Lecture Notes in Control and Information Sciences

Edited by M. Thoma and A. Wyner

89

G. K. H. Pang
A. G. J. MacFarlane

An Expert Systems Approach to Computer-Aided Design of Multivariable Systems

Lecture Notes in
Control and
Information Sciences
Edited by M. Thoma and A. Wyner

89

G. K. H. Pang
A. G. J. MacFarlane

An Expert Systems Approach to Computer-Aided Design of Multivariable Systems

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo

Authors
Dr. Grantham K. H. Pang
Department of Electrical Engineering
University of Waterloo
Waterloo
Ontario N2L 3G1
Canada

Professor Alistair G. J. MacFarlane
Engineering Department
University of Cambridge
Trumpington Street
Cambrigde CB2 1PZ
England

ISBN 3-540-17356-0 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-17356-0 Springer-Verlag New York Heidelberg Berlin

Offsetprinting: Mercedes-Druck, Berlin
Binding: B. Helm, Berlin
2161/3020-543210

ACKNOWLEDGEMENTS

G.K.H. Pang would like to thank his wife Fanny for helping with the preparation of this manuscript, and the Croucher Foundation for financial support. We are grateful to J-M. Boyle for many helpful discussions and detailed comments, and to Cambridge University Engineering Department for the use of its computing facilities.

NOTATION

Unless otherwise stated, the following notation will be adopted:

$a \simeq b$ means a is approximately equal to b

$a := b$ means a is defined to be b or a denotes b

$\mathbb{R}$, $\mathbb{C}$:= field of real and complex numbers, respectively

$\mathbb{C}_+$:= $\{ z \in \mathbb{C} \mid \text{Re } z \geq 0 \}$, the closed right-half complex plane

For $z \in \mathbb{C}$

$| z |$:= modulus (or magnitude) of z

$\angle z$, argz := argument of z

Rez, Imz := real, imaginary part of z, respectively

For $k \in \mathbb{R}$

$\sqrt{k}$:= square root of k ; also written as $k^{1/2}$; unless otherwise stated, the value is taken to be positive

E^k := k dimensional Euclidean space

$\max_k$:= the maxmium with respect to k

$\mathbb{R}(s)$, $\mathbb{C}(s)$:= field of rational functions in s with coefficients in $\mathbb{R}$, $\mathbb{C}$

$O(s^i)$:= a quantity of order s^i (or less)

Let F be any one of $\mathbb{R}$, $\mathbb{C}$, $\mathbb{R}(s)$ or $\mathbb{C}(s)$, then :

$F^{m \times \ell}$:= set of $m \times \ell$ matrices with elements in F

$F^{m \times \ell}(s)$:= set of $m \times \ell$ matrices with elements in F(s)

F^n := vector space of nx1 column vectors with elements in F, over an appropriate field

Let $M \in F^{m \times \ell}$ where F is either $\mathbb{R}$ or $\mathbb{C}$, then :

m_{ij} := (i,j)th entry of M ; we also write $M = (m_{ij})$

$\{ g_i \}$:= set of eigenvalues (spectrum) of M ; also known as characteristic values or gains ; generally, g_i are arranged in descending order of their magnitude

$\{ \sigma_i \}$:= set of singular values of M ; also known as principal gains ; generally, σ_i are arranged in descending order of their magnitude

$\bar{\sigma}(M)$:= maximum singular value of M

$\underline{\sigma}(M)$:= minimum singular value of M

M^t := transpose of M

M^{-1} := inverse of M

M^* := conjugate transpose of M

$|M|$:= (x_{ij}) where $x_{ij} = |m_{ij}|$

argM := (x_{ij}) where $x_{ij} = \arg m_{ij}$

$\| M \|_F$:= $(\sum_{j=1}^{\ell} \sum_{i=1}^{m} |m_{ij}|^2)^{1/2}$, the Frobenius norm of M

$\| M \|_2$:= $\bar{\sigma}(M)$, spectral norm or maximum singular value of M

I_m := mxm unit matrix

Let $u \in F^{\ell}$ where F is either $\mathbb{R}$ or $\mathbb{C}$, then

$\| u \|_2$:= $(u^* \cdot u)^{1/2} = (\sum_{i=1}^{\ell} |u_{ij}|^2)^{1/2}$, the Euclidean vector norm of u

u^t := transpose of the vector u

$\mathrm{diag}\{d_i\}_{i=1}^{n}$:= nxn diagonal matrix with $d_1, \ldots, d_n$ along the diagonal ; also written as $\mathrm{diag}\{d_1, \ldots, d_n\}$ or $\mathrm{diag}\{d_i\}$

Let $A \in \mathbb{R}^{nxn}$, $B \in \mathbb{R}^{nx\ell}$, $C \in \mathbb{R}^{mxn}$, $D \in \mathbb{R}^{mxl}$ and s be the frequency variable ($s \in \mathbb{C}$), then :

G(s) := $C(sI_n - A)^{-1}B + D$, the plant open-loop gain (transfer function) matrix

Also, let g be the gain variable ($g \in \mathbb{C}$) and $\ell = m$, then:

S(g) := $B(gI_m - D)^{-1}C + A$, the closed-loop frequency matrix

Let $\Omega \subset \mathbb{C}$ and $G(s) \in \mathbb{R}(s)^{mx\ell}$, then:

#SMP[G(s),Ω] := number of Smith-McMillan poles of G(s) in Ω

Let ς be a (finite number of) closed curve(s) in $\mathbb{C}$, then:

#E(ς,a) := number of encirclements of ς around the point a ; anti-clockwise encirclements are taken as positive

List of Symbols:

0	zero; zero vector; zero matrix
i	integer
j	$\sqrt{-1}$; integer
ω	angular frequency
D_{NYQ}	Nyquist D-contour ; Section 3.2
MS(G)	measure of skewness, a normality indicator ($G \in \mathbb{C}^{mxm}$); Section 3.4.4
MS(k)	MS(G(jk)) where $k \in \mathbb{R}$, measure of skewness of G(jk); Section 6.6
$\kappa(G)$	spectral condition number ($G \in \mathbb{C}^{mxm}$); Section 3.4.5
cond(g_i)	condition number for an eigenvalue g_i ; Section 3.4.5
ρ_i	gain ratio; Section 3.4.8
==>,<==	implies, is implied by
□	marks the end of a proof

List of Abbreviations:

AI	Artificial Intelligence; Chapter 2
AIRC	Aircraft Dynamics Model; Section 7.4.3
AUTO	Automobile Gas Turbine Model; Section 6.9.1
CACSD	Computer-Aided Control System Design; Section 2.2
CS, CSi	Misalignment Angle, i^{th} branch of; Section 3.4.7
CVD	Characteristic Value Decomposition; Section 3.2
E, Ei	Eigenloci, Characteristic Gain Loci, i^{th} branch of; Fig. 5.1.1
FLOW	Flow-box Model; Section 5.5.3
FOO	Full-Order Observer; Fig. 8.1
GHEL	Helicopter Model; Section 6.9.2
GROC	Rocket Engine Model; Section 5.5.4

HF	High Frequency; Section 8.1
HFS	High Frequency Sub-controller; Section 5.2.1
IF	Intermediate Frequency; Section 8.1
KBF	Kalman-Bucy Filter; Chapter 2
KEE	Knowledge Engineering Environment; Section 8.8
LF	Low Frequency; Section 8.1
LFS	Low Frequency Sub-controller; Section 5.2.2
LHP	Left-Half Plane; Section 7.2
LQR	Linear Quadratic Regulator; Chapter 2
LTR	Loop Transfer Recovery; Chapter 2
MAID	Multivariable Analytical & Interactive Design; Section 8.4.1
MIMO	Multi-Input, Multi-Output; Section 4.2
NSRE	Non-Square Chemical Reactor Model; Section 6.10.2.1
OBC	Observer-Based Controller; Section 7.2
P, Pi	Principal Gain Loci, i^{th} branch of; Fig. 5.1.1
P+I	Proportional plus Integral; Section 5.4
REAC	Chemical Reactor Model; Section 5.5.5
RFA	Reverse Frame Approximation; Section 6.1
RFAT	Reverse Frame Alignment Technique; Section 8.1
ROO	Reduced-Order Observer; Fig. 8.1
SDT	Simple Design Technique; Section 5.2
SISO	Single-Input, Single-Output; Chapter 2
SVD	Singular Value Decomposition; Section 3.3
STD	Schur Triangular Decomposition; Section 3.4
TGEN	Turbo-Generator Model; Section 6.9.3
w.r.t.	with respect to
iff	if and only if

<u>Units</u>:

m	meter
N	newton
rad	radian
s	second
rev	revolution ($2 \cdot \pi$ radians)
min	minute (60 seconds)
kN	kilonewton (1000 N)

CONTENTS

Tables and Figures are placed at the end of each chapter except Fig. 6.12.

CHAPTER ONE

INTRODUCTION

The theory and codified practice of automatic control is an organised body of shareable knowledge, and the importance of developing appropriate interactive computing environments lies largely in making such a specialised body of knowledge easily usable and easily accessible, and therefore easily shareable. Expert and knowledge-based systems have a key role to play in the creation of such environments. The work presented here is concerned with the investigation of expert system techniques for the design of linear multivariable feedback control systems. It is important that the procedures used by such an expert system to manipulate models and their attributes are formulated in terms of a set of individual functions which a designer can cause to be executed on the computer. Only in such circumstances will one be able to formulate any high-level machine-based procedure which would "explain" its actions. For the same reason the procedures used by the machine must be coherent with the principles in terms of which the man thinks about the tasks which are being carried out. In order to achieve these key attributes of referential transparency and coherence, the analysis and design techniques presented here are based on an appropriate generalisation of classical feedback methods. This enables a comprehensive and accurate representation of the behaviour of a multivariable feedback system to be given in terms of a basic set of graphical indicators.

1.1 The Interactive Design Process

The relationship between man and machine in the interactive design process is summarised in Fig. 1.1. We consider data passed from machine to man in terms of indicators and data passed from man to machine in terms of drivers. The man works in terms of a high level conceptual framework and accesses in the machine a powerful manipulative framework. The basic task in creating a satisfactory interactive computing system is to get these two frameworks to mesh together satisfactorily via an appropriate set of indicators and drivers.

It is important to realise, as illustrated by Fig. 1.2, that design is a feedback process, and that in general both the object being created and the specification against which it is being manipulated are being iteratively adjusted in a feedback cycle of dependence as the design proceeds. Design can also be described as a process of instantiation: the progessive generation of a specific fully defined object from an initial incomplete general description. In creating a specific instance of the general class of object desired, the designer is grappling with both uncertainty and complexity, and it is for coping with these twin sources of difficulty that the interactive man/machine combination is well suited: the man to handle uncertainty and the machine to handle complexity. In seeking to define the relative roles of man and machine one must start from a consideration of their strengths and weaknesses in respect of the tasks involved. The man has as strengths:

- the ability to abstract, simplify and conceptualise
- the ability to handle incomplete and ill-defined descriptions
- experience and common sense
- adaptability and flexibility
- skills in pattern-recognition and association.

He has as weaknesses:

- short-term memory limitations
- slowness in executing complex procedures
- tendency to fatigue and distraction

- varying responses to similar stimuli
- inability to handle many disparate activities at the same time
- difficulties in long-term memory retrieval.

The machine has as strengths:

- speed and reliability
- extensive and accurate short-term and long-term memory
- indifference to fatigue
- predicability of response
- ability to handle large amounts of data and to perform a number of unrelated tasks simultaneously
- ability to accurately execute extremely complex formally-specified procedures.

It has as weaknesses:

- inability to generalise
- no conceptual level and no common sense
- inability to disambiguate and to handle uncertainty
- lack of flexibility and adaptability.

As has already been emphasised, design is a feedback process and this feedback is critically important in progressively stripping away the uncertainty in the original design specification. In the following sub-section design methods are considered in three categories associated with different amounts of initial uncertainty about the behaviour of the objects being handled: analytical, procedural and experimental. Although the requirements are markedly different in the three cases, the same general principles apply. The man sets and refines goals, argues from general principles in terms of abstract concepts, and handles ambiguity, conflict of objectives and uncertainties in description and performance. The machine evaluates functions, executes complex procedures, searches through complex data sets, generates and manipulates indicators, and accepts and acts on drivers.

When developing an interactive computing environment, we have to take proper account of the man as well as the machine. In discussing this it is useful to talk in terms of principles and procedures. Principles are the organisers of high-level declarative knowledge, and procedures are the implementors of low-level imperative knowledge; a man thinks in terms of general principles, and a machine functions in terms of formally specified procedures. In the interactive computing context we have to handle both formal and informal knowledge, and also declarative and imperative knowledge, and somehow we have to make them all fit together in an effective and efficient way, as illustrated by Fig. 1.3.

The basic problem of an automatic control system designer is to create or modify a given dynamical system so that it has a specified behaviour or set of attributes. In doing so he wants to use the interactive computing environment to:

- handle formal declarative knowledge by evaluating for him the behaviour and attributes of any given dynamical model
- handle formal imperative knowledge by executing appropriate sequences of procedures in order to attain specified objectives
- handle informal declarative knowledge in the form of textual descriptions of background theory, codes of practice, design data bases etc.
- handle informal imperative knowledge in the form of design guidelines, rules of practice, mandatory design requirements, etc.

To do all these satisfactorily will require a wide range of software, display