MODERN CONTROL SYSTEMS

A Manual of Design Methods

John A. Borrie

MODERN CONTROL SYSTEMS: A Manual of Design Methods

JOHN A. BORRIE

PHI Prentice/Hall International

Englewood Cliffs, N.J. London Mexico New Delhi
Rio de Janeiro Singapore Sydney Tokyo Toronto Wellington

Library of Congress Cataloging-in-Publication Data

Borrie, John A., 1937–
Modern control systems.

Bibliography: p.
Includes index.
1. Automatic control. 2. Control theory. I. Title.
TJ213.B6 1986 629.8 85–19270

ISBN 0-13-590290-8

British Library Cataloguing in Publication Data

Borrie, John A.
Modern control system: a manual of design methods.
1. Control theory
I. Title
629.8'312 QA402.3

ISBN 0-13-590290-8
ISBN 0-13-590282-7 Pbk

Prentice-Hall Inc., *Englewood Cliffs, New Jersey*
Prentice-Hall International (UK) Ltd, *London*
Prentice-Hall of Australia Pty Ltd, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanomericana S.A., *Mexico*
Prentice-Hall of India Private Ltd, *New Delhi*
Prentice-Hall of Japan Inc., *Tokyo*
Prentice-Hall of Southeast Asia Pte Ltd, *Singapore*
Editora Prentice-Hall do Brasil Ltda, *Rio de Janeiro*
Whitehall Books Ltd, Wellington, *New Zealand*

Printed and bound in Great Britain for
Prentice-Hall International (UK) Ltd,
66 Wood Lane End, Hemel Hempstead, Hertfordshire, HP2 4RG
by A. Wheaton and Company Ltd, Exeter

1 2 3 4 5 90 89 88 87 86

ISBN 0-13-590290-8
ISBN 0-13-590282-7 PBK

MODERN CONTROL SYSTEMS:
A Manual of Design Methods

PREFACE

This book contains an ordered presentation of practical modern control engineering techniques with explanations, formulas, and examples, but without mathematical proofs. Many detailed suggestions are made for the construction of computer aided design (CAD) algorithms suited to readily available microcomputers supporting BASIC. At the end of each chapter a limited but carefully selected bibliography helps the reader to explore further. Continuous and discrete time systems are given equal emphasis.

Chapter 1 sets the mathematical background with topics such as differential and difference equations, Laplace, *z*, Fourier transforms, and stochastic system definitions. It includes CAD algorithm designs which are useful in themselves and as subroutines for larger programs. Chapter 2 deals with 'classical' techniques for continuous and discrete time systems. Chapters 3 and 4 describe modern control ideas for linear systems including pole shifting, state estimation, stochastic systems, and Kalman filters. Chapter 5 is devoted to computing methods for function optimization and system identification since these are keys to the future development of this subject. The appendixes contain basic definitions, methods, and algorithms.

This material, which has been warmly welcomed by many short-course students, is suited to professional engineers, postgraduates, and advanced undergraduates. It is not intended as a first introduction to control engineering.

The author is grateful to the considerable number of Cranfield students who have investigated the methods and algorithms outlined in this book. He is also indebted to the staff members who have developed the laboratory experiments. These have provided valuable insight without being unduly complex, and can perhaps be copied by the interested reader.

J. A. B.

MODERN CONTROL SYSTEMS:
A Manual of Design Methods

CONTENTS

1 BASIC DEFINITIONS AND MATHEMATICAL TECHNIQUES

1.1 INTRODUCTION

Some basic definitions and mathematical techniques are set out in this chapter in a compact form, together with suggestions for the design of CAD algorithms. Topics covered include linear differential and difference equations, Laplace, z, and Fourier transforms and basic stochastic system definitions. The reference books listed at the end provide thorough introductions to these topics from a fairly elementary level.

1.2 ORDINARY DIFFERENTIAL EQUATIONS

The behavior of many dynamical systems can be modeled by one or more differential equations of the form:

$$F(y, y^{(1)}, y^{(2)}, \ldots, y^{(m)}, t) = 0 \tag{1.1}$$

where t represents time, y represents some aspect of the system, typically its output, and

$$y^{(1)} = \frac{dy}{dt}; \quad y^{(2)} = \frac{d^2y}{dt^2}; \quad \ldots \quad ; \quad y^{(m)} = \frac{d^my}{dt^m}.$$

Equation (1.1) is mth order (the highest order of derivative) and ordinary (only ordinary derivatives involved).

The general solution, i.e. the function $y(t)$ which satisfies Eq. (1.1), is of the form:

$$y(t) = y(t, c_0, c_1, \ldots, c_{m-1})$$

where $c_0, c_1, \ldots, c_{m-1}$ are constants which can usually be determined if values of $y, y^{(1)}, y^{(2)}, \ldots, y^{(m-1)}$ are known for some specific value of t, say t_0. Typically, 'initial' values $y(0), y^{(1)}(0), \ldots, y^{(m-1)}(0)$ are known.

EXAMPLE

Consider the first-order differential equation:

$$\frac{dy}{dt} + 2yt - e^{-t^2} = 0. \tag{1.2}$$

This has the general solution:

$$y = e^{-t^2}(t + c_0)$$

where c_0 is a constant.

Given the 'initial' value $y(0) = 1$, it is easy to show by substitution that the solution is:

$$y = e^{-t^2}(t + 1)$$

While some useful types of differential equation do have analytic solutions (ref. 1), numerical solutions can very often be found for these and less tractable cases. This topic is of interest here.

If Eq. (1.1) can be rewritten with the highest-order derivatives on the left-hand side (LHS) – and it usually can – it can be recast as two equations, one first and one $(m-1)$th order. Repeating this process, Eq. (1.1) can finally be cast as a set of m first-order equations which in vector format (App. A.1) is:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \tag{1.3}$$

with initial conditions:

$$\mathbf{x}(0) = \mathbf{x}_0$$

where

$$\mathbf{x} = (x_1 x_2 x_3 \ldots x_m)^{\mathrm{T}}.$$

EXAMPLE

Consider the third-order differential equation:

$$\frac{d^3y}{dt^3} + 9\frac{d^2y}{dt^2} + 26\frac{dy}{dt} + 24y - 1 = 0 \tag{1.4}$$

with initial conditions:

$$y(0) = 4, \quad y^{(1)}(0) = 3, \quad y^{(2)}(0) = 2.$$

By setting $x_1 = y$, rewriting Eq. (1.4) with d^3x_1/dt^3 on the LHS and substituting $x_2 = dx_1/dt$, two equations are generated, and by repeating the process with $x_3 = dx_2/dt$, Eq. (1.4) is recast:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_3 \\ -9x_3 - 26x_2 - 24x_1 + 1 \end{bmatrix}$$

$$y = x_1$$

with initial conditions:

$$\mathbf{x}(0) = \begin{bmatrix} 4 \\ 3 \\ 2 \end{bmatrix}.$$

This is in the form of Eq. (1.3)

Once a differential equation, or set of equations, has been cast in this form, it can be fed to a fairly simple CAD algorithm to yield a numerical solution.

CAD Facility

Equations in the form of Eq. (1.3) with $\mathbf{x}(0) = \mathbf{x}_0$ can be solved numerically; i.e. the value $\mathbf{x}(nh)$, $n = 0, 1, 2, \ldots$, h a calculation interval, can be found by several well-known methods.

An interactive CAD algorithm is shown in the flow diagram of Fig. 1.1. The operation of the algorithm is as follows.

BLOCK 1

The differential equation is input in the form of Eq. (1.3) or in a manner allowing easy conversion to this format. It is displayed and corrected if necessary.

BLOCK 2

The calculation interval h, a printout ratio, R, and the total number of iterations required, N, are input.

BLOCK 3

$\mathbf{x}(nh)$, $n = 0, 1, 2, \ldots, N$, are calculated. Two types of mathematically stable method are outlined here.

(a) *Runge–Kutta Methods* (*refs.* 5, 7). These are based on the Taylor expansion:

$$\mathbf{x}((n+1)h) = \mathbf{x}(nh) + h\mathbf{x}^{(1)}(nh) + \frac{h^2}{2!}\mathbf{x}^{(2)}(nh) + \frac{h^3}{3!}\mathbf{x}^{(3)}(nh) + \cdots \quad (1.5)$$

Commonly, four terms in this series are used to derive the fourth-order Runge–Kutta algorithm:

Calculate:

$$\mathbf{k}_1 = h\mathbf{f}(\mathbf{x}(nh), nh)$$

$$\mathbf{k}_2 = h\mathbf{f}(\mathbf{x}(nh) + \tfrac{1}{2}\mathbf{k}_1, nh + \tfrac{1}{2}h)$$

$$\mathbf{k}_3 = h\mathbf{f}(\mathbf{x}(nh) + \tfrac{1}{2}\mathbf{k}_2, nh + \tfrac{1}{2}h)$$

$$\mathbf{k}_4 = h\mathbf{f}(\mathbf{x}(nh) + \mathbf{k}_3, nh + h).$$

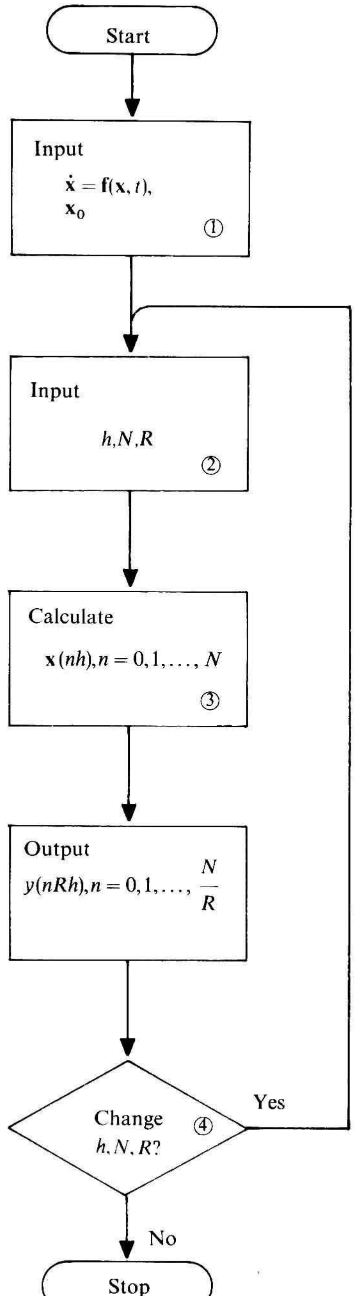

Fig. 1.1 CAD program for solving differential equations.

Then:

$$\mathbf{x}((n+1)h) = \mathbf{x}(nh) + \tfrac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4).$$

Higher accuracy, at the cost of added complexity, can be achieved using a fifth-order Runge–Kutta algorithm (ref. 7). For simple cases, only two terms need be considered to yield the second-order algorithm:

Calculate:

$$\mathbf{k}_1 = h\mathbf{f}(\mathbf{x}(nh), nh)$$

$$\mathbf{k}_2 = h\mathbf{f}(\mathbf{x}(nh) + \tfrac{1}{2}\mathbf{k}_1, nh + \tfrac{1}{2}h).$$

Then:

$$\mathbf{x}((n+1)h) = \mathbf{x}(nh) + \mathbf{k}_2.$$

In each of these algorithms, since $\mathbf{x}((n+1)h)$ can be found from $\mathbf{x}(nh)$, the starting data $\mathbf{x}(0)$ are sufficient to allow the algorithm to proceed.

The Runge–Kutta method yields estimates of calculation errors only with some difficulty (ref. 7). Predictor–corrector methods are better in this respect, and one of these is outlined next.

(b) *Predictor–Corrector Methods.* Perhaps the simplest such method, due to Adams–Moulton (ref. 5), uses the information $\mathbf{x}(nh)$, $\mathbf{x}((n+1)h)$, to predict $\mathbf{x}((n+2)h)$ by linear extrapolation.

$$\bar{\mathbf{x}}(n+2)h = \mathbf{x}(nh) + \frac{h}{2}\{3\mathbf{f}[\mathbf{x}((n+1)h, (n+1)h] - \mathbf{f}(\mathbf{x}(nh), nh)\}.$$

This is then used in the correction formula:

$$\mathbf{x}((n+2)h) = \mathbf{x}((n+1)h) + \frac{h}{2}\{\mathbf{f}[\mathbf{x}((n+1)h), (n+1)h] + \mathbf{f}[\bar{\mathbf{x}}((n+2)h), (n+2)h)]\}.$$

The second equation can be used a number of times to improve $\mathbf{x}((n+2)h)$ until a stable value is obtained.

More sophisticated algorithms of this kind are based on nonlinear extrapolation formulas using three or more initial known values, typically $\mathbf{x}(0)$, $\mathbf{x}(h)$, $\mathbf{x}(2h)$. The initial information required by such algorithms is usually generated by Runge–Kutta methods.

BLOCK 4

The calculation interval may be shortened after a run, and the process repeated. If compatible results are obtained, the solution may be judged satisfactory and the longer interval selected for further runs. Otherwise the process is repeated.