Classical Control Using H^∞ Methods
Rough Draft

J. William Helton
Department of Mathematics
University of California, San Diego

Orlando Merino
Department of Mathematics
University of Rhode Island

November 27, 1994

Contents

Preface 11

I Short Design Course 1

1 A method for solving system design problems 3
- 1.1 Rational functions 3
- 1.2 The closed loop system $\mathcal{S}$ 5
- 1.3 Designable transfer function 6
- 1.4 A system design problem 6
- 1.5 The method 7
- 1.6 Exercises 9
- 1.7 References 10

2 Internal Stability 11
- 2.1 Control and stability. 11
- 2.2 Interpolation. 14
- 2.3 Systems with a stable plant 15
- 2.4 Exercises 17

3 Frequency domain performance requirements 19
- 3.1 Introduction 19
  - 3.1.1 The closed loop system $\mathcal{S}$ 19
  - 3.1.2 Frequency domain performance requirements 20
  - 3.1.3 Disk inequalities 20
- 3.2 Measures of performance 22
  - 3.2.1 Gain-phase margin 22
  - 3.2.2 Tracking error 23
  - 3.2.3 Bandwidth 25
  - 3.2.4 Closed loop roll-off 26

3.2.5 Fundamental tradeoffs 27
3.2.6 Choosing sets of performance requirements 27
3.3 Disk inequalities 29
3.4 More performance measures 30
3.4.1 Peak magnitude 30
3.4.2 Compensator bound 31
3.4.3 Plant bound 32
3.4.4 Disturbance rejection 33
3.4.5 More on tracking error 34
3.5 A fully constrained problem 36
3.6 References and further reading 38

4 Optimization 41
4.1 Review of concepts 41
4.2 Generating a performance function 43
4.3 Finding T with best performance 44
4.3.1 Example 45
4.4 Good performance functions 47
4.5 Performance not of circular type 50
4.6 Optimization 51
4.6.1 The optimization problem $OPT_{\mathcal{I}}$ 51
4.7 Internal stability and optimization 52
4.7.1 The optimization problem OPT 52
4.7.2 OPT with circular Γ 53
4.8 Exercises 53
4.9 References and further reading 53

5 A Design Example with OPTDesign 55
5.1 Introduction 55
5.1.1 The package OPTDesign 55
5.1.2 How to use this notebook 55
5.1.3 Starting a session 56
5.2 The problem 56
5.2.1 Specifying the problem. 57
5.2.2 The requirements envelope 58
5.3 OPTDesign iteration 60
5.3.1 Generating a rational closed loop 61
5.4 Producing a rational compensator 62
5.5 How good is the answer that we obtained? 64
5.5.1 Plots of the calculated closed loop function 65
5.6 Reducing the error 67
5.6.1 Improving the fit 67

5.6.2 When the relative degree of the compensator is not zero . . . 68
5.7 Other performance functions . . . 69
5.7.1 Modifying the performance . . . 69
5.7.2 Non-rational weight functions . . . 69
5.8 OPTDesign[] output . . . 70
5.8.1 Interpretation of output diagnostics . . . 70
5.9 Controlling the numerical error . . . 71
5.9.1 Options Ngrid and GridSpread . . . 71
5.9.2 Option Nsmth . . . 72

II Advanced Design 75

6 Examples 77
6.1 Numerical Practicalities . . . 78
6.1.1 Discontinuous functions and smoothing . . . 78
6.1.2 Vanishing radius function . . . 79
6.1.3 Sampling functions on the $j\omega$ axis . . . 80
6.2 Design Example 1 . . . 81
6.2.1 Electro-mechanical and electrical models . . . 81
6.2.2 Mathematical model . . . 83
6.2.3 Statement of the problem . . . 84
6.2.4 Reformulation of requirements . . . 84
6.2.5 Formulation of the performance function . . . 85
6.2.6 Optimization . . . 88
6.3 Time domain performance requirements . . . 88
6.3.1 Two common time domain requirements . . . 90
6.3.2 A naive method . . . 91
6.3.3 Limitations of the naive method . . . 92
6.4 Design Example 2 . . . 93
6.4.1 Statement of the problem . . . 93
6.4.2 Translation of time domain requirements . . . 93
6.4.3 Forming the performance function Γ . . . 96
6.4.4 Optimization . . . 97
6.5 Design problems with competing constraints . . . 99
6.5.1 Competing constraints . . . 99
6.5.2 Performance from competing constraints . . . 101
6.5.3 Solving OPT . . . 102
6.5.4 Smoothing the performance function . . . 103
6.5.5 Weight functions . . . 103
6.6 Design Example 3 . . . 104

6.7 References 104

7 Time domain requirements 105

7.1 Requirements on the step response function 105
  7.1.1 Step response indices 105
  7.1.2 Time domain performance requirements 106
  7.1.3 Classical time domain requirements 107
  7.1.4 Envelope-type requirements 107
7.2 Translation of time-domain requirements 108
  7.2.1 Completely monotonic functions 108
  7.2.2 Known boundary values 110
  7.2.3 Application to system design 111
7.3 Design Example 4 112
  7.3.1 Statement of the problem 112
  7.3.2 Strategy for defining the performance function . . . 113
  7.3.3 Generating an envelope requirement 113
  7.3.4 Weak form of requirements envelope 114
  7.3.5 Choosing weight functions 115
  7.3.6 Results of the optimization 116
7.4 References 116

8 Internal Stability II 119

8.1 Calculating interpolants 119
  8.1.1 Calculating one interpolant 120
  8.1.2 Parameterization of all interpolants 121
  8.1.3 Interpolation with a relative degree condition 123
8.2 Plants with simple RHP zeros and poles 123
8.3 Parameterization: the general case 126
  8.3.1 Higher order interpolation 126
  8.3.2 Plants with multiple RHP zeros and poles 127
8.4 Exercises 129
8.5 References and further reading 130

III H^∞ Theory 133

9 H^∞-Optimization and Control 137

9.1 The problem OPT 137
9.2 The fundamental H^∞ problem of control 138
9.3 Change of coordinates 140
9.4 Performance functions 141

10 Solutions to OPT **145**
10.1 Complex partial derivatives 145
10.2 Winding number 147
10.3 Main result 147
10.4 Properties of solutions 148

11 Facts about analytic functions **151**
11.1 Any function on an arc 151
11.2 Spectral factorization 153
11.3 Analytic functions with prescribed phase 154
11.4 The fundamental mistake of H^∞-control 156

12 Proof of the main result **159**
12.1 Taylor expansions 159
12.2 Solutions make the performance flat 160
12.3 Solutions satisfy the winding number condition 161
12.4 Flatness and winding number conditions are sufficient . . . 162

13 Computer Solutions to OPT **165**
13.1 Computer diagnostics 165
13.2 Spaces of functions 167
13.3 A numerical algorithm 169

14 References **173**

IV H^∞ Theory: Vector Case **177**

15 Many Analytic Functions **179**
15.1 The OPT problem 180
15.2 Solutions only a keystroke away 180
15.3 Necessary conditions for solutions 181
15.4 An Example 183
15.5 Computer diagnostics 184
15.6 Algorithms for OPT 186
15.7 References 188

16 More Numerical Algorithms **191**
16.1 Notation 191
16.2 Nehari's problem 192
16.3 Disk Iteration algorithms 193
  16.3.1 The Power Method for solving Nehari's Problem . . 193
  16.3.2 The Algorithm 194

16.4 More on the Newton iteration algorithm 195
16.5 Derivation of the Nehari 197
16.6 References 199

17 A sufficient condition for optimality **201**
17.1 Complete characterization of solutions 201
17.2 An Example, continued 202
17.3 Existence 204
17.4 Proof of Theorem 23 205
17.5 References 208

V Appendices **209**

A History and Perspective **211**

B Other Software **219**

C Pure Mathematics and Problems **221**

D Uncertainty **225**
D.1 Introduction 225
D.2 Types of uncertainty 226
D.3 Dealing with uncertainty: worst case analysis 227
D.4 A method to treat plant uncertainty 228
D.5 An example with quasicircular Γ 229
D.6 Performance and uncertainty 230
D.7 Extensions 232

E Getting OPTDesign and Anopt via anonymous ftp **237**

F Anopt.ma Notebook **239**

G NewtonInterpolant.ma Notebook **253**

H NewtonFit.ma Notebook **259**

List of Figures

0.1 The designable part of the system. 12
0.2 Paths to be followed in order to read this book. 17

1.1 Plot of $|f(j\omega)|$ vs. frequency ω, where $f(s) = 1/((s-.5)^2+4)$. The dotted line marks $||f||_\infty$. 4
1.2 The closed loop system $\mathcal{S}$. 5

2.1 The closed loop system $\mathcal{S}$. 11
2.2 Plot of the unstable pole s_δ of P^δ as a function of δ. 13

3.1 The closed loop system $\mathcal{S}$. 20
3.2 Tube like region $\{(\omega, z) : \ |k(j\omega) - z| \leq r(j\omega)\}$ determined by a disk inequality $|k(j\omega) - T(j\omega)| \leq r(j\omega)$. 21
3.3 Region determined by piecing together disk inequalities. . . 21
3.4 The gain margin g, phase margin ϕ, and g-p. margin m . . 22
3.5 Region defined by the gain-phase margin constraint. 23
3.6 Region defined by the tracking error constraint 25
3.7 Region defined by the bandwidth constraint. 26
3.8 Region defined by the closed loop roll-off constraint. 27
3.9 The region defined by the constraints in Table 3.1. 30
3.10 Region defined by the peak magnitude constraint. 31
3.11 Region defined by the compensator bound constraint 32
3.12 Region defined by the plant bound constraint 33
3.13 Disturbance in a feedback system 34
3.14 Section of the envelope defined by center and radius functions 38
3.15 Envelope defined by the constraints. 39
3.16 Envelope as seen from another vantage point. 40

4.1 The closed loop system $\mathcal{S}$. 41
4.2 Plot of the function $g(\omega) = 1/r(\omega)^2|k(j\omega) - T(j\omega)|^2$. 44
4.3 Envelope defined by center and radius functions in Example 1. 47

4.4 Error function 48
4.5 Magnitude of T_0. 48
4.6 Phase of T_0. 48
4.7 The zeros and poles of the compensator 49

6.1 Sample values of a discontinuous function 79
6.2 Smoothed sample values of a discontinuous function 79
6.3 High frequency undersampled 80
6.4 Low frequency undersampled 80
6.5 Slide drive apparatus. 81
6.6 Electro-mechanical model 82
6.7 Electrical model 83
6.8 Derivation of m in Design Example 1. 85
6.9 Requirements envelope. 86
6.10 Requirement envelopes: before and after smoothing. 87
6.11 Modified envelope. 88
6.12 Bode plots of optimal closed loop T in design example 1. . . 89
6.13 Bode plots of optimal open loop L in design example 1. . . 89
6.14 Polar plot of optimal T in design example 1. 90
6.15 Polar plot of optimal L in design example 1. 90
6.16 Step Response curve of optimal system in design example 1. 91
6.17 Magnitude of the closed loop transfer function for different plants 91
6.18 A step response function that satisfies time domain requirements 92
6.19 Bode plots of T_{ref}. 94
6.20 Step response for T_{ref}. 94
6.21 Plot of $|T_{ref}|$. 95
6.22 Plot of $|1 - T_{ref}|$. 95
6.23 Requirements envelope. 98
6.24 Log-log plot of the function $K + R$. 98
6.25 Modified envelope. 99
6.26 BodePlots of optimal T for Design Problem 2. 100
6.27 Plots of magnitude of optimal sensitivity S and of step response for design problem 2. 101
6.28 Sublevel sets S_ω from competing constraints 102

7.1 The step response function indices 106
7.2 Bounds for $c(t)$. 108
7.3 Bounds for $c'(t)$. 109
7.4 A nondecreasing step response function. 111

7.5 Bound g for envelope requirement in Design Example 4. . . 114
7.6 Plot of $\Gamma_2(\omega, T_1^*(j\omega))$. 115
7.7 Bode plots of optimal closed loop T^* in Design Example 4. 117
7.8 Plot of $|T^*(j\omega) - T_1(j\omega)|$. 117
7.9 Step response function corresponding to T_1. 118

9.1 The given and designable parts of the system 139
9.2 The function f makes performance at least as good as c at $\omega = \omega_0$. 139

10.1 The sublevel set $S_\theta(c)$ and the gradient 146
10.2 The winding number of a function about 0. 147
10.3 The optimal performance surface 149

12.1 The function $g(e^{j\theta})$ dips below $\gamma - \delta$. 160
12.2 The function ah_n winds about 0. 162

D.1 The uncertainty set $\mathcal{P}_\omega$ 226
D.2 The uncertainty set and the direction of largest increase in G. 229
D.3 Plot of both the worst case uncertainty performance 232
D.4 Plot of the levels 0.01, 0.05, 0.1, 0.5 233
D.5 Plot of G and G^* when T_0 ranges on the x-axis. 234

Preface

Purpose of this book

One of the main accomplishments of control in the 1980's was the development of a subject called H^∞-control. This book teaches control system design using H^∞ techniques. The first parts of the book are aimed at students of control at any level who want an H^∞ supplement to their control course. The later parts of the book treat the theory of H^∞ control.

Since the book is highly modular the theory parts can be read independently of the design parts and vice versa. Thus the book should be useful for three purposes.

Control systems design (Parts I and II).

We believe that the approach presented here has three virtues. First it is conceptually simple. Secondly it corresponds directly to what is often called "classical control", which is good simply because of the widespread appeal of classical frequency domain methods. Finally, classical control has always been presented as trial and error applied to specific cases; this book lays out a much more precise (almost axiomatic) approach. This has the tremendous advantage of converting an engineering problem to one which can be put directly into a mathematical optimization package. While some trial and error may be needed, our approach greatly reduces it.

A student who learns Part I should have a good feel for how engineering specs are encoded as precise mathematical constraints.

A theory for frequency domain CAD (Parts III and IV).

Part III gives a theory of H^∞ control which is no more difficult to understand than other approaches but which applies to much more general problems. This approach uses a few simple qualitative principles about analytic functions (which have of broad engineering value) to produce a seemingly qualitative picture of what properties an optimal controller should have. Rather surprisingly this physically appealing characterization of an optimal controller leads directly to highly effective computer algorithms.

Part IV and the appendices extend the basics presented in Part I, II and III in several ways.

Research in engineering and mathematics
The core of H^∞ control is optimization of a performance measure over the family of analytic functions. Parts III and IV integrate results that have appeared in print since the early 80's and while a powerful and unified theory is emerging the area is new enough that much remains to be done. The theory is interesting in itself, and has connections and applications to other fields, e.g. operator theory and several complex variables in mathematics or amplifier design and antenna design in engineering. Who turns directly to Part III and Part IV should find them a readable account of recent developments.

Optimal design

We shall always deal with a situation where part of a linear system is given (called the plant in control theory). We want to find the additional part f (the designable part) so that the whole system meets some requirements.

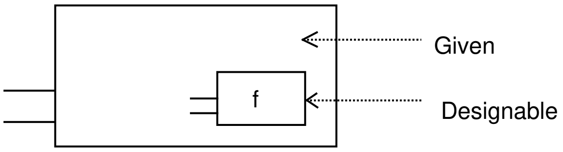

Figure 0.1: The designable part of the system.

A high-level description of the design method described in this book is:

I. List requirements carefully and find a mathematical representation for them.

II. Obtain a performance function Γ from the given part of the system and the requirements listed in I, in terms of the designable variable f. The performance function is a function Γ of frequency ω and f. In other words, the performance of the closed loop system at frequency ω is $\Gamma(j\omega, f(j\omega))$. The function Γ is actually a cost function; the smaller its value, the better the design.

III. Optimize the performance to obtain an acceptable system, or conclude that no such system exists with the given requirements.

This book presents a list of requirements that are sound from the physical point of view yet simple mathematically. The list of requirements discussed here is by no means complete: as more is understood about systems, more requirements can be added to those in this book. It is our contention that the framework is flexible enough to accommodate them. Several examples illustrate the practical solution of design problems using steps I – III.

H^∞-Control optimizes the performance over all frequencies, not just averages (mean square) performance over frequencies. This approach is true to the physical problem where the older mean square optimization approaches distorts engineering specifications in order to produce a mathematically easy problem. Fortunately in the last 15 years the mathematics of H^∞ has become powerful enough to solve engineering problems.

Software

This book also can serve to introduce the control software package OPTDesign, which runs under Mathematica. The book is independent of the software; however, the companionship of the software provides a richer experience. With OPTDesign the reader can easily reproduce the calculations done here in the solved examples, and even try variations on them.

SISO and MIMO

This book emphasizes single-input–single-output (SISO) systems. However, the methods used here can be generalized readily to multiple-input–multiple-output systems (MIMO). MIMO design problems are not supported by the current version of the software package OPTDesign. However, the optimization routine Anopt used by OPTDesign has the capability of solving optimization problems of the kind that arise in MIMO design problems.

Frequency domain and time domain

One of the fundamental limitations of frequency domain techniques for system design has been the lack of knowledge about how to translate performance measures in the time domain to the frequency domain. This is no simple problem, for there are deep and complicated connections between a function and its Laplace transform. Some results have been obtained by mathematicians since 1930, but they have not been incorporated in practical design [1]. This may be explained by the fact that to take full advantage

---

[1] See the comment in [Hor, page 190]

of these results requires well developed optimization theory, computational methods, and computer capabilities. Recent advances in these areas make it possible to use some of the aforementioned results. In this book, we take advantage of this and present a method for dealing with time domain requirements using frequency domain tools.

Low order models

The mathematics presented here and implemented in OPTDesign produces solutions in the form of values of a function on a grid on the $j\omega$ axis, rather than as a rational function. Although it is possible to obtain rational functions here, order reduction and identification is not the main ingredient of the method, and thus it is not emphasized in this exposition[2].

Recommended background

The reader should know about basic control concepts such as those contained in the popular textbooks [FPE], [D] and [O].

Advanced books that present H^∞ theory but with a different approach include [Fr], [LG], [BGR], [FF], [GM], [DFT], [BB], [H], [Dy],[L]. All of these books are based on the same type of mathematics, which are extensions of something called Nevanlinna-Pick, Nehari and Commutant Lifting theories. The mathematical core of our book, presented in the later parts is fundamentally different.

A book that that can be used at the undergraduate level and that contains many recent developments in the field is [DFT]. However, we feel that the approach taken in our book is simpler and more direct.

Thanks

Thanks to Trent Walker, David Ring, Robert O'Barr, David Schwartz, Neola Crimmins, Joy Kirsch, Rosetta Fuller, Mike Swafford and John Flattum. John Boyd, Dan Cunningham, Frank Hauser, and Alan Schneider made valuable comments on the exposition. Julia Myers read many drafts of the book and found many typographical and grammatical errors, and cases of unclear exposition. We thank Julia for her wonderful work. We are especially grateful to Prof. Fred Bailey of the University of Minnesota, his student Brett Larson, and to Prof. A. T. Shenton and research assistant Zia Shafiei of the University of Liverpool. Finally, we would like to

---

[2] The software package OPTDesign has procedures for approximating discrete data with rational functions.

thank the Air Force Office of Scientific Research and the National Science Foundation for partially supporting the writing of this book through grants.

Code development

OPTDesign and its companion Anopt are based on algorithms due to Helton, Merino and Walker, and written in Mathematica by Orlando Merino, Julia Myers and Trent Walker. Also Daniel Lam, Robert O'Barr, Mark Paskowitz and Mike Swafford contributed. Earlier versions of Anopt (Fortran, 1989) were designed and written by Jim Bence, J. William Helton, Julia Myers, Orlando Merino, Robert O'Barr and David Ring. An even earlier Fortran package is Approxih by David Schwartz and J. W. Helton (1985).

OPTDesign

The software package OPTDesign is available in the language Mathematica. To obtain OPTDesign, send electronic mail to *anopt@.ucsd.edu* or see the instructions in the Appendix.

J. William Helton *Orlando Merino*

References and further reading

[BGR] J. A. Ball, I. Gohberg and L. Rodman. Interpolation of rational matrix functions. Birkhauser-Verlag, Basel, 1990.

[DFT] J. C. Doyle, B. A. Francis and A. T. Tannenbaum. Feedback Control Theory. Maxwell Macmillan Int., N. Y., 1992.

[Hor] I. M. Horowitz. *Synthesis of Feedback Systems.* Academic Press, 1963.

[FF]C. Foias and Frazho. The commutant lifting approach to interpolation problems. Birkhauser-Verlag, Basel, 1991.

[FPE] G. F. Franklin, J. D. Powell and A. Emami-Naeini. *Feedback Control of Dynamic Systems.* Addison-Wesley, 1986.

[BB] S. Boyd and C. P. Barrat. *Linear compensator design: limits of performance.* Prentice Hall, 1991.

[Dy] H. Dym. J contractive matrix functions, reproducing kernel Hilbert spaces and interpolation. CBMS Reg. Conf. Ser. in Math., vol 71, Amer. Math. Soc., Providence, RI, 1989.

[Fr] B. Francis. *A First Course in* H^∞ *control.* Springer Verlag, 1986.

[GL]K. Glover and D. C. McFarlane. Robust Controller Design Using Normalized Coprime Factor Pland Descriptions. Vol. 138 in Lecture Notes in Control and Information Sciences. Springer Verlag, New York. 1990.

[GM] M. Green and D. J. N. Limebeer. Book

[O] K. Ogata. *Modern Control Engineering.* Prentice Hall, Second Edition, 1990.

[H1] J. William Helton. Worst case analysis in the frequency domain: the H^∞ approach to control. IEEE Transactions on Automatic Control, Vol AC-30, No. 12, pp 1154-1170, 1985.

[H2] J. W. Helton. Broadbanding: Gain equalization directly from data. IEEE Transactions on Circuits and Systems, Vol CAS-28, No. 12, pp 1125-1137, 1981.

[SS] Z. Shafiei and A. T. Shenton. Theory and application of H^∞ disk method. Report No. MES/ATS/BAE/002/90, Department of Mechanical Engineering, University of Liverpool, U.K., October 1990.

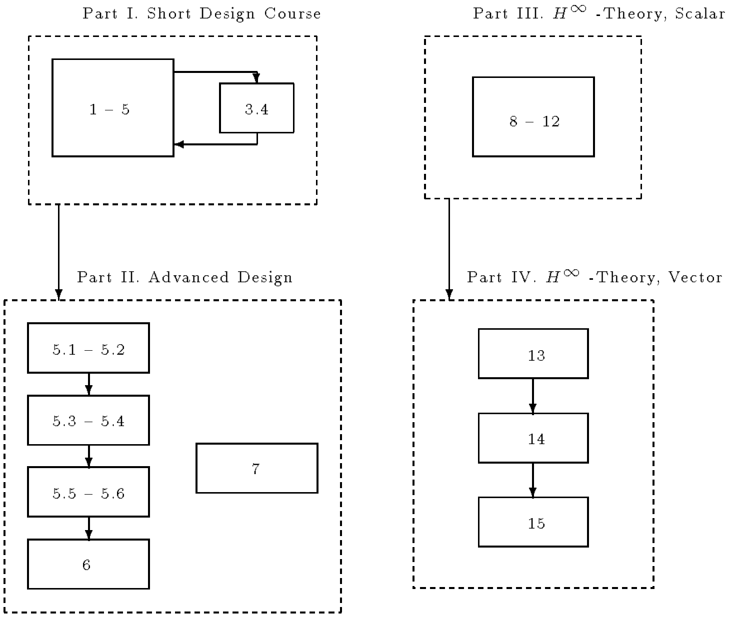

Figure 0.2: Paths to be followed in order to read this book. The short course includes chapters 1–5. More advanced studies may include one or more of the topics in Part II in the order indicated by the arrows. Graduate studies may include the H^∞ theory, presented in parts III and IV.