TP
155.75
.M67
1989

ROBUST PROCESS CONTROL

AERO

Manfred Morari
Chemical Engineering
California Institute of Technology

Evanghelos Zafiriou
Department of Chemical and Nuclear Engineering
and Systems Research Center
University of Maryland



Prentice Hall
Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging-in-Publication Data

Morari, Manfred.
Robust process control / Manfred Morari, Evanghelos Zafiriou.
p. cm.
Bibliography: p.
Includes index.
ISBN 0-13-782153-0
I. Chemical process control. I. Zafiriou, Evanghelos.
II. Title.
TP155.75.M67 1989 88-22928
660.2'81—dc19 CIP

Cover design: Photo Plus Art
Manufacturing buyer: Mary Ann Gloriande



© 1989 by Prentice-Hall, Inc.
A Division of Simon & Schuster
Englewood Cliffs, New Jersey 07632

The fonts in this book are Computer Modern Roman, set using Leslie Lamport's LaTeX document preparation facility, with the help of Jan Owen. Some of the figures were prepared by Glenn C. Smith on an Apple Macintosh with the program MacDraw.

TeX is a trademark of the American Mathematical Society.
Macintosh and MacDraw are trademarks of Apple Computer, Inc.

All rights reserved. No part of this book may be
reproduced, in any form or by any means,
without permission in writing from the publisher.

Printed in the United States of America
10 9 8 7 6 5 4 3 2

ISBN 0-13-782153-0

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte. Ltd., *Singapore*
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

In memoriam

CONSTANTIN G. ECONOMOU

1958 – 1986

Contents

PREFACE xv

NOMENCLATURE xix

1 INTRODUCTION 1
- 1.1 The Evolution of Control Theory 1
- 1.2 Controller Parametrization: The IMC Structure 3
- 1.3 Robustness 4
- 1.4 Scope of Book 4
- 1.5 Some Hints for the Reader 5

Part I: CONTINUOUS SINGLE-INPUT SINGLE-OUTPUT SYSTEMS 9

2 FUNDAMENTALS OF SISO FEEDBACK CONTROL 11
- 2.1 Definitions 11
- 2.2 Formulation of Control Problem 13
  - 2.2.1 Process Model 14
  - 2.2.2 Model Uncertainty Description 15
  - 2.2.3 Input Specification 19
  - 2.2.4 Control Objectives 21
- 2.3 Internal Stability 22
- 2.4 Nominal Performance 23
  - 2.4.1 Sensitivity and Complementary Sensitivity Function 24
  - 2.4.2 Two-Degree-of-Freedom Controller 25
  - 2.4.3 Asymptotic Properties of the Closed-Loop Response (System Type) 27
  - 2.4.4 Linear Quadratic (H_2-) Optimal Control 28
  - 2.4.5 H_∞-Optimal Control 29
- 2.5 Robust Stability 31
- 2.6 Robust Performance 34
  - 2.6.1 H_2 Performance Objective 34
  - 2.6.2 H_∞ Performance Objective 35

2.7 Summary 36
2.8 References 38

3 THE SISO IMC STRUCTURE FOR STABLE SYSTEMS 39
3.1 IMC Structure 39
3.2 Stability Conditions for IMC 41
3.2.1 Internal Stability 41
3.2.2 Relationship with "Classic Feedback" 42
3.2.3 Implementation: Classic Feedback Versus IMC 44
3.3 Performance of IMC 45
3.3.1 Sensitivity and Complementary Sensitivity Function 45
3.3.2 Two-Degree-of-Freedom Controller 46
3.3.3 Asymptotic Properties of Closed-Loop Response (System Type) 46
3.3.4 The Concept of "Perfect Control" 47
3.4 Outline of the IMC Design Procedure 50
3.4.1 Basic Design Philosophy 50
3.4.2 Two-Degree-of-Freedom Design 52
3.4.3 Design in the Presence of Measurement Device Dynamics . 54
3.5 Summary 55
3.6 References 55

4 SISO IMC DESIGN FOR STABLE SYSTEMS 57
4.1 Nominal Performance 57
4.1.1 H_2-Optimal Control 58
4.1.2 Design for Specific Input Forms 60
4.1.3 Minimum Error Norm for Step Inputs 61
4.2 The IMC Filter 64
4.2.1 Filter Form 64
4.2.2 A Qualitative Interpretation of the Function of the Filter . 65
4.3 Robust Stability 66
4.3.1 Norm-Bounded Uncertainty Regions 66
4.3.2 General Uncertainty Regions 68
4.4 Robust Performance 70
4.4.1 Norm-Bounded Uncertainty Regions 70
4.4.2 General Uncertainty Regions 73
4.5 Summary of IMC Design Procedure 74
4.6 Application: IMC Design for a First-Order Deadtime System . . . 75
4.6.1 Deadtime Uncertainty 75
4.6.2 Three-Parameter Uncertainty 81

4.7 References 84

5 SISO IMC DESIGN FOR UNSTABLE SYSTEMS 85

5.1 Parametrization of All Stabilizing Controllers 85
5.1.1 Conditions for Internal Stability 85
5.1.2 Controller parametrization 87
5.2 Nominal Performance 88
5.2.1 H_2-Optimal Controller 88
5.2.2 Design for Common Input Forms 92
5.2.3 Minimum Error Norm for Step Inputs to Stable Systems . . 94
5.2.4 Two-Degree-of-Freedom Controller 94
5.3 The IMC Filter 96
5.3.1 Filter Form 96
5.3.2 Qualitative Interpretation of the Filter Function 99
5.4 Robust Stability 101
5.4.1 Norm-Bounded Uncertainty Regions 101
5.4.2 General Uncertainty Regions 102
5.5 Robust Performance 102
5.6 Summary of the IMC Design Procedure 103
5.7 Applications 104
5.7.1 Distillation Column Base Level Control 104
5.7.2 NMP Unstable Systems 107
5.8 References 110

6 ISSUES IN SISO IMC DESIGN 113

6.1 Implications of IMC for Classic Feedback Controllers 113
6.1.1 General Relationships 113
6.1.2 PID Settings for Simple Models 114
6.1.3 PID Settings for a First-Order System with Deadtime . . . 121
6.1.4 Summary 125
6.2 IMC Interpretation of Smith Predictor Controller 126
6.2.1 General Relationships 126
6.2.2 Some Myths about the Tuning of Smith-Predictor Controllers 128
6.2.3 Robust Tuning of Smith-Predictor Controller for First-Order System with Deadtime 130
6.2.4 Summary 131
6.3 Feedforward Control 131
6.3.1 Objectives and Structure 131
6.3.2 Design 132
6.3.3 Summary 135

6.4 Cascade Control 135
6.4.1 Objectives, Structure, and Design 135
6.4.2 Implementation 138
6.4.3 Summary 138
6.5 References 139

Part II: SAMPLED DATA SINGLE-INPUT SINGLE-OUTPUT SYSTEMS 141

7 FUNDAMENTALS OF SAMPLED-DATA SYSTEMS CONTROL 143
7.1 Sampled-Data Feedback Structure 143
7.2 IMC Structure 147
7.3 Formulation of Control Problem 147
7.3.1 Process Model 149
7.3.2 Model Uncertainty Description 149
7.4 Internal Stability 150
7.5 Nominal Performance 151
7.5.1 Sensitivity and Complementary Sensitivity Function 151
7.5.2 Asymptotic Properties of Closed-Loop Response 153
7.5.3 Limitations on Achievable Performance 155
7.5.4 Discrete Linear Quadratic (H_2^*-) Optimal Control 158
7.5.5 H_∞ Performance Objective 159
7.6 Robust Stability 160
7.7 Robust Performance 161
7.7.1 H_2 Performance Objective 161
7.7.2 H_∞ Performance Objective 162
7.8 Summary 162
7.9 References 164

8 SISO IMC DESIGN FOR STABLE SAMPLED-DATA SYSTEMS 165
8.1 Nominal Performance 165
8.1.1 H_2^*-Optimal Control 165
8.1.2 Design of the IMC Controller $\tilde{q}(z)$ 168
8.2 The Discrete IMC Filter 170
8.3 Robust Stability 174
8.3.1 Filter Design 174
8.3.2 Effect of Sampling 176
8.4 Robust Performance 176

8.4.1 Filter Design . . . 177
8.4.2 Sampling Time Selection . . . 178
8.4.3 Example . . . 178
8.5 Summary . . . 181
8.6 References . . . 182

9 SISO DESIGN FOR UNSTABLE SAMPLED-DATA SYSTEMS 183
9.1 Parametrization of All Stabilizing Controllers . . . 183
9.1.1 Internal Stability . . . 183
9.1.2 Controller Parametrization . . . 184
9.2 Nominal Performance . . . 184
9.2.1 H_2^*-Optimal Controller . . . 185
9.2.2 Design of the IMC Controller $\tilde{q}(z)$ . . . 188
9.2.3 Anti-aliasing Prefilter . . . 189
9.2.4 Design for Common Input Forms . . . 190
9.2.5 Integral Squared Error (ISE) for Step Inputs to Stable Systems . . . 192
9.3 The Discrete IMC Filter . . . 193
9.3.1 Filter Form . . . 193
9.3.2 Qualitative Interpretation of the Filter Function . . . 195
9.4 Robust Stability . . . 198
9.5 Robust Performance . . . 198
9.6 Summary of the IMC Design Procedure . . . 199
9.7 Application: Distillation Column Base Level Control . . . 200
9.8 References . . . 201

Part III: CONTINUOUS MULTI-INPUT MULTI-OUTPUT SYSTEMS 203

10 FUNDAMENTALS OF MIMO FEEDBACK CONTROL 205
10.1 Definitions and Basic Principles . . . 205
10.1.1 Modeling . . . 205
10.1.2 Poles . . . 206
10.1.3 Zeros . . . 207
10.1.4 Vector and Matrix Norms . . . 208
10.1.5 Singular Values and the Singular Value Decomposition . . . 211
10.1.6 Norms on Function Spaces . . . 215
10.2 Classic Feedback . . . 217
10.2.1 Definitions . . . 217
10.2.2 Multivariable Nyquist Criterion . . . 217

10.2.3 Internal Stability 221
10.2.4 Small Gain Theorem 221
10.3 Formulation of Control Problem 222
10.3.1 Process Model 222
10.3.2 Model Uncertainty Description 223
10.3.3 Input Specifications 225
10.3.4 Control Objectives 226
10.4 Nominal Performance 227
10.4.1 Sensitivity and Complementary Sensitivity Function 227
10.4.2 Asymptotic Properties of Closed-Loop Response (System Type) 228
10.4.3 Linear Quadratic (H_2-) Optimal Control 228
10.4.4 H_∞-Optimal Control 230
10.5 Summary 232
10.6 References 233

11 ROBUST STABILITY AND PERFORMANCE 235
11.1 Robust Stability for Unstructured Uncertainty 236
11.1.1 Uncertainty Description 236
11.1.2 General Robust Stability Theorem 237
11.1.3 Multiplicative Output Uncertainty 239
11.1.4 Multiplicative Input Uncertainty 240
11.1.5 Inverse Multiplicative Output Uncertainty 242
11.1.6 Example: Input Uncertainty for Distillation Column 242
11.1.7 Integral Control and Robust Stability 244
11.2 Robust Stability for Structured Uncertainty 246
11.2.1 Uncertainty Description 246
11.2.2 Structured Singular Value 248
11.2.3 Simultaneous Multiplicative Input and Output Uncertainty 252
11.2.4 Batch Reactor: Simultaneous Parametric and Unstructured Uncertainty 254
11.2.5 Independent Uncertainty in the Transfer Matrix Elements . 257
11.2.6 Condition Number and Relative Gain Array as Sensitivity Measures 258
11.3 Robust Performance 262
11.3.1 H_∞-Performance Objective 262
11.3.2 Multiplicative Output Uncertainty 264
11.3.3 Multiplicative Input Uncertainty 266
11.3.4 H_2-Performance Objective 268
11.3.5 Application: High-Purity Distillation 271

11.4 Robustness Conditions in Terms of Specific Transfer Matrices . . . 275
11.4.1 How to find the LFT . . . 278
11.4.2 New Properties of μ . . . 281
11.4.3 Examples . . . 285
11.5 Summary . . . 287
11.6 References . . . 290

12 MIMO IMC DESIGN 293
12.1 IMC Structure . . . 293
12.2 Conditions for Internal Stability . . . 295
12.3 Parametrization of All Stabilizing Controllers . . . 296
12.4 Asymptotic Properties of Closed-Loop Response . . . 297
12.5 Outline of the IMC Design Procedure . . . 298
12.6 Nominal Performance . . . 301
12.6.1 Assumptions . . . 301
12.6.2 H_2-Optimal Control for a Specific Input . . . 302
12.6.3 H_2-Optimal Control for a Set of Inputs . . . 306
12.6.4 Algorithm for "Inner-Outer" Factorization . . . 309
12.7 Robust Stability and Performance . . . 310
12.7.1 Filter Structure . . . 311
12.7.2 General Interconnection Structure with Filter . . . 313
12.7.3 Robust Control: H_∞ Performance Objective . . . 315
12.7.4 Robust Control: H_2-Type Performance Objective . . . 318
12.8 Application: High-Purity Distillation . . . 320
12.9 Summary . . . 324
12.10Discussion and References . . . 325

13 PERFORMANCE LIMITATIONS FOR MIMO SYSTEMS 327
13.1 Effect of Plant Gain . . . 328
13.1.1 Constraints on Manipulated Variable . . . 328
13.1.2 Disturbance Condition Number . . . 329
13.1.3 Implications of κ_d for Closed-Loop Performance . . . 330
13.1.4 Decomposition of d along Singular Vectors . . . 332
13.1.5 Summary . . . 338
13.2 NMP Characteristics . . . 339
13.2.1 Zero Direction . . . 339
13.2.2 Implications of Zero Direction for Achievable Performance . 342
13.2.3 Summary . . . 343
13.3 Sensitivity to Model Uncertainty . . . 344
13.3.1 Sensitivity to Diagonal Input Uncertainty . . . 344

13.3.2 Sensitivity with Different Controller Structures 346
13.3.3 "Worst-Case" Uncertainty 348
13.3.4 Example 349
13.3.5 Summary 351
13.4 References 356

14 DECENTRALIZED CONTROL **359**
14.1 Motivation 359
14.2 Definitions 361
14.3 Necessary Conditions for Controllability 362
14.3.1 Results 362
14.3.2 Proofs 365
14.4 Stability Conditions – Interaction Measures 367
14.4.1 Necessary and Sufficient Stability Conditions 368
14.4.2 Sufficient Stability Conditions 370
14.4.3 Diagonal Dominance Interaction Measures 372
14.4.4 Generalized Diagonal Dominance Interaction Measures . . . 373
14.4.5 The μ-Interaction Measure 374
14.4.6 Interaction Measures for 2×2 Systems 375
14.4.7 Examples 376
14.5 Robust Performance Conditions 378
14.5.1 Sufficient Conditions for Robust Performance 379
14.5.2 Design Procedure 382
14.5.3 Example 383
14.6 Summary 388
14.7 References 389

Part IV: SAMPLED DATA MULTI-INPUT MULTI-OUTPUT SYSTEMS **391**

15 MIMO Sampled-Data Systems **393**
15.1 Fundamentals of MIMO Sampled-Data Systems 393
15.1.1 Sampled-Data Feedback 393
15.1.2 Poles and Zeros 395
15.1.3 Internal Stability 396
15.1.4 IMC Structure 397
15.1.5 Model Uncertainty Description 397
15.2 Nominal Internal Stability 399
15.2.1 IMC Structure 399
15.2.2 Feedback Structure 400

15.3 Nominal Performance 401
15.3.1 Sensitivity and Complementary Sensitivity Function 401
15.3.2 H_2^* Performance Objectives 402
15.3.3 H_∞ Performance Objective 404
15.4 Robust Stability 404
15.5 Robust Performance 405
15.5.1 Sensitivity Function Approximation 405
15.5.2 H_∞ Performance Objective 406
15.6 IMC Design: Step 1 ($\tilde{Q}$) 406
15.6.1 H_2^*-Optimal Control 406
15.6.2 Setpoint Prediction 413
15.6.3 Intersample Rippling 413
15.6.4 Inner-Outer Factorization 416
15.7 IMC Design: Step 2 (F) 419
15.7.1 Filter Structure 420
15.7.2 Robust Stability Interconnection Structure 421
15.7.3 Robust Performance Interconnection Structure 422
15.7.4 Robust H_∞ Performance Objective 423
15.8 Illustration of the Design Procedure 424
15.8.1 System Description 424
15.8.2 Design of $\tilde{Q}$ 426
15.8.3 Design of F 428
15.9 Summary 433
15.10References 434

Part V: CASE STUDY **435**

16 LV-CONTROL OF A HIGH-PURITY DISTILLATION COLUMN **437**
16.1 Features 437
16.1.1 Uncertainty 437
16.1.2 Nonlinearity 438
16.1.3 Logarithmic Compositions 438
16.1.4 Choice of Nominal Operating Point 439
16.2 The Distillation Column 439
16.2.1 Modelling 440
16.2.2 Simulations 442
16.3 Formulation of the Control Problem 443
16.3.1 Performance and Uncertainty Specifications 443
16.3.2 Analysis of Controllers 443

16.3.3 Controllers 444
16.4 Results for Operating Point A 445
16.4.1 Discussion of Controllers 446
16.4.2 Conclusions 451
16.5 Effect of Nonlinearity (Results for Operating Point C) 452
16.5.1 Modelling 452
16.5.2 μ-Analysis 453
16.5.3 Logarithmic Versus Unscaled Compositions 453
16.5.4 Transition from Operating Point A to C 455
16.6 Conclusions 456
16.7 References 456

Appendix **459**

References **469**

Index **479**

PREFACE

The development of the state space approach in the early 1960s made it possible for the first time to solve general linear multivariable control problems with relative ease. The new techniques seemed to hold high promise for practical applications and attracted much interest — at least in the academic community. Fifteen years later several review papers (e.g., Foss, 1973; Kestenbaum et al., 1976) concluded that the impact on the industrial practice had been negligible.

A number of possible reasons for this failure can be identified. For example, there was no smooth transition from the established and proven techniques and tools (PID controllers, Smith Predictor, Relative Gain Array, Decoupler, etc.) to the new ones. This led to great educational difficulties and from today's perspective pointless debates in which the classic frequency domain approach was pitched against the modern state space techniques. More serious than this lack of understanding which persisted for almost two decades was that the new techniques did not address some very fundamental issues which are at the very heart of feedback design. For example, the concept of "nonminimum phase behavior" seemed forgotten. Also, the problem of model uncertainty and model error, which dominates process control, was only addressed via sensitivity analysis, whose validity is limited to infinitesimal model perturbations.

After years of isolated but persistent criticism a new understanding and approach started to emerge in the late 1970s. Some of the key theoretical contributions were the new formulation and parametrization of the optimal control problem by Youla and coworkers (1976), the definition of the H_∞ control problem by Zames (1981), and the introduction of the structured singular value by Doyle (1982). These and other discoveries sparked again much enthusiasm and a flurry of research activity almost comparable to the 1960s. Only time will show the impact on the control practice.

The new approach does address many issues of practical importance like model uncertainty (robustness). Many research challenges remain however. New results appear monthly but the ideas have not matured enough to form a coherent pic-

ture. Moreover, the level of abstraction makes the new developments inaccessible to anybody but the researchers in this area.

This book attempts to bridge the gap between a good undergraduate training in process control and the new arriving generation of robust control theory. Though not a textbook, it is intended as a supplement for a graduate level control course. Because the emphasis is on practical control system design methods, sections of the book should also be of interest to the industrial control engineer who wants to learn about the new powerful techniques. Some basic concepts from linear algebra and complex variables are assumed.

The book motivates the new theory with a series of typical process control examples: single-input single-output systems with time delay, multivariable distillation column models, reactor models, etc. However, the applicability of the new techniques is not limited to process control. The emphasis is on the derivation of new analysis and design tools. Preference is given to simple effective techniques over the most recent and most general theoretical discoveries. There is no doubt that the next few years will bring about a number of refinements which will affect some of the details in the book. We strongly believe, however, that the key ideas of robust control as presented here will profoundly impact process control understanding, teaching, and practice in the future. Indeed, some of the concepts have been applied already to industrial systems and some of the algorithms have become integral parts of industrial process control software.

Many colleagues have influenced our thinking on the topic of robust process control. We wish to acknowledge in particular Reuel Shinnar, Coleman Brosilow, and John Doyle. Shinnar made us first aware of the robustness problems. Brosilow convinced us of the advantages of the Internal Model Control structure. Doyle familiarized us with the new powerful theory which makes the practical robustness analysis of multivariable systems possible.

The initial outline for this book was developed with our late colleague Constantin Economou. We missed his high standards, his flair for communicating complex ideas, and his humor. The project was not the same without him.

A number of present and former Caltech graduate students and research fellows have contributed either directly or indirectly to the material. In particular, we would like to mention Sigurd Skogestad, Dan Rivera, Dan Laughlin, Dan Lewin, and Claudio Scali. Brad Holt, Yaman Arkun, and Christos Georgakis have classroom-tested parts of this book and have provided us with valuable criticism. The continuous encouragement by our friends in industry was most important. We wish to acknowledge, in particular, Dave Prett and Carlos Garcia of Shell Development, Dave Smith and Bjorn Tyreus of DuPont, and John Hamer of Kodak.

Much of the research reported in this book was carried out at Caltech. The extraordinary scholarly environment and supportive administration made this endeavor possible. The final sections of the book were completed while the first author enjoyed an appointment as the Gulf Visiting Professor of Chemical Engineering at Carnegie Mellon University. During the past year, the second author held a joint appointment with the Chemical and Nuclear Engineering Department and the Systems Research Center of the University of Maryland. The excellent atmosphere at this institution allowed him to continue his work on the book.

Over the years we received continuous financial support from the National Science Foundation, the Department of Energy, Shell Development, DuPont, and Kodak which enabled us to pursue our research objectives.

Finally we would like to thank Glenn C. Smith for preparing most of the figures, Evangelos Petroutsos for assisting with TEX and LATEX and Jan Owen who did such an excellent job typing the manuscript after suffering through three different word processing systems.

M. Morari
E. Zafiriou

NOMENCLATURE

Abbreviations

DIC	Decentralized Integral Controllable
IC	Integral Controllable
IM	Interaction Measure
IMC	Internal Model Control
IS	Integral Stabilizable
ISE	Integral Squared Error
LFT	Linear Fractional Transformation
LHP	Left-Half Plane
MIMO	Multi-Input Multi-Output
MP	Minimum Phase
NMP	Nonminimum Phase
NP	Nominal Performance
NS	Nominal Stability
PFE	Partial Fraction Expansion
RGA	Relative Gain Array
RHP	Right-Half Plane
RHS	Right Hand Side
RP	Robust Performance
RS	Robust Stability
SISO	Single-Input Single-Output
SSE	Sum of Squared Errors
SSV	Structured Singular Value
SVD	Singular Value Decomposition
UC	Unit Circle

Symbols

$B(z)$	Factor to preserve system type in $\tilde{q}(z)$ and $\tilde{Q}(z)$
$b_p(s)(b_p^*(z))$	Allpass with the open RHP (outside the UC) poles of $\tilde{p}(s)(\tilde{p}^*(z))$
$b_v(s)(b_v^*(z))$	Allpass with the open RHP (outside the UC) poles of $v(s)(v^*(z))$
$\mathcal{C}$	Field of complex numbers
$\mathcal{C}^{n\times m}$	Field of complex matrices of dimension $n \times m$
$c(s), c(z)(C(s), C(z))$	SISO (MIMO) continuous and discrete classic feedback controllers
$c_f(s)$	Classic feedforward controller
$d(s)$	Effect of $d'(s)$ on $y(s)$
$d'(s)$	Disturbance entering $p_d(s)$
$d^*(z)(d_\gamma^*(z))$	z-transform of $d(s)$ before (after) passing through $\gamma(s)$
$e(s)(e^*(z))$	Continuous (discrete) error signal: Difference between $y(s)(y^*(z))$ and $r(s)(r^*(z))$
$e'(s)$	Weighted error signal $W_2(s)e(s)$
$f(s), f(z)(F(s), F(z))$	SISO (MIMO) continuous and discrete IMC filters
$f_1(s)$	Type 1 one-parameter discrete SISO IMC filter
$G^F(s)$	Transfer function matrix used as argument of μ in the RP SSV criterion for the continuous case
$G_v^F(s)$	The equivalent of $G^F(s)$ for the discrete case; it depends on whether $v = r$ or $v = d$
$G_{11}^F(s)(G_{11}^{*F}(z))$	Transfer function matrix used as argument of μ in the RS SSV condition for the continuous (discrete) case
$h_0(s)(H_0(s))$	SISO (MIMO) zero-order hold
$H_2(H_2^*)$	$H_2^{n\times m}(H_2^{*,n\times m})$ for $n = m = 1$
$H_2^n(H_2^{*,n})$	$H_2^{n\times m}(H_2^{*,n\times m})$ for $m = 1$
$H_2^{n\times m}$	Subspace of $L_2^{n\times m}$, containing the functions with analytic continuations in the RHP (includes all rational, proper, stable transfer function matrices)
$H_2^{*,n\times m}$	Subspace of $L_2^{*,n\times m}$ defined as the orthogonal complement of the $L_2^{*,n\times m}$ subspace which contains functions with analytic continuations inside the UC ($H_2^{*,n\times m}$ contains all rational, strictly proper, stable z-transfer function matrices).
Im	Imaginary part

$\ell_a(s)(L_A(s))$	Additive SISO (MIMO) uncertainty
$\bar{\ell}_a(\omega)(\bar{\ell}_a^*(\omega))$	Bound on the additive uncertainty for the continuous (discrete) case
$L_E(s)(\bar{\ell}_E(\omega))$	(Bound on the) MIMO inverse multiplicative output uncertainty
$L_I(s)(\bar{\ell}_I(\omega))$	(Bound on the) MIMO input multiplicative uncertainty
$\ell_m(s)$	Multiplicative SISO uncertainty
$\bar{\ell}_m(\omega)(\bar{\ell}_m^*(\omega))$	Bound on the multiplicative SISO uncertainty for the continuous (discrete) case
$L_O(s)(\bar{\ell}_O(\omega))$	(Bound on the) MIMO output multiplicative uncertainty
$L_2(L_2^*)$	$L_2^{n\times m}(L_2^{*,n\times m})$ for $m = n = 1$
$L_2^n(L_2^{*,n})$	$L_2^{n\times m}(L_2^{*,n\times m})$ for $m = 1$
$L_2^{n\times m}(L_2^{*,n\times m})$	Space of $n \times m$ matrix valued functions that are square integrable on the imaginary axis (UC)
$p(s)(P(s))$	SISO (MIMO) plant; also denotes the process model in sections where $p = \tilde{p}(P = \tilde{P})$ is assumed
$\tilde{p}(s)(\tilde{P}(s))$	SISO (MIMO) process model
$p^*(z), p_\gamma^*(z)(P^*(z), P_\gamma^*(z))$	Zero-order hold discrete equivalent of $p(s)(P(s))$ without and with $\gamma(s)(\Gamma(s))$ included, respectively
$\tilde{p}^*(z), \tilde{p}_\gamma^*(z)(\tilde{P}^*(z), \tilde{P}_\gamma^*(z))$	Zero-order hold discrete equivalents of $\tilde{p}(s)(\tilde{P}(s))$ without and with $\gamma(s)(\Gamma(s))$ included, respectively
$\tilde{p}_A(s), \tilde{p}_A^*(z)(\tilde{P}_A(s), \tilde{P}_A^*(z))$	Allpass with the NMP elements of $\tilde{p}(s)$ and $\tilde{p}^*(z)(\tilde{P}(s)$ and $\tilde{P}^*(z))$ respectively
$p_d(s), \tilde{p}_d(s)$	True and model of the disturbance effect plant transfer function
$p_m(s), \tilde{p}_m(s)$	True and model of the measurement device transfer function
$\tilde{p}_M(s), \tilde{p}_M^*(z)(\tilde{P}_M(s), \tilde{P}_M^*(z))$	MP factors of $\tilde{p}(s)$ and $\tilde{p}^*(z)$ $(\tilde{P}(s)$ and $\tilde{P}^*(z))$ respectively
$q(s), q(z)(Q(s), Q(z))$	SISO (MIMO) continuous and discrete IMC controllers
$\tilde{q}(s), \tilde{q}(z)(\tilde{Q}(s), \tilde{Q}(z))$	SISO (MIMO) continuous and discrete 1st step IMC controllers
$\tilde{q}_d(s), q_d(s)(\tilde{q}_r(s), q_r(s))$	Disturbance rejection (setpoint tracking) 1st step and overall IMC controllers of the two-degree-of-freedom structure
$q_f(s)$	Feedforward IMC controller

$\tilde{q}_H(z)(\tilde{Q}_H(z))$	SISO (MIMO) H_2^*-optimal IMC controller
$\tilde{q}_-(z)$	Factor included in $\tilde{q}(z)$ and $\tilde{Q}(z)$ to avoid intersample rippling
$\mathcal{R}$	Field of real numbers
Res	Residue
Re	Real part
$r(s)(r^*(z))$	Continuous (discrete) setpoint
T	Sampling time
$u(s)(u(z))$	Continuous (discrete) input to the plant
$v(s)(v'(s))$	External input $r(s)$ or $d(s)$ (normalized to impulse)
$v^*(z)$	z-transform of $v(s)$
$V(s)(V(z))$	Square matrix whose columns consist of external inputs $v(s)(v^*(z))$
$\mathcal{V}(\mathcal{V}')$	Set of inputs $v(s)(v'(s))$ or $v^*(z)$
$v_A(s), v_A^*(z)(V_A(s), V_A(z))$	Allpass with the NMP elements of $v(s), v^*(z)(V(s), V(z))$ respectively
$v_M(s), v_M^*(V_M(s), V_M(z))$	MP factors of $v(s), v^*(z)$ $(V(s), V(z))$ respectively
$w(s)$	Performance weight for the SISO case
$w_d(s)(w_r(s))$	Performance weight for disturbance rejection (setpoint tracking) in the two-degree-of-freedom structure
$W_1(s)(W_2(s))$	Input (Output) weights for the MIMO case
$y(s)(\tilde{y}(s))$	Continuous plant (model) output
$y^*(z), y_\gamma^*(z)(\tilde{y}^*(z), \tilde{y}_\gamma^*(z))$	z-transform of $y(s)(\tilde{y}(s))$ before and after passing through $\gamma(s)$ respectively
$y_m(s)$	Output of measurement device

Greek Characters

$\gamma(s)(\Gamma(s))$	SISO (MIMO) anti-aliasing prefilter
$\Delta(s)(\Delta(z))$	Block diagonal matrix containing the uncertainty for the continuous (discrete) case
$\mathbf{\Delta}$	Set of possible Δ's
$\Delta_u(s)(\Delta_u(z))$	Uncertainty block in $\Delta(s)(\Delta(z))$
$\Delta_p(s)$	RP block in $\Delta(s)$
$\epsilon(s), E(s)(\tilde{\epsilon}(s), \tilde{E}(s))$	(Nominal) sensitivity function for the continuous SISO and MIMO cases, respectively
$\bar{\epsilon}(s)$	first step IMC nominal sensitivity function

$\tilde{\epsilon}^*(z)(\tilde{E}^*(z))$	SISO (MIMO) discrete sensitivity function
$\tilde{\epsilon}_r(s), \tilde{E}_r(s)(\tilde{\epsilon}_d(s), \tilde{E}_d(s))$	Setpoint tracking (disturbance rejection) SISO and MIMO (approximate) continuous nominal sensitivity function for sampled-data systems
$E_v(s)(\tilde{E}_v(s))$	(Nominal) MIMO continuous sensitivity function for sampled-data systems; it depends on whether $v = d$ (approximation) or $v = r$
$\zeta(s)(\zeta(z))$	Zero polynomial for the continuous (discrete) case
$\eta(s), H(s)(\tilde{\eta}(s), \tilde{H}(s))$	(Nominal) Complementary sensitivity function for the continuous SISO and MIMO cases respectively
$\bar{\eta}(s)$	first-step IMC complementary sensitivity function
$\eta^*(z)(H^*(z))$	SISO (MIMO) discrete complementary sensitivity function
$\tilde{\eta}_r(s)(\tilde{H}_r(s))$	Setpoint tracking SISO (MIMO) continuous nominal complementary sensitivity function for sampled-data systems
$\kappa(\kappa^*)$	(Minimized) condition number
κ_d	Disturbance condition number
κ_R	Rijnsdorp IM
$\Lambda(A)$	RGA matrix of the matrix A
$\lambda_i(A)$	i^{th} eigenvalue of A
$\lambda_{ij}(A)$	ij-element of $\Lambda(A)$
$\mu(A)$ or $\mu_{\Delta}(A)$	SSV of A for same specific set Δ
Π	Set of possible plants
$\pi(s)(\pi(z))$	Pole polynomial for the continuous (discrete) case
$\pi(\omega)(\pi^*(\omega))$	Uncertainty region on the Nyquist plane for the continuous (discrete) case
$\rho(A)$	Spectral radius of A
$\bar{\sigma}(A)$	Maximum singular value of A
$\underline{\sigma}(A)$	Minimum singular value of A
$\sigma_i(A)$	i^{th} singular value of A
$\phi(z)$	Factor included in $f(z)$ to satisfy the requirements on the IMC filter
ω_B	Closed-loop system bandwidth
ω_s	Sampling frequency

Superscripts

H	Complex conjugate transpose of a matrix
T	Transpose of a matrix
$\perp$	Orthogonal complement of a function space

Special Notation

$F(A, \Delta)$	Matrix resulting from the application of a linear fractional transformation on A, Δ with Δ in the feedback path
$\mathcal{L}\{\cdot\}$	Laplace transform of $(\cdot)$
$\mathcal{Z}\{\cdot\}$	Z-transform of $(\cdot)$
$\|\cdot\|$	Norm on $\mathcal{C}^{n\times m}$
$\|\cdot\|_F$	Frobenius norm on $\mathcal{C}^{n\times m}$
$\|\cdot\|_2$	$L_2^{n\times m}$ or $L_2^{*,n\times m}$ norm
$\|\cdot\|_\infty$	∞-norm of a function
$\|\cdot\|_{i2}$	Operator norm induced by $\|\cdot\|_2$
$\{g(s)\}_+(\{g(z)\}_+)$	Projection of an $L_2^n(L_2^{*,n})$ function $g(s)(g(z))$ on $H_2^n(H_2^{*,n})$
$\{g(s)\}_-(\{g(z)\}_-)$	Projection of an $L_2^n(L_2^{*,n})$ function $g(s)(g(z))$ on $(H_2^n)^\perp((H_2^{*,n})^\perp)$
$\forall$	For all
$\ni$	Such that

We also wish to remind the reader that the following statements are equivalent:

$$A \Leftarrow B$$

$$A \text{ if } B$$

$$B \text{ is sufficient for } A$$

$$B \Rightarrow A$$

$$B \text{ only if } A$$

$$A \text{ is necessary for } B$$

$$\text{not } A \quad \Rightarrow \quad \text{not } B$$

Chapter 1

INTRODUCTION

The main ideas covered in this book are motivated and put in historical perspective. A new controller parametrization is outlined and the concept of "robustness" is explained. The scope of the book is defined and hints for its study are provided.

1.1 The Evolution of Control Theory

Feedback control mechanisms have been used for millenia. The first applications of the feedback principle (Mayr, 1970) can be traced back to ancient Greece: the water clock of Ktesibios (ca. 300 B.C.) employed a float regulator. The first automatic feedback controller used in an industrial process was James Watt's flyball governor invented in 1769 for controlling the speed of a steam engine. As these centrifugal governors were refined, they developed some serious problems, then referred to as "hunting" and now known as "instability." This in turn prompted the first mathematical analysis of a feedback system via differential equations by Maxwell (1868). During the same period Vyshnegradskii (1877) formulated a mathematical theory of regulators.

The invention of the negative feedback amplifier at Bell Laboratories by Black prior to World War II marks another milestone in the use of feedback. The observed stability problems ("singing") were explained through the frequency domain analysis techniques by Bode and Nyquist (Nyquist, 1932; Bode, 1964; Black, 1977).

Throughout the 1940s and 1950s feedback design by trial and error was predominant and even today this approach plays a major role. Typically, performance specifications are defined initially and then the controller design is modified iteratively until the best compromise between the frequently conflicting objectives is reached. Every design step is followed by an analysis step and the iterations are guided by experience and rules of thumb. In the late 1950s Newton, Gould, and Kaiser (1957) showed that if the control objectives are formulated in terms of the integral square error (ISE) the "optimal" feedback controller can be found

"analytically" — i.e., directly and without trial and error. This approach, which is based on the Wiener-Hopf factorization technique, was subsequently greatly generalized, in particular by Kalman.

The deficiency of these "synthesis" techniques is that typical practical performance objectives are usually much more complex than ISE. Frequently the designer wishes to impose constraints on the closed-loop system response characteristics like overshoot, rise time, and decay ratio. Also a certain "robustness" against changes in the dynamic characteristics of the plant to be controlled is desirable. Often it is possible to achieve these attributes indirectly by frequency-weighting the ISE and by adding to the objective a penalty for excessive movements of the manipulated variable. For that purpose the weights and penalties have to be chosen again iteratively by trial and error. Thus, for a practical design the advantages of such an "optimal control synthesis" procedure over an ad hoc adjustment of controller parameters and a test via simulation can become quite small.

The ISE or, in the more general context, the H_2-optimal control formulation dominated the literature for about 20 years. In the late seventies three major and in some sense related discoveries started a new era of feedback control theory.

- Youla and coworkers (1976) showed that it is possible to parametrize all stabilizing controllers for a particular system in a very effective manner: when searching for a controller with specific properties one can simply and without loss search over the space of *all stable* transfer functions. The parametrization guarantees that the resulting feedback controller *automatically* yields a closed-loop stable system. Thus, the search for a good controller is greatly simplified.

- Zames (1981) postulated that measuring performance in terms of the ∞-norm rather than the traditional 2-norm (ISE) might be much closer to the practical needs. This ushered in the era of H_∞ optimal control.

- Doyle argued that model uncertainty is often described very effectively in terms of norm-bounded perturbations. For these perturbations and the H_∞ performance objective he developed a powerful tool (the structured singular value) for testing "robust stability" (i.e., stability in the presence of model uncertainty) and "robust performance" (i.e., performance in the presence of model uncertainty). This is probably the primary motivation for the modern ∞-norm objective.

In parallel to these developments in mathematics and electrical engineering a new type of algorithm exemplified by Model Algorithmic Control (Richalet et al.,

1978) and Dynamic Matrix Control (Cutler and Ramaker, 1979) was invented in the process industries and applied successfully to complex process control problems. Though these algorithms had a heuristic basis Garcia and Morari (1982) discovered that some of the modern robust control characteristics had been incorporated in them in an ad hoc fashion. Thus, the time was ripe to put these empirical algorithms on a firmer footing and to have them benefit from the rich new theory. The objective of this book is to do that while retaining the features which make these algorithms easy to understand and easy to adjust on-line. At times this required mathematical generality to be sacrificed.

1.2 Controller Parametrization: The IMC Structure

Consider the block diagram shown in Fig. 1.2-1 where the control system is shaded: it includes the two blocks labeled *controller* and *model.* The control system has as its inputs the setpoint and process output (measurement) and as its output the manipulated variable (process input). Let us discuss qualitatively the advantages of such a structure over the classic feedback structure shown in Fig. 1.2-2.

The effect of the parallel path with the *model* is to subtract the effect of the manipulated variables from the process output. If we assume for the moment that the *model* is a *perfect* representation of the *process*, then the feedback signal is equal to the *influence of disturbances* and is not affected by the action of the manipulated variables. Thus, the system is effectively open-loop and the usual stability problems associated with feedback have disappeared. The overall system is stable simply if and only if both the *process* and the *IMC controller* are stable.

Moreover, the IMC controller plays the role of a feedforward controller and can be designed as such. But the IMC controller does not suffer from the disadvantages of feedforward controllers: it can cancel the influence of (unmeasured) disturbances because the feedback signal is equal to this influence and modifies the controller setpoint accordingly.

If the *model* does not mimic the dynamic behavior of the *process* perfectly then the feedback signal expresses both the *influence of (unmeasured) disturbances* and the effect of this model error. The model error gives rise to feedback in the true sense and leads to possible stability problems. This forces the designer to "detune" the ideal feedforward controller for "robustness."

The controller form described here is a special case of the Youla-parametrization mentioned above. It is inherent in all "model predictive" control schemes, in particular Model Algorithmic Control and Dynamic Matrix Control.

1.3 Robustness

Regardless of what design technique is used, controllers are always designed based on (necessarily incomplete) information about the dynamic behavior of the process. This information (i.e., the "model") can have the form of a system of coupled partial differential equations or be simply the process gain and the settling time experienced by the plant operator. The accuracy of this information varies but is never perfect. Moreover, the behavior of the plant itself changes with time (feedstock changes, catalyst activity changes, etc.) and these changes are rarely captured in the models. It is most desirable that the controller be insensitive to this kind of *model uncertainty*, i.e., the controller should be *robust*.

Though the design objective *robustness* seems most practical and reasonable it is essentially absent from the control literature from about 1960 to 1980. This is probably one of the reasons why the design techniques developed in this time period have had negligible effect on the industrial control practice. Since the late 1970s robustness has become a major objective of control research. This book will demonstrate the importance of robustness considerations for process control and will propose design techniques which include robustness as one of the objectives.

1.4 Scope of Book

The book has several goals:

- to alert the reader to the key role model uncertainty and robustness play in the design and successful operation of feedback control systems.
- to outline the basic ideas behind the most recent theoretical advances for addressing robustness systematically in feedback design.
- to describe simple and effective analysis and design techniques for robust feedback controllers.

The different parts of the book aim at different groups of readers:

- the first-year graduate student who wants to broaden his/her horizon, gain some basic insights into feedback control, and get a taste of the direction of the new theory.
- the beginning researcher in the area of robust control who wishes to master the transition from the undergraduate process control course to the monographs by Vidyasagar (1985) and Francis (1987) as well as the current research papers in this area.

- the industrial practitioner who needs easily applicable and proven controller tuning techniques and who is interested in learning about the basic dos and don'ts of multivariable control.

Depending on the reader's objectives and background some chapters might be of more interest than others. Chapters 2 through 4 discuss the basic trade-offs in single-input single-output (SISO) feedback control systems and how they can be addressed in the Internal Model Control framework. These chapters are essential for understanding the rest of the book. Chapter 5 generalizes the results to unstable systems and can be skipped by the application-oriented reader. Chapter 6 proposes some very effective PID tuning rules and discusses more general SISO control structures like Smith predictor, cascade and feedforward control systems.

Chapters 7 through 9 deal with sampled-data systems and mimic Chapters 1 through 6. In the derivations particular emphasis is placed on the behavior of the *continuous* output — e.g., provisions against intersample rippling are built into the controllers.

Chapters 10 and 11 cover basic aspects of robustness of multi-input multi-output (MIMO) systems. These concepts are the key to understanding the unusual phenomena observed in the control of MIMO systems. The MIMO design technique proposed in Chapter 12 is complex and tentative. Undoubtedly, more effective techniques will become available in the future. Chapter 13 discusses the *system properties* which limit the achievable performance of MIMO systems. A knowledge of these properties helps when screening designs according to their operability characteristics. Chapter 14 deals with the design of controllers for MIMO systems with a restricted information structure — e.g., multiloop controllers. Chapter 15 extends the discussion of Chapters 10 through 12 to sampled-data systems. The book concludes with the detailed report on an application to a high-purity distillation column (Chapter 16).

1.5 Some Hints for the Reader

The book assumes that the reader has mastered the material covered in a typical undergraduate control text like the one authored by Stephanopoulos (1984) or Franklin et al. (1986). Furthermore some knowledge of linear algebra (e.g., Strang, 1980) and complex variables (e.g., Churchill & Brown, 1984) is required. Finally, some familiarity with linear operator theory (e.g., Desoer & Vidyasagar, 1975; chapters 1 through 6 of Ramkrishna & Amundson, 1985) is advantageous, but not necessary.

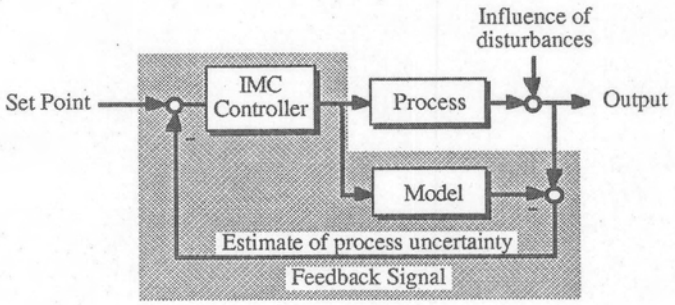

Fig. 1.2-1. Internal Model Control Structure.

Chapters 5, 9 and 13 and in particular the proofs in these chapters are mathematically more demanding and can be skipped at first reading. The chapters on sampled-data systems (7 through 9 and 15) assume that the corresponding material on continuous systems has been studied first. Also the study of the chapters on MIMO systems requires the prior understanding of the SISO concepts. In principle, the SISO material follows as a special case from the MIMO material. For tutorial reasons it is treated separately.

With the exception of this first chapter all references are collected at the chapter end for continuity. Most chapters contain one or several summary sections, usually located immediately before the reference section. This summary section allows the reader to review the main concepts. Often, mastering this summary section is sufficient for the application of the basic techniques covered in the chapter. The equation numbers in the summary section refer to the equation numbers in the main body of the text.

The end of proofs and examples is marked with the symbol □.

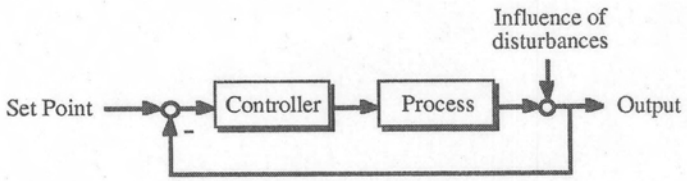

Fig. 1.2-2. Classic Feedback Structure.

Part I

CONTINUOUS SINGLE-INPUT SINGLE-OUTPUT SYSTEMS

Chapter 2

FUNDAMENTALS OF SISO FEEDBACK CONTROL

After a review of some basic definitions, the controller design problem will be formulated in general terms. All controller design procedures are based on models of one form or another. These models are necessarily inaccurate. It is of great practical importance that the controller performs well even when the dynamic behavior of the real process differs from that described by its model. In order to accomplish this objective, not only must a process model and the performance specifications be provided for the design, but also some indication of the model accuracy should be made. The attribute "robust" will be used for a property which holds not only for the model but also in the presence of model uncertainty. Mathematical conditions for robust stability and robust performance will be derived.

2.1 Definitions

The block diagram of a typical classic feedback loop is shown in Fig. 2.1-1A. Here c denotes the controller and p the plant transfer function. The transfer function p_d describes the effect of the disturbance d' on the process output y. The measurement device transfer function is symbolized by p_m. Measurement noise n corrupts the measured variable y_m. The controller determines the process input (manipulated variable) u on the basis of the error e. The objective of the feedback loop is to keep the output y close to the reference (setpoint) r.

Commonly we will use the simplified block diagram in Fig. 2.1-1B. Here d denotes the effect of the disturbance on the output. Exact knowledge of the output y is assumed ($p_m = 1, n = 0$).

In general, the transfer functions will be allowed to be rational or irrational — i.e., they may include time delays. In order to be physically realizable, the transfer functions have to be *proper* and *causal*.

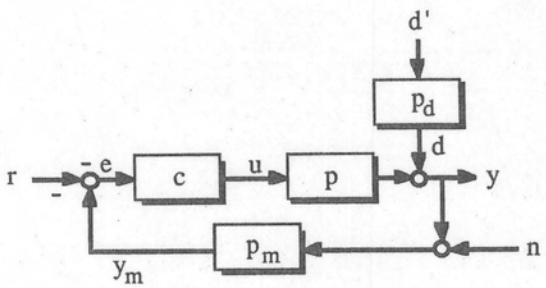

Figure 2.1-1. General (A) and simplified (B) block diagram of feedback control system.

Definition 2.1-1. *A system $g(s)$ is proper if $\lim_{s\to+\infty} g(s)$ is finite. A proper system is strictly proper if $\lim_{s\to+\infty} |g(s)| = 0$ and semi-proper if $\lim_{s\to+\infty} |g(s)| > 0$. All systems which are not proper are improper.*

A system $g(s)$ is improper if the order of the numerator polynomial exceeds the order of the denominator polynomial and proper otherwise. An improper system cannot be realized physically because it contains pure differentiators.

Definition 2.1-2. *A system requiring prediction ($e^{+s\theta}$) is noncausal.*[1] *A system which does not require prediction is causal.*

Some care has to be used in determining if a system is causal or not. A time delay in the denominator of a transfer function does not necessarily imply lack of causality. For example,

$$g(s) = \frac{y(s)}{u(s)} = \frac{e^{s\theta}}{2e^{s\theta} - 1} = \frac{1}{2 - e^{-s\theta}}$$

is causal because the present output y is determined solely by past values of y and the present value of u:

$$y(s) = \frac{1}{2}(e^{-s\theta}y(s) + u(s))$$

[1]Generally an improper system is also referred to as noncausal. In this book we will reserve noncausal to denote prediction.

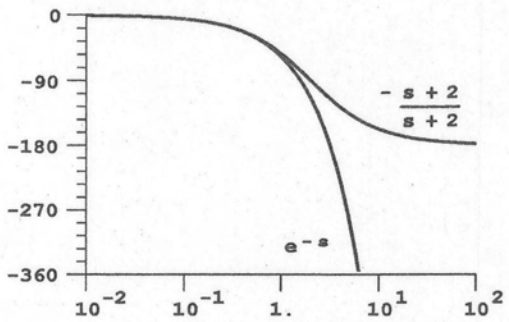

Figure 2.1-2. Phase behavior of two nonminimum phase (NMP) allpass elements.

Definition 2.1-3. *A system $g(s)$ is nonminimum phase (NMP) if its transfer function contains zeros in the Right Half Plane (RHP) or time delays or both. Otherwise a system is minimum phase (MP).*

Systems with an odd number of RHP zeros display *inverse response* characteristics: the initial direction of the step response is opposite to the direction of the final steady state.

The origin of the term "minimum phase" can be best understood in the context of stable systems: if a system $g(s)$ is MP, then there is no other system $g'(s)$ which has the same magnitude ($|g(i\omega)| = |g'(i\omega)|\ \forall\omega$) but a smaller phase lag. In other words, a NMP system $\bar{g}(s)$ does not have the smallest phase lag that is possible for a system with this magnitude ($|\bar{g}(i\omega)|$). For example, the *allpass* systems

$$g_1(s) = e^{-s}, \qquad g_2(s) = \frac{-s+2}{s+2}$$

have

$$|g_1(i\omega)| = |g_2(i\omega)| = 1$$

Their phase behavior is shown in Fig. 2.1-2. Clearly there is another system with the same magnitude (for example, $g'(s) = 1$) which has a smaller (zero) phase lag. Therefore, $g_1(s)$ and $g_2(s)$ are NMP.

2.2 Formulation of Control Problem

For any design procedure to yield a control algorithm which works satisfactorily in a real environment, the following must be specified:

- process model

- model uncertainty bounds
- type of inputs (i.e., setpoints and disturbances)
- performance objectives

Omission of any of these four items invariably leads to controllers which fail in practice. *Every* design or tuning procedure is centered around a *process model* whose complexity can vary. The experience of the operator who "knows" how the plant responds to certain inputs is the simplest kind of model. At the other end of the scale would be, for example, a coupled system of nonlinear partial differential equations. Neglecting *model uncertainty* leads to controllers which are too "tight" and which are likely to become unstable in the real operating environment. It is physically impossible to design controllers which work well for *all* types of setpoint changes and disturbances. The designer has to decide which *inputs* are most important and most frequent, and accept inferior performance if the inputs encountered during the operation are not exactly equal to the inputs assumed for the design. Finally, the designer has to specify what is meant by "good *performance*" for the particular problem at hand. What is good in one case might be entirely unacceptable in another.

2.2.1 Process Model

Chemical processing systems are often most conveniently described by infinite dimensional linear time invariant models (transfer functions with delays). Thus, any book on process control must discuss how to design controllers based on this type of model. However, a rigorous theoretical treatment of time-delay systems is very complex and more of mathematical than of practical interest. The reason is that there are extremely few true time-delay processes encountered in practice. In general, a process modeled with a delay is actually a finite dimensional system of very high order whose dynamic behavior can be *approximated* by a delay. The delay in this case is simply a convenient modeling tool which does not have to be treated rigorously in the design procedure. Also, as we will argue in more detail in the next section, *all* models are inadequate at high frequencies. In the low-frequency range, the delay can be approximated arbitrarily well by a finite dimensional (i.e., delay-free) model. In the high-frequency range the difference between the delay and the finite dimensional approximation can be treated as "model uncertainty" and the controller can be designed accordingly. (One key objective of this book is to show how to account for model uncertainty in the controller design procedure.)

Strictly speaking, most results obtained in this book are only applicable to systems described by rational transfer function models. In a less rigorous manner

we will extrapolate the results to systems involving time delays, in particular when this allows us to obtain compact analytical expressions. The mathematically precise reader should interpret the time delays in our equations as a shorthand notation for a very high-order Padé approximation.

2.2.2 Model Uncertainty Description

Linear time invariant models of the type used throughout this book describe actual plant dynamics only approximately. The "model uncertainty" can have several different sources. Most important, real processes are nonlinear. If the process model is obtained via *linearization*, then it is accurate only in the neighborhood of the reference state chosen for the linearization.

In other cases the process might be represented quite accurately by a linear model. However, *different operating conditions* could lead to changes in the parameters in the linear model. For example, increased throughput and flowrates usually result in smaller deadtimes and time constants.

In the two cases above, the sources and the structure of the "uncertainty" are known quite accurately. However, there is always some "true" uncertainty even when the underlying process is essentially linear: the physical parameters are never known exactly and fast dynamic phenomena (e.g., valve dynamics) are usually neglected in the model. Therefore, at high frequencies, even the model order is unknown.

Uncertainty can be described in many different ways: bounds on the parameters of a linear model, bounds on nonlinearities, frequency domain bounds, etc. To account for model uncertainty we will assume in this book that the dynamic behavior of a plant is described not by a single linear time invariant model but by a *family* Π of linear time invariant models. This is a somewhat primitive uncertainty description especially when the effect of nonlinearities is to be captured, but it is the only feasible approach at present.

Throughout this book the family Π of plants will be defined in the frequency domain. We will assume that the transfer function magnitude and phase at a particular frequency ω is not confined to a point but can lie in a *region* $\pi(\omega)$ on the Nyquist plane. In general, the region can have a very complex shape as the following examples illustrate.

Assume, for example, that frequency response experiments allowed the designer to establish an upper and a lower bound on both the magnitude $|p(i\omega)|$ and the phase $\arg\{p(i\omega)\}$ of the real plant. At a particular frequency these bounds give rise to the sector shaped region $\pi(\omega)$ in Fig. 2.2-1A.

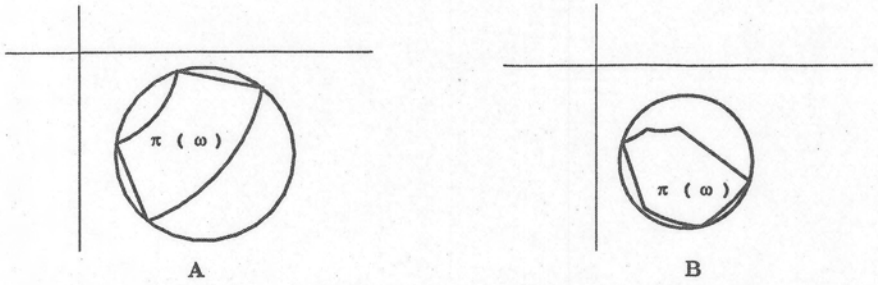

Figure 2.2-1. Uncertainty regions resulting from gain and phase bounds (A) and parametric bounds (B).

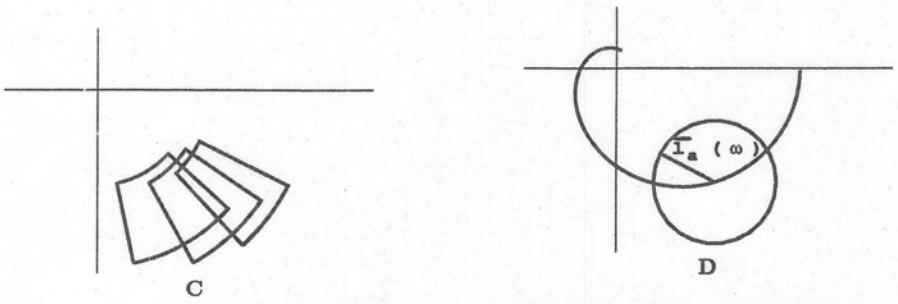

Figure 2.2-1. Uncertainty "band" (C). Uncertainty disk for additive uncertainty (D).

On the other hand, step response experiments might have yielded a first-order model with deadtime ($\tilde{p}(s) = ke^{-s\theta}(\tau s+1)^{-1}$) with upper and lower bounds on the parameters k, θ and τ. The resulting uncertainty region is shown for a particular frequency in Fig. 2.2-1B. The union of all regions $\pi(\omega)$ constitutes the family of possible plants Π and can be viewed as a "fuzzy" Nyquist plot or Nyquist band (Fig. 2.2-1C).

Regions of the general shape seen in these examples have complex mathematical descriptions and are very difficult to deal with in the context of control system design. Therefore, in most of this book we will assume the regions to be disk shaped with radius $\bar{\ell}_a(\omega)$ (Fig. 2.2-1D). Any complex region can be approximated by a disk with more or less conservatism (see Fig. 2.2-1A and B). Algebraically, the family Π of plants described by the disk is defined by:

$$\Pi = \{p : |p(i\omega) - \tilde{p}(i\omega)| \leq \bar{\ell}_a(\omega)\} \qquad (2.2-1)$$

Here $\tilde{p}(i\omega)$ is the *nominal plant* or the *model* defining the center of all the disk-shaped regions. Any member of the family Π satisfies

$$p(i\omega) = \tilde{p}(i\omega) + \ell_a(i\omega) \qquad (2.2-2)$$

with

$$|\ell_a(i\omega)| \leq \bar{\ell}_a(\omega) \qquad (2.2-3)$$

Equation (2.2–2) is referred to as an *additive* uncertainty description and (2.2–3) states a bound on the allowed additive uncertainty.

If we define

$$\ell_m(i\omega) = \frac{\ell_a(i\omega)}{\tilde{p}(i\omega)} \qquad (2.2-4)$$

and

$$\bar{\ell}_m(\omega) = \frac{\bar{\ell}_a(\omega)}{|\tilde{p}(i\omega)|} \qquad (2.2-5)$$

then the family Π (2.2–1) can be represented as

$$\Pi = \left\{ p : \frac{|p(i\omega) - \tilde{p}(i\omega)|}{|\tilde{p}(i\omega)|} \leq \bar{\ell}_m(\omega) \right\} \qquad (2.2-6)$$

Thus, any member of the family Π satisfies

$$p(i\omega) = \tilde{p}(i\omega)(1 + \ell_m(i\omega)) \qquad (2.2-7)$$

with

$$|\ell_m(i\omega)| \leq \bar{\ell}_m(\omega) \qquad (2.2-8)$$

Equation (2.2–7) is referred to as a *multiplicative* uncertainty description and (2.2–8) states a bound on the allowed multiplicative uncertainty. A typical plot of $\bar{\ell}_m(\omega)$ is shown in Fig. 2.2-2. The uncertainty usually increases with frequency and eventually exceeds or becomes equal to unity. The reason is that our models tend to describe well the steady state and low-frequency behavior of processes but become inaccurate for high-frequency inputs.

If a zero is located on the imaginary axis at ω^* for some $p \in \Pi$ then $p(i\omega^*) = 0$ and $|\ell_m(i\omega^*)| = 1$. If $\bar{\ell}_m(\omega) > 1$ for $\omega > \omega^*$, then the set Π allows zeros of $p(s)$ sufficiently far from the the origin "to cross" the imaginary axis. Also, for $\omega > \omega^*$ the disk-shaped regions include the origin which implies that the phase is completely unknown.

On the other hand, if a pole is located on the imaginary axis at ω^* for some $p \in \Pi$ then $|p(i\omega^*)| = \infty$ and therefore $\bar{\ell}_a(\omega^*) = \bar{\ell}_m(\omega^*) = \infty$. Thus the additive and multiplicative uncertainty are unsuitable to describe sets of plants where poles can cross the imaginary axis or equivalently, the number of RHP poles can vary.

Figure 2.2-3 shows a block diagram representation of the two types of uncertainties.

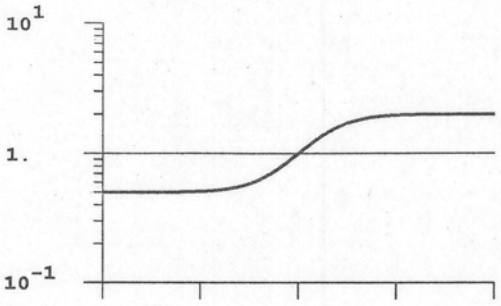

Figure 2.2-2. Typical form of multiplicative uncertainty bound $\bar{\ell}_m$.



Figure 2.2-3. Additive (ℓ_a) and multiplicative (ℓ_m) uncertainty.

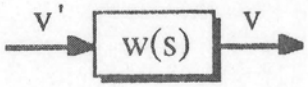

Figure 2.2-4. Weight w transforming normalized input v' to physical input v.

2.2.3 Input Specification

We refer to all external signals entering the feedback loop at some point as inputs. The symbol v will be used to symbolize these inputs which could be setpoint changes or disturbances. For the control system design the inputs have to be specified. The inputs will be *normalized* for notational convenience (Fig. 2.2-4). It will be assumed that an input v entering the control loop is generated by passing the normalized input v' through a transfer function block $w(s)$, sometimes referred to as an *input weight.*

Our design procedures will be able to deal with two types of *normalized* input specifications:

Specific input (impulse):

$$v'(s) = 1 \tag{2.2-9}$$

Set of bounded inputs (all inputs with 2-norm bounded by unity):

$$\mathcal{V}' = \left\{ v' : \|v'\|_2^2 \triangleq \int_0^\infty v'^2(t)dt \leq 1 \right\} \tag{2.2-10}$$

Very often in process control we know quite well what type of inputs to expect. For example, setpoint changes usually have the shape of steps or ramps, disturbances can often be modeled as steps, entering a lag. If the control system design is to be carried out to accommodate a *specific* expected *input* v, such as a step setpoint change, then the designer specifies the input weight w to be an integrator $(1/s)$.

Specific input:

$$v = wv' = w \tag{2.2-11}$$

However, sometimes the closed-loop performance can be quite sensitive to the choice of input; if the input assumed for the design is not exactly equal to the input encountered in practice the performance can deteriorate significantly. Then it is more meaningful to consider a set of inputs containing the input which is encountered most frequently together with other "similar" inputs. According to (2.2–10) we will consider the

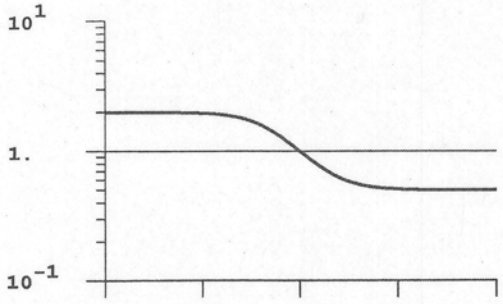

Figure 2.2-5. Typical form of input weight w.

Set of bounded inputs:

$$\mathcal{V} = \left\{ v : \|v'\|_2^2 = \left\|\frac{v}{w}\right\|_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left|\frac{v(i\omega)}{w(i\omega)}\right|^2 d\omega \leq 1 \right\} \qquad (2.2-12)$$

Note that the frequency integral used in (2.2–12) for the 2-norm is equal to the time integral in (2.2–10) through Parseval's theorem.

The expression (2.2–12) can be interpreted as follows: if the spectrum of the input v is narrow and concentrated near ω^* (i.e., the input looks almost like sin $\omega^* t$) then the input power (amplitude) is limited to $|w(i\omega^*)|$. A typical input weight is shown in Fig. 2.2-5 indicating that high-frequency inputs are expected to have small amplitudes.

Assume, for example, that we expect unit step inputs v. Then we can define a weight w such that a step lies in the set (2.2–12). For example the weight

$$w(s) = \frac{s+\beta}{s\sqrt{2\beta}} \quad \beta > 0 \qquad (2.2-13)$$

has the desired characteristics: the input

$$v' = \frac{\sqrt{2\beta}}{s+\beta} \qquad (2.2-14)$$

satisfies

$$\|v'\|_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty}\frac{2\beta}{\omega^2+\beta^2}\,d\omega = 1 \qquad (2.2-15)$$

and gives rise to a step v when passing through the weight (2.2–13). However, the characterization (2.2–12) with the weight (2.2–13) contains also many other

signals v which are "similar" to steps. For example, consider the input

$$v' = \frac{\sqrt{2\alpha}}{s+\alpha} \qquad \alpha > 0, \alpha \neq \beta \qquad (2.2-16)$$

which also satisfies (2.2–10). After passing through the weight the signal becomes

$$v = \sqrt{\frac{\alpha}{\beta}} \frac{s+\beta}{(s+\alpha)s} \qquad (2.2-17)$$

which is a step modified by a lead (for $\alpha > \beta$) or a lag (for $\alpha < \beta$). Thus, if the controller is designed for the set (2.2–12) with weight (2.2–13) it will work well not only for steps but also for "modified steps," for example of the form (2.2–17).

In order for a physically realizable controller to exist, the external signals which are assumed to enter the control loop have to be bounded. Unbounded external signals can only be corrected by unbounded control actions. Therefore, they do not give rise to a meaningful control problem formulation. In general, the effect of the disturbances on the output (d) is bounded and the disturbance transfer function p_d (Fig. 2.1-1A) is stable. If the plant is open loop unstable and a (bounded) disturbance enters at the plant *input*, its effect on the output is usually unbounded and p_d is unstable. In this special case the unstable poles of p_d are a subset of the unstable poles of the plant p. Therefore, a controller with bounded control action can be designed, which provides assymptotic disturbance rejection.

2.2.4 Control Objectives

The ultimate objective of control system design is clearly that the controller works "well" when implemented on the real plant. This goal can be best understood if we decompose it into a series of subobjectives. Because we assume that the model is at least an approximate description of the true plant it is reasonable to require stability when the controller is applied to the plant *model*. Thus, the minimal requirement on the closed-loop system is *nominal* stability. In Sec. 2.3 we will introduce the concept of *internal stability* to derive the conditions for nominal stability which must be satisfied through appropriate controller design.

As a measure of performance we will generally use the Integral Square Error (ISE) because of its mathematical convenience. In Sec. 2.4 we will formulate the optimal control problem for minimizing the ISE.

It is essential that our knowledge about the model uncertainty be incorporated into the controller design procedure. Otherwise the controllers are bound to fail in the real world where the actual plant behavior can be quite different from that

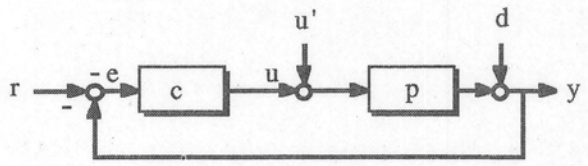

Figure 2.3-1. Block diagram for discussion of internal stability.

of the model. We will require that the controller be designed such that the closed-loop system is stable and meets the performance specifications for *all* members of the family Π of possible plants defined in Sec. 2.2.2. If a closed-loop property holds for a *family* of plants we will refer to it as *robust*. Thus, our controller design objectives are *robust stability* and *robust performance*. In Secs. 2.5 and 2.6 we will derive conditions for robust stability and robust performance, respectively.

2.3 Internal Stability

The signals between the blocks constituting a control system are subject to (possibly very small) errors. In practice it cannot be tolerated that these small errors lead to unbounded signals at some other location in the control system. This motivates the following definition:

Definition 2.3-1. *A control system is internally stable if bounded signals injected at any point of the control system generate bounded responses at any other point.*

Definition 2.3-2. *A linear time invariant control system is internally stable if the transfer functions between any two points of the control system are stable — i.e., have all poles in the open Left Half Plane (LHP).*

In a control system many different points can be selected for signal injection and observation but most of the choices are equivalent for checking internal stability. For example, for the system in Fig. 2.3-1, y and e differ only by a bounded signal (r) and their observation reveals the same information about internal stability.

Also, from the point of view of internal stability, the effect of d and r on u is equivalent. Simple arguments of this type reveal that there are only two "independent" outputs, which can be chosen as y and u, and two "independent" inputs, which can be chosen as r and u'. Thus, the classic feedback system is

stable if and only if all elements in the 2×2 transfer matrix in (2.3–1) have all their poles in the open LHP.

$$\begin{pmatrix} y \\ u \end{pmatrix} = \begin{pmatrix} \frac{pc}{1+pc} & \frac{p}{1+pc} \\ \frac{c}{1+pc} & \frac{-pc}{1+pc} \end{pmatrix} \begin{pmatrix} r \\ u' \end{pmatrix} \qquad (2.3-1)$$

It should be noted that the concept of internal stability is more complete than the usual stability concept of undergraduate textbooks where system stability is checked by examining the roots of the characteristic equation $1+pc = 0$. Consider the example

$$c = \frac{-s+1}{s+1} \qquad (2.3-2)$$

$$p = \frac{1}{-s+1} \qquad (2.3-3)$$

for which the characteristic equation

$$1 + pc = 1 + \frac{1}{s+1} = 0 \qquad (2.3-4)$$

has a single root at $s = -2$ which would indicate stability.

On the other hand, the transfer matrix in (2.3–1) evaluated for this example

$$\begin{pmatrix} y \\ u \end{pmatrix} = \begin{pmatrix} \frac{1}{s+2} & \frac{s+1}{(-s+1)(s+2)} \\ \frac{-s+1}{s+2} & \frac{-1}{s+2} \end{pmatrix} \begin{pmatrix} r \\ u' \end{pmatrix} \qquad (2.3-5)$$

shows that any bounded input u' leads to an unbounded output y. Thus, the system is not internally stable.

Note that the RHP zero of the controller (2.3–2) cancels exactly the RHP pole of the plant (2.3–3) when (2.3–4) is formed. The internal stability concept clarifies the fact that even *exact* cancellation is not enough to guarantee the stability of the system.

If both p and c are stable then the unstable poles in (2.3–1) can arise from $(1 + pc)^{-1}$ only. In this special case it is necessary and sufficient for internal stability that all the roots of the characteristic equation $1 + pc = 0$ are in the open LHP.

2.4 Nominal Performance

The most basic objective of a feedback controller is to keep the error between the plant output y and the reference r small when the overall system is affected by external signals r and d. In order to quantify performance, a measure of

"smallness" for the error has to be defined. Furthermore, the set of external input signals has to be specified for which the error is to be made small. Control system design techniques differ in the way they measure "magnitude" and how they define the permissible set of external inputs. Two particularly popular approaches are outlined in Secs. 2.4.4 and 2.4.5.

2.4.1 Sensitivity and Complementary Sensitivity Function

If we set $p_d = p_m = 1$, then the most important relationships between the inputs and outputs in Fig. 2.1-1A are

$$\frac{e}{d-r} = \frac{y}{d} = \frac{1}{1+pc(s)} \triangleq \epsilon(s) \qquad (2.4-1)$$

$$\frac{y}{r-n} = \frac{pc(s)}{1+pc(s)} \triangleq \eta(s) \qquad (2.4-2)$$

where the error e is defined as $e = y - r$

The *sensitivity function* $\epsilon(s)$[2] relates the external inputs $d - r$ to the error e. It also expresses the effect of the disturbance d on the output y. The sensitivity $\epsilon(s)$ is of primary importance in judging the performance of a feedback controller. It is desirable to make $\epsilon(s)$ as "small" as possible. If pc is strictly proper (which is always the case for physical systems) then

$$\lim_{s\to+\infty} pc = 0 \qquad (2.4-3)$$

This implies that

$$\lim_{\omega\to\infty} |\epsilon(i\omega)| = \lim_{\omega\to\infty} \left|\frac{1}{1+pc(i\omega)}\right| = 1 \qquad (2.4-4)$$

Thus $|\epsilon|$ can be made small only over a finite frequency range. Physical limitations do not allow "perfect control" ($\epsilon = 0$). A typical plot of $|\epsilon|$ is shown in Fig. 2.4-1.

The frequency ω_B at which $|\epsilon|$ exceeds $1/\sqrt{2}$ will be called the *system bandwidth.*

$$|\epsilon(\omega)| < 1/\sqrt{2} \qquad \forall\omega < \omega_B \qquad (2.4-5)$$

The bandwidth ω_B can serve as a simple closed-loop performance measure.

The *complementary sensitivity function* $\eta(s)$ derives its name from the equality

$$\epsilon(s) + \eta(s) = 1 \qquad (2.4-6)$$

[2]The symbol ϵ was chosen because of the Greek word $\epsilon\upsilon\alpha\iota\sigma\vartheta\eta\sigma\iota\alpha$ for sensitivity. It was then most natural to denote the complementary sensitivity by η.

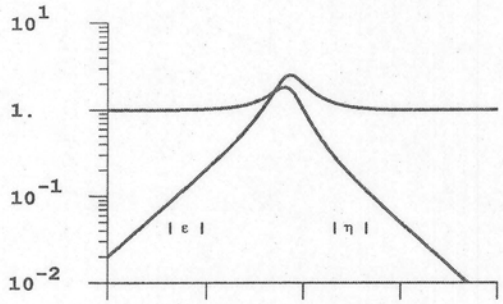

Figure 2.4-1. Typical form of sensitivity ϵ and complementary sensitivity η.

The complementary sensitivity $\eta(s)$ relates the reference r to the output y (2.4–2). From this point of view, $\eta(s)$ should be made as close to unity as possible. However, because of (2.4–3)

$$\lim_{\omega\to\infty} |\eta(i\omega)| = \lim_{\omega\to\infty} \left|\frac{pc(i\omega)}{1+pc}\right| = 0 \qquad (2.4-7)$$

Thus $|\eta|$ can be made equal to unity over a finite frequency range only.

The complementary sensitivity $\eta(s)$ also expresses the effect of measurement noise n on y (2.4–2). From this point of view $\eta(s)$ should be made small. This illustrates one of the basic trade-offs in feedback design: good setpoint tracking and disturbance rejection ($\epsilon \approx 0, \eta \approx 1$) has to be traded off against suppression of measurement noise ($\epsilon \approx 1, \eta \approx 0$). The optimal compromise can be found via stochastic optimal control theory. Experience has shown that this particular trade-off is usually irrelevant in process control. Measurement noise is often small and η resulting from stochastic optimal control considerations is close to unity over a wide frequency range. As we will learn later, model uncertainty also imposes an upper bound on the magnitude of η. With the poor models generally available for process control, this bound is much more restrictive than the measurement noise constraint. Therefore, measurement noise will be largely neglected in this book.

2.4.2 Two-Degree-of-Freedom Controller

From (2.4–1) we see that apart from the sign, r and d have the *same* effect on e.

$$e = \epsilon(d-r) = \frac{1}{1+pc}(d-r) \qquad (2.4-8)$$

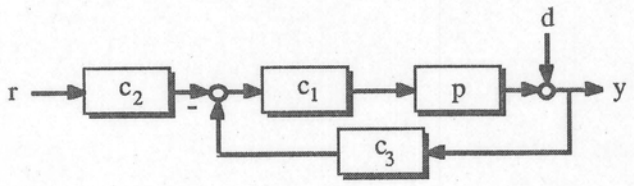

Figure 2.4-2. Block diagram for two-degree-of-freedom controller.

The control system has only "one degree of freedom." If the reference signal r and the disturbance d vary in a similar manner then c can be chosen such that e is satisfactory. If r and d behave differently, for example, if one expects step changes in r and ramp changes in d, then c has to be chosen either for good response to steps in r or good response to ramps in d or else some compromise has to be found. If there are strict requirements on both setpoint tracking and disturbance suppression, an acceptable compromise might not exist. Then additional controller blocks have to be introduced into the system (Fig. 2.4-2) to allow independent controller adjustments for both r and d. The relationship between the inputs and the error, which demonstrates the "two degrees of freedom," is:

$$e = \frac{1}{1 + pc_1c_3} d + \left(\frac{pc_1c_2}{1 + pc_1c_3} - 1 \right) r \qquad (2.4 - 9)$$

Here c_1c_3 can be selected first for good disturbance rejection and then the *prefilter* c_2 can be chosen independently for good setpoint tracking. As we will show in Chap.5, all three controller blocks are necessary, in general, to minimize the impact of the controller structure on the achievable trajectory tracking and disturbance rejection performance. In some special cases two blocks (c_1 and c_2, or c_1 and c_3) are sufficient.

We see from (2.4–8) that the effects of the inputs d and $-r$ on the error e are the same. Therefore, in the rest of the chapter we will use the symbol v to denote an input, where $v = d$ or $v = -r$. If the control system is to be designed for both trajectory tracking and disturbance rejection, then these two design tasks can be handled sequentially and independently through the two-degree-of-freedom structure.

2.4.3 Asymptotic Properties of the Closed-Loop Response (System Type)

The disturbances encountered in the chemical process industries are frequently slowly varying and can be approximated well by steps or ramps. As a basic closed loop performance specification the error is often required to vanish asymptotically for these types of inputs. Obviously, it is only meaningful to check if this specification is met when the closed loop system is stable. Then the asymptotic properties can be assessed from the Final Value Theorem. For example, for $\epsilon(s)$ not to exhibit any offset to step inputs it is necessary and sufficient that

$$\lim_{s\to 0}\left[s(\epsilon(s)\frac{1}{s})\right] = 0 \qquad (2.4-10)$$

It has become customary to define "System Types" to classify the asymptotic behavior.

Definition 2.4-1. *Let $g(s)$ be the open loop transfer function and let m be the largest integer for which*

$$\lim_{s\to 0} s^m g(s) \neq 0$$

Then the system $g(s)$ is said to be of Type m. (In other words, a Type m system has m poles at the origin.)

Theorem 2.4-1. *Let the open loop system $g(s)$ be of Type m and assume that the closed-loop system with unity feedback is stable. Then the sensitivity function $\epsilon(s) = (1+g(s))^{-1}$ satisfies*

Type m:

$$\lim_{s\to 0}\frac{\epsilon(s)}{s^k} = 0 \qquad 0 \leq k < m \qquad (2.4-11)$$

Also, as $t \to \infty$ the closed loop system tracks perfectly setpoint changes (rejects perfectly disturbances) of the form $\sum_{k=0}^{m} a_k s^{-k}$ where a_k are real constants.

Proof. Follows directly from Final Value Theorem.

Specifically, in order for a closed loop system to track steps without offset the sensitivity function $\epsilon(s)$ has to have a zero at the origin. This is the case if the open loop system $g(s)$ has a pole at the origin. That is, $g(s)$ has to be of Type 1. Similarly offset free tracking of ramps requires $g(s)$ to have two poles at the origin (Type 2).

2.4.4 Linear Quadratic (H_2-) Optimal Control

The idea of optimal control is to express the control objectives in the form of an optimization problem and to determine the controller structure and parameters from its solution. In the formulation practical relevance always has to be sacrificed for mathematical convenience – i.e., an optimization problem which can be solved. Thus the "optimality" of a controller does not imply its usefulness. Whether a controller determined from optimal control principles solves a specific practical problem, depends to a large extent on how well the control system designer has approximated reality to fit the mathematical formalism. For several decades only the so-called *Linear Quadratic (H_2-) Optimal Control* was available. In the seventies the new paradigm of *H_∞-Optimal Control* emerged. Depending on the problem, the assumptions inherent in one or the other optimal control formalism can be more applicable and can allow a better approximation and thus a better solution.

In this section we will concentrate on the H_2-optimal control formulation. The controller c is determined such that the integral square error

$$\|e\|_2^2 = \int_0^\infty e^2(t)dt \qquad (2.4-12)$$

is minimized for a particular input v. Referring back to the introductory discussion of Sec. 2.4. we note that the "measure of smallness" is the 2-norm of the error ($\|e\|_2$) and the "input set" is just one specific input v. Using Parseval's theorem, we can state the optimization problem in the frequency domain

$$\min_c \|e\|_2^2 = \min_c \frac{1}{2\pi}\int_{-\infty}^{\infty} |e(i\omega)|^2 d\omega \qquad (2.4-13)$$

or upon substitution for e from (2.4–8) and for v from (2.2–11)

$$\min_c \|\epsilon w\|_2^2 = \min_c \frac{1}{2\pi}\int_{-\infty}^{\infty} |\epsilon(i\omega)w(i\omega)|^2 d\omega \qquad (2.4-14)$$

For a comparison with $H_\infty-$ optimal control and for the extension to the multivariable case an alternate interpretation of (2.4–14) is useful. The integrand is the magnitude of the sensitivity function *weighted* by w, where w depends on the input v. The integral operation generates the *average* of this weighted magnitude. Thus, the H_2-optimal controller minimizes the average magnitude or, in mathematical terms, *minimizes the 2-norm of the sensitivity function ϵ weighted by w.*

Throughout the book we will adopt this latter more mathematical interpretation of (2.4–14) because it is more practical. If w is chosen to represent a particular input according to (2.2–11) then it is rare that the controller which

solves (2.4–14) satisfies *all* the practical design requirements. For example, the minimum ISE response might be too oscillatory or the overshoot too large. Therefore, it is common practice for the designer to use w as a "tuning parameter." The weight w is then varied until the closed-loop response characteristics are satisfactory. Naturally, the input type is a very effective guide for the weight selection. The advantage of using w for tuning rather than setting up an alternate optimization problem which addresses the practical requirements more directly, is that very effective methods are available for solving (2.2–14).

For the optimization (2.4–14) clearly only the magnitude of the weight w matters but not its phase characteristics — i.e., whether it is MP or NMP, stable or unstable. However, if w reflects the effect of a specific unstable input v, then the optimal ϵ has to satisfy certain internal stability requirements to guarantee that the error e vanishes as $t \to \infty$ and the objective (2.4–12) is bounded (see Chap. 5).

Usually the problem (2.4–13) leads to controllers which are too aggressive because emphasis is placed on the error only and not on the manipulated variable. More general objective functions than (2.4–12), including, for example, a term u^2, to penalize excessive input variations, are well known and are not discussed here. Entirely equivalent stochastic formulations are also available.

2.4.5 H_∞-Optimal Control

Here, the inputs v are assumed to belong to the set $\mathcal{V}$ of norm-bounded functions with a frequency dependent weight w as discussed in Sec. 2.2.3.

$$\mathcal{V} = \left\{ v : \|v'\|_2^2 = \left\|\frac{v}{w}\right\|_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left|\frac{v(i\omega)}{w(i\omega)}\right|^2 d\omega \le 1 \right\} \qquad (2.4-15)$$

The input class defined by (2.4–15) contains a large variety of spectra including possibly steps, pulses and narrow-band signals of various frequencies. This input characterization is sometimes more meaningful than what is required for the H_2-optimal control problem. Recall that there we assumed a single fixed input.

Each input $v \in \mathcal{V}$ gives rise to an error e. The H_∞-optimal controller is designed to minimize the *worst* error which can result from any input $v \in \mathcal{V}$:

$$\min_c \sup_{v\in\mathcal{V}} \|e\|_2 = \min_c \sup_{v'\in\mathcal{V}'} \|\epsilon w v'\|_2 \qquad (2.4-16)$$

Here we performed the same substitution for e as in (2.4–14). The worst error can be bounded as follows

$$\sup_{v'\in\mathcal{V}'} \|\epsilon w v'\|_2^2 = \sup_{v'\in\mathcal{V}'} \frac{1}{2\pi}\int_{-\infty}^{\infty} |\epsilon w v'|^2 d\omega \le \sup_{\omega} |\epsilon w|^2 \sup_{v'\in\mathcal{V}'} \frac{1}{2\pi}\int_{-\infty}^{\infty} |v'|^2 d\omega$$

Then, using the definition of the set $\mathcal{V}'$ (2.2–10) we find

$$\sup_{v' \in \mathcal{V}'} \|\epsilon w v'\|_2^2 \leq \sup_{\omega} |\epsilon w|^2 \qquad (2.4-17)$$

It can be shown that there exists a sequence of inputs $v_k^* \in \mathcal{V}'$ for which $\lim_{k\to\infty} \|\epsilon w v_k^*\|_2 = \sup_\omega |\epsilon w|$. Thus (2.4–17) is actually an equality. The characteristics of v_k^* can be understood from the following argument. Let us assume that $|\epsilon w(i\omega)|$ reaches its maximum for $\omega = \omega^*$. Then, the spectrum of the input v_k^* which makes (2.4–17) an equality as $k \to \infty$ has to be increasingly narrow-band and concentrated near ω^* as k increases.

We define the ∞-norm of the weighted sensitivity function

$$\|\epsilon w\|_\infty \triangleq \sup_{\omega} |\epsilon w| \qquad (2.4-18)$$

and from (2.4–16)-(2.4–18) the H_∞-optimal control problem is

$$\min_c \|\epsilon w\|_\infty = \min_c \sup_{\omega} |\epsilon w(i\omega)| \qquad (2.4-19)$$

Thus the H_∞-optimal controller minimizes the maximum magnitude or, in mathematical terms, *minimizes the ∞-norm of the sensitivity function ϵ weighted by w.*

According to this frequency domain interpretation the H_2-optimal controller minimizes the *average* value and the H_∞-optimal controller the *peak value* of the magnitude of the weighted sensitivity function. For the optimization (2.4–19) clearly only the magnitude of the weight w matters, but not its phase characteristics — i.e., whether it is MP or NMP, stable or unstable.

As in the case of H_2-optimal control it is more practical to think of w as a tuning parameter affecting the solution of (2.4–19) rather than to adhere to the formal problem definition (2.4–16). Let us assume that for a particular w the optimal value of the objective function (2.4–18) is k. Then (2.4–19) implies that $|\epsilon| \leq k|w|^{-1}, \forall\omega$. This suggests that the designer can impose a bound on the sensitivity function

$$|\epsilon(i\omega)| < |w(i\omega)|^{-1} \qquad \forall\omega \qquad (2.4-20)$$

which the H_∞-optimal controller aims to satisfy. If a controller is found such that

$$\|\epsilon w\|_\infty < 1 \qquad (2.4-21)$$

then the bound (2.4–20) is met. Therefore, the H_∞-performance requirement is usually written in the form (2.4–21).

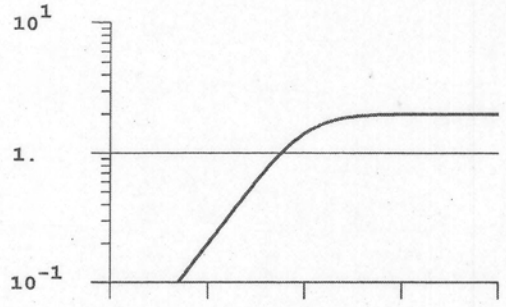

Figure 2.4-3. Typical bound (w^{-1}) imposed on sensitivity function.

Naturally, the types of inputs which are expected will guide the selection of the weight w. Figure 2.4-3 shows a typical weight. It implies that the expected inputs are mostly slowly varying and that high frequency signals have smaller amplitudes. The bound w^{-1} forces the sensitivity function to be small in the frequency range where large inputs are anticipated. The weight w in Fig. 2.2-5 would indicate much milder performance requirements. Expressing the control objective as a bound on the sensitivity function ϵ has much practical appeal. Often the designer wishes to specify a minimum bandwidth and to limit the maximum peak of the sensitivity function to avoid excessive disturbance amplification. In the H_∞-optimal control formulation this can be done explicitly. Most important, however, this new formalism allows one to address the robust controller design problem in a consistent framework, as will be discussed in the next section.

Recall that for high frequencies $|pc|$ is small and therefore

$$|\epsilon| = \left|\frac{1}{1+pc}\right| \cong 1 \quad \omega \quad \text{large} \qquad (2.4-22)$$

Thus tight performance specifications are only meaningful in the low-frequency range, where $|\tilde{p}c|$ is large and $|\epsilon| \cong |pc|^{-1}$. Then the performance specification (2.4–21) reduces to

$$|pc| > |w| \quad \omega \quad \text{small} \qquad (2.4-23)$$

That is, the loop gain $|pc|$ has to be shaped to fall above the performance weight $|w|$.

2.5 Robust Stability

We wish to derive conditions for the robust stability of the family Π of plants defined by (2.2–6). For this purpose we will employ the Nyquist stability criterion.

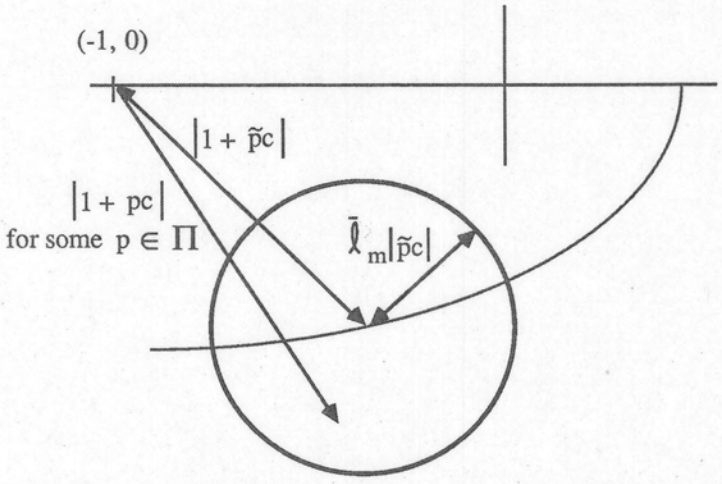

Figure 2.5-1. Diagram for graphical derivation of robust stability and robust performance conditions.

Let us denote by $N(k, g(s))$ the net number of clockwise encirclements of the point $(k, 0)$ by the image of the Nyquist D contour under $g(s)$. We will assume that all $p \epsilon \Pi$ have the same number (n) of RHP poles. Then we have robust stability for a specific controller c if and only if

$$N(-1, pc) = -n \qquad \forall p \in \Pi \tag{2.5-1}$$

In particular it is necessary that the nominal plant $\tilde{p}$ be closed loop stable:

$$N(-1, \tilde{p}c) = -n \tag{2.5-2}$$

Assuming that (2.5–2) holds, (2.5–1) will hold if and only if the Nyquist band which comprises all $p \in \Pi$ does not include the point (-1,0). From the simple geometric argument in Fig. 2.5-1 it is clear that this will be the case if and only if the distance of $\tilde{p}c$ from the point (-1,0) — i.e., $|1 + \tilde{p}c|$, exceeds the disk radius $|\tilde{p}c(i\omega)|\bar{\ell}_m(\omega)$:

$$|1 + \tilde{p}c(i\omega)| > |\tilde{p}c(i\omega)|\bar{\ell}_m(\omega) \qquad \forall\omega$$

or

$$\left|\frac{\tilde{p}c(i\omega)}{1 + \tilde{p}c(i\omega)}\right|\bar{\ell}_m(\omega) = |\tilde{\eta}(i\omega)|\bar{\ell}_m(\omega) < 1 \qquad \forall\omega \tag{2.5-3}$$

where $\tilde{\eta} = \tilde{p}c(1 + \tilde{p}c)^{-1}$. We have proven the following theorem.

Theorem 2.5-1 (Robust Stability). *Assume that all plants p in the family Π*

$$\Pi = \left\{ p : \left| \frac{p - \tilde{p}(i\omega)}{\tilde{p}} \right| \leq \bar{\ell}_m(\omega) \right\} \qquad (2.2-6)$$

have the same number of RHP poles and that a particular controller c stabilizes the nominal plant $\tilde{p}$. Then the system is robustly stable with the controller c if and only if the complementary sensitivity function $\tilde{\eta}(s)$ for the nominal plant $\tilde{p}$ satisfies the following bound

$$\|\tilde{\eta}\bar{\ell}_m\|_\infty \triangleq \sup_\omega |\tilde{\eta}\bar{\ell}_m(\omega)| < 1 \qquad (2.5-4)$$

Let us clarify what it means that Thm. 2.5-1 is not only sufficient for robust stability but also necessary. If condition (2.5–4) is violated then in the set Π defined by (2.2–6) there exists a plant p for which the closed-loop system with the controller c is unstable. However, if the set Π was obtained by approximating the true uncertainty regions with disks (see Sec. 2.2.2) and therefore contains plants not present in the original uncertainty set then condition (2.5–4) is generally only sufficient for the original uncertainty set. Nevertheless, the necessity implies that at least for the set Π defined by (2.2–6), this is the tightest robust stability condition that can be derived.

Theorem 2.5-1 is most instructive for several reasons. First note that robust stability imposes a bound on the *∞-norm of the complementary sensitivity function $\tilde{\eta}$ weighted by $\bar{\ell}_m$*. Recall, on the other hand, that for the H_2-optimal control formulation the performance specifications were expressed in terms of the *2-norm of the sensitivity function $\tilde{\epsilon}$ weighted by w*. The different norms are an indication that performance, robust stability and robust performance cannot be treated "jointly" when the ISE is employed as a performance objective. One of the main motivations for introducing H_∞-optimal control is that the robust stability condition and the performance specification are expressed in terms of the same norm.

Furthermore recall that measurement noise also tends to impose a bound on the magnitude of η (see 2.4–2). However, in process control the constraint imposed by model uncertainty (2.5–4) tends to dominate.

Also note that for high frequencies $|\tilde{p}c|$ is small and therefore

$$|\tilde{\eta}| = \left| \frac{\tilde{p}c}{1 + \tilde{p}c} \right| \cong |\tilde{p}c| \quad \omega \quad \text{large} \qquad (2.5-5)$$

Thus, for large ω (2.5–4) reduces to

$$|\tilde{p}c| < \frac{1}{|\bar{\ell}_m|} \quad \omega \quad \text{large} \qquad (2.5-6)$$

The design implication is that the controller gain for high frequencies is limited by uncertainty. The loop gain $|\tilde{p}c|$ has to be shaped to fall below the uncertainty bound $|\bar{\ell}_m|^{-1}$.

Finally note that in the H_∞-optimal control framework we want to minimize $\|\tilde{\epsilon}w\|_\infty$ for performance (see (2.4–19)) and $\|\tilde{\eta}\bar{\ell}_m\|_\infty$ for robustness. A trade-off between performance and robustness arises from the fact that $\tilde{\epsilon}$ and $\tilde{\eta}$ are not independent (see (2.4–6)) and making one small will automatically make the other one large. This problem is inherent in feedback control and cannot be removed by clever controller design. The objective of a design method for robust control systems is to reach the best compromise between the conflicting objectives of performance and robustness.

2.6 Robust Performance

Robust stability is the minimum requirement a control system has to satisfy to be useful in a practical environment where model uncertainty is an important issue. However, robust stability alone is not enough. If the bound (2.5–4) is satisfied for a family Π, then there exists a particular plant $p \in \Pi$ for which the closed-loop system is on the verge of instability and for which the performance is arbitrarily poor. Thus we also have to make sure that some performance specifications are met for *all* plants in the family Π. We will derive conditions for robust performance when the performance is measured in terms of both the 2-norm and the ∞-norm.

2.6.1 H_2 Performance Objective

Expanding the scope of Sec. 2.4.4 dealing with a single plant p, we are now looking at the family of plants Π defined by (2.2–6). We set as our objective to design the controller such that the error resulting from a specific input d is minimized for the "worst" plant in the family. The worst plant is the one giving rise to the largest error.

$$\min_c \max_{p\in\Pi} \int_0^\infty e^2 dt = \min_c \max_{p\in\Pi} \frac{1}{2\pi} \int_{-\infty}^{\infty} \left|\frac{1}{1+pc}\right|^2 |v|^2 d\omega \qquad (2.6-1)$$

The integral is maximized by maximizing the integrand at every frequency. From geometric arguments (Fig. 2.5-1) we find

$$|1+pc| \geq |1+\tilde{p}c| - |\tilde{p}c|\bar{\ell}_m, \qquad \forall p \in \Pi \qquad (2.6-2)$$

or

$$|\epsilon| = \left|\frac{1}{1+pc}\right| \leq \frac{|\tilde{\epsilon}|}{1-|\tilde{\eta}|\bar{\ell}_m}, \qquad \forall p \in \Pi \qquad (2.6-3)$$

where $\tilde{\epsilon} = (1+\tilde{p}c)^{-1}$. Note that when the robust stability condition (2.5–4) holds the denominator of the RHS of (2.6–3) is positive. Thus (2.6–1) can be rewritten as

$$\min_c \frac{1}{2\pi}\int_{-\infty}^{\infty}\left|\frac{1}{1-|\tilde{\eta}|\bar{\ell}_m}\right|^2|\tilde{\epsilon}v|^2 d\omega \qquad (2.6-4)$$

Because the controller c appears in a very complex manner in (2.6–4), there is little hope of finding a simple solution to this problem. However, for a specific controller c, we can use (2.6–3) to determine the worst error which can result from a specific input v for a set of plants Π.

$$\max_{p\in\Pi}\|e\|_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left|\frac{1}{1-|\tilde{\eta}|\bar{\ell}_m}\right|^2|\tilde{\epsilon}v|^2 d\omega \qquad (2.6-5)$$

The first factor in the integral accounts for the effect of model error. Because of the robust stability condition (2.5–4) the integrand is always bounded. As expected, the observed error increases relative to the nominal error as the stability boundary is approached.

Equation (2.6–5) is a handy analysis tool. After a controller has been designed, a performance bound in terms of ISE can be established. The error which is observed on the unknown "real" system as a result of a specific input v is always less than the established bound – provided the uncertainty description is correct.

2.6.2 H_∞ Performance Objective

If the performance specifications are stated in the H_∞ framework, then we will require that (2.4–21) be met for the "worst" plant

$$\max_{p\in\Pi}\|\epsilon w\|_\infty = \max_{p\in\Pi}\sup_\omega|\epsilon w(i\omega)| < 1 \qquad (2.6-6)$$

or

$$\|\epsilon w\|_\infty = \sup_\omega|\epsilon w(i\omega)| < 1 \qquad \forall p\in\Pi \qquad (2.6-7)$$

Using (2.6–3), (2.6–7) can be rewritten as

$$\frac{|\tilde{\epsilon}w|}{1-|\tilde{\eta}|\bar{\ell}_m} < 1 \qquad \forall\omega \qquad (2.6-8)$$

or

$$|\tilde{\eta}\bar{\ell}_m| + |\tilde{\epsilon}w| < 1 \qquad \forall\omega \qquad (2.6-9)$$

Theorem 2.6-1 (Robust Performance). *Assume that all plants in the family*

$$\Pi = \left\{p : \left|\frac{p(i\omega)-\tilde{p}(i\omega)}{\tilde{p}(i\omega)}\right| \le \bar{\ell}_m(\omega)\right\} \qquad (2.2-6)$$

have the same number of RHP poles. Then the closed-loop system will meet the performance specification

$$\|\epsilon w\|_\infty = \sup_\omega |\epsilon w| < 1 \qquad \forall p \in \Pi$$

if and only if the nominal system is closed-loop stable and the sensitivity function $\tilde{\epsilon}$ *and the complementary sensitivity function* $\tilde{\eta}$ *satisfy*

$$|\tilde{\eta}\bar{\ell}_m| + |\tilde{\epsilon} w| < 1 \qquad \forall \omega \tag{2.6-10}$$

Trivially, robust performance (2.6–10) implies robust stability (2.5–4) and nominal performance (2.4–20). The interdependence of $\tilde{\epsilon}$ and $\tilde{\eta}$ makes it a challenge to meet (2.6–10). Improving the nominal performance (decreasing $|\tilde{\epsilon} w|$) worsens the robustness (increases $|\tilde{\eta}\bar{\ell}_m|$) and pushes the system closer to the point of instability for some plant $p \in \Pi$.

If $\bar{\ell}_m(0) < 1$ then the bound (2.6–10) can be satisfied for low frequencies by making $|\tilde{p}c|$ large; see (2.4–23). If $w < 1$ for high frequencies the bound (2.6–10) is met by choosing $|\tilde{p}c|$ small; see (2.5–6). By shaping the loop gain $|\tilde{p}c|$ to be large for low frequencies and small for high frequencies the robust performance specifications can be met assuming that they are not too tight (w large) in a frequency range where the uncertainty $\bar{\ell}_m$ is large (design by "loop shaping").

A controller which optimizes robust performance solves

$$\min_c \sup_\omega (|\tilde{\eta}\bar{\ell}_m| + |\tilde{\epsilon} w|) \tag{2.6-11}$$

Compare problem (2.6–11) for optimizing robust performance with problem (2.4–19) for optimizing nominal performance. For *nominal performance* the weighted ∞-norm of the sensitivity function is to be minimized. For *robust performance* the ∞-norm ceases to be a suitable measure. As we will learn in Chap. 11 the appropriate measure for robust performance is the Structured Singular Value (SSV) which will be discussed in the context of multivariable systems. The LHS of (2.6–10) turns out to be the SSV of a particular matrix.

We see from (2.6–10) that we can attain robust performance simply by satisfying both the robust stability test (2.5–4) and the nominal performance test (2.4–20) with some margin: If $|\tilde{\eta}\bar{\ell}_m| < \alpha(\omega)$ and $|\tilde{\epsilon} w| < 1 - \alpha(\omega)$ where $\alpha \leq 1, \forall \omega$, then robust performance is automatically guaranteed. Thus, for SISO systems, robust performance is not an objective which is particularly difficult to deal with. This will not be true for MIMO systems.

2.7 Summary

For the design of a feedback controller the following essentials have to be specified:

- process model
- model uncertainty bounds
- type of inputs
- performance objectives

All design techniques aim to make the error resulting from certain external inputs small. This is equivalent to making the sensitivity function

$$\epsilon = \frac{e}{d-r} = \frac{1}{1+pc} \qquad (2.4-1)$$

small. The H_2-optimal controller minimizes the ISE for a particular input or equivalently it minimizes the 2-norm ("average") of ϵ weighted by the input.

H_2:

$$\min_c \int_0^\infty e^2 dt = \min_c \|\epsilon w\|_2^2 \qquad (2.4-14)$$

The H_∞-optimal controller minimizes the worst ISE which can result from a set of inputs or equivalently it minimizes the ∞-norm ("peak") of ϵ weighted by w.

H_∞:

$$\min_c \|\epsilon w\|_\infty = \min_c \sup_\omega |\epsilon w| \qquad (2.4-19)$$

For the suppression of measurement noise and in particular for robust stability the complementary sensitivity $\tilde{\eta}$ should be small in the frequency range where the model uncertainty measured by $\bar{\ell}_m$ is large.

Robust Stability:

$$\|\tilde{\eta}\bar{\ell}_m\|_\infty = \sup_\omega |\tilde{\eta}\bar{\ell}_m| < 1 \qquad (2.5-4)$$

Because

$$\tilde{\epsilon} + \tilde{\eta} = 1 \qquad (2.4-6)$$

a basic tradeoff between nominal performance and robust stability arises.

Robust Performance:

$$|\tilde{\eta}\bar{\ell}_m| + |\tilde{\epsilon}w| < 1 \qquad \forall\omega \qquad (2.6-10)$$

Robust performance is achieved simply by satisfying both nominal performance ($|\tilde{\epsilon}w| < 1$) and robust stability ($|\tilde{\eta}\bar{\ell}_m| < 1$) with some margin.

2.8 References

2.1. The inverse response behavior of NMP systems is proven, for example, by Holt & Morari (1985a).

2.2.2. The uncertainty description via norm-bounded perturbations ("disks") was popularized by Doyle & Stein (1981).

2.3. Desoer & Lin (1985) investigated the stability of a large variety of feedback structures in the same manner as in this section.

2.4.2. Horowitz (1963) is generally credited with having first proposed and studied the two-degree-of-freedom controller.

2.4.4. Newton first presented an analytic feedback design technique to minimize an H_2 objective. The procedure is described in the book by Newton, Gould & Kaiser (1957).

2.4.5. The H_∞-optimal control problem was first formulated by Zames (1981). Closely related ideas proposed by other authors are discussed in his paper. A possible limiting response which proves that (2.4–17) is an equality is proposed by Desoer & Vidyasagar (1975).

2.5. The robust stability condition (at least for the MIMO case) is generally attributed to Doyle & Stein (1981).

2.6.2. The robust performance condition follows as a special case from the MIMO Structured Singular Value Analysis proposed by Doyle (1984, 1985).

Chapter 3

THE SISO IMC STRUCTURE FOR STABLE SYSTEMS

The Internal Model Control (IMC) structure will be introduced as an alternative to the classic feedback structure. Its main advantage is that closed-loop stability is assured simply by choosing a stable IMC controller. Also, closed-loop performance characteristics (like settling time) are related *directly* to controller parameters, which makes on-line tuning of the IMC controller very convenient. A two-step design procedure will be proposed. In the first step the controller is designed for optimal setpoint tracking (disturbance rejection) without regards for input saturation or model uncertainty. In the second step the controller is detuned for robust performance. While this procedure has no inherent optimality characteristics, it constitutes a simple and transparent approach for finding controllers which satisfy all typical practical requirements.

3.1 IMC Structure

The block diagram of the IMC loop is shown in Fig. 3.1-1A. Here p denotes the plant and p_m the measurement device transfer functions. In general neither p nor p_m are known exactly but only their nominal models $\tilde{p}$ and $\tilde{p}_m$ are available. The transfer function p_d describes the effect of the disturbance d' on the process output y. The measurement of y is corrupted by measurement noise n. The controller q determines the value of the input (manipulated variable) u. The control objective is to keep y close to the reference (setpoint) r.

Commonly we will use the simplified block diagram in Fig. 3.1-1B. Here d denotes the effect of the disturbance on the output. Exact knowledge of the output y is assumed ($p_m = 1, n = 0$).

Note that the complete control system to be implemented through computer software or analog hardware is contained in the shaded box in Fig. 3.1-1B. Because in addition to the controller q it includes the plant model $\tilde{p}$ explicitly we

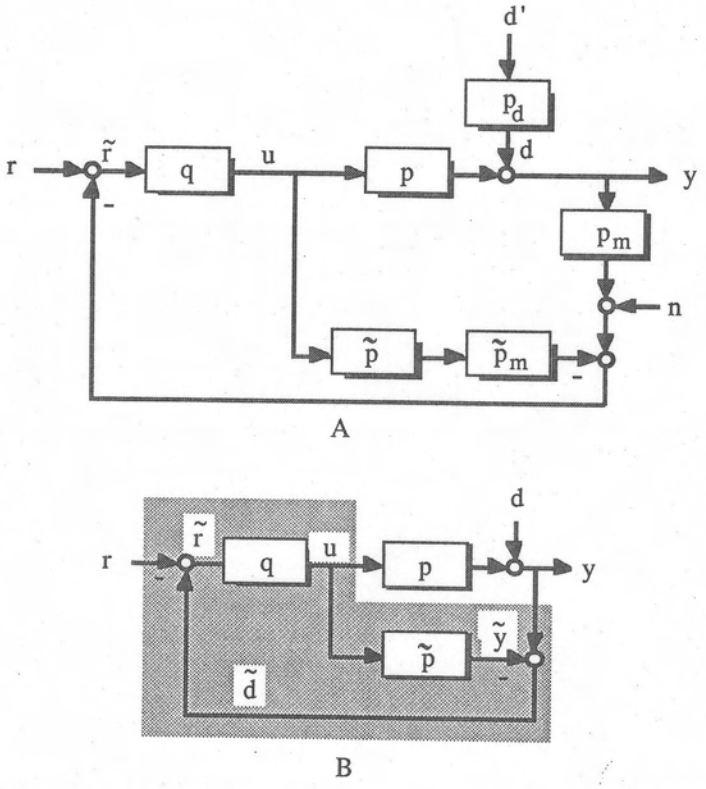

Figure 3.1-1. General (A) and simplified (B) block diagram of Internal Model Control system.

refer to this feedback configuration as *Internal Model Control* (IMC).

The feedback signal is

$$\tilde{d} = (p - \tilde{p})u + d \qquad (3.1-1)$$

If the model is exact ($p = \tilde{p}$) and there are no disturbances ($d = 0$), then the model output $\tilde{y}$ and the process output y are the same and the feedback signal $\tilde{d}$ is zero. Thus, the control system is open-loop when there is no uncertainty — i.e., no model uncertainty and no unknown inputs d. This demonstrates very instructively that for open-loop stable processes feedback is *only* needed because of uncertainty. If a process and all its inputs are known perfectly, there is no need for feedback control. The feedback signal $\tilde{d}$ expresses the uncertainty about the process.

3.2 Stability Conditions for IMC

3.2.1 Internal Stability

In order to test for internal stability we examine the transfer functions between all possible system inputs and outputs. From the block diagram in Fig. 3.2-1 we find that there are three independent system inputs and three independent outputs. As shown in Fig. 3.1-1 we choose the independent inputs to be r, u_1, and u_2 and the independent outputs y, u, and $\tilde{y}$. If there is no model error ($p = \tilde{p}$), then the inputs and outputs are related through the following transfer matrix.

$$\begin{pmatrix} y \\ u \\ \tilde{y} \end{pmatrix} = \begin{pmatrix} pq & (1-pq)p & p \\ q & -pq & 0 \\ pq & -p^2q & p \end{pmatrix} \begin{pmatrix} r \\ u_1 \\ u_2 \end{pmatrix} \qquad (3.2-1)$$

Theorem 3.2-1 follows trivially by inspection.

Theorem 3.2-1. *Assume the model is perfect ($p = \tilde{p}$). Then the IMC system in Fig. 3.1-1B is internally stable if and only if both the plant p and the controller q are stable.*

This result is not unexpected. Recall that we pointed out in the last section that the IMC system is effectively open-loop when there is no uncertainty. Because the stabilization of open-loop unstable systems requires feedback the IMC structure cannot be applied in this case.

It can be argued that the lack of model uncertainty is an artificial assumption. Uncertainty gives rise to feedback and thus it could be possible to stabilize an unstable system with IMC. However in any practical situation it is unacceptable to rely on model uncertainty for stability.

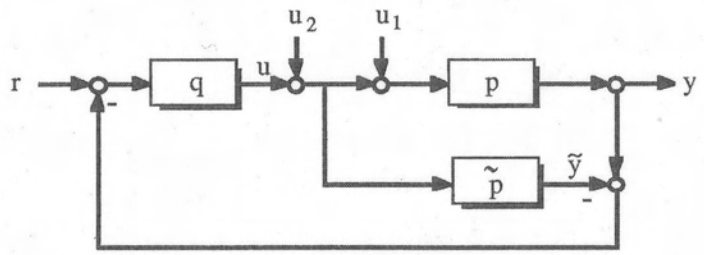

Figure 3.2-1. Block diagram for derivation of internal stability conditions.

3.2.2 Relationship with "Classic Feedback"

The manipulations necessary to transform the block diagram in Fig. 3.2-2A into the one in Fig. 3.2-2B leave the signals u and y unaffected. If we combine the two blocks q and $\tilde{p}$ in Fig. 3.2-2B, which are both part of the control system, into one block c then we obtain a classic feedback control system with

$$c = \frac{q}{1 - \tilde{p}q} \qquad (3.2-2)$$

On the other hand if we add the two blocks $\tilde{p}$ to the classic feedback system in Fig. 3.2-2C the signals u and y also remain unaffected. The IMC structure follows with

$$q = \frac{c}{1 + \tilde{p}c} \qquad (3.2-3)$$

Thus, in the way the outputs u and y react to inputs r and d the classic feedback structure and the IMC structure are *entirely equivalent* and the controllers c and q are related through (3.2–2) and (3.2–3) respectively.

Consider specifically the case that p is stable and that $p = \tilde{p}$.

i. Assume that the IMC structure (Fig. 3.2-2A) is internally stable — i.e., that q is stable. Then the equivalent classic feedback structure (Fig. 3.2-2B) is stable because the internal signals u and y are unaffected by the transformation.

ii. Assume that the classic feedback structure (Fig. 3.2-2B) is internally stable. Then (2.3–1) is stable and therefore q defined by (3.2–3) is stable. Thus the equivalent IMC structure is internally stable.

We have proven the following theorem.

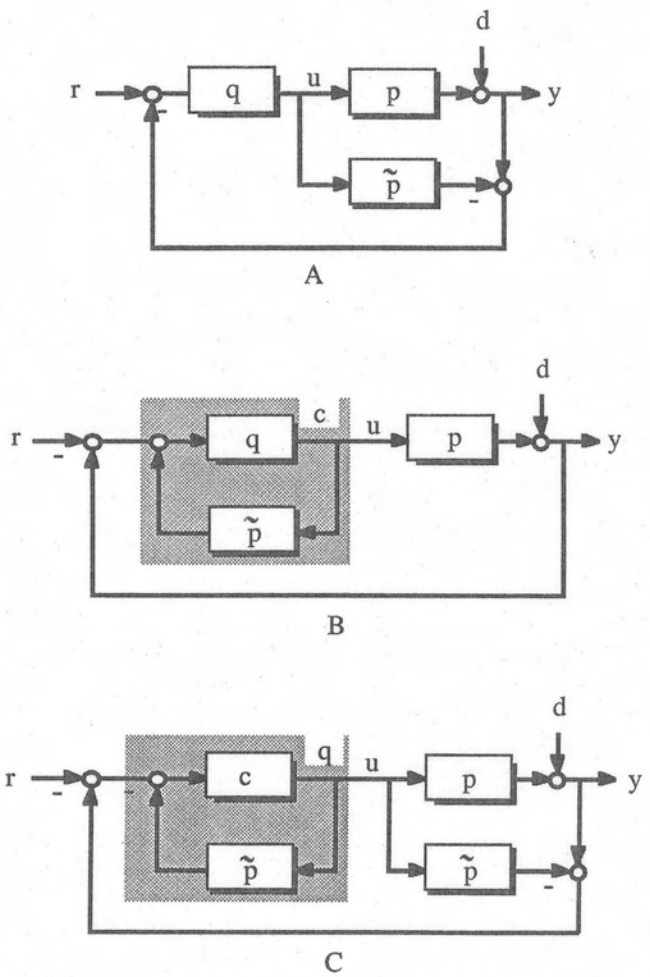

Figure 3.2-2. Alternate representations of the IMC structure.

Theorem 3.2-2. *Assume that p is stable and that $p = \tilde{p}$. Then the classic feedback system with controller*

$$c = \frac{q}{1 - \tilde{p}q} \qquad (3.2-2)$$

is internally stable if and only if q is stable.

This theorem has very profound implications for control system design. Usually a design procedure involves the search for a controller c such that the closed-loop system has certain desired properties. This search is greatly complicated by the fact that only those controllers c are allowed for which the closed-loop system is stable. Theorem 3.2-2 provides a simple parametrization of *all* stabilizing controllers c for the plant p in terms of the stable transfer function q. Thus, instead of searching for c it is possible to search for q without any loss of generality. This search is much simpler because stability of q automatically guarantees the stability of the closed loop system with c determined from (3.2–2).

3.2.3 Implementation: Classic Feedback Versus IMC

In the previous section we showed that IMC provides a convenient parametrization of the classic feedback controller. However, there are significant advantages to not only *designing* the controller via IMC but also *implementing* it in the IMC structure as a model block $\tilde{p}$ and a controller block q.

In any practical situation u is constrained by some upper and lower bounds (e.g., valve saturation). It is well known that these constraints can cause problems for the classic feedback controller. If the controller includes integral action they lead to "reset windup." Even if there is no integral component, a system which is stable without constraints can be destabilized when constraints become active. In the classic feedback configuration these issues have to be dealt with in an ad-hoc fashion ("anti-reset windup").

If the controller is implemented in the IMC configuration, input constraints do not cause any problems provided the *actual* (constrained) plant input is sent to the model $\tilde{p}$ rather than the input *computed* by the controller q (Fig. 3.2-3). Then, under the assumption that $p = \tilde{p}$, the IMC structure remains effectively open-loop and stability is guaranteed by the stability of p and q, as is the case in the absence of constraints. Omitting the saturation block in front of the model block would cause the model to behave differently from the plant when the inputs reach constraints. This model/plant mismatch can lead to instability as we will show later.

In the structure in Fig. 3.2-3 there is no information feedback: the feedback signal is always $\tilde{d}$ and the controller q is entirely unaware of the effect of its

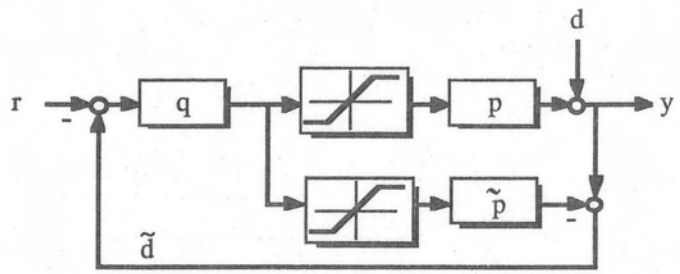

Figure 3.2-3. IMC implementation in the presence of actuator constraints.

actions. In particular, it does not know if and when the manipulated variable saturates. It is reasonable to assume that the closed-loop performance could be improved if the controller adjusted its actions when the manipulated variable has saturated. Thus, while the structure in Fig. 3.2-3 guarantees global stability (for $p = \tilde{p}$), performance might suffer somewhat compared to other anti-windup schemes.

It should also be noted that even if q, p, and $\tilde{p}$ are general *nonlinear* operators the IMC system would still be effectively open-loop (as long as $p = \tilde{p}$) and the stability of p and q would imply closed-loop stability. Thus the IMC structure offers the opportunity of implementing complex nonlinear control algorithms without generating complex stability issues.

3.3 Performance of IMC

3.3.1 Sensitivity and Complementary Sensitivity Function

For IMC (Fig. 3.1-1) we easily find the following transfer functions relating inputs and outputs:

$$y = \frac{pq}{1 + q(p - \tilde{p})} r + \frac{1 - \tilde{p}q}{1 + q(p - \tilde{p})} d \qquad (3.3-1)$$

The sensitivity function $\epsilon(s)$ relates the external inputs r and d to the error $e = y - r$

$$\frac{e}{d - r} = \frac{y}{d} = \frac{1 - \tilde{p}q}{1 + q(p - \tilde{p})} \triangleq \epsilon(s) \qquad (3.3-2)$$

We find the complementary sensitivity function $\eta(s)$ by subtracting $\epsilon(s)$ from unity:

$$\frac{y}{r} = \frac{pq}{1 + q(p - \tilde{p})} \triangleq \eta(s) \qquad (3.3-3)$$

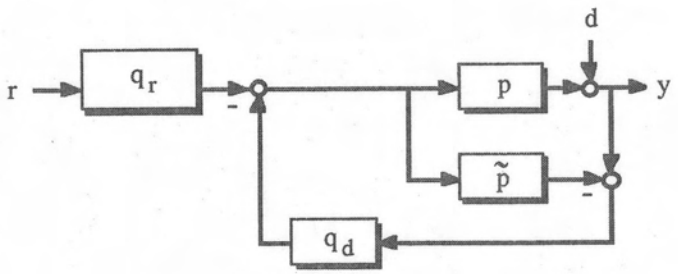

Figure 3.3-1. Two-degree-of-freedom IMC.

When the model is exact ($p = \tilde{p}$), (3.3–2) and (3.3–3) reduce to

$$\tilde{\epsilon}(s) = 1 - \tilde{p}q \tag{3.3-4}$$

$$\tilde{\eta}(s) = \tilde{p}q \tag{3.3-5}$$

We concluded in Chap.2 that $\epsilon(s)$ determines performance and $\eta(s)$ robustness. Through the IMC parametrization the controller q is related to $\epsilon(s)$ and $\eta(s)$ in a very simple (linear) manner which will make the design of q easy. The effect of the classic controller c on $\epsilon(s)$ and $\eta(s)$ (2.4–1 and 2.4–2) is much more complex.

3.3.2 Two-Degree-of-Freedom Controller

If both good tracking of r and good disturbance rejection are important and if the dynamic characteristics of the two inputs r and d are substantially different it is advantageous to introduce another controller block (Fig. 3.3-1). The effects of r and d on the error are described by

$$e = \frac{1 - \tilde{p}q_d}{1 + q_d(p - \tilde{p})}d - \left[1 - \frac{pq_r}{1 + q_d(p - \tilde{p})}\right]r \tag{3.3-6}$$

For $p = \tilde{p}$ (3.3–6) becomes

$$e = (1 - pq_d)d - (1 - pq_r)r \tag{3.3-7}$$

Here q_d is designed for disturbance rejection and q_r for setpoint tracking.

3.3.3 Asymptotic Properties of Closed-Loop Response (System Type)

"System types" were defined in Sec. 2.4.3 to classify the asymptotic closed-loop behavior.

Type m:

$$\lim_{s\to 0} \frac{\epsilon(s)}{s^k} = 0 \qquad 0 \leq k < m \tag{3.3-8}$$

Using (3.3–2) this definition becomes

Type m:

$$\lim_{s\to 0} \frac{1-\tilde{p}q}{1+q(p-\tilde{p})} \frac{1}{s^k} = 0 \qquad 0 \leq k < m \tag{3.3-9}$$

Condition (3.3–9) is satisfied if and only if $(1-\tilde{p}q)$ has m zeros at the origin which is the case if and only if

Type m:

$$\lim_{s\to 0} \frac{d^k}{ds^k}(1-\tilde{p}q) = 0 \qquad 0 \leq k < m \tag{3.3-10}$$

Specifically we obtain

Type 1:

$$\lim_{s\to 0} \tilde{p}q = 1 \tag{3.3-11}$$

Thus, in order to track asymptotically constant inputs error free the controller steady state gain has to be the inverse of the model steady-state gain.

Type 2:

$$\lim_{s\to 0} \tilde{p}q = 1, \quad \lim_{s\to 0} \frac{d}{ds}(\tilde{p}q) = 0 \tag{3.3-12}$$

Note that (3.3–11) and (3.3–12) are necessary and sufficient for off set free tracking of steps and ramps respectively even when model error is present.

3.3.4 The Concept of "Perfect Control"

In Chap.2 we identified the sensitivity function ϵ as a direct indicator of performance. The feedback controller is designed to make some measure (e.g., a weighted norm) of the sensitivity function small. For IMC

$$\epsilon = \frac{e}{d-r} = \frac{1-\tilde{p}q}{1+q(p-\tilde{p})} \tag{3.3-2}$$

Algebraically ϵ can be made identically equal to zero by choosing the model inverse as the controller

$$q = \frac{1}{\tilde{p}} \tag{3.3-13}$$

This would imply "perfect control"

$$e(t) = 0 \qquad \forall \quad r, d, t \tag{3.3-14}$$

which we know from practical experience to be impossible. A reason for the infeasibility of perfect control is evident from the classic feedback controller c generated by (3.3–13).

$$c = \frac{q}{1 - \tilde{p}q} = \infty \quad \text{for } q = \tilde{p}^{-1} \tag{3.3 - 15}$$

Though "perfect control" cannot be achieved, it is of great theoretical and practical interest to determine how closely this ideal can be approached. Obviously the closeness depends on the controller q (or c). If c is a PI controller, then better performance (a "smaller" ϵ) is expected than when c is just a P controller. In the age of digital control when very complex controllers can be implemented with relative ease, it does not seem to be meaningful to investigate what performance is achievable with particular simplistic controllers. What is much more important is how *plant characteristics* limit the achievable performance *independent* of controller complexity.

This information would be useful in several different ways: If the potential performance improvement through an advanced control strategy is to be assessed, the current performance (for example, with a PI controller) could be compared against the *best* achievable performance with *any* controller. If this comparison does not reveal any significant differences, *no* advanced control algorithm will be able to generate significant benefits. This judgment can be made without actually considering or designing any particular advanced controllers. The analysis would also imply that real performance improvements can come only from a design change of the plant itself rather than from modifications of the control system.

Our earlier discussions of $q = \tilde{p}^{-1}$ indicate that achievable performance is related to plant invertibility. If inversion were feasible then perfect control would be possible. Limits on invertibility are limits on achievable performance. There are three reasons why the use of the model inverse as controller is generally infeasible and therefore perfect control impossible.

i. *Nonminimum-phase (NMP) Characteristics of the Plant.* If p contains a time delay, p^{-1} is noncausal and cannot be used as a controller. If p contains RHP zeros then p^{-1} is unstable. However, for the stability of IMC q has to be stable and therefore $q = p^{-1}$ is not viable. Thus, when NMP elements are present in the plant, the sensitivity function cannot be made zero — not even in an approximate sense.

ii. *Constraints on the Inputs.* If p is strictly proper then q is improper — i.e., it involves pure differentiators which cannot be realized. This does not appear to be a severe handicap because poles can always be added to the

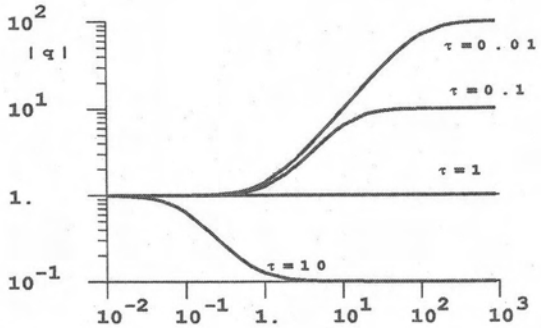

Figure 3.3-2. q' approaches the improper q as $\tau \to 0$.

denominator of q such that q approximates p^{-1} arbitrarily closely without being improper. For example if $p = (s+1)^{-1}$, $q = (s+1)$ is improper but

$$q' = \frac{s+1}{\tau s + 1}$$

is proper and "approaches" q as τ approaches zero. The Bode plot in Fig. 3.3-2 indicates that the excursions of the input u are excessive for high frequency inputs when τ is small. Thus, the constraints on the manipulated variable impose a limit on how closely q' can approximate q.

iii. *Model Uncertainty.* The controller is designed based on a process model $\tilde{p}$ which is an approximate description of the real plant p. We saw in Chap. 2 that model uncertainty constrains the choice of the complementary sensitivity function $\tilde{\eta}(s)$. In particular, for robust stability $\tilde{\eta}$ has to satisfy

$$|\tilde{\eta}(i\omega)\bar{\ell}_m(\omega)| < 1 \qquad \forall\omega \qquad (3.3-16)$$

when the multiplicative uncertainty is bounded by $\bar{\ell}_m(\omega)$:

$$\left|\frac{p(i\omega) - \tilde{p}(i\omega)}{\tilde{p}(i\omega)}\right| \leq \bar{\ell}_m(\omega) \qquad (3.3-17)$$

When we rewrite (3.3–16) in terms of $\tilde{\epsilon}$

$$|(1-\tilde{\epsilon})\bar{\ell}_m| < 1 \qquad \forall\omega \qquad (3.3-18)$$

we see that this bound is satisfied for $\tilde{\epsilon} = 0$ ("perfect control") as long as $\bar{\ell}_m < 1$. However, for large frequencies $\bar{\ell}_m$ always exceeds unity because

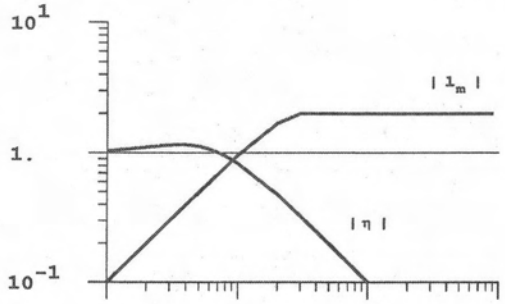

Figure 3.3-3. Complementary sensitivity $\tilde{\eta}$ satisfying robustness bound $\bar{\ell}_m$.

there is complete phase uncertainty, and therefore $\tilde{\epsilon}$ cannot be zero. Figure 3.3-3 shows a typical plot of how $|\tilde{\eta}|$ has to decrease as $\bar{\ell}_m$ increases. The frequency range over which good control ($|\tilde{\eta}| \approx 1$) is possible, is limited by model uncertainty.

The general conclusion is that perfect control is only feasible in the low frequency range. For high frequencies the perfect controller has to be detuned.

One of the attractive features of IMC is that it reveals the fundamental limitations on performance with clarity and simplicity. This insight is difficult to obtain from a study of the classic feedback structure.

3.4 Outline of the IMC Design Procedure

3.4.1 Basic Design Philosophy

In Chap.2 we discussed the objectives of control system design.

Nominal Performance:

Specification:

$$\|\tilde{\epsilon}w\|_\alpha < 1 \qquad \alpha = 2, \infty \tag{3.4-1}$$

Optimal Design Problem:

$$\min_q \|\tilde{\epsilon}w\|_\alpha = \min_q \|(1 - \tilde{p}q)w\|_\alpha \quad \alpha = 2, \infty \tag{3.4-2}$$

Robust Stability:

Specification:

$$\|\tilde{\eta}\bar{\ell}_m\|_\infty = \|\tilde{p}q\bar{\ell}_m\|_\infty < 1 \tag{3.4-3}$$

Robust Performance:

Specification:

$$\|\epsilon w\|_\infty < 1 \qquad \forall p \in \Pi \tag{3.4-4}$$

or

$$|\tilde{\epsilon} w| + |\tilde{\eta}\bar{\ell}_m| = |(1 - \tilde{p}q)w| + |\tilde{p}q\bar{\ell}_m| < 1 \quad \forall \omega \tag{3.4-5}$$

Optimal Design Problem:

$$\min_q \sup_\omega (|\tilde{\epsilon} w| + |\tilde{\eta}\bar{\ell}_m|) = \min_q \sup_\omega (|(1 - \tilde{p}q)w| + |\tilde{p}q\bar{\ell}_m|) \tag{3.4-6}$$

For good nominal performance a weighted norm of the sensitivity function $\tilde{\epsilon}$ should be made small (3.4–1). In the popular optimal control formulations either the 2-norm or the ∞-norm is minimized (3.4–2). For robust stability the complementary sensitivity function $\tilde{\eta}$ is constrained by modeling error (3.4–3). For robust performance a more complex constraint (3.4–5) involving both $\tilde{\epsilon}$ and $\tilde{\eta}$ has to be met.

Clearly what we need in practice is robust performance. However, there are several hurdles, which prevent us at present from designing controllers by solving the optimal robust performance problem (3.4–6). First, the efficient solution of (3.4–6) is still an active area of research. For the general multivariable case no reliable solution procedures are available yet. Furthermore, the specification of a frequency-dependent weight w, which results in desirable time domain response characteristics, when the ∞-norm is minimized, involves much trial and error. Finally, in an application environment it is desirable to make the "degree of robustness" adjustable on-line because the plant-model mismatch tends to increase with time after the controller has been installed. Solving (3.4–6) repeatedly with different functions w and $\bar{\ell}_m$ is not practical under these circumstances.

Therefore, for IMC design we use a two-step approach which has no inherent optimality characteristics but should provide a good engineering approximation to the optimal solution. It guarantees robustness but the performance is generally not optimal in any sense.

STEP 1: Nominal Performance

The controller $\tilde{q}$ is selected to yield a "good" system response for the input(s) of interest, without regard for constraints and model uncertainty. Generally we will choose $\tilde{q}$ such that it is Integral-Square-Error (ISE) or H_2-optimal

$$\min_{\tilde{q}} \|(1 - \tilde{p}\tilde{q})w\|_2 \tag{3.4-7}$$

The optimal sensitivity function becomes

$$\bar{\epsilon} \triangleq 1 - \tilde{p}\tilde{q} \qquad (3.4-8)$$

and the optimal complementary sensitivity function

$$\bar{\eta} \triangleq \tilde{p}\tilde{q} \qquad (3.4-9)$$

STEP 2: Robust Stability and Performance

At high frequencies, when $\bar{\ell}_m$ exceeds unity $\bar{\eta}$ has to be rolled off. Therefore, $\tilde{q}$ is augmented by a low-pass filter f

$$q \triangleq \tilde{q}f \qquad (3.4-10)$$

The order of f is such that q is proper and its roll-off frequency low enough to satisfy the robust stability constraint (3.4–3). With the filter, $\tilde{\eta}$ and $\tilde{\epsilon}$ become

$$\tilde{\eta} = \tilde{p}q = \tilde{p}\tilde{q}f \qquad (3.4-11)$$

$$\tilde{\epsilon} = 1 - \tilde{p}q = 1 - \tilde{p}\tilde{q}f \qquad (3.4-12)$$

The function of f is to detune the controller, to sacrifice performance (increase $|\tilde{\epsilon}|$) for robustness (decrease $|\tilde{\eta}|$). Clearly, an open-loop stable system can be made arbitrarily robust, simply by turning off the controller. Thus (3.4–3) can always be satisfied. However, because of the associated loss of performance (3.4–5) will generally not be met. To design the filter for robust performance (3.4–5) is difficult. Indeed, there might not exist *any* filter to satisfy (3.4–5) for the particular choices of weights w and $\bar{\ell}_m$. In our design procedures we will tend to relax performance (decrease w) in exchange for simple search techniques for the appropriate filter. This will yield robust stability and reasonable, albeit not optimal, performance.

3.4.2 Two-Degree-of-Freedom Design

For the two-controller structure in Fig. 3.3-1 we derived the relationship

$$e = \frac{1 - \tilde{p}q_d}{1 + q_d(p - \tilde{p})}d - \left[1 - \frac{pq_r}{1 + q_d(p - \tilde{p})}\right]r \qquad (3.3-6)$$

which becomes for $p = \tilde{p}$

$$e = (1 - pq_d)d - (1 - pq_r)r \qquad (3.3-7)$$

The design problems for nominal performance are independent. If the H_∞ performance objective is chosen then q_r and q_d are designed to satisfy

$$\|(1-\tilde{p}q_r)w_r\|_\infty < 1 \qquad (3.4-13)$$

$$\|(1-\tilde{p}q_d)w_d\|_\infty < 1 \qquad (3.4-14)$$

Robust stability depends only on q_d

$$\|\tilde{p}q_d\bar{\ell}_m\|_\infty < 1 \qquad (3.4-15)$$

Because the "prefilter" q_r is not in the feedback path, it has no effect on stability.

Consider now the design for robust performance. Let us first design q_d for robust performance just as described in the last section. The robust performance specification for disturbance rejection is

$$\|\epsilon w_d\|_\infty < 1 \qquad \forall p \in \Pi \qquad (3.4-16)$$

which is equivalent to

$$|\tilde{\epsilon}w_d| + |\tilde{\eta}\bar{\ell}_m| < 1 \qquad \forall\omega \qquad (3.4-17)$$

Next let us design q_r for robust tracking performance. The performance specification is

$$\left|\left[1-\frac{\tilde{p}q_r(1+\ell_m)}{1+\tilde{p}q_d\ell_m}\right]w_r\right| < 1 \qquad \forall\ell_m\in\Lambda_m, \quad \forall\omega \qquad (3.4-18)$$

where the set $\Lambda_m(i\omega)$ is defined by

$$\Lambda_m(i\omega) = \{\ell_m(i\omega): \quad |\ell_m(i\omega)| < \bar{\ell}_m(\omega)\} \qquad (3.4-19)$$

From (3.4–18) we derive the condition

$$\Leftrightarrow |\,[1+\tilde{p}q_d\ell_m - \tilde{p}q_r(1+\ell_m)]\cdot w_r| < |1+\tilde{p}q_d\ell_m| \qquad \forall\ell_m\in\Lambda_m,\forall\omega$$

$$\Leftarrow |\,[1+\tilde{p}q_d\ell_m - \tilde{p}q_r(1+\ell_m)]\cdot w_r| + |\tilde{p}q_d\ell_m| < 1 \qquad \forall\ell_m\in\Lambda_m,\forall\omega$$

$$\Leftarrow |(1-\tilde{p}q_r)w_r| + |\tilde{p}q_d\bar{\ell}_m|\left[1+\left|1-\frac{q_r}{q_d}\right|w_r\right] < 1 \qquad \forall\omega \qquad (3.4-20)$$

Note that because of the last two steps in the derivation, (3.4–20) is only *sufficient* for robust setpoint tracking performance. A necessary and sufficient condition can be derived via the Structured Singular Value, which will be introduced in the discussion of multivariable systems.

When $q_r = q_d$, (3.4–20) reduces to the necessary and sufficient condition for robust performance derived in Sec. 2.6. The first term in (3.4–20) expresses nominal performance. The second term is proportional to the multiplicative uncertainty $\bar{\ell}_m$ and to the disturbance controller magnitude — i.e., the *disturbance* controller q_d affects the robust *setpoint tracking* performance. Thus the designs of q_d and q_r are *not* independent when the objective is robust performance. For the proper tradeoff they have to be designed jointly rather than sequentially as we did here.

3.4.3 Design in the Presence of Measurement Device Dynamics

If the measurement device dynamics can be identified independent from the process dynamics (for example a time delay caused by a chromatograph employed for composition analysis) and if the real process output rather than the measured one is to be controlled then the IMC structure should be modified as shown in Fig. 3.1-1A. Through the addition of the measuring device the closed-loop relationship becomes

$$y = \frac{1 - \tilde{p}'q}{1 + q(p' - \tilde{p}')}d + \frac{pq}{1 + q(p' - \tilde{p}')}r \qquad (3.4-21)$$

where the augmented plant and model are defined as

$$p' = pp_m \qquad (3.4-22)$$

$$\tilde{p}' = \tilde{p}\tilde{p}_m \qquad (3.4-23)$$

For $p = \tilde{p}, p_m = \tilde{p}_m$ we find from (3.4–21)

$$e = (1 - p'q)d - (1 - pq)r \qquad (3.4-24)$$

The design of q for nominal tracking performance is unaffected by the measuring device. This should be intuitively obvious from the "open-loop nature" of IMC. For nominal disturbance-rejection performance q must be designed for the augmented plant p'.

Note that even when d and r are identical signals (e.g., steps) the optimal q's for setpoint tracking and disturbance rejection are generally different because of the measurement device dynamics. Thus a two-degree-of-freedom controller is needed for optimal performance.

The robust stability and robust disturbance rejection problem are in the standard form studied in Sec.3.4.1. Using similar arguments as in Sec.3.4.2 a sufficient condition for robust tracking can be derived. A necessary and sufficient condition can be expressed in terms of the Structured Singular Value which we will discuss later.

3.5 Summary

In the absence of model uncertainty, the IMC structure (Fig. 3.1-1) is effectively open-loop. Therefore, it can only be implemented on open-loop stable processes. For a stable process and a perfect model, closed-loop stability is assured by a stable IMC controller. The performance achievable with IMC is identical to that achievable with any classic feedback controller selected such that

$$c = \frac{q}{1 - \tilde{p}q} \qquad (3.2-2)$$

and vice versa.

The discussion of the IMC structure revealed the following factors which prevent perfect control: NMP plant characteristics, input constraints and model uncertainty.

A two-step design procedure is proposed. In the first step the controller $\tilde{q}$ is selected for good response without regard for constraints and model uncertainty. Often $\tilde{q}$ will be chosen to be H_2-optimal:

$$\min_{\tilde{q}} \|(1 - \tilde{p}\tilde{q})w\|_2 \qquad (3.4-7)$$

In the second step the controller $\tilde{q}$ is augmented by a low-pass filter f

$$q = \tilde{q}f \qquad (3.4-10)$$

to provide the rolloff necessary for robustness and milder action of the manipulated variables. The filter can be adjusted on-line if necessary.

3.6 References

An excellent historical review of the origins of the IMC structure which has as its special characteristic a model in parallel with the plant is provided by Frank (1974). It appears to have been discovered by several people simultaneously in the late fifties. Newton, Gould and Kaiser (1957) use the structure to transform the closed loop system into an open loop one so that the results of Wiener can be applied to find the H_2-optimal controller q. The Smith Predictor (Smith, 1957) also contains a process model in parallel with the plant (see Chap.6). Independently Zirwas (1958) and Giloi (1959) suggested the predictor structure for the control of systems with time delay.

Frank (1984) first realized the general power of this structure, fully exploits it and extends the work by Newton et al. (1957) to handle persistent disturbances and setpoints. Youla et al. (1976) extend the convenient "q-parametrization" of

the controller c to handle unstable plants. In 1981 Zames ushers in the era of H_∞-control utilizing for his developments the q-parametrization. At present it is used in all robust controller design methodologies.

Unaware of all these developments the process industries both in France (Richalet et al., 1978) and in the United States (Cutler and Ramaker, 1979; Prett and Gillette, 1979) exploit the advantages of the parallel model/plant arrangement, Brosilow (1979) utilizes the Smith predictor parametrization to develop a robust design procedure and Garcia and Morari (1982, 1985 a,b) unify all these concepts and refer to the structure in Fig. 3.1-1 as Internal Model Control (IMC) because the process *model* is explicitly an *internal* part of the controller.

3.2.3. Alternate ways to deal with the windup problem are discussed by Doyle, Smith & Enns (1987). A general anti-windup scheme was proposed recently by Hanus, Kinnaert & Henrotte (1987).