Linear Quadratic Control Package Part II - the Discrete Problem

Mårtensson, Krister

1969

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Mårtensson, K. (1969). *Linear Quadratic Control Package Part II - the Discrete Problem*. (Research Reports TFRT-3010). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

LINEAR QUADRATIC CONTROL PACKAGE

PART II - THE DISCRETE PROBLEM [†]

K. Mårtensson

ABSTRACT

In this report we consider the linear quadratic control problem under the restriction that the control variable is constant over the sampling intervals. Algorithms and flow chart for numerical solution are presented. The program can be used to design optimal control systems and to compute optimal filters and predictors for implementation on process control computers.

[†] This work has been supported by the Swedish Board for Technical Development under Contract 68-336-f.

TABLE OF CONTENTS

1. INTRODUCTION

In a previous report {2} we have discussed the solution of the continuous version of linear quadratic optimal control (linear system equations, quadratic cost functional). It was shown that the optimal control is a linear time-varying or, in the asymptotic case, constant feedback from the state variables of the system. The success of this control strategy thus continuously requires complete information about the state of the system, and possibility for the control variable to change according to the control law. However, in many applications at least one of these conditions is violated. Complete information about the state for example, may either be too expensive from an instrumentation point of view or even impossible to get continuously (e.g. chemical reactors). A reconstruction of the states using a Kalman filter then requires off-line digital computations, and if a process control computer is used to implement the optimal control, the estimates will be available only at discrete sampling events. The control variable is then changed only at these sampling events, and thus both conditions mentioned above are violated.

It is then convenient to state the problem in a somewhat different way. Instead of looking for the best continuous control, we shall try to find the control in the class of piecewise constant functions, and it shall be the best in this class in the sense that the criteria is still as small as possible. The solution will then be directly applicable on a process controller and the estimates can be computed between two subsequent sampling intervals.

The problem is stated and solved in section 2. Notice that, despite of the discrete nature of the problem, we have chosen an integral criteria. This will punish the overall behaviour of the system, while a discrete criteria just considers the state of the system at the sampling events. Moreover, very little about the influence of different criteria parameters can be predicted in the discrete case, while at least some simple rules are known in the continuous (Butterworth patterns, etc.). The connection to the continuous case is justified in section 3. In section 4 we consider the numerical solution of the problem. Algorithms and flow chart for a complete program package are presented. Finally, in section 5, some typical examples are solved and briefly discussed.

2. THE DISCRETE CONTROL PROBLEM

Consider the linear time-varying system

$$\frac{d}{dt} x(t) = A(t) \cdot x(t) + B(t) \cdot u(t) \tag{2.1}$$

where $x(t)$ is the n-dimensional state vector, $u(t)$ is the r-dimensional unconstrained control vector, and $A(t)$ and $B(t)$ are nxn respectively nxr matrices. Moreover, $A(t)$ and $B(t)$ are assumed to be piecewise continuous and bounded. Let t_o and $t_f > t_o$ be fixed initial and final times, and let

$$x(t_o) = x_o \tag{2.2}$$

be the initial condition for (2.1). Denote by $V(x_o,t_o;t_f,u)$ the functional

$$V(x_o,t_o;t_f,u) = x^T(t_f)\cdot Q_o \cdot x(t_f) + \int_{t_o}^{t_f} \begin{pmatrix} x(s) \\ u(s) \end{pmatrix}^T \cdot Q(s) \cdot \begin{pmatrix} x(s) \\ u(s) \end{pmatrix} ds \tag{2.3}$$

Q_o and $Q(s)$ are assumed to be nonnegative definite symmetric matrices of order n x n and (n + r) x (n + r), and $Q(s)$ is a composite matrix

$$Q(s) = \begin{pmatrix} Q_{11}(s) & Q_{12}(s) \\ Q_{21}(s) & Q_{22}(s) \end{pmatrix} \tag{2.4}$$

where

$$Q_{21}(s) = Q_{12}^T(s) \tag{2.5}$$

The dimensions of Q_{11}, Q_{12} and Q_{22} are n x n, n x r and r x r, and they are all bounded and piecewise continuous.

The optimal control problem then consists of determining the control function $u(t)$ on the interval $[t_o,t_f]$ in such a way that the nonnegative quantity (2.3) becomes as small as possible. (2.3) will be referred to as the cost functional, and Q_o, Q_{11}, Q_{12} and Q_{22} as the cost or criteria matrices.

Denote by $V^o(x_o,t_o;t_f)$ the minimal value of (2.3) with respect to the class of all piecewise continuous functions defined on $[t_o,t_f]$. Let this class be denoted by U_c. Then

$$V^o(x_o,t_o;t_f) = \min_{u\varepsilon U_c} V(x_o,t_o;t_f,u) \tag{2.6}$$

Under these assumptions the solution of the problem is well known {2}, and the minimal value of (2.3) is a quadratic function of the initial state

$$V^o(x,t;t_f) = x^T \cdot S(t;t_f)x \tag{2.7}$$

where $S(t;t_f)$ is the solution of the Riccati equation with $S(t_f;t_f) = Q_o$. We will now consider the problem under the restriction that $u \varepsilon U_d$, and U_d is the class of all piecewise constant functions defined on $[t_o,t_f]$. Let $\tilde{V}^o(x_o,t_o;t_f)$ be the analogue of (2.6), that is

$$\tilde{V}^o(x_o,t_o;t_f) = \min_{u\varepsilon U_d} V(x_o,t_o;t_f,u) \tag{2.8}$$

Then obviously the following holds

$$V^o(x_o,t_o;t_f) \leq \tilde{V}^o(x_o,t_o;t_f) \tag{2.9}$$

since $U_d \subset U_c$. More detailed results concerning the difference between V^o and $\tilde{V}^o$ are given in section 3.

Now consider the final time t_f as fixed, and let the interval $[t_o,t_f]$ be split into N semiopen intervals $[t_i,t_i+\tau_i)$ in such a way that

$$t_k = t_o + \sum_{i=0}^{k-1} \tau_i \tag{2.10}$$

and

$$t_N = t_f \tag{2.11}$$

The set $\{\tau_i\}$ will in the sequel be called the sampling intervals. To guarantee uniqueness of the control variable we assume that u(t) is constant over the intervals $[t_i,t_i+\tau_i)$. (Fig. 1).

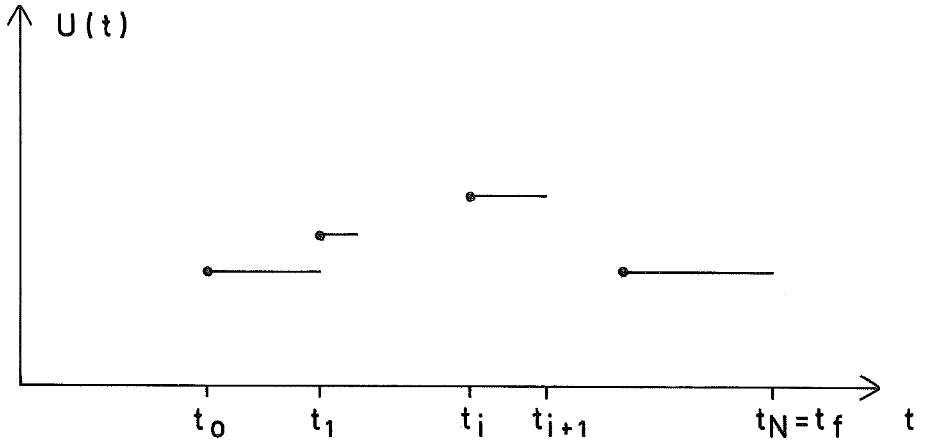

Fig. 1

The solution of (2.1) will then be

$$x(t) = \phi(t,t_k)\; x(t_k) + \Gamma(t,t_k)\; u(t_k) \qquad t_k \leq t \leq t_{k+1} \tag{2.12}$$

where $\phi(t,t_k)$ is the fundamental matrix of (2.1) satisfying

$$\begin{aligned} \frac{d}{dt}\,\phi(t,t_k) &= A(t)\;\phi(t,t_k) \qquad t_k \leq t \leq t_{k+1} \\ \phi(t_k,t_k) &= I \end{aligned} \tag{2.13}$$

and

$$\Gamma(t,t_k) = \int_{t_k}^{t} \phi(t,s)\; B(s)\; ds \qquad t_k \leq t \leq t_{k+1} \tag{2.14}$$

Splitting up the functional (2.3) into a sum of integrals over the sampling intervals we get

$$\tilde{V}(x_o,t_o;t_f,u) = x^T(t_f)Q_o x(t_f) + \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \begin{pmatrix} x(s) \\ u(s) \end{pmatrix}^T \begin{pmatrix} Q_{11}(s) & Q_{12}(s) \\ Q_{21}(s) & Q_{22}(s) \end{pmatrix} \begin{pmatrix} x(s) \\ u(s) \end{pmatrix} ds \tag{2.15}$$

or

$$V(x_o,t_o;t_f,u)=x^T(t_f)Q_o x(t_f)+\sum_{k=0}^{N-1}\int_{t_k}^{t_{k+1}}\{x^T(s)Q_{11}(s)x(s)+$$
$$+u^T(s)Q_{22}(s)u(s)+x^T(s)Q_{12}(s)u(s)+$$
$$+u^T(s)Q_{21}(s)x(s)\}ds \qquad (2.16)$$

Using the relation (2.12) and the fact that $u(t) = u(t_k)$ if $t_k \le t \le t_{k+1}$ the terms under the integral can be made independent of $x(s)$ and $u(s)$.

$$\int_{t_k}^{t_{k+1}} x^T(s)Q_{11}(s)x(s)ds = \int_{t_k}^{t_{k+1}} \{x^T(t_k)\phi^T(s,t_k)Q_{11}(s)\phi(s,t_k)x(t_k) +$$
$$+ x^T(t_k)\phi^T(s,t_k)Q_{11}(s)\Gamma(s,t_k)u(t_k) + u^T(t_k)\Gamma^T(s,t_k)Q_{11}(s)$$
$$\phi(s,t_k)x(t_k) + u^T(t_k)\Gamma^T(s,t_k)Q_{11}(s)\Gamma(s,t_k)u(t_k)\}ds$$
$$(2.17)$$

$$\int_{t_k}^{t_{k+1}} u^T(s)Q_{22}(s)u(s)ds = \int_{t_k}^{t_{k+1}} u^T(t_k)Q_{22}(s)u(t_k)ds \qquad (2.18)$$

$$\int_{t_k}^{t_{k+1}} x^T(s)Q_{12}(s)u(s)ds = \int_{t_k}^{t_{k+1}} \{x^T(t_k)\phi^T(s,t_k)Q_{12}(s)u(t_k) +$$
$$+ u^T(t_k)\Gamma^T(s,t_k)Q_{12}(s)u(t_k)\}ds \qquad (2.19)$$

Finally

$$\int_{t_k}^{t_{k+1}} u^T(s)Q_{21}(s)x(s)ds = \int_{t_k}^{t_{k+1}} x^T(s)Q_{12}(s)u(s)ds \qquad (2.20)$$

since

$$Q_{21}(s) = Q_{12}^T(s)$$

Define the matrices $\tilde{Q}_{11}(t_k)$, $\tilde{Q}_{12}(t_k)$, $\tilde{Q}_{21}(t_k)$ and $\tilde{Q}_{22}(t_k)$ of dimensions n x n, n x r, r x n and r x r in the following way

$$\tilde{Q}_{11}(t_k) = \int_{t_k}^{t_{k+1}} \phi^T(s,t_k)Q_{11}(s)\phi(s,t_k)ds \qquad (2.21)$$

$$\tilde{Q}_{12}(t_k) = \int_{t_k}^{t_{k+1}} \{\phi^T(s,t_k)Q_{11}(s)\Gamma(s,t_k) + \phi^T(s,t_k)Q_{12}(s)\}ds$$
$$(2.22)$$

$$\tilde{Q}_{21}(t_k) = \tilde{Q}_{12}^T(t_k) \tag{2.23}$$

$$\tilde{Q}_{22}(t_k) = \int_{t_k}^{t_{k+1}} \{\Gamma^T(s,t_k)Q_{11}(s)\Gamma(s,t_k) + Q_{21}(s)\Gamma(s,t_k) + \\ + \Gamma^T(s,t_k)Q_{12}(s) + Q_{22}(s)\}ds \tag{2.24}$$

Then from (2.16) to (2.24) follows that

$$\int_{t_k}^{t_{k+1}} \begin{pmatrix} x(s) \\ u(s) \end{pmatrix}^T \begin{pmatrix} Q_{11}(s) & Q_{12}(s) \\ Q_{21}(s) & Q_{22}(s) \end{pmatrix} \begin{pmatrix} x(s) \\ u(s) \end{pmatrix} ds = \\ = \begin{pmatrix} x(t_k) \\ u(t_k) \end{pmatrix}^T \begin{pmatrix} \tilde{Q}_{11}(t_k) & \tilde{Q}_{12}(t_k) \\ \tilde{Q}_{21}(t_k) & \tilde{Q}_{22}(t_k) \end{pmatrix} \begin{pmatrix} x(t_k) \\ u(t_k) \end{pmatrix} \tag{2.25}$$

and consequently

$$\tilde{V}(x_o,t_o;t_f,u) = x^T(t_f)Q_o x(t_f) + \sum_{k=0}^{N-1} \begin{pmatrix} x(t_k) \\ u(t_k) \end{pmatrix}^T \cdot \\ \begin{pmatrix} \tilde{Q}_{11}(t_k) & \tilde{Q}_{12}(t_k) \\ \tilde{Q}_{21}(t_k) & \tilde{Q}_{22}(t_k) \end{pmatrix} \begin{pmatrix} x(t_k) \\ u(t_k) \end{pmatrix} \tag{2.26}$$

The problem of minimizing the continuous functional (2.3) given the system (2.1) under the restriction $u \in U_d$, thus is equivalent to the problem of minimizing the discrete functional (2.26) given the system (2.12) - (2.14).

It is interesting to notice that the crossproduct term $\tilde{Q}_{12}(t_k)$ may well differ from zero in the discrete functional, although it may be zero in the continuous.

The discrete optimal control problem is solved by a straightforward application of linear dynamic programming. From (2.8) follows

$$\tilde{V}^o(x(t_n), t_n;t_f) = \min_{u \in U_d} \tilde{V}(x(t_n),t_n;t_f,u) \tag{2.27}$$

with the boundary condition

$$\tilde{V}^o(x(t_N),t_N;t_f) = x^T(t_f)\; Q_o\; x(t_f) \tag{2.28}$$

Then according to the principle of optimality $\tilde{V}^o$ should satisfy the functional equation

$$\tilde{V}^o(x(t_n),t_n,t_f)=\min_{u\varepsilon U_d}\left\{\begin{pmatrix}x(t_n)\\u(t_n)\end{pmatrix}^T \begin{pmatrix}\tilde{Q}_{11}(t_n) & \tilde{Q}_{12}(t_n)\\ \tilde{Q}_{21}(t_n) & \tilde{Q}_{22}(t_n)\end{pmatrix}\begin{pmatrix}x(t_n)\\u(t_n)\end{pmatrix} + \right.$$

$$\left. + \tilde{V}^o(x(t_{n+1}),\ t_{n+1},t_f)\right\} \tag{2.29}$$

with the boundary condition (2.27).

Since the boundary condition is quadratic in the initial state, we make the approach

$$\tilde{V}^o(x(t_n),t_n,t_f) = x^T(t_n)\ \tilde{S}(t_n,t_f)\ x(t_n) \tag{2.30}$$

where $\tilde{S}(t_n,t_f)$ is a symmetric nonnegative definite matrix of order n x n. Then

$$x^T(t_n)\tilde{S}(t_n,t_f)x(t_n)=\min_{u\varepsilon U_d}\left\{\begin{pmatrix}x(t_n)\\u(t_n)\end{pmatrix}^T \begin{pmatrix}\tilde{Q}_{11}(t_n) & \tilde{Q}_{12}(t_n)\\ \tilde{Q}_{21}(t_n) & \tilde{Q}_{22}(t_n)\end{pmatrix}\begin{pmatrix}x(t_n)\\u(t_n)\end{pmatrix} + \right.$$

$$\left. + x^T(t_{n+1})\tilde{S}(t_{n+1},t_f)x(t_{n+1})\right\} \tag{2.31}$$

which is equivalent to

$$x^T(t_n)\tilde{S}(t_n,t_f)x(t_n)=\min_{u\varepsilon U_d}\ \{x^T(t_n)[\phi^T(t_{n+1},t_n)\tilde{S}(t_{n+1},t_f)\phi(t_{n+1},t_n)+$$

$$+\tilde{Q}_{11}(t_n)]x(t_n)+2\ x^T(t_n)[\tilde{Q}_{12}(t_n)+\phi^T(t_{n+1},t_n)\tilde{S}(t_{n+1},t_f)\Gamma(t_{n+1},t_n)]$$

$$u(t_n)+u^T(t_n)[\Gamma^T(t_{n+1},t_n)\tilde{S}(t_{n+1},t_f)\Gamma(t_{n+1},t_n)+\tilde{Q}_{22}(t_n)]u(t_n)\}$$

$$\tag{2.32}$$

since

$$x(t_{n+1}) = \phi(t_{n+1},t_n)x(t_n) + \Gamma(t_{n+1},t_n)u(t_n) \tag{2.33}$$

To simplify the notations we introduce

$x(n) = x(t_n)$

$u(n) = u(t_n)$

$\phi \quad = \phi(t_{n+1},t_n)$

$\Gamma \quad = \Gamma(t_{n+1},t_n)$

$$\tilde{S}(n) = \tilde{S}(t_n,t_f)$$
$$\tilde{Q}_{11} = \tilde{Q}_{11}(t_n)$$
$$\tilde{Q}_{12} = \tilde{Q}_{12}(t_n)$$
$$\tilde{Q}_{22} = \tilde{Q}_{22}(t_n) \tag{2.34}$$

Completing the square of the right hand of (2.33) yields

$$x^T(n)\tilde{S}(n)x(n)=\min_{u\varepsilon U_d} \{[u(n)+(\Gamma^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{22})^{-1}(\Gamma^T\tilde{S}(n+1)\phi+\tilde{Q}_{12}^T)x(n)]^T$$
$$[\Gamma^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{22}][u(n)+(\Gamma^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{22})^{-1}(\Gamma^T\tilde{S}(n+1)\phi+\tilde{Q}_{12}^T)x(n)] -$$
$$- x^T(n)[\phi^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{12}][\Gamma^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{22}]^{-1}[\Gamma^T\tilde{S}(n+1)\phi+\tilde{Q}_{12}^T]\ x(n) +$$
$$+ x^T(n)[\phi^T\tilde{S}(n+1)\phi + \tilde{Q}_{11}]x(n) \tag{2.35}$$

A regularity condition that guarantees a unique control law is that

$$\Gamma^T\ \tilde{S}(n+1)\Gamma + \tilde{Q}_{22} \tag{2.36}$$

is strictly positive and thus invertible. The minimum of (2.35) is achieved if the control variable u(n) is chosen as a linear feedback from the state variables

$$u(n) = - \tilde{L}(n)\ x(n) \tag{2.37}$$

where $\tilde{L}(n) = \tilde{L}(t_n,t_f)$ is of order r x n and

$$\tilde{L}(n) = [\Gamma^T\tilde{S}(n+1)^T + \tilde{Q}_{22}]^{-1}[\Gamma^T\tilde{S}(n+1)\phi + \tilde{Q}_{12}^T] \tag{2.38}$$

Substituting (2.37) into (2.35) yields a recursive equation for $\tilde{S}(n)$

$$\tilde{S}(n) = \phi^T\tilde{S}(n+1)\phi + \tilde{Q}_{11} - [\phi^T\tilde{S}(n+1)\Gamma + \tilde{Q}_{12}][\Gamma^T\tilde{S}(n+1)\Gamma + \tilde{Q}_{22}]^{-1}$$
$$[\Gamma^T\tilde{S}(n+1)\phi + \tilde{Q}_{12}^T] \tag{2.39}$$

since (2.35) holds for all x(n). The boundary condition is

$$\tilde{S}(n) = Q_o \tag{2.40}$$

More compact (2.35) can be written

$$\tilde{S}(n) = \phi^T\tilde{S}(n+1)\phi+\tilde{Q}_{11}-\tilde{L}^T(n)[\Gamma^T\tilde{S}(n+1)\Gamma+\tilde{Q}_{22}]\tilde{L}(n) \tag{2.41}$$

Notice that $\tilde{S}(n)$ and the feedback parameters $\tilde{L}(n)$ are computed backwards in time, the recursive computation starting at the terminal time t_f. Substituting the control law (2.38) into the system equation gives the optimal system

$$x(t_{n+1}) = [\phi(t_{n+1},t_n) - \Gamma(t_{n+1},t_n)\tilde{L}(t_n)]\ x(t_n) \qquad (2.42)$$

or

$$x(t_{n+1}) = \psi(t_{n+1},t_n)x(t_n) \qquad (2.43)$$

Now let t_f approach infinity. Then under what conditions will $\tilde{S}(t_k,t_f)$ converge towards a stationary solution $\tilde{S}(t_k)$, and when will this result in an asymptotic stable optimal system (2.43)? In the continuous case, that is $u \varepsilon U_c$, sufficient conditions are well known {1}, and then analogue results could be established for the discrete case. Without any proofs we will in the next section make comparisons between the continuous and the discrete case, and then show that the continuous regularity conditions guarantee regular behaviour of the discrete problem, as soon as the sampling interval is small enough.

3. COMPARISONS WITH THE CONTINUOUS CONTROL PROBLEM

The results in this section are due to Åström {4}. Let $W(x,t;t_f)$ be the difference between the minimal values of the functional (2.3) with respect to the class of piecewise continuous and the class of piecewise constant control variables. Then

$$W(x,t;t_f) = \tilde{V}^o(x,t;t_f) - V^o(x,t;t_f) \qquad (3.1)$$

It is clear that $W(x,t;t_f) \geq 0$. Since in both cases the minimal values are quadratic in the initial state we have

$$W(x,t;t_f) = x^T\tilde{S}(t;t_f)x - x^TS(t;t_f)x = x^TT(t;t_f)x \qquad (3.2)$$

where $T(t;t_f)$ is a nonnegative definite symmetric matrix. In {4} it is shown that

$$\lambda_{min}(TS^{-1}) \leq \frac{\tilde{V}^o - V^o}{V^o} \leq \lambda_{max}(TS^{-1}) \leq ||TS^{-1}|| \qquad (3.3)$$

provided that the inverse S^{-1} exists. The quantity $\lambda_{max}(TS^{-1})$ can be considered as the maximum relative increase of the loss function due to the restriction $u \varepsilon U_d$. (The existence of S^{-1} is guaranteed by the controllability and observability criterias. {1}).

From {4} we further have

$$\tilde{V}(x,t;t_f) \rightarrow V(x,t;t_f) \tag{3.4}$$

as

$$\max_i (\tau_i) \rightarrow 0 \tag{3.5}$$

This is equivalent to

$$\tilde{S}(t;t_f) \rightarrow S(t;t_f) \tag{3.6}$$

when

$$\max_i (\tau_i) \rightarrow 0 \tag{3.7}$$

This implies that the minimum value of the discrete loss function can be made arbitrarily close to the continuous loss function if the lengths of the sampling intervals are made small enough. Now consider the time-invariant case, that is, constant parameters A, B, Q_{11}, Q_{12} and Q_{22}, and let the sampling intervals be of equal length h. Then we get a measure of the difference

$$T(t;t_f) = O(h^2) \tag{3.8}$$

or

$$\tilde{S}(t;t_f) = S(t;t_f) + O(h^2) \tag{3.9}$$

This is illustrated by an example in section 5, where the solutions for different sampling intervals are compared with the corresponding continuous solution.

(3.9) together with some more elementary results {4} now makes it possible to give sufficient regularity conditions for the time-invariant discrete case.

Assume that the pair (A,B) is completely controllable and that (C,A), where $Q_{11} = C^TC$, is completely observable. If h is chosen small enough, then $\tilde{S}(t_k,t_f)$ converges towards a positive definite symmetric matrix $\tilde{S}$ as t_f approaches infinity, and the optimal system $\psi(t_{k+1},t_k)$ converges towards an asymptotic stable constant matrix ψ.

The restriction to small sampling intervals is important, since the controllability (observability) may be lost by an improper choice of h. For large values of h, it is then necessary to consider the transformed system and criteria parameters in a manner similar to {1}.

4. NUMERICAL SOLUTION

A complete computer program package for solving the problem has been developed. For simplicity the program is restricted to time-invariant systems and criterias, and the sampling intervals are assumed to be of equal length. The package consists of:

LIOPSAMP	Main program
TRANS	Subroutine
NORM	"
DYNPROG	"
SYMINV	"

Referring to the flow chart shown in fig. 2, we give a description of the different parts of the package.

PROGRAM LIOPSAMP

This is the main program administrating inputs and outputs. It also makes the appropriate calls to the subroutines TRANS and DYNPROG. Input data required:
N-order of the system (max 10).
NU-number of control variables (max 10).
ITYPE - is set 1 if the system and loss function are given in continuous form and a transformation to the discrete form is required. Option exists to apply the dynamic programming routine directly to a discrete system and criteria. In this case ITYPE is set 2.
ITIME - number of sampling events in which $\tilde{S}$ and $\tilde{L}$ are computed.
TSAMP - length of sampling intervals. In the discrete case this is arbitrary.

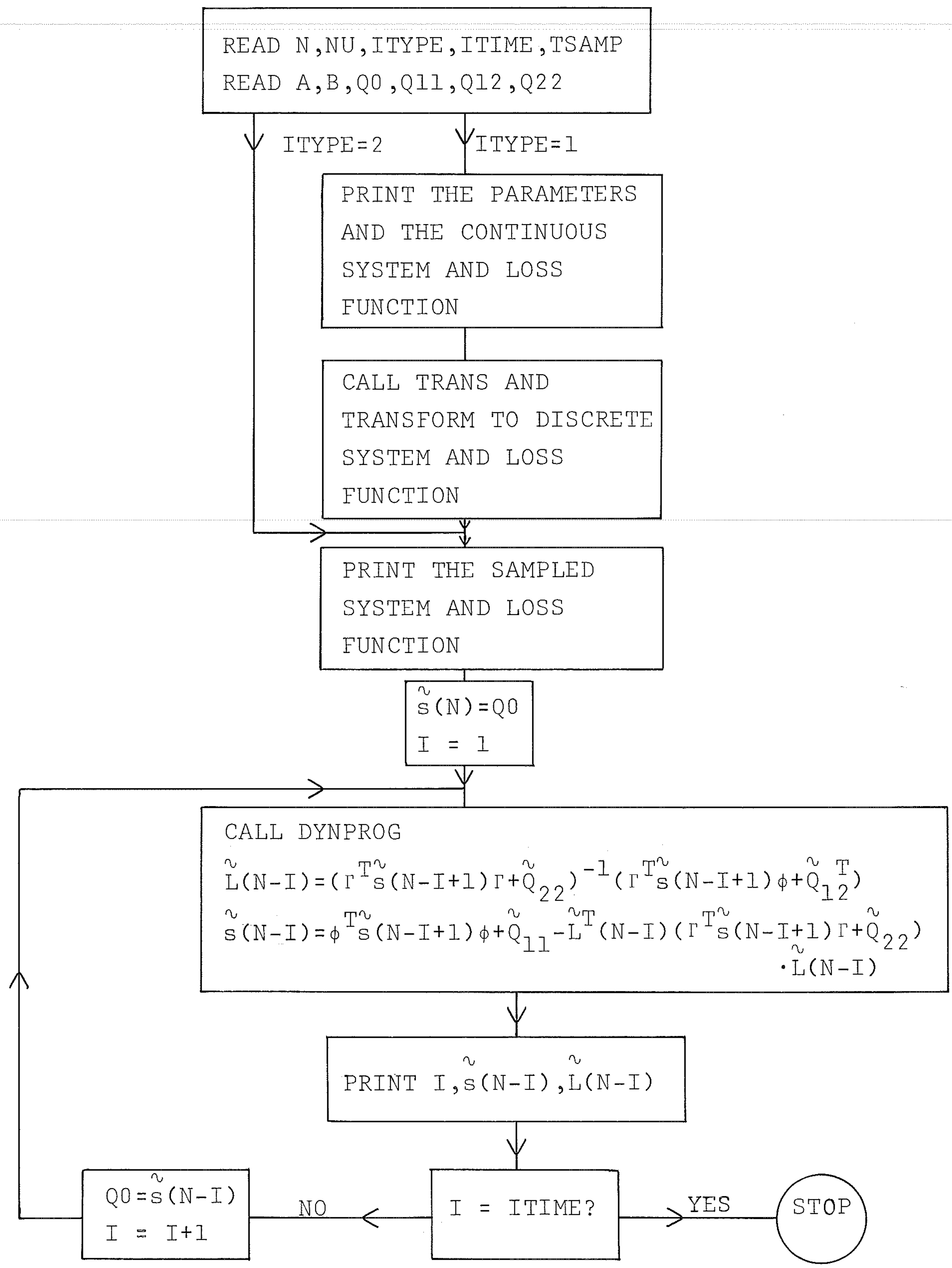

Fig. 2 - Flow chart

A - system matrix of order N x N (ϕ in the discrete case).

B - system matrix of order N x NU (Γ).

Q0, Q11 - criteria matrices of order N x N ($\tilde{Q}0$ and $\tilde{Q}11$).

Q12 - criteria matrix of order N x NU ($\tilde{Q}12$).

Q22 - criteria matrix of order NU x NU ($\tilde{Q}22$).

SUBROUTINE TRANS (N, NU, TSAMP, NOCONV)

This subroutine performs the transformation of the system and criteria matrices. From (2.12) we have

$$x(t) = \phi(t,t_k)x(t_k) + \Gamma(t,t_k)u(t_k) \qquad t_k \leq t \leq t_{k+1} \tag{4.1}$$

where

$$\frac{d}{dt} \phi(t,t_k) = A(t)\phi(t,t_k) \tag{4.2}$$

$$\phi(t_k,t_k) = I$$

and

$$\Gamma(t,t_k) = \int_{t_k}^{t} \phi(t,s)B(s)ds \qquad t_k \leq t \leq t_{k+1} \tag{4.3}$$

With the restriction to time-invariant parameters (4.1) and (4.3) are equal to

$$\phi(t,t_k) = e^{A(t-t_k)} \tag{4.4}$$

and

$$\Gamma(t,t_k) = \int_{t_k}^{t} e^{A(t-s)}ds\ B \tag{4.5}$$

In the dynamic programming routine we are only interested in $\phi(t_{k+1},t_k)$ and $\Gamma(t_{k+1},t_k)$.

Let τ be the length of the sampling intervals, then (4.4) and (4.5) are reduced to

$$\phi(t_{k+1},t_k) = e^{A\tau} \qquad \forall\ k \tag{4.6}$$

$$\Gamma(t_{k+1},t_k) = \int_{0}^{\tau} e^{As}ds\ B \qquad \forall\ k \tag{4.7}$$

(4.6) and (4.7) could be computed by straightforward series expansions since the exponential series is uniformly convergent for all constant matrices A. However, before discussing numerical aspects we shall consider the transformed criteria matrices. To simplify the notations we introduce

$$\phi(t) = e^{At} \tag{4.8}$$

and

$$\Gamma(t) = \int_0^t e^{As} ds \; B \tag{4.9}$$

Then (2.21) - (2.24) are reduced to

$$\tilde{Q}_{11}(t_k) = \int_0^\tau \phi^T(s) Q_{11} \phi(s) ds \qquad \forall \; k \tag{4.10}$$

$$\tilde{Q}_{12}(t_k) = \int_0^\tau \{\phi^T(s) Q_{11} \Gamma(s) + \phi^T(s) Q_{12}\} ds \qquad \forall \; k \tag{4.11}$$

$$\tilde{Q}_{21}(t_k) = \tilde{Q}_{12}^T(t_k) \qquad \forall \; k \tag{4.12}$$

$$\tilde{Q}_{22}(t_k) = \int_0^\tau \{\Gamma^T(s) Q_{11} \Gamma(s) + \Gamma^T(s) Q_{12} + Q_{21} \Gamma(s) + Q_{22}\} ds \quad \forall \; k \tag{4.13}$$

(4.10) - (4.13) could also be computed by series expansions, and thus the transformation problem could be solved by five straightforward but time-consuming expansions. However, we will now prove that just two series are enough.

Introduce the uniformly convergent series

$$T(t) = I \frac{t^2}{2!} + A \frac{t^3}{3!} + A^2 \frac{t^4}{4!} + \ldots = \sum_{n=0}^{\infty} A^n \frac{t^{n+2}}{(n+2)!} \tag{4.14}$$

and

$$Y(t) = \int_0^t \{Is + A\,T(s)\}^T Q_{11} \{Is + A\,T(s)\} ds \tag{4.15}$$

Let

$$P(t) = It + A\,T(t) \tag{4.16}$$

Then

$$\phi(\tau) = I + A\,P(\tau) \tag{4.17}$$

$$\Gamma(\tau) = P(\tau)\,B \tag{4.18}$$

and

$$\frac{d}{dt} P(t) = I + A\,P(t) = \phi(t) \tag{4.19}$$

(4.10) is now reduced to

$$\tilde{Q}_{11}(t_k) = \int_0^\tau \phi^T(s) Q_{11}\,\phi(s)ds = \Big[P^T(s) Q_{11}\,\phi(s) \Big]_0^\tau - \{ \int_0^\tau P^T(s) Q_{11}\,\phi(s)ds \} A \tag{4.20}$$

Introduce

$$S(\tau) = \int_0^\tau P^T(s) Q_{11}\,\phi(s)ds \tag{4.21}$$

Then

$$\tilde{Q}_{11}(t_k) = P^T(\tau) Q_{11}\,\phi(\tau) - S(\tau)\,A \tag{4.22}$$

since $P(0) = 0$. But

$$S(\tau) = \int_0^\tau P^T(s) Q_{11}\,\phi(s)ds = \int_0^\tau \{Is + AT(s)\}^T Q_{11} \{I + As + A^2 T(s)\}ds =$$

$$= [\int_0^\tau \{Is + A\,T(s)\}^T Q_{11} \{Is + A\,T(s)\}ds]\,A +$$

$$+ \int_0^\tau \{Is + A\,T(s)\}^T Q_{11}\,ds \tag{4.23}$$

which is reduced to

$$S(\tau) = Y(\tau)\,A + T^T(\tau) Q_{11} \tag{4.24}$$

since

$$\frac{d}{dt} T(t) = It + A\,T(t) \tag{4.25}$$

Then $\tilde{Q}_{11}(t_k)$ can be computed from the series (4.14) and (4.15). Making use of (4.18) and (4.19), $\tilde{Q}_{12}(t_k)$ can be written

$$\tilde{Q}_{12}(t_k) = \int_0^\tau \phi^T(s) Q_{11}\,\Gamma(s)ds + \int_0^\tau \phi^T(s) Q_{12}\,ds =$$

$$= [\int_0^\tau \phi^T(s) Q_{11}\,P(s)ds]\,B + \int_0^\tau \frac{dP}{ds}^T Q_{12}\,ds \tag{4.26}$$

or

$$\tilde{Q}_{12}(t_k) = S^T(\tau)\,B + P^T(t)\,Q_{12} \tag{4.27}$$

which again only involves the series (4.14) and (4.15). From (4.12) immediately follows

$$\tilde{Q}_{21}(t_k) = \tilde{Q}_{12}^T(t_k) \tag{4.28}$$

For $\tilde{Q}_{22}(t_k)$ we have

$$\tilde{Q}_{22}(t_k)=\int_0^\tau \Gamma^T(s)Q_{11}\ \Gamma(s)ds+\int_0^\tau \Gamma^T(s)Q_{12}ds+\int_0^\tau Q_{21}\ \Gamma(s)ds+\int_0^\tau Q_n ds \tag{4.29}$$

But

$$\Gamma(t) = \frac{d}{dt}\ T(t)\ B \tag{4.30}$$

and then

$$\int_0^\tau \Gamma^T(s)Q_{12}ds = B^T\ T^T(\tau)Q_{12} \tag{4.31}$$

$$\int_0^\tau Q_{21}\ \Gamma(s)ds = (B^T\ T^T(\tau)Q_{12})^T \tag{4.32}$$

The first term of (4.29) is reduced to

$$\int_0^\tau \Gamma^T(s)Q_{11}\ \Gamma(s)ds = B^T \int_0^\tau [Is+AT(s)]^T\ Q_{11}[Is+AT(s)]ds\ B =$$
$$= B^T\ Y(\tau)\ B \tag{4.33}$$

and finally

$$\int_0^\tau Q_{22}ds = Q_{22}\ \tau \tag{4.34}$$

Then

$$\tilde{Q}_{22}(t_k)=B^T\ Y(\tau)B+B^T\ T^T(\tau)Q_{12}+(B^T\ T^T(\tau)Q_{12})^T+Q_{22}\tau \tag{4.35}$$

Summarizing the computations required we have

1. Compute $T(\tau)$ and $Y(\tau)$
2. $P(\tau) = I\tau + A\ T(\tau)$
   $S(\tau) = Y(\tau)A + T^T(\tau)Q_{11}$
3. $\phi(\tau) = I + A\ P(\tau)$
   $\Gamma(\tau) = P(\tau)\ B$
   $\tilde{Q}_{11}(t_k) = P^T(\tau)Q_{11}\ \phi(\tau) - S(\tau)\ A$ (4.36)
   $\tilde{Q}_{12}(t_k) = S^T(\tau)\ B + P^T(\tau)\ Q_{12}$
   $\tilde{Q}_{21}(t_k) = \tilde{Q}_{12}^T(t_k)$
   $\tilde{Q}_{22}(t_k) = B^T\ Y(\tau)B + B^T\ T^T(\tau)Q_{12} + Q_{12}^T\ T(\tau)\ B + Q_{22}\ \tau$

The series expansions are computed in the following way

$$T(t) = I \frac{t^2}{2!} + A \frac{t^3}{3!} + \ldots =$$

$$= t^2 \; \frac{I}{2!} + \frac{(At)}{3!} + \frac{(At)^2}{4!} + \ldots =$$

$$= t^2 \sum_{n=1}^{\infty} T_n \qquad (4.37)$$

where

$$T_1 = \frac{T}{2!}$$

$$T_2 = \frac{(At)}{3!}$$

$$\vdots$$

$$T_n = \frac{(At) \; T_{n-1}}{(n+1)!} \qquad (4.38)$$

Although Y(t) is given in the integral form (4.15), it is easy to show that

$$Y(t) = t^3 \sum_{n=1}^{\infty} Y_n \qquad (4.39)$$

where

$$Y_1 = \frac{1}{3!} (2Q_{11})$$

$$Y_2 = \frac{1}{4!} (3 \; Q_{11} \; (At) + 3 \; (At)^T \; Q_{11})$$

$$Y_3 = \frac{1}{5!} (4 \; Q_{11} \; (At)^2 + 6 \; (At)^T \; Q_{11} \; (At) + 4 \; (At)^{T^2} \; Q_{11})$$

$$Y_n = \frac{1}{(n+2)} (Y_{n-1} \; (At) + (At)^T \; Y_{n-1} + Q_{11} \; T_n + T_n^T \; Q_{11}) \qquad (4.40)$$

The series are truncated when

$$\max \left\{ \frac{||T_k||}{|| \sum_{n=1}^{k-1} T_n ||} , \frac{||Y_k||}{|| \sum_{n=1}^{k-1} Y_n ||} \right\} < 10^{-10} \qquad (4.41)$$

or after 35 terms if this condition is not yet satisfied.

The parameters of TRANS are:

N - order of the system (max 10).

NU - number of control variables (max 10).

TSAMP - length of sampling intervals (τ).

NOCONV - returned 1 if the series expansions have not converged to the desired accuracy (4.41), otherwise 0.

The system and criteria matrices A, B, Q_{11}, Q_{12}, Q_{22} are provided via a common block, and upon return they contain ϕ, Γ, $\tilde{Q}_{11}$, $\tilde{Q}_{12}$ and $\tilde{Q}_{22}$.

SUBROUTINE NORM (A,N,IA,S)

This routine computes the norm used in the criteria (4.41). The norm is chosen as

$$S = \min \{\max_i \sum_{j=1}^{n} |a_{ij}| \ , \ \max_j \sum_{i=1}^{n} |a_{ij}| \ \} \tag{4.42}$$

Parameters:

A - quadratic matrix.

N - order of A.

IA - dimension parameter.

S - resulting norm.

SUBROUTINE DYNPROG (F,G,Q11,Q12,Q22,N,NU,S,UL,IA,IB,IERR)

Given $\tilde{S}(n+1)$ the subroutine computes $\tilde{S}(n)$ and $\tilde{L}(n)$ according to (2.35) and (2.36), that is

$$\tilde{L}(n) = [\Gamma^T\tilde{S}(n+1)\Gamma + \tilde{Q}_{22}]^{-1} \ [\Gamma^T\tilde{S}(n+1)\phi + \tilde{Q}_{12}^T] \tag{4.43}$$

and

$$\tilde{S}(n) = \phi^T\tilde{S}(n+1)\phi + \tilde{Q}_{11} - \tilde{L}^T(n) \ [\Gamma^T\tilde{S}(n+1)\Gamma + \tilde{Q}_{22}] \ \tilde{L}(n) \tag{4.44}$$

For computational reasons a more suitable form of (4.44) is

$$\tilde{S}(n) = \phi^T\tilde{S}(n+1)\phi + \tilde{Q}_{11} - \tilde{L}^T(n) \ [\Gamma^T\tilde{S}(n+1)\phi + \tilde{Q}_{12}^T] \tag{4.45}$$

Parameters:

F,G,Q11,Q12,Q22 - System and criteria matrices $\phi,\Gamma,\tilde{Q}_{11},\tilde{Q}_{12},\tilde{Q}_{22}$.

N - order of the system (max 10).

NU - number of control variables (max 10).

S - When calling DYNPROG S contains $\tilde{S}(n+1)$ upon return S contains $\tilde{S}(n)$.

UL - returned containing $\tilde{L}(n)$.

IA,IB - dimension parameters.

IERR - returned -1 if the inversion has failed.

SUBROUTINE SYMINV (N,IA,IFAIL,A)

A fast routine for the inversion of a symmetric matrix A. {3}. SYMINV is used in subroutine DYNPROG for the computation of $(\Gamma^T\tilde{S}(n+1)\Gamma + \tilde{Q}_{22})^{-1}$.

Parameters:

A - symmetric matrix, returned containing A^{-1}.

N - order of A.

IA - dimension parameter.

IFAIL - returned 0 if the subroutine has executed correctly, 1 if not.

The accuracy of the program varies with the parameters involved. A useful measure of the accuracy is the symmetry of $\tilde{S}(n)$, and therefore no attempts to make $\tilde{S}(n)$ symmetric after each step are made. As soon as the series expansions in subroutine TRANS have converged within 35 terms, the results have shown good accuracy. If the transformations fail, a decrease of the length of the sampling intervals will give faster convergence. All the programming is done in CDC-3600 FORTRAN, and total memory requirement is about 5k.

5. EXAMPLES

A. Double-integral plant.

The system is

$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = u \qquad (5.1)$$

and the cost functional to be minimized is chosen as

$$V = x_1^2(10) + \int_o^{10} \frac{1}{2} u^2(s)ds \qquad (5.3)$$

This corresponds to

$$A=\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad B=\begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad Q_o=\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad Q_{11}=\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad Q_{12}=\begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad Q_{22}=(0.5)$$

The sampling intervals are chosen as $\tau = 1.0$, and then we have the sampling events $t_n = 0,1,\ldots 9,(10)$. Computed feedback parameters and a complete output from the program are given in appendix.

B. Double-integral plant, asymptotic behaviour. Consider the system (5.1) with the criteria

$$V=x^T(t_f)Q_o\ x(t_f)+ \int_o^{t_f} \{x_1^2(s)+2x_1(s)x_2(s)+2x_2^2(s)+u^2(s)\}ds \qquad (5.4)$$

Then

$$A=\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad B=\begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad Q_{11}=\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \quad Q_{12}=\begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad Q_{22}=1.0$$

and Q_o arbitrary. In the continuous case it is well known that independent of Q_o, $S(t;t_f)$ approaches a stationary value S as t_f tends to infinity. Although we have in section III proved the analogue behaviour only for small values of τ, this holds if $\tau = 1.0$, and is illustrated in fig. 3. The elements of the feedback matrix $\tilde{L}(t_n,t_f)$ are converging towards the same stationary value independent of Q_o.

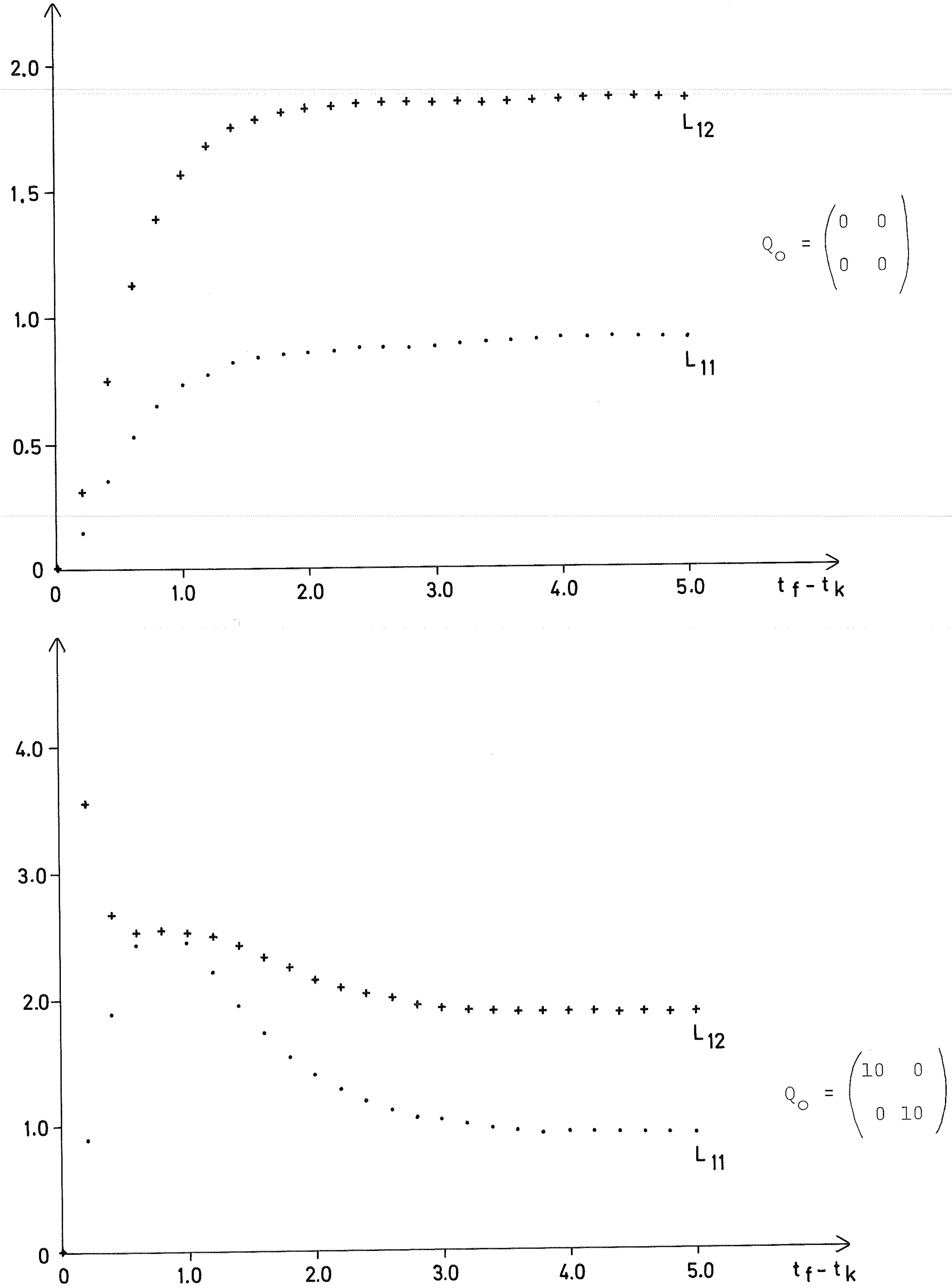

Fig. 3 - Components of the feedback matrix L for various Q_o

C. Double-integral plant, comparison with the continuous case.

This example illustrates the relations stated in section III between the continuous and the discrete problem. We consider the same system and loss functional as in example A. $\tilde{S}(8,10)$ has been computed for τ = 1.0,0.1,0.01, and are presented in fig. 4 together with the continuous solution S(8,10). We notice that $\tilde{S}(8,10)$ equals S(8,10) in about two more digits at each step, which well confirms the $O(h^2)$-behaviour.

$$\tilde{S}(8,10)\ , \quad u \in U_d$$

$$\tau=1.0 \quad \begin{pmatrix} 0.16666\ 66667 & 0.33333\ 33333 \\ 0.33333\ 33333 & 0.66666\ 66666 \end{pmatrix}$$

$$\tau=0.1 \quad \begin{pmatrix} 0.15797\ 78831 & 0.31595\ 57662 \\ 0.31595\ 57662 & 0.63191\ 15324 \end{pmatrix}$$

$$\tau=0.01 \quad \begin{pmatrix} 0.15789\ 55679 & 0.31579\ 11359 \\ 0.31579\ 11359 & 0.63158\ 22720 \end{pmatrix}$$

$$S(8,10)\ , \quad u \in U_c$$

$$\begin{pmatrix} 0.15789\ 47369 & 0.31578\ 94737 \\ 0.31578\ 94737 & 0.63157\ 89474 \end{pmatrix}$$

6. REFERENCES

1. Kalman R.E., Contributions to the Theory of Optimal Control, Bol.Soc.Mat.Mex., vol 5, 1960.

2. Mårtensson K., Linear Quadratic Control Package, Part I - The Continuous Problem, Report 6802, 1968, Lund Institute of Technology, Division of Automatic Control.

3. Rutishauser H., Comm. ACM, Alg.nr 150, 1963.

4. Åström K.J., On the Choice of Sampling Rates in Optimal Linear Systems, Report RJ 243, 1963, IBM San Jose Research Laboratory, California, USA.

```
PRINTOUTS FROM PROGRAM LIOPSAMP

THE CONTINUOUS SYSTEM IS

MATRIX A

  -0.0000000000+000    1.0000000000+000
  -0.0000000000+000   -0.0000000000+000

MATRIX B

  -0.0000000000+000
   1.0000000000+000

MATRIX Q0

   1.0000000000+000   -0.0000000000+000
  -0.0000000000+000   -0.0000000000+000

MATRIX Q11

  -0.0000000000+000   -0.0000000000+000
  -0.0000000000+000   -0.0000000000+000

MATRIX Q12

  -0.0000000000+000
  -0.0000000000+000

MATRIX Q22

   5.0000000000-001

NUMBER OF SAMPLING INTERVALS=  10

LENGTH OF SAMPLING INTERVAL=   1.00000
```

PRINTOUTS FROM PROGRAM LIOPSAMP

THE DISCRETE SYSTEM IS

MATRIX AD

```
   1.0000000000+000     1.0000000000+000
   0.0000000000+000     1.0000000000+000
```

MATRIX BD

```
   5.0000000000-001
   1.0000000000+000
```

MATRIX Q0D

```
   1.0000000000+000    -0.0000000000+000
  -0.0000000000+000    -0.0000000000+000
```

MATRIX Q11D

```
   0.0000000000+000     0.0000000000+000
   0.0000000000+000     0.0000000000+000
```

MATRIX Q12D

```
   0.0000000000+000
   0.0000000000+000
```

MATRIX Q22D

```
   5.0000000000-001
```

NUMBER OF SAMPLING INTERVALS= 10

LENGTH OF SAMPLING INTERVAL=   1.00000

COMPUTED FEEDBACK PARAMETERS

SAMPLING EVENT=T1-   1*TSAMP

COMPUTED S-MATRIX

6.6666666665-001	6.6666666665-001
6.6666666665-001	6.6666666665-001

COMPUTED L-MATRIX(U=-L*X)

6.6666666669-001	6.6666666669-001

SAMPLING EVENT=T1-   2*TSAMP

COMPUTED S-MATRIX

1.6666666666-001	3.3333333331-001
3.3333333331-001	[illegible].6666666663-001

COMPUTED L-MATRIX(U=-L*X)

5.0000000001-001	1.0000000000+000

SAMPLING EVENT=T1-   3*TSAMP

COMPUTED S-MATRIX

5.4054054050-002	1.6216216215-001
1.6216216215-001	4.8648648645-001

COMPUTED L-MATRIX(U=-L*X)

2.7027027027-001	8.1081081082-001

SAMPLING EVENT=T1-   4*TSAMP

COMPUTED S-MATRIX

2.3255813953-002	9.3023255810-002
9.3023255810-002	3.7209302324-001

COMPUTED L-MATRIX(U=-L*X)

1.6279069767-001	6.5116279067-001

SAMPLING EVENT=T1-   5*TSAMP

COMPUTED S-MATRIX

1.1976047904-002	5.9880239518-002
5.9880239520-002	2.9940119759-001

COMPUTED L-MATRIX(U=-L*X)

1.0778443114-001	5.3892215568-001

SAMPLING EVENT=T1- 6*TSAMP

COMPUTED S-MATRIX

6.9444444447-003	4.1666666666-002
4.1666666666-002	2.4999999999-001

COMPUTED L-MATRIX(U=-L*X)

7.6388888886-002	4.5833333333-001

SAMPLING EVENT=T1- 7*TSAMP

COMPUTED S-MATRIX

4.3763676152-003	3.0634573304-002
3.0634573304-002	2.1444201312-001

COMPUTED L-MATRIX(U=-L*X)

5.6892778993-002	3.9824945295-001

SAMPLING EVENT=T1- 8*TSAMP

COMPUTED S-MATRIX

2.9325513201-003	2.3460410557-002
2.3460410557-002	1.8768328445-001

COMPUTED L-MATRIX(U=-L*X)

4.3988269796-002	3.5190615836-001

SAMPLING EVENT=T1- 9*TSAMP

COMPUTED S-MATRIX

2.0597322352-003	1.8537590114-002
1.8537590113-002	1.6683831101-001

COMPUTED L-MATRIX(U=-L*X)

3.5015447993-002	3.1513903192-001

SAMPLING EVENT=T1- 10*TSAMP

COMPUTED S-MATRIX

1.5015015019-003	1.5015015016-002
1.5015015016-002	1.5015015015-001

COMPUTED L-MATRIX(U=-L*X)

2.8528528530-002	2.8528528529-001