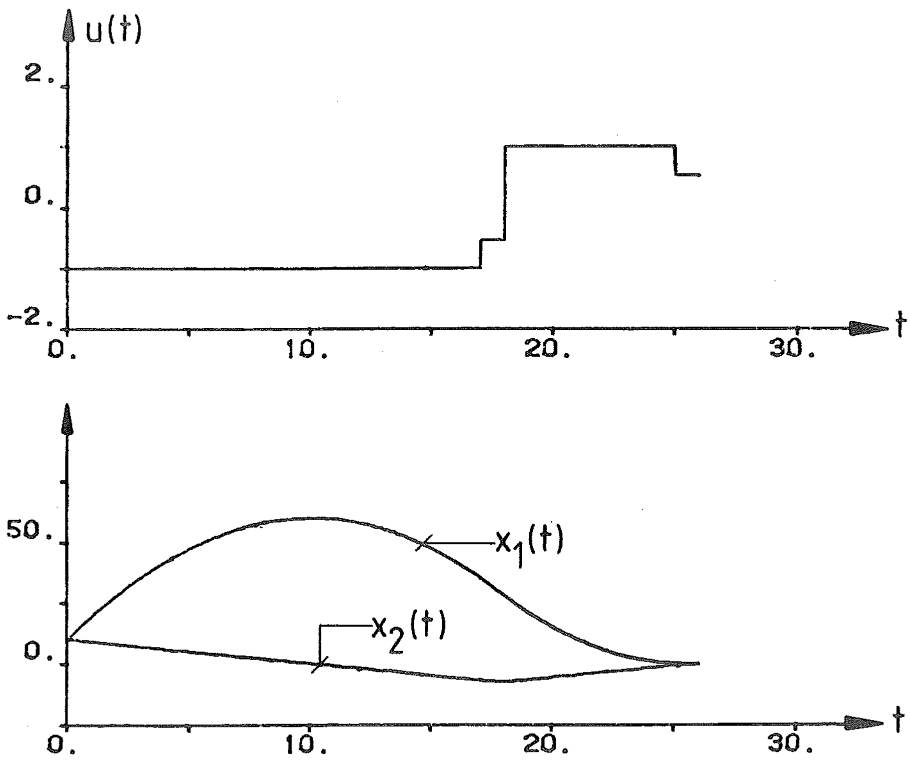

Fig. 4:2. The double integrator: u(t) and x(t).

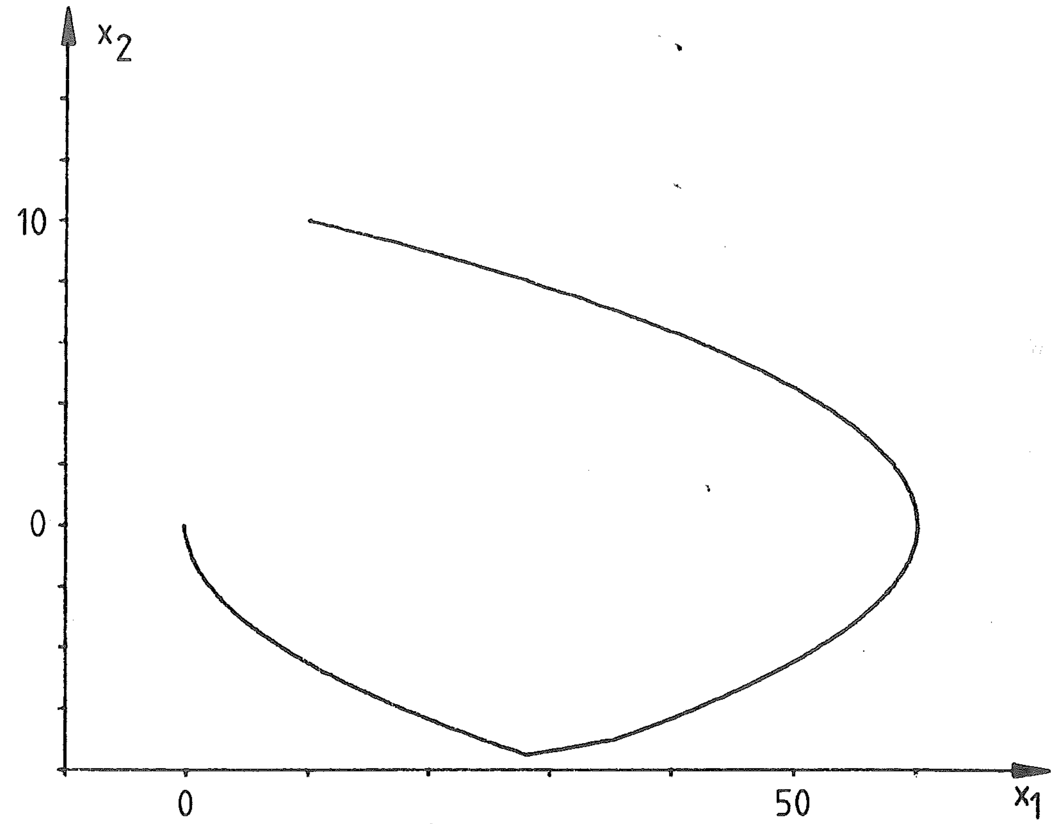

Fig. 4:3. Phase portrait for the double integrator starting at $x(0) = (10\ \ 10)^T$.

The sampling interval of the controller is 1 second. The sampled model does not include any time-delay possibly generated in the regulator. The weights in (4.1) and (4.5) are chosen as c = d = (1 1) and ε = 0.66.

In Fig. 4:2 the computed controls and the state trajectories from the initial condition $(10\ 10)^T$ are shown. Fig. 4:3 depicts the phase portrait.

As expected, the LP-regulator behaves extremely well for this simple example. Given the sampling interval, the trajectories are indeed time optimal. The computation times in "steady state" are very low, less than 0.08 seconds. A possible time problem occurs at start-up: the initial time horizon, τ_0 was set (guessed) to 22, and the program needed 0.23 seconds (0.79 seconds when $\tau_0 = 1$) to find the optimal $\tau = 25$. The computation time is not included as a delay in the simulation, which means that u(t) was assumed available immediately after the receipt of x(t).

□

Example 4.2: The plant is the model for the lateral motion of an airplane, found in Bryson (1969). The plant has 5 states and 2 control inputs:

$$\dot{x} = \begin{bmatrix} -0.0297 & -1 & 0 & 0.0438 & 0 \\ 0.331 & -0.00416 & -0.0461 & 0 & 0 \\ -1.13 & 0.129 & -0.795 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ -0.381 & 0.0671 \\ -0.0404 & 1.59 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} u \qquad (4.8)$$

$$-1 \le u_j(t) \le 1 \qquad j = 1,2$$

where x_1 = sideslip angle, x_2 = yaw angular velocity, x_3 = roll angular velocity, x_4 = roll angle, x_5 = yaw angle, u_1 = rudder deflection, and u_2 = aileron deflection.

7
S
H

The sampling interval of the controller is (highly unrealistically) set to 1 second. The sampled model does not include any time delay possibly generated by the regulator. The weights are c = d = (2 0.5 0.5 1 2), and ε = 0.001.

The control trajectories can be found in Fig. 4:4. In Fig. 4:5 you find the ensuing state trajectories from the initial condition $(1\ 0\ 0\ 0\ 1)^T$.

The computing times in "steady state" are 0.4 CPU-seconds or less. However, the start-up computation time, with $\tau_0 = \tau_{opt} + 4$ s is longer than the sampling period: 1.18 CPU-seconds (1.42 seconds for $\tau_0 = 1$). The computation

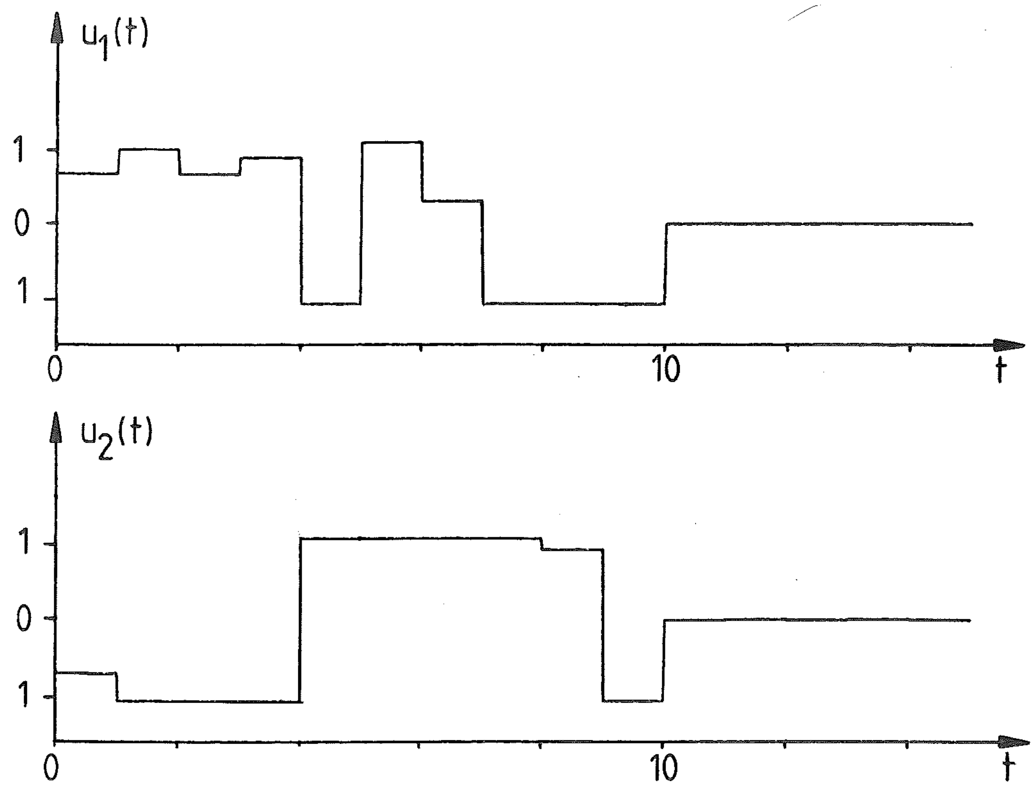

Fig. 4:4. The airplane: the controls.

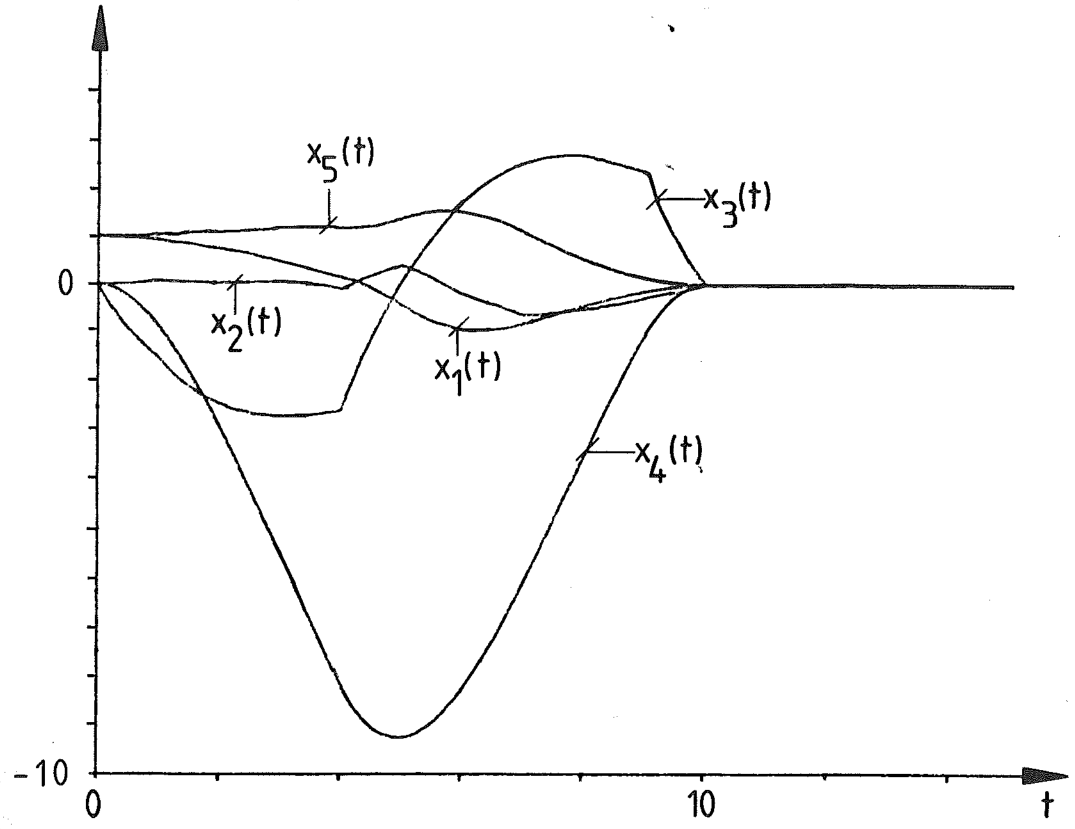

Fig. 4:5. The airplane: the states.

times are not included as a time delay in the simulations.

□

5. THE CONSTRAINTS ON BOTH INPUT AND STATE

Before reformulating (3.3) for our subroutine PRIMAL, we investigate two possible difficulties that might occur when constraints are placed on the states.

Assume that a measurement of a state variable is outside the allowed region. See Fig. 5:1. This might occur for instance if the plant is non-linear or noise affects the system. It might happen that the state variable is so far out in the forbidden region, that there exists no admissible control that will take it back into the allowed region in one sampling interval. In such a case the LP-routine will not find a feasible solution for any τ. But there may still be a control that will take it back in more than one sampling interval.

We therefore propose that the state measurements are "filtered", so that

$$\begin{aligned} x_i(0) > \beta_x(i) \;&\Rightarrow\; x_i(0) := \beta_x(i), \quad i = 1, \dots n \\ x_i(0) < \alpha_x(i) \;&\Rightarrow\; x_i(0) := \alpha_x(i), \quad i = 1, \dots n \end{aligned} \tag{5.1}$$

The controller will believe that the state is on a boundary, and compute a feasible control. This control, however, may or may not bring the actual state towards the allowed

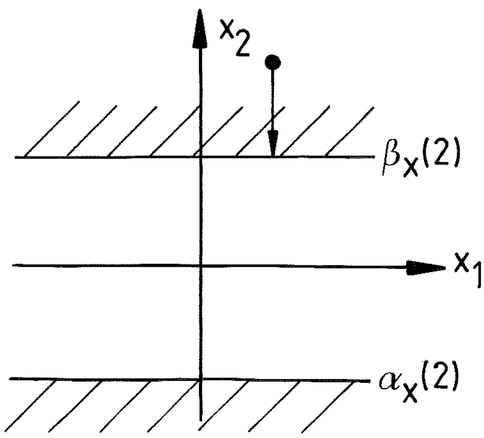

Fig. 5:1. The filter.

region. Because of the filter, the controller will not know if the state passes further into the forbidden region. Since we do not want to rely on luck to bring the state back to the allowed region, we try to keep it slightly away from the boundary, so that the chance it passes outside is decreased. Further, even if a state is not inside the forbidden region at the sampling instants, it might be there between the sampling instants. There is also a risk for oscillatory inputs along a state boundary, similar to chattering.

We therefore propose that a penalty is added to the loss function if the state belongs to the "cordon sanitaire", a strip along the boundaries, see Fig. 5:2. The minimization criterion replacing (4.1), will thus be:

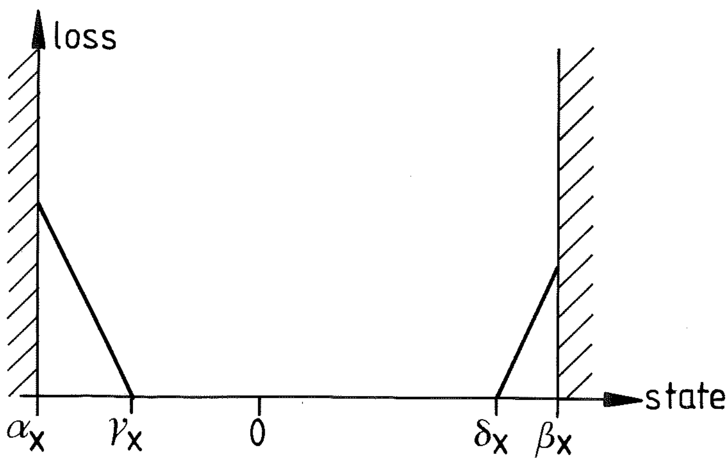

Fig. 5:2. The "cordon sanitaire".

7 S V

$$\min \sum_{i,t} \Big[r_i \max[-x_i(t)+\gamma_x(i),0] + w_i \max[x_i(t)-\delta_x(i),0] \Big] + $$

$$+ \sum_i c_i |x_i(\tau)| \qquad (5.2)$$

with $r = [r_1, r_2, \ldots, r_n]$ and similarily for γ_x^T, w, δ_x^T, and c.

The stopping criterion is still (4.5).

If the penalty weights and the "cordon sanitaire" itself are chosen judiciously, we might be able to assure that the state stays inside the allowed region for all times. We propose simulations to verify this.

<u>Remark 5.1</u> If noise or non-linearities affect the constrained input plant (2.1), controlled by the controller based on (4.4), the state $x(0)$ might be such that there exists no control sequence leading to the origin. To decrease the risk that the state is thrown too far out, the state trajectory should be kept well inside the safe region. This will most likely happen if the control is not allowed to assume its boundary values, except in "emergency" situations. Therefore a cordon sanitaire on the control might be useful.

□

Introducing the same type of dummy variables, (v, y, p_t, q_t, ρ_t, s_t) as in (4.2) and using the loss function (5.2), (3.3) is rewritten into the form:

$$A = \left[\begin{array}{cccccccccccccccc}
-1 & c^T & c^T & 0 & 0 & r^T & w^T & 0 & 0 & \cdot\cdot & 0 & r^T & w^T & 0 & 0 \\
0 & I & -I & -\Gamma & -\Phi & \Phi & 0 & 0 & 0 & \cdot\cdot & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -I & I & I & -I & 0 & \cdot\cdot & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & I & -I & 0 & 0 & -\Gamma & \cdot\cdot & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdot\cdot & -\Phi & \Phi & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdot\cdot & -I & I & I & -I & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdot\cdot & I & -I & 0 & 0 & -\Gamma
\end{array}\right]$$

$$z = \left[\text{loss} \ \ v^T \ \ y^T \ \ u^T_{\tau-1} \ \middle| \ p^T_{\tau-1} \ \ q^T_{\tau-1} \ \ \varrho^T_{\tau-1} \ \ s_{\tau-1} \ \ u^T_{\tau-2} \ \middle| \ \cdots \ \ u^T_0\right]^T$$

$$b = \left[0 \ \ (\Phi\gamma_x)^T \ \ (\gamma_x - \delta_x)^T \ \ (\Phi\gamma_x - \gamma_x)^T \ \ \cdots \ \ (\Phi x(0) - \gamma_x)^T\right]^T$$

$$v = \begin{bmatrix} \text{maxloss} \\ \beta_x \\ -\alpha_x \\ \beta_u - \alpha_u \\ \beta_x - \alpha_x \\ \gamma_x - \alpha_x \\ \beta_x - \delta_x \\ \delta_x - \alpha_x \\ \beta_u - \alpha_u \\ \vdots \\ \beta_u - \alpha_u \end{bmatrix} \qquad (5.3)$$

with

$$p_t - q_t = x(t) - \gamma_x, \quad t = 1, 2, \ldots, \tau-1$$

$$\varrho_t - s_t = x(t) - \delta_t, \quad t = 1, 2, \ldots, \tau-1$$

Each block in the matrix A is of dimension 3n*(4n+m) where n is the number of states, and m the number of control inputs. There are (τ-1) blocks in the A-matrix.

We will treat two examples, similar to those of chapter 4.

Example 5.1: Consider the double integrator plant defined by (4.6) and (4.7) together with the state constraint

$$\begin{aligned} -100 \le x_1(t) \le 100 \\ -3 \le x_2(t) \le 3 \end{aligned} \tag{5.4}$$

The sampling interval of the controller is 1 second. The sampled model does not include any time delay, possibly generated in the regulator. In (5.2) and (4.5),

$$\begin{aligned} r = w = c = d = [1 \; 1] \\ \gamma_x = [-90 \; -2]^T \\ \delta_x = [\; 90 \;\; 2]^T \end{aligned} \tag{5.5}$$

and ε = 0.5. The computation time is included as a delay in the simulation.

In Fig. 5:3 and 5:4 the reference value for x_1, the computed controls and the state trajectories are shown for a simulated experiment. The computation times are clearly visible. At t = 0, the measured state is (0 0) while the

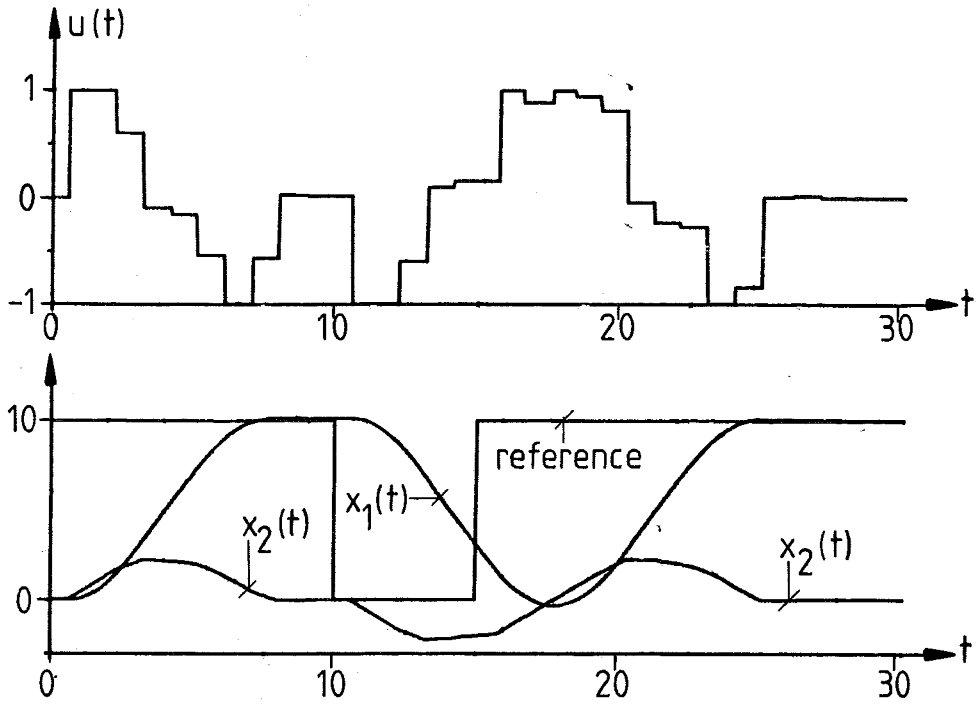

Fig. 5:3. The double integrator: u(t), x(t) and reference.

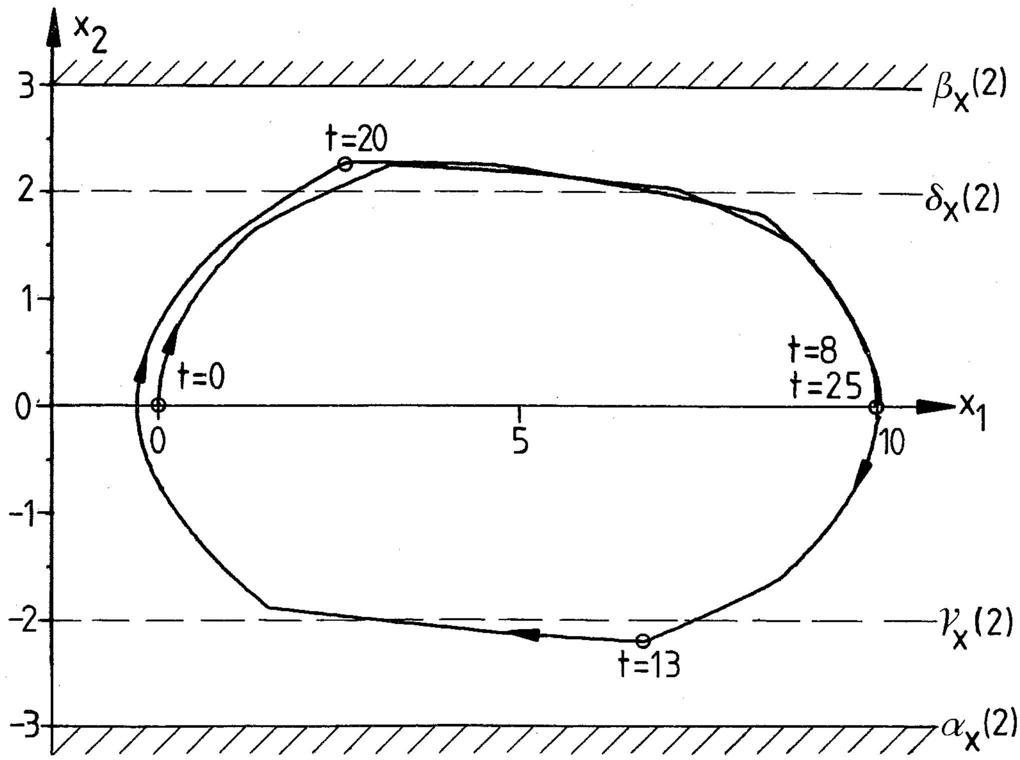

Fig. 5:4. The double integrator: Phase portrait. Note the change of reference point.

desired steady state is (10 0). The computation, starting at the initial time horizon $\tau_0 = 1$ and ending with the optimal time horizon $\tau_{opt} = 7$, takes 0.63 seconds. In the subsequent sampling intervals the computations take between 0.3 and 0.1 seconds, while at t = 7 and up to t = 10, when the state is at (10 0), the computations take about 0.05 seconds. When the reference change at t = 10 occurs, the computations take 0.58 seconds, and at the reference change at t = 15, 0.78 seconds. The time horizon is reset to $\tau = \tau_0 = 1$ at the reference changes.

The controller behaves as expected: full speed forwards until x_2 reaches its "cordon sanitaire", then a suitable control to keep it there, and finally full speed backwards in the unconstrained optimal fashion to reach the desired steady state, whereafter the control is zero. The phase portrait shows that the state bounds are not transgressed.

This simulation is very encouraging since it shows how well the LP-regulator behaves, even when the long computation times affect the control.

The simulation experiment was repeated with $\tau_0 \neq 1$ at the reference changes. However, unless τ_0 was very judiciously chosen, the computation times got longer than those reported above.

□

Example 5.2: Consider the plant given in Example 4.2 together with the state constraints

$$-2.05 \leq x_i(t) \leq 2.05 \qquad i = 1,2,\ldots,5 \qquad (5.6)$$

The sampling interval of the controller is 1 second. The sampled model does not include any time delay, possibly generated by the regulator. In (5.2) and (4.5),

$$r = w = [1\ 1\ 1\ 1\ 1]$$
$$c = d = [2\ 0.5\ 0.5\ 1\ 2] \qquad (5.7)$$
$$\gamma_x = -\delta_x = [-2\ -2\ -2\ -2\ -2]^T$$

and $\varepsilon = 0.001$.

The initial condition is (1 0 0 0 1) and the desired steady state is the origin. With $\tau_0 = 1$, the computation time to get the optimal $\tau = 20$ was 65.1 seconds. With $\tau_0 = 24$, the computation time was 450.2 seconds. Clearly the desired sampling interval of 1 second is far too short for this system, and consequently no OLOF simulation was done.

However, as an illustration only, the solution was applied as an open loop control sequence. The result can be seen in Fig. 9:5. The time to reach the origin, 20 seconds, should be compared with the approximately 50 seconds it takes if the airplane is controlled by a regulator based on linear-quadratic theory with the control and state constraints in mind. The LP-controller will be appealing when the computation time is brought down.

□

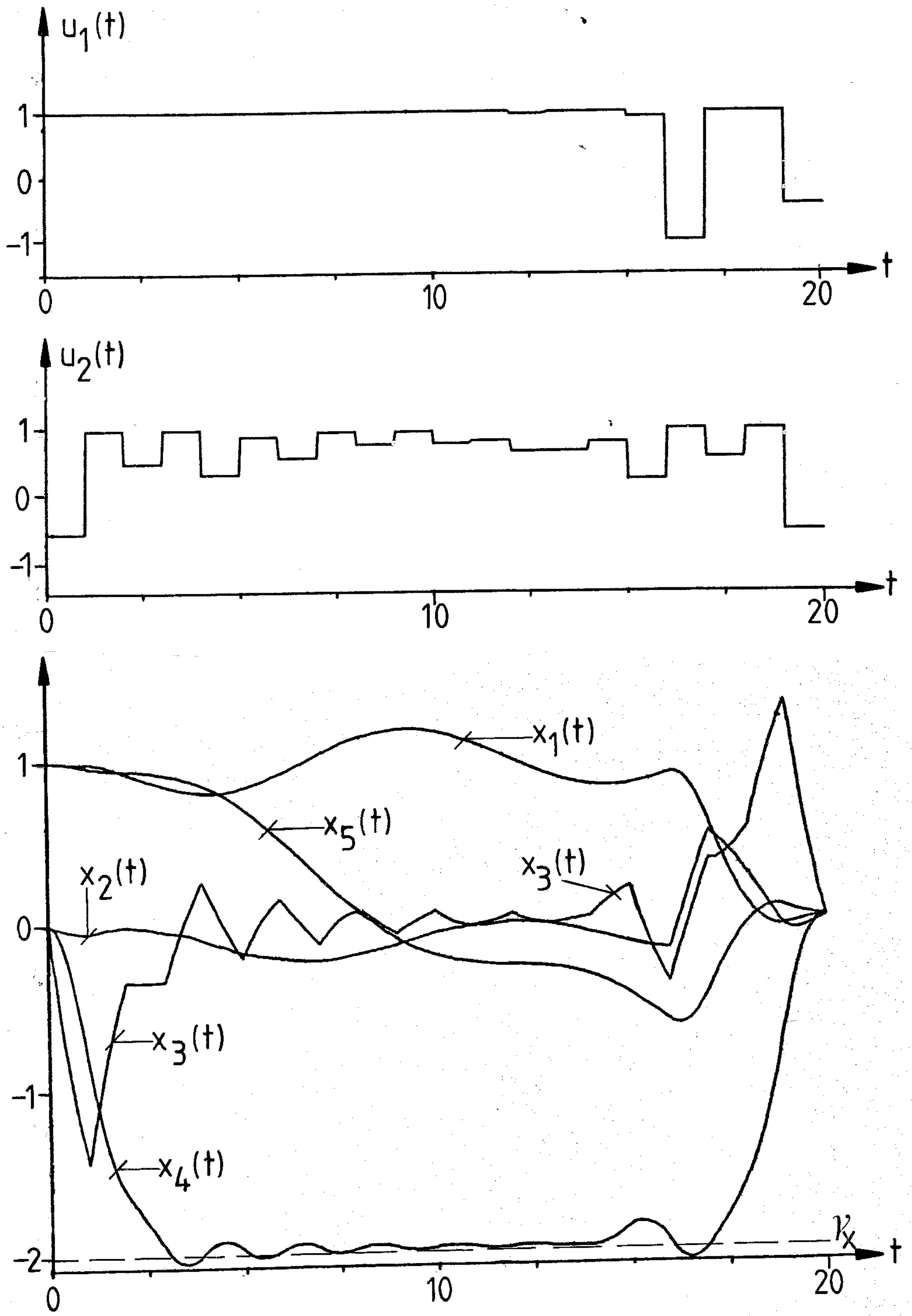

Fig. 5:5. The airplane: the controls and the states.

6. AN EXPERIMENT: THE DOUBLE TANK PROCESS

The LP-regulator was used to control a laboratory process consisting of two water tanks, as drawn in Fig. 6:1.

The tanks were identical: height 20 cm, width 10 cm, and breadth 3 cm. There was free flow from the upper tank (tank 1) into the lower tank (tank 2) through a circular hole with the diameter 3 mm. The flow out of tank 2 was free through a circular hole (diameter 2 mm). A cog wheel pump was used. Its rotational velocity (the tacho signal) was proportional to the flow. The pump characteristics from the input voltage to the flow out was highly non-linear. The water heights in both tanks were measured; h1 in tank 1, h2 in tank 2.

The control objective was to get h2 as fast as possible to a reference value, h2ref, without causing overflow in either tank. The input voltage to the pump was limited between 0 and 10 volts.

The non-linear pump characteristic was found so harmful that we introduced a local, analogue PID-feedback around the pump. The time constant of the pump + PID-controller is less than 1 second which is much smaller than 12 seconds, the sampling interval of the LP-regulator. The dynamics of the

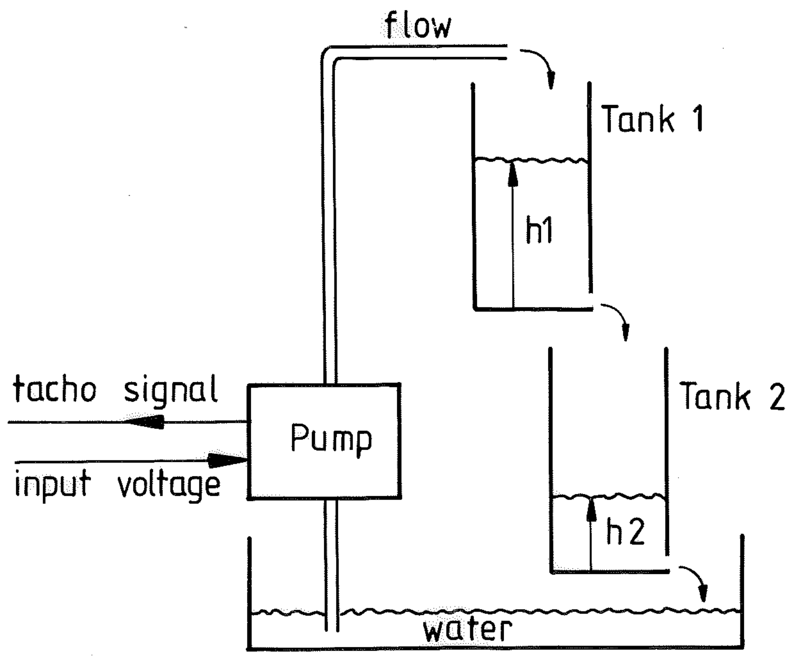

Fig. 6:1. The double tank process: physical setup.

pump with PID-regulator are neglected. Fig. 6:2 depicts the block diagram of the process.

As our VAX 11/780 computer lacks AD/DA-ports, we used another computer, an LSI-11, as an interface. The LSI-11 was connected to a VAX terminal port. The interface program included the sampling clock, and a primitive operator communication facility for inserting h2ref via the LSI-11 terminal. The interface program is written in Pascal extended with the real time kernel described in Elmqvist (1981). The computer structure of the set-up is found in Fig. 6:3.

The double tank system is non-linear. The time constant (from u to h2), when filling the tanks is about 125 seconds; but only about 60 seconds when emptying. The sampling interval for the LP-regulator was set to 12 seconds. The first reason is that it is then a suitable fraction of the shorter time constant. The second reason is that 12 seconds suffices as computation time in the LP-regulator for most reference changes.

Dimensionless units were introduced for u, h1, and h2:

$$0 \le u \le 1$$
$$0 \le h1 \le 1 \qquad (6.1)$$
$$0 \le h2 \le 1$$

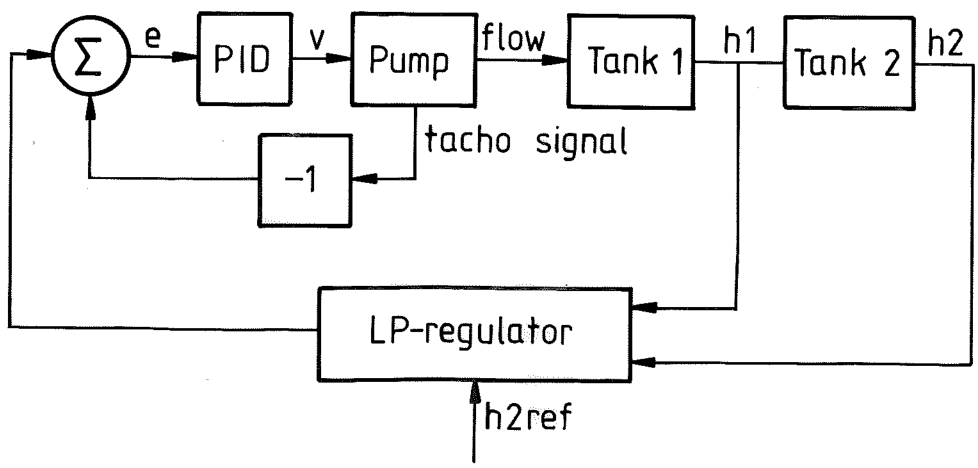

Fig. 6:2. The double tank process: Block diagram.

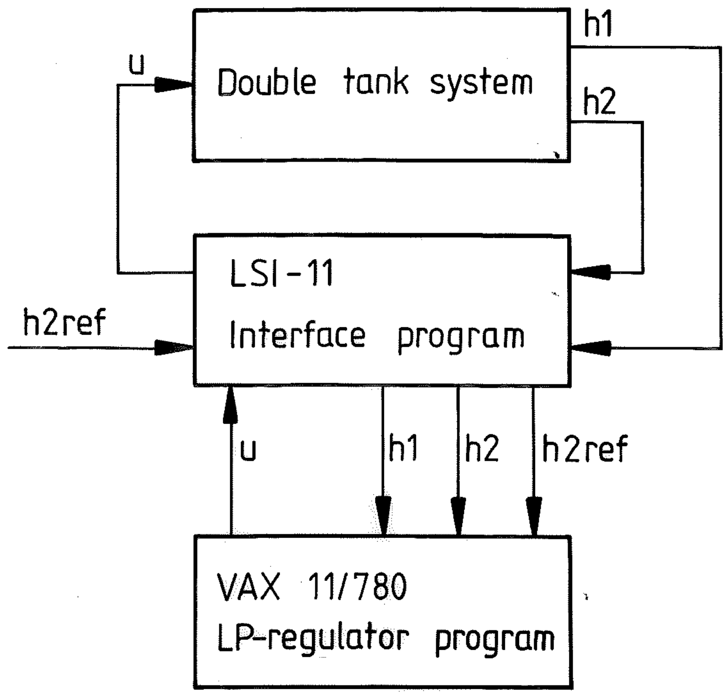

Fig. 6:3. The double tank process: Computer structure.

u = 0 means no water flow, u = 1 means maximum flow. For the tank heights h1 and h2, 0 indicates that a tank is empty, 1 that it is full.

We obtained the following model from some step input experiments. IDPAC, an interactive identification program for linear systems (Wieslander, 1980), was used. We stretched the theory a little in order to retain physical state variables:

$$\dot{h}(t) = \begin{bmatrix} -0.0197 & 0 \\ 0.0178 & -0.0129 \end{bmatrix} h(t) + \begin{bmatrix} 0.0263 \\ 0 \end{bmatrix} u(t) \qquad (6.2)$$

where $h(t) = [h1(t) \quad h2(t)]^T$.

The continuous model was discretized with a sampling period of 12 seconds:

$$h(t+12) = \begin{bmatrix} 0.777 & 0 \\ 0.174 & 0.857 \end{bmatrix} h(t) + \begin{bmatrix} 0.330 \\ 0.0348 \end{bmatrix} u(t) \qquad (6.3)$$

This model was used in the LP-regulator, with

$$u_{LP} = u - uref$$
$$x_1 = h1 - h1ref \qquad (6.4)$$
$$x_2 = h2 - h2ref$$

with h1ref and uref computed from the stationary solution of (6.2), given h2ref. The input and state constraints are given in (6.1). In (5.2) and (4.5) we chose

$$r = w = [1 \;\; 1]$$
$$c = d = [1 \;\; 4]$$
$$\gamma_x = [-h1ref \;\; 0]^T \qquad (6.5)$$
$$\delta_x = [-h1ref+0.9 \;\; 0]^T$$

and $\varepsilon = 0.05$.

Since h2ref = 0 is a perfectly acceptable reference, which means that h1 = 0 in stationarity, we did not want to have a cordon sanitaire for low tank 1 levels. Therefore, $\gamma_x(1) = -h1ref$.

The values of $\gamma_x(2)$ and $\delta_x(2)$ mean that |h2 - h2ref| is included in the loss for all t, not only at the time horizon τ. This is not the intended way to define a cordon sanitaire, but we settled for it after a few preliminary experiments. Especially the behaviour of the controller for small deviations from h2ref was improved. Moreover, for those occasions when the sampling interval is not sufficient to find the optimal time horizon (at large reference value changes), it is good to penalize the deviation from h2ref along the trajectory to ensure that the control leads the system in the right direction. Since the second tank is essentially a first order system, this type of penalty function will not affect the time optimality of the LP-solution.

With $\gamma_x(2) \neq 0$ and $\delta_x(2) \neq 0$, the control again tended to become a dead-beat control for small deviations from h2ref. This caused oscillations since the model was not perfect. We also found that a minimum time horizon of 2 sampling intervals was beneficial for the local control, in addition to the above cordon sanitaire.

The tanks were initially empty during a real life experiment

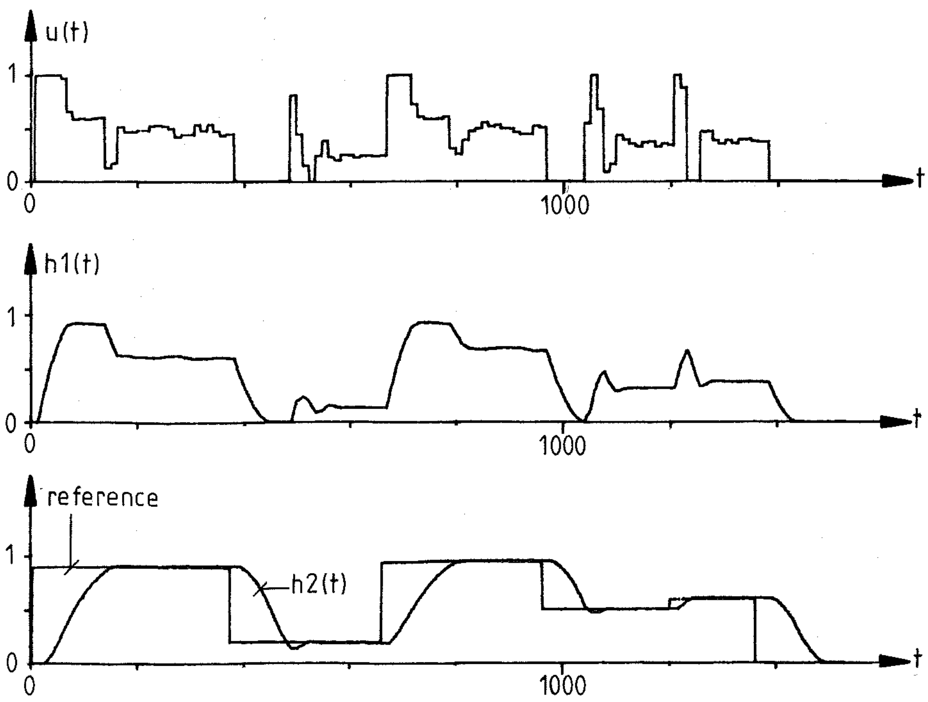

Fig. 6:4. The double tank process: u(t), h1(t), h2ref(t), and h2(t) during a real life experiment.

and h2 was to follow h2ref, as seen in Fig. 6:4. No outer disturbances affected the system. The computation times are discernible in the figure. At reference value changes they were up to 4 seconds, except at the last reference value change (the one at $t = 1362$ seconds), when the computations were interrupted after 10 seconds, and the suboptimal solution, $u = 0$, corresponding to $\tau = 23$, was used in the ensuing sampling interval. (The control $u = 0$ happens to be equal to the optimal control.) The interruption came after 10 seconds, because it was estimated that the next iteration would make the total computation time longer than 12 seconds. In "steady state", the computation times were less than 1.5 seconds. At reference value changes, the time horizon was restarted at 1. At the first reference value change (0 to 0.9) the optimal time horizon was found to be $\tau = 11$, and it took about 12 sampling intervals (144 seconds) for h2 to reach 0.9. At the next reference value change (0.9 to 0.2), the optimal $\tau = 15$, but it took only about 11 sampling intervals (132 seconds) for h2 to reach 0.2. For reference value changes upwards, τ roughly reflected the settling time, while it was too long for downward changes. This explains the overshoot at downward reference changes.

The computation times at the reference changes were: for $\tau = 4$, 0.23 s; for $\tau = 9$, 1.1 s; for $\tau = 11$, 1.8 s; for $\tau = 15$, 3.4 s and for $\tau = 23$ (suboptimal), 10 s. Compare fig. 6:5. These data suggest that the computation time is approximately proportional to $\tau^2 \cdot \ln \tau$.

The controller behaves as expected: maximum flow until tank 1 is full, then a suitable constant flow to keep tank 1 full until h2 reaches the vicinity of h2ref, then a drop in u so that h1 reaches h1ref when h2 = h2ref, and finally u := uref. We notice that the state constraints are not violated, and that the local control is smooth.

As a comparison, this LP-regulator was simulated against the linear model (6.2). The computation times were included in the simulation, but not in the sampled model. The result can be seen in Fig. 6:5, and a comparison between the real lower tank level h2 and the simulated one is found in Fig. 6:6. Notice especially the computation times in Fig. 6:5, which are roughly equal to the real life computation times. As expected, the LP-regulator behaves perfectly in the simulated situation. There are two main deviations between real life and simulation: In the real life experiment there is a slight stationary error in h2. The real life downwards dynamics are faster than the simulated. These discrepances are due to the use of the simple linear model (6.3). A set of different linear models for various reference changes could be used if better control is desired.

What is remarkable is instead how closely the real life experiment resembles the simulation. Even if a simple linear model is used for a non-linear system, the LP-regulator may have nice properties.

8 P V

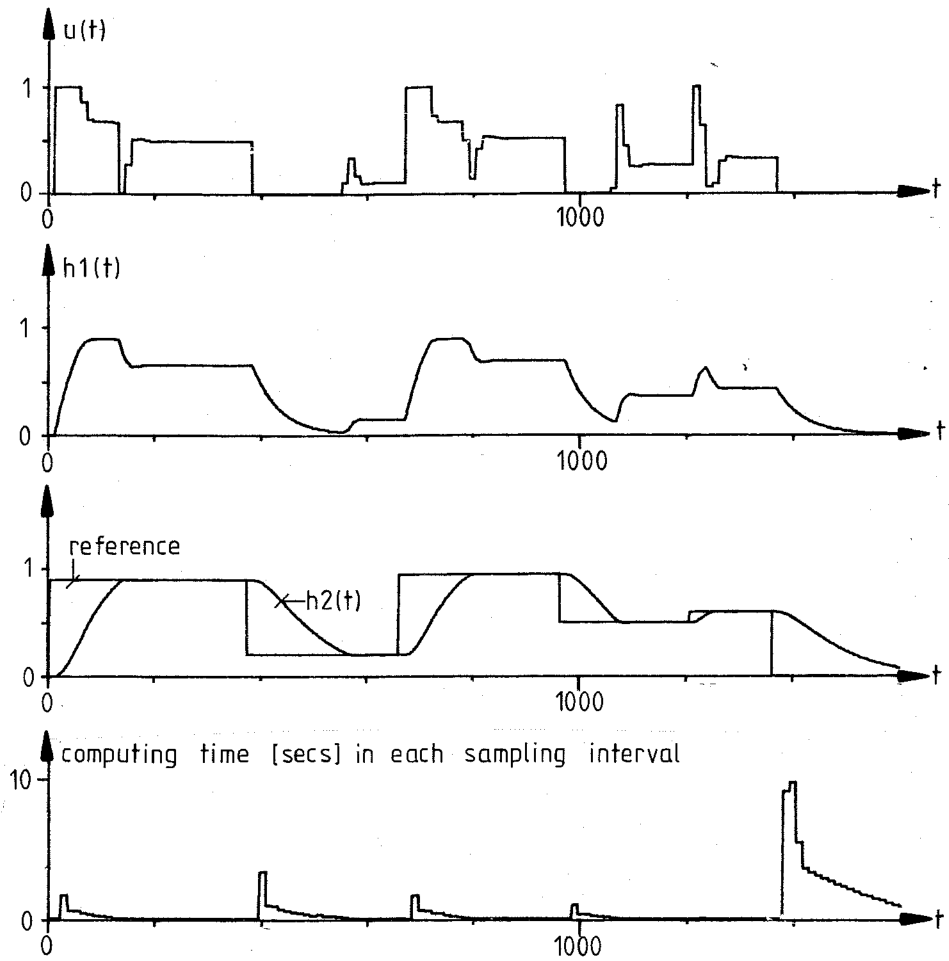

Fig. 6:5. The double tank process: u(t), h1(t), h2ref(t), h2(t), and the computation time in a simulated experiment.

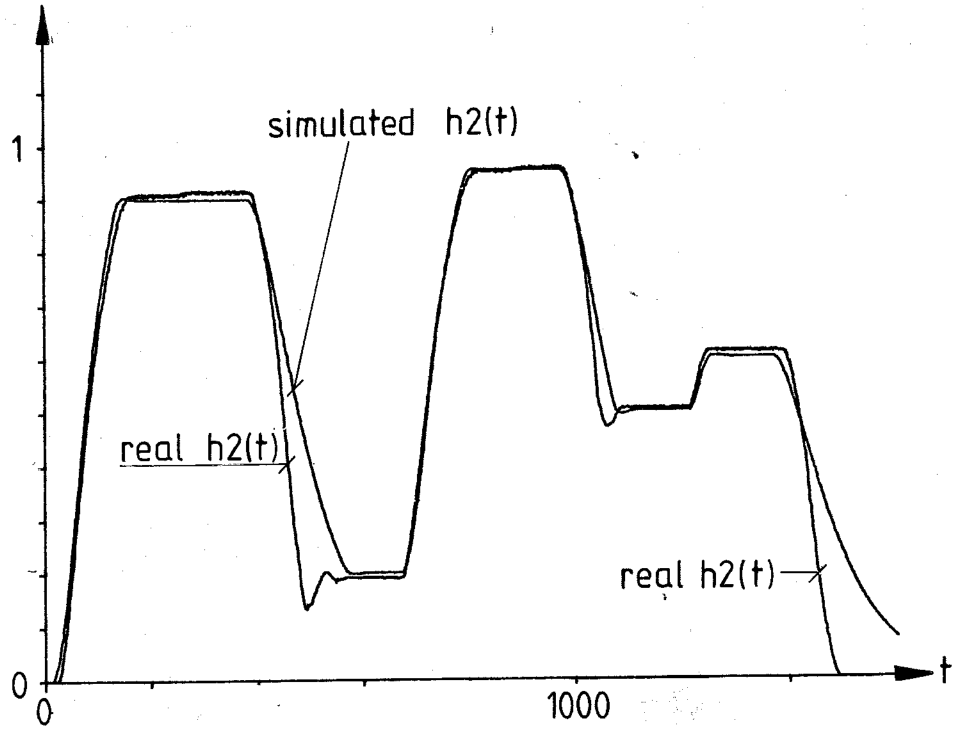

Fig. 6:6. The double tank process: h2(t) in the real life and the simulated experiments.

7. PROPERTIES OF THE LP-REGULATOR

In this chapter some general experience of the proposed LP-regulator will be given. The observations are based on the simulations and experiments performed for this study, and not on analysis. A lot of theoretical work remains to be done to gain a solid knowledge about the regulator.

The LP-regulator is used to control plants that can be modelled linearly with linear state and control constraints. It is easy to assure that the constraints are never violated, by properly choosing weights, bounds and "cordons sanitaires". It is possible to introduce further boundary layers with different penalties, and to penalize the controls. The price you pay is that the LP-problem gets larger and more time consuming, and that you have more parameters to tune.

It is not more difficult to control multi-input systems with the LP-method than single-input systems. This contrasts favourably with other methods for constrained control. It is also easy to introduce feedforward from known disturbances, and model time delays.

The user defines loss functions of the type given in (4.1), (4.5), and (5.2), and chooses the weights. The weights do have physical significance but are sometimes difficult to tune to get the desired time domain behaviour of the system. Clearly, the loss function must include a penalty on the final state. If the state is not penalized _en route_, and if the state and input stay free from the bounds or "cordons sanitaires", the regulator behaves as a dead-beat controller. This typically happens near the origin. In general the dead-beat behaviour is undesirable, in particular when noise affects the system, or when there is a modelling error. To penalize the state _en route_, however, will influence the global control. In general it will not be time optimal. So, there seems to be a conflict between global control, i.e. when the state is far from the origin, and local control.

For certain loss criteria the LP-algorithm may give non-unique solutions. In our simulations this did not seem to adversely influence the control. The controller always stabilized the plant, even when the computation times occasionally were nearly as long as the sampling interval.

It is well known that open loop optimal control strategies are sensitive to model errors. This does not seem to be the case in the feedback version. We performed several simulations with incorrect linear models in addition to the double tank experiment of chapter 6. In all cases the LP-regulator was robust.

We performed a few simulations in which measurement noise and state noise were introduced. The degradation of the performance was not unexpectedly high. No unexpected features arose when a Kalman filter was used.

The computation times on a VAX 11/780 computer are given in the following table. "Steady state" means as above that the optimal time horizon and the solution is known approximately in advance from the previous sampling interval.

			Computation time [s]	
State constraints	System order	No. of inputs	Steady state	Ref. value change
no	2	1	< 0.1	< 0.8
no	5	2	< 0.5	< 1.5
yes	2	1	< 0.3	< 0.8
yes	2	1	< 1.5	4 - 20
yes	5	2	-	< 70

The table gives some typical computation times. At present we have no rules of thumb regarding the connection between system complexity, constraint complexity and computation time. Thus it is necessary to investigate the individual problem.

When we had tuned the test quantities of our LP-routine, there were no numerical difficulties.

8. DISCUSSION

This chapter contains a discussion of a few of the problems with the LP-regulator, and some ideas for further improvements.

In chapter 7 it was mentioned that there may be a conflict between local and global control. It would be favourable to have different controllers for these situations. The LP-regulator is most advantageous for global control, when optimal trajectories will touch the state and input constraints. However, for local control there is an abundance of good regulators, such as PID-regulators, linear quadratic state feedback regulators, etc. We propose that such a controller, rather than a LP-regulator, is used for local control. There are potential advantages with such a choice. The sampling interval can be chosen shorter. Integral action can be achieved in a more natural way. Existing regulators, which people are more used to tune, do not have to be removed.

We want to stress that it is possible to use a LP-regulator for local control, but with a loss function different from the loss function of the global controller, so that the dead-beat behaviour is avoided. It is also possible to introduce integrating action into the local LP-regulator.

When switching between the local and global controllers, hysteresis should be used in order to avoid limit cycles around the switching surface.

At set point changes or when large disturbances act on the system, the correct time horizon is unknown and the computation time might increase beyond the sampling interval. This is a serious problem which might lead to the dilemma that Rasmy (1977) touched upon (see section 3.2). Occasional long computation times, and the subsequent choice of a suboptimal solution, might not be harmful as the experiment in chapter 6 shows. But we would like to warn have different controllers for these situations. The LP-regulator is most advantageous for global control, when optimal trajectories will touch the state and input constraints. However, for local control there is an abundance of good regulators, such as PID-regulators, linear quadratic state feedback regulators, etc. We propose that such a controller, rather than a LP-regulator, is used for local control. There are potential advantages with such a choice. The sampling interval can be chosen shorter. Integral action can be achieved in a more natural way. Existing regulators, which people are more used to tune, do not have to be removed.

We want to stress that it is possible to use a LP-regulator for local control, but with a loss function different from the loss function of the global controller, so that the dead-beat behaviour is avoided. It is also possible to introduce integrating action into the local LP-regulator.

When switching between the local and global controllers, hysteresis should be used in order to avoid limit cycles around the switching surface.

At set point changes or when large disturbances act on the system, the correct time horizon is unknown and the computation time might increase beyond the sampling interval. This is a serious problem which might lead to the dilemma that Rasmy (1977) touched upon (see section 3.2). Occasional long computation times, and the subsequent choice of a suboptimal solution, might not be harmful as the experiment in chapter 6 shows. But we would like to warn against a systematic use of suboptimal solutions in every or almost every sampling interval, as proposed by Chang (1981).

The computation time is obviously the big problem with the LP-regulator. There seem to be four routes to slash it:

* If the large computation times occur only at set point changes, indicate the set point change a short while before it should take place. Then use the spare time during the following sampling intervals, while the regulator still operates in "steady state", to precompute an approximately optimal time horizon and an approximately optimal control trajectory.

* Implement the LP-regulator in a multi-processor configuration, operating on a common data area. The computations for different time horizons can be performed in parallell. One or more processors can handle precomputations.

* Find faster LP-algorithms. As far as we know, there are no algorithms especially geared towards plants with constraints both on inputs and states.

* Wait until faster computers are built (!).

Other improvements would be

* design schemes for weights and other parameters with a stability proof in mind.

* increase the robustness to measurement errors and model errors by including state filters and adaptivity.

9. CONCLUSIONS

In a near future there will appear microcomputers more powerful than the VAX 11/780. appear. It will then be attractive to use the LP-regulator on processes with sampling periods of 10 seconds or more, i.e. for many industrial applications involving flow, heat, and chemical processes, and climate control.

The experience up to date indicates that LP-regulator is a robust and flexible controller for linear systems with or without time delays and with state and control constraints.

10. ACKNOWLEDGEMENT

I am very happy with the cooperation with my advisor, Prof. Per Hagander who is the best possible source of inspiring and unending support, good ideas and a constructively critical view. I have had vivid and fruitful discussions with Prof. Karl Johan Åström, often during Saturday afternoons and late at night. My collegues, especially Sven Erik Mattsson, have provided me with a lot of good advice. Leif Andersson and Rolf Braun helped me setting up the Double Tank experiment. Karl Erik Årzén provided me with his excellent data logger, (Årzén, 1981). Leif Andersson's work to perfect the text editing programs TEVE and REPORT, and his willingness to be consulted about them any time day or night was indispensible to get this report ready. Special thanks to Gunnar Holmberg of the Royal Intitute of Technology, Stockholm, who provided me with PRIMAL and fixed a few remaining bugs in it. Inga, the attractive librarian at the University Library in Lund is remembered for the computer literature search she conducted for me. And, of course, Britt Marie Carlsson's curves are as delicious as never before.

11. REFERENCES

Abdelmalek N.N. (1978): Solutions of minimum time problem and minimum fuel problem for discrete linear admissible control systems. Int. J. Systems Sci., vol. 9, no. 8, 849-855.

Årzén, K.E. (1981): Logger - an interactive program for data logging on PDP-11. Report, Dept. Aut. Contr., Lund Inst. Techn., Lund, Sweden. CODEN: LUTFD2/ (TFRT-5262)/ 1-126/ (1981).

Åström K.J. (1971): Olinjära system (in Swedish). Lund Institute of Technology: TLTH/VBV.

Baba Y. (1980): Choice of sampling period for minimum-time computer control system (in Japanese). Trans. Soc. Instrum. & Control Eng. (Japan), vol. 16, no. 4, 491-497.

Bar-Shalom Y., Tse E. (1976): Concepts and Methods in Stochastic Control. In Control and dynamic systems, Advances in Theory and Applications (ed.: C.T. Leondes), vol. 12. New York: Academic.

Bashein G. (1971): a simplex algorithm for on-line computation of time optimal controls. IEEE Trans. Auto. Control., vol. AC-16, 479-482.

Bryson A.E., Ho Y.C. (1969): Applied Optimal Control. New York:Blaisdell.

Cadzow J.A. (1974): Minimum amplitude control of linear discrete systems. Int. J. Control, vol. 19, 765.

Canon M.D., Cullum C.D., Polak E. (1970): Theory of Optimal Control and Mathematical Programming. New York:McGraw-Hill.

Chang T.S., Seborg D.E. (1981): A linear programming approach for multivariable feedback control with inequality constraints. Proc. 20th IEEE Conf. on Decision and Control Dec. 16-18 1981, San Diego, California. ISSN 0191-2216, vol 1, WP-6, 426.

Dreyfus S. (1964): Some types of optimal control of stochastic systems. SIAM J. Control, vol.2, 120.

Elmqvist H. (1975): SIMNON - An interactive simulation program for nonlinear systems, User's manual. Report 7502, Dept. Auto. Control, Lund Inst. Techn., Lund, Sweden.

Elmqvist H., Mattsson S.E. (1981): A Real-time Kernel for Pascal. Dept. Aut. Contr., Lund Inst. Techn, Lund, Sweden. CODEN: LUTFD2 (TFRT-7231)/ 1-23/ (1981).

Glattfelder A.H. (1974): Regelungsysteme mit Begrenzungen. Munchen:Oldenburg.

Gutman P.-O. (1982): A new design of contrained controllers for linear systems. Part 2 in "Controllers for bilinear and constrained linear systems", Ph.D. dissertation, Dept. Auto. Contr., Lund Inst. of Techn., Lund, Sweden.

Holmberg G. (1981): PRIM2 - ett subrutinpaket som löser LP-problem (in Swedish). UTM-Program nr 50. Uppdragsgruppen för tillämpad matematik. Royal Institute of Technology, Stockholm, Sweden.

Kim B.K., Bien Z. (1981): An efficient algorithm for a time-suboptimal problem for systems with delay in state. Int. J. Control, vol. 33, no. 5, 871-889.

Kolev L.V. (1978): Minimum-fuel minimum-time control of linear discrete systems. Int. J. Control, vol. 27, no. 1, 21-29.

Knudsen J.K.H. (1975): Time-Optimal Computer Control of a Pilot Plant Evaporator. Proc. IFAC 6th World Congress, part 2, 45.3/1-6.

Lack G.N.T., Enns M. (1967): Optimal Control Trajectories with Minimax Objective Functions by Linear Programming. IEEE Trans. Auto. Control, vol. AC-12, 749-752.

Nieman R.E., Fisher D.G. (1973a): Experimental evaluation of time-optimal open-loop control. Trans. Instn Chem. Engrs, vol. 51, 132-140.

Nieman R.E., Fisher D.G. (1973b): Experimental evaluation of optimal multivariable servo control in conjunction with conventional regulatory control. Chemical Engineering Communications, vol. 1, 77-87.

Orchard-Hays W. (1968): Advanced linear programming computing techniques. New York:McGraw-Hill.

Propoi A.I. (1963): Use of LP methods for synthesizing sampled-data automatic systems. Automation and Remote Control, vol. 24, 837.

Propoi A., Zimiu I. (1981): Dynamic Linear Programming Models of Energy, Resource, and Economic Development Systems. Report RR-81-14, IIASA, Laxenburg, Austria.

Rasmy M.E., Hamza M.H. (1975): Minimum-effoert time-optimal control of linear discrete systems. Int. J. Control, vol. 21, no. 2, 293-304.

Rasmy M.E., Hamza M.H. (1977): Subopotimal control of minimum-time linear discrete systems. Int. J. Control, vo. 25, no. 3, 361-373.

Sakawa M., Narutaki R., Suwa T. (1977): Optimal control of linear systems with svereal cost functionals through a multicriteria simplex method. Int. J. Control, vol. 25, no. 6, 901-914.

Shapiro E.Y. (1972): Suboptimal design of minimal-time feedback regulator for lilnear, time-invariant plants. Sc.D. dissertation, Dep. Elec. Eng. Comput. Sci., Columbia University, New York.

Torng H.C. (1964): Optimization of Discrete Control Systems Through Linear Programming. J. Franklin Inst., vol. 278, no. 1, 28-44.

Tracht R. (1980): Ein Regelalgoritmus fur zeitoptimale Abtastsysteme mit Stellgrössenbeschränkungen. Regelungstechnik, vol. 28, no. 10, 339-346.

Weischedel H.R. (1970): A solution to the Discrete Minimum-time Control Problem. J. Optimiz. Theory and Applic., vol. 5, no.2, 81-96.

Wieslander J. (1980): Idpac Commands - User's Guide. Dept. Aut. Contr., Lund Inst. Techn., Lund, Sweden. CODEN: LUTFD2/ (TFRT-3157)/ 1-108/ (1980).

Wittenmark B. (1975): Stochastic adaptive control methods: a survey. Int. J. Control, vol. 21, no. 5, 705.

Zadeh L.A., Whalen B.H. (1962): On optimal control and linear programming. IRE Trans. Auto. Cont., vol. 7, no. 4, 45.

9 PΣ V