Robust Optimal Receding Horizon Control

by
Margaret Bell, B.Sc.(Eng.)

A thesis submitted for the degree of Doctor of Philosophy of the University of London
and for the Diploma of Membership of the Imperial College

Centre for Process Systems Engineering
and
Department of Chemical Engineering and Chemical Technology
Imperial College of Science, Technology and Medicine
University of London
London SW7 2BY

March 1997

To Dad and Mom for always being there...

even when I wasn't.

Abstract

Receding horizon optimal control has a long history and provides the basis for several of the currently popular techniques for model predictive control of chemical processes. Based on a prediction of the current state of the process, derived from plant measurements, and predictions of the likely input conditions over a certain period ahead (the time horizon), optimal controls are computed to maximise a given "performance function", which can be either an economic or technical measure of performance. The state predictions are regularly updated as new measurements are made and the controls are re-computed if either the current state or the predicted input conditions change significantly.

An important advantage of computing optimal controls is that this can take direct account of process operating constraints. However, general inequality constraints have always posed a difficult problem in numerical optimal control algorithms and we propose a new algorithm for dealing with this problem for systems described by DAEs of any index.

Due to both the excessive time required for model development and the excessive computational time required for a full model, the on-line model is generally of a much lower complexity than the true system and although parameters in the model may be re-estimated using plant measurements, there is inevitably some uncertainty in the model predictions as well as in the input predictions. The controls must therefore be robust, giving reasonable performance in the face of variations of actual conditions from the predictions.

To obtain a sufficiently simple controller, the $\mathcal{H}_\infty$ approach has been to design a controller to perform optimally if the worst possible set of inputs and model predictions actually occur, and this can also be applied within the receding horizon optimal control

framework. The result is a game, with the controls playing against the uncertain inputs and model parameters. However, this does not guarantee robustness in the presence of inequality constraints and here we propose a new simple technique for dealing with this problem, which exploits the sampling interval of the receding horizon formulation of the problem.

Acknowledgements

This work would not have been possible without the inspiration and support of Professor Roger Sargent. I thank him for supervising me in his "retirement" and for sharing with me a small fraction of his knowledge and experience. I feel privileged to be a member of his "family". I thank Professor David Limebeer for introducing me to the field of $\mathcal{H}_\infty$ control (and for his frank sense of humour). Finally, I thank Professor Richard Vinter for guiding me with his expertise in the area of existence theory. Financial support from the Centre for Process Systems Engineering in the form of an Engineering and Physical Sciences Research Council (EPSRC) Research Assistantship is gratefully acknowledged.

I was fortunate to share the last few years with a colourful group of colleagues who ensured the years were filled with many interesting moments. They all have contributed to the maintenance of my sanity, either by useful technical discussions or by general silliness. It is impossible to thank everyone, but many deserve mention:

"The Greek Majority": Vassilis (Silly Billy, Billy Kat) Katsouros, Thanassis (Arnie) Pantas, Takis and Kallia (The Smiling Greeks) Fotopoulos, Alexis Coutlis, Pavlos Mavrikis, Michaelis Zervos and Stavros Stefanis.

"The British Minority": Andy (The Joker) Buxton, Darryl (Mr. Talkative) Copsey, Jonathan (P.M.) Heath, Nefyn (Mr. President) Jones, Vince (Auch!) White and Patrick (Froggie) Woodford.

"The Odd Balls": Nick Alsop (and his many surprising comments!), Jezri (Chill Out) Mohideen, Soraya Ismail, Jae Cho, Geoff Bell, Haitham El-Zobaidi, Crina Manolescu and Patricia Hoch.

Words cannot express my gratitude to my family, especially my Mom and Dad, for their love, encouragement, support and, most importantly, for understanding and accepting my absence at important family occasions over the last 3 years.

Finally, I thank Rob simply for being Rob - for supporting me and continually reminding me of the importance of laughing.

Now it's full night, clear, moonless and filled with stars, which are not eternal as was once thought, which are not where we think they are. If they were sounds, they would be echoes, of something that happened millions of years ago: a word made of numbers. Echoes of light, shining out of the midst of nothing.

It's old light, and there's not much of it. But it's enough to see by.

Margaret Atwood in *Cat's Eye*

Contents

List of Figures **11**

Nomenclature **13**

1 Thesis Outline **17**

2 Literature Review **20**
2.1 Receding Horizon Control 20
2.1.1 A History of Receding Horizon Control 21
2.1.2 Linear Discrete Time Receding Horizon Control 22
2.1.3 Robustness and Stability for Finite Horizons 24
2.1.4 Robustness and Stability for Infinite Horizons 24
2.1.5 Cost Functions 25
2.1.6 Considering Nonlinear Systems 25
2.1.7 Choice of Horizon 30
2.1.8 Conclusions 31
2.2 Optimal Control 31
2.2.1 Statement of Optimal Control Problem 32
2.2.2 Equivalent Optimal Control Problem Formulations 32
2.2.3 Conditions for Existence of Optimal Controls on a Finite Horizon . 33
2.2.4 Necessary Conditions for Optimality 34
2.2.5 Conclusions 35
2.3 Computational Methods 36
2.3.1 Control Parameterisation 36
2.3.2 Collocation Methods 38
2.3.3 Direct Solution of Necessary Conditions 39
2.3.3.1 Single Shooting 40
2.3.3.2 Multiple Shooting 40
2.3.3.3 Quasilinearisation 40
2.3.3.4 Comparison of Single versus Multiple Shooting 40
2.3.4 Conclusions 41
2.4 Dealing with State Endpoint Equality Constraints 42
2.4.1 Control Parameterisation 42

2.4.2 Collocation 42
2.4.3 Direct Solution of Necessary Conditions 42
2.4.3.1 Bryson and Ho (1975) Method 42
2.4.3.2 Miele's (1973) Gradient-Based Method 44
2.4.4 Conclusions 46
2.5 Dealing with High-index DAE Systems 47
2.5.1 The Gear (1988) Algorithm 48
2.5.2 The Campbell (1988) Algorithm 49
2.5.3 The Bachmann *et al.* (1990) Algorithm 49
2.5.4 The Chung and Westerberg (1990) Algorithm 49
2.5.5 The Mattsson and Söderlind (1993) Algorithm 49
2.5.6 The Pantelides *et al.* (1994) Algorithm 50
2.5.7 The Jarvis (1994) Algorithm 50
2.5.8 Conclusions 50
2.6 Constraint Handling 51
2.6.1 Collocation 52
2.6.2 Conversion to Point Constraint 53
2.6.2.1 Integration of Violation 53
2.6.2.2 Vassiliadis' (1993) Hybrid Approach 54
2.6.2.3 Gritsis' (1990) Bound on Minimum Value of Constraint . 55
2.6.3 Conversion to Equality Constraint 56
2.6.3.1 Valentine's (1937) Slack Variable Approach 56
2.6.3.2 Implicit Constraints 56
2.6.4 Conversion to Unconstrained Problem 57
2.6.4.1 Penalty Methods 57
2.6.4.2 Barrier Functions and the Interior Point Method 58
2.6.5 Conclusions 59
2.7 Control in the Presence of Process Uncertainty and Plant/Model Mismatch 59
2.7.1 Defining Robustness in a Process Systems Context 59
2.7.2 Controlling Deviations: 1 and 2−norms versus ∞−norm 61
2.7.3 $\mathcal{H}_\infty$−Optimisation and Worst Case Design 61
2.7.4 Combining Receding Horizon Control with $\mathcal{H}_\infty$−Optimisation . . . 62
2.7.5 Dealing with Constrained Uncertain Systems 64
2.7.6 Conclusions 65
2.8 Concluding Remarks 65

3 Existence of Optimal Controls **67**
3.1 Introduction and General Problem Definition 67
3.2 Choice of Cost Function in a Process Systems Context 69
3.3 Formulating The Infinite Horizon Optimal Control Problem 69
3.4 Useful Definitions and Concepts 71
3.5 Conditions for the Existence of Infinite Horizon Optimal Controls 79
3.5.1 Comments on Existence Conditions 80
3.5.2 Proof of Theorem 3.5.1 81
3.6 Conclusions 86

4 Necessary Conditions for Optimal Controls **87**
4.1 Problem Definition 87
4.1.1 Original System Definition 87
4.1.2 Modified System Definition 89
4.2 Development of Adjoint System 90
4.3 Variational Analysis 92
4.3.1 Augmented Cost Functional 92
4.3.2 Proof that higher order terms in variational expansion are negligible 94
4.3.3 Integration of DAE system 96
4.3.4 Transversality Conditions 97
4.3.4.1 Initial Transversality Conditions 97
4.3.4.2 Endpoint Transversality Conditions 98
4.3.5 Path Conditions 98
4.3.6 Necessary Conditions in the Limit 99
4.4 Conclusions 100

5 Algorithm Development **101**
5.1 Introduction of Slack Variables 102
5.1.1 Modified Transversality Conditions 104
5.1.1.1 Modified Initial Transversality Conditions 104
5.1.1.2 Modified Endpoint Transversality Conditions 105
5.1.2 Modified Path Conditions 105
5.2 The Basic Algorithm 106
5.3 Inequality Constraints and The Interior Point Method 107
5.3.1 Determining $\Delta\mathcal{X}_k$ 108
5.3.2 Determining the Damping Factor α_k 113
5.3.3 Interior Point Algorithm 116
5.3.3.1 Initialisation 117
5.3.3.2 General Major Iteration 117
5.3.4 Implementation Details 117
5.3.5 Convergence of the Interior Point Method 118
5.4 Overall Convergence of Algorithm 123
5.5 Singular Solutions 124
5.6 Free Endtime Problems 125
5.7 Parameter Optimisation 126
5.8 Dealing with Equality Constraints at the Endpoints 126
5.8.1 Partition of the Adjoint Equations 126
5.8.2 Miele's Combined Gradient-Restoration Scheme 129
5.8.3 Conclusions 130
5.9 Infeasibility of Inequality Path Constraints 130
5.9.1 Method of Solution using Active Sets 131
5.9.1.1 Modified Basic Algorithm - I 132
5.9.1.2 Modified Basic Algorithm - II 132
5.10 An Illustrative Example 134
5.10.1 Direct Application of the Necessary Conditions 135
5.10.2 An Alternative Method of Handling State Inequality Constraints . 138
5.11 Conclusions 140

6 Numerical Validation **142**
6.1 Consecutive First Order Reactions 143
6.2 Batch Reactor with Control Saturation 145
6.3 Catalyst Mixing Problem 147
6.4 Car Problem 150
6.5 Minimum Time Car Problem 152
6.6 State Constrained Car Problem 153
6.7 Conclusions 156

7 Robust Control **158**
7.1 Dealing with Uncertainty 158
7.1.1 Differential Game Theory Approach for Unconstrained Systems . . 158
7.1.2 New Modified Inequality Approach for Constrained Systems 159
7.2 An Illustrative Example 162
7.3 Conclusions 163

8 An Algorithm for High Index DAE Systems **164**
8.1 Partitioning of High Index DAE Systems 164
8.2 Numerical Partitioning 165
8.2.1 Example of partitioning 167
8.3 Formulating the Augmented Optimal Control Problem 167
8.4 Numerical Partitioning versus Structural Partitioning 168
8.5 An Illustrative Example 170
8.5.1 Parallel Reaction Scheme 170
8.6 Conclusions 171

9 Conclusions and Future Directions **173**
9.1 Existence of Optimal Controls 173
9.2 New Necessary Conditions for Constrained DAE Systems 174
9.3 Optimal Control Algorithm 174
9.4 Robustness 177
9.5 High Index DAE systems 177
9.6 Computational Considerations 178
9.7 Conclusions 179

References **180**

List of Figures

2.1 Receding Horizon Control Structure 21
2.2 Receding Horizon Control Strategy 23
2.3 Single shooting versus Multiple shooting 39
2.4 SISO Feedback Loop 61
2.5 Saddle Point 64
2.6 Contours of Saddle Point 65
2.7 Infeasibility of Saddle Point Solution 66

3.1 Riemann Integral 74
3.2 Lebesgue Integral 75
3.3 Hausdorff Distance 78
3.4 Hausdorff Continuity 79
3.5 Graph of $u_i(\cdot)$, $i = 1, 2, ...$ 81
3.6 Graph of $x_i(\cdot)$, $i = 1, 2, ...$ 81
3.7 Diagonalisation Procedure: S_1 on $[0, T_1]$ 84

5.1 Determining α_k 116
5.2 Demonstrating $\alpha_k > 0$ 122
5.3 DASOLV: "Locking" of an implicit discontinuity 134

6.1 Consecutive Reactor: Temperature Profile 1 144
6.2 Consecutive Reactor: Temperature Profile 2 144
6.3 Consecutive Reactor: Concentration Profiles 145
6.4 Consecutive Reactor: Cost and Error Profile 145
6.5 Batch Reactor: Control Profile 146
6.6 Batch Reactor: Concentration Profiles 147
6.7 Batch Reactor: Cost and Error Profiles 147
6.8 Catalyst Mixing: Control Profiles 149
6.9 Catalyst Mixing: State Profiles 149
6.10 Catalyst Mixing: Cost and Error Profiles 150
6.11 Car: Distance and Velocity Profiles 151
6.12 Car: Acceleration Profile 151
6.13 Minimum Time Car: Distance and Velocity Profiles 153

6.14 Minimum Time Car: Acceleration Profile 153
6.15 Constrained Car: Distance and Velocity Profiles 155
6.16 Constrained Car: Acceleration Profile 155

7.1 Modified Inequality Bounds 162
7.2 Constrained Car: Distance and Velocity Profiles with Robustness Considerations 162
7.3 Constrained Car: Acceleration Profile with Robustness Considerations . . 163

8.1 Parallel Reaction Scheme: Concentration Profiles 170
8.2 Parallel Reaction Scheme: Inlet and Outlet Profiles 171

Nomenclature

Chapter 3

A	a set
$\bar{\alpha}$	objective function parameter, to penalise excessive control action
α	lower bound on range of function f
β	upper bound on range of function f
a	lower bound on time domain
b	upper bound on time domain
a_i	lower bound on control, u_i
b_i	upper bound on control, u_i
$\mathbb{B}^m$	open unit ball in $\mathbb{R}^m$
B	Borel set
$d_A(x)$	distance of a point x from a set A
$d(A, B)$	Hausdorff distance between two sets
ϕ	state dynamics
f	state dynamics
$F(x)$	feedback law
ϕ	differential equation
γ	convexity bound on cost integrand L
$G(x)$	multifunction
I	time interval
J	cost function
$k(t)$	Lipschitz function
L	integrand of integral cost function
L_u	upper Lebesgue integral
L_l	lower Lebesgue integral
$m(E_k)$	measure of set E_k
$\mathbb{N}$	the Natural numbers
Ω	control space (i.e. $u \in \Omega$)
$\mathcal{O}$	null set
P	optimal control problem
$\mathbb{R}$	the Real numbers
R_u	upper Riemann integral

R_l	lower Riemann integral
S	upper Riemann/Lebesgue sum
s	lower Riemann/Lebesgue sum
$S(t,x)$	extended velocity set
u	control
v	state velocity
X_0	set of admissible initial states
x	state

Chapter 4

f_x	Jacobian of f with respect to differential states x
ϕ^0	equality constraint at t_0
ϕ^f	equality constraint at t_f
g	inequality constraint on $[t_0, t_f]$
$\overline{g}$	upper bound on inequality constraint on $[t_0, t_f]$
$\underline{g}$	lower bound on inequality constraint on $[t_0, t_f]$
μ	barrier function parameter associated with g on (t_0, t_f)
μ^0	barrier function parameter associated with g at t_0
μ^f	barrier function parameter associated with g at t_f
ν^0	barrier function parameter associated with ψ^0 at t_0
ν^f	barrier function parameter associated with ψ^f at t_f
P	objective function defined at t_f
ψ^0	inequality constraint at t_0
$\overline{\psi}^0$	upper bound on inequality constraint at t_0
$\underline{\psi}^0$	lower bound on inequality constraint at t_0
$\overline{\psi}^f$	upper bound on inequality constraint at t_f
$\underline{\psi}^f$	lower bound on inequality constraint at t_f
p	adjoint variable
t_0	initial time
t_f	final time
V	vector of time-invariant optimisation parameters, v
$\overline{v}$	upper bound on time-invariant optimisation parameters
$\underline{v}$	lower bound on time-invariant optimisation parameters
u'	control variables
u	vector of control variables and algebraic variables, $u = [u', x']$
x	differential state variables
x'	algebraic state variables

Chapter 5

α	interior point damping parameter
α'	backwards differentiation formula parameter (in DASOLV)

β	interior point parameter
β'	backwards differentiation formula parameter (in DASOLV)
δ	interior point parameter
f_x	Jacobian of f with respect to differential states x
ϕ^0	equality constraint at t_0
ϕ^f	equality constraint at t_f
γ	interior point parameter
γ'	backwards differentiation formula parameter (in DASOLV)
g	inequality constraint on $[t_0, t_f]$
$\overline{g}$	upper bound on inequality constraint on $[t_0, t_f]$
$\underline{g}$	lower bound on inequality constraint on $[t_0, t_f]$
h_n	time step in DASOLV
μ	barrier function parameter associated with g on (t_0, t_f)
μ^0	barrier function parameter associated with g at t_0
μ^f	barrier function parameter associated with g at t_f
ν^0	barrier function parameter associated with ψ^0 at t_0
ν^f	barrier function parameter associated with ψ^f at t_f
P	objective function defined at t_f
ψ^0	inequality constraint at t_0
$\overline{\psi}^0$	upper bound on inequality constraint at t_0
$\underline{\psi}^0$	lower bound on inequality constraint at t_0
$\overline{\psi}^f$	upper bound on inequality constraint at t_f
$\underline{\psi}^f$	lower bound on inequality constraint at t_f
p	adjoint variable
ρ	interior point parameter
r	slack variable associated with parameter inequality V
s	slack variable associated with inequality g
σ	slack variable associated with inequality ψ
t_0	initial time
t_f	final time
V	vector of time-invariant optimisation parameters, v
$\overline{v}$	upper bound on time-invariant optimisation parameters
$\underline{v}$	lower bound on time-invariant optimisation parameters
u'	control variables
u	vector of control variables and algebraic variables, $u = [u', x']$
x	differential state variables
x'	algebraic state variables
$\mathcal{X}$	vector of interior point iterates

Chapter 7

f	DAE system
$f_{\dot{x},x'}$	Jacobian of system equations with respect to state velocity and algebraic states
g	path inequality constraint
$\parallel g_x \parallel$	norm of Jacobian of g with respect to x

$\overline{g}$	upper bound on path inequality constraint
$\underline{g}$	lower bound on path inequality constraint
γ	disturbance attenuation term
K_x	bound on deviation of initial state from nominal value
K_v	bound on deviation of parameter from nominal value
K_w	bound on deviation of disturbance from nominal value
τ	sampling interval of process
$u(t)$	control trajectory
v	time-invariant optimisation parameter
$\overline{v}$	nominal value for optimisation parameter
$w(t)$	disturbance trajectory
$\overline{w}(t)$	nominal value for disturbance
$\overline{x}(t_0)$	nominal value for initial state x
$z(t)$	performance objective

Chapter 8

x	differential state variables
$\tilde{x}$	partition of differential state variables corresponding to index one subsystem $\tilde{f}$
$\hat{x}$	partition of differential state variables corresponding to "controlled" subsystem $\hat{f}$
x'	algebraic state variables
$\tilde{x}'$	partition of algebraic state variables corresponding to index one subsystem
$\hat{x}'$	partition of algebraic state variables corresponding to "controlled" subsystem $\hat{f}$
f	DAE system
$\tilde{f}$	index one subsystem
$\hat{f}$	"controlled" subsystem
L	lower triangular matrix with unit diagonal
R	row permutation matrix
C	column permutation matrix
$\tilde{U}$	nonsingular upper triangular submatrix of reduced Jacobian $[f_{\dot{x}}\ f_{x'}]$
U	nonzero submatrix of Jacobian $[f_{\dot{x}}\ f_{x'}]$
$\hat{u}$	control variable generated in index reduction procedure, $\hat{u} = \dot{\hat{x}}$
u'	control variable

CHAPTER 1

Thesis Outline

Optimal control of process systems presents many challenging problems. The most problematic are:

- **Nonlinearity**: The traditional approach of control has been to assume a linear model provides an adequate description of the system dynamics. However, chemical processes are prone to extremely nonlinear behaviour.

- **Stiffness**: In some cases the dynamics of a process may be stiff. Stiff systems are characterised by mixed states of slow and fast dynamics.

- **Path Constraints**: Inevitably, process systems will be subject to constraints on both the control variables as well as the state variables, due to the physical limitations of the plant (eg. valve sizing) and safety limitations (eg. pressure and temperature). While much work has been done for systems with constrained control activity, less has been done to consider the more difficult problem of state restrictions.

- **Endpoint Constraints**: It is often the case that the optimal control policy is required to bring the system to some specified state at the end of the control horizon (eg. steady state after grade changeover).

- **Modelling Errors and Disturbances**: Model predictive control strategies must assume the model to be an adequate description of the system dynamics. However, a true description of the system may require a computationally infeasible

large model. Furthermore, instrumentation noise may introduce drift in the system. thereby changing the system characteristics. It is important for a control algorithm to recognise this possibility, if only for safety considerations and operating restrictions.

- **Model Characteristics**: In general, process systems cannot be modelled by simple ordinary differential equation (ODE) systems due to the presence of algebraic equations (eg. mass balances). Furthermore, the differential algebraic equation systems are generally assumed to be index one. However, high index systems do arise in a process systems context.

- **Multivariable Systems**: Most process control problems involve several variables to *be* controlled as well as several manipulated variables to *do* the controlling.

This thesis will address these issues in the following way:

Chapter 2

In Chapter 2 we present a literature survey outlining previous research in receding horizon control, constrained systems, existence of optimal controls, robust optimal control and integration of high index DAEs.

Chapter 3

In Chapter 3 we consider the general formulation of continuous time optimal control problems and present a proof of existence of optimal controls for problems defined on a free, and perhaps infinite, horizon. This chapter includes an introduction to the necessary theory and provides intuitive definitions of the theoretical tools.

Chapter 4

In this chapter, we develop necessary conditions for optimal control of constrained DAE systems of a very general nature. The development of the necessary conditions involves the use of logarithmic barrier functions to inhibit deviations from the constrained feasible region. Thus, we augment the cost function with a logarithmic penalty term and minimise this augmented cost.

Chapter 5

In Chapter 5 we present a new algorithm for dealing with the control of systems defined by index one constrained DAEs. The approach is new in its use of logarithmic barrier functions to inhibit deviations from the constrained feasible region. Slack variables are introduced as a means of transforming the original inequality constrained optimal control problem to a sequence of equality constrained optimal control problems. The interior point method is introduced as an innovative way of dealing with such constraints in optimal control. Path constraints are dealt with using an active set strategy.

Chapter 6

In Chapter 6 we present a series of illustrative examples of the utility of this new approach.

Chapter 7

In Chapter 7 we consider the effect of model mismatch and measurement error on the control of constrained systems and develop a new method of dealing with inequality constrained systems subject to uncertainty.

Chapter 8

In Chapter 8 we show that the algorithm proposed in Chapter 5 can be extended to high index DAEs by reformulating the system equations as a control problem with endpoint equality constraint.

Chapter 9

Finally, in Chapter 9 we summarise the research and outline future directions.

CHAPTER 2

Literature Review

This chapter aims to provide a comprehensive and critical review of aspects of process control relevant to this thesis. The first section provides an introduction to the receding horizon control paradigm of model-based predictive control. Subsequent sections will introduce general optimal control theory; outline the various computational methods of optimal control; introduce high index differential algebraic equations (DAEs) and discuss the methods available for dealing with systems of this form; provide an overview of methods to handle constraints; and, finally, discuss notions of robustness to model mismatch and disturbances. The review demonstrates the various approaches one can take when designing a robust dynamic optimisation algorithm.

2.1 Receding Horizon Control

Many of the recent advances in process control research have been a direct result of practical need within industry. The desire for higher product quality, coupled with the environmental issues of efficient use of energy and raw materials, has resulted in the need for advanced control methodologies. Model-based predictive control (MBPC) techniques are a direct result of this need and are among the most widely used advanced control techniques in industry. The practical advantage of model-based predictive controllers is their ability to yield high performance control systems requiring little intervention (Garcia *et al.*, 1989). Research in model-based process control has demonstrated that controllers employing a process model have superior performance over traditional methods in many chemical engineering processes (Lee *et al.*, 1989).

2.1.1 A History of Receding Horizon Control

The receding horizon control methodology evolved from general model-based predictive control techniques. The technique uses an explicit dynamic model of a plant to predict the effect of future manipulated variable moves. The moves are then calculated to minimise a specified performance objective, while satisfying all constraints. Therefore, it has two central tasks:

1. *prediction* of future plant behaviour over a defined time horizon

2. *computation* of control action to achieve a desired objective.

It is well suited to process applications, since most process systems are slow to respond to control changes. This allows the necessary time for the online control optimisation to be carried out between implementations of control variable changes.

The general structure of the receding horizon controller is illustrated in Figure 2.1. Measurements from the plant are passed to the optimiser which first generates the initial states from a state estimator. Having these initial states, the optimiser then computes the optimal control trajectory which satisfies the process constraints over some finite horizon $[t_0, t_f]$ and implements this control strategy until the next measurements are available.

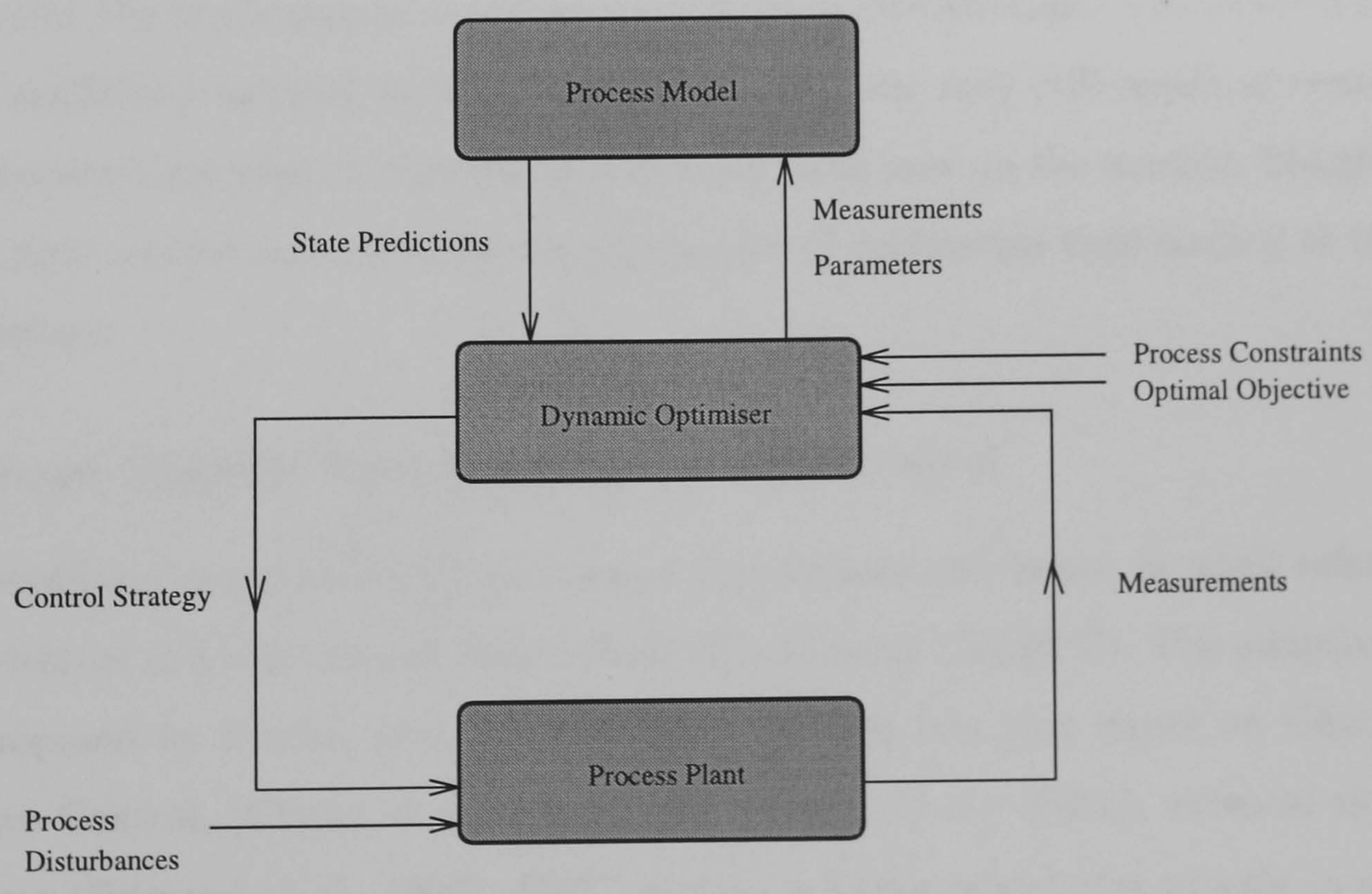


Figure 2.1: Receding Horizon Control Structure

The long history of these techniques is outlined in a survey paper written by (Garcia *et al.*, 1989). The combination of predictive and adaptive techniques was proposed by many, including Richalet *et al.* with the "output smoothing" approach of IDCOM (Richalet *et al.*, 1978); Cutler and Ramaker's (1980) step-response model approach, known as DMC and later modified to include a quadratic objective (QDMC) (Garcia and Morshedi, 1986); and Garcia and Morari (1986) with the IMC algorithm which reduced the number of adjustable parameters of DMC to produce a more easily tuned controller and, according to (Ogunnaike and Ray, 1994), was the first algorithm to place predictive techniques in a "classical" framework. All of these algorithms share two basic elements:

1. Each performs a finite (time) horizon optimisation at each time interval. As new measurements become available, the horizon moves forward and a new control trajectory is computed. This shifting of the optimised interval is labelled a *receding horizon* and is illustrated in Figure 2.2.

2. Each method incorporates a *model* used to predict the process response over this fixed horizon.

A further point is that each of these techniques is implemented in discrete time, using discrete time models. While model development from first principles yields continuous time models, the implementation will necessarily be in discrete time. The discrete nature, then, is artificially induced by computers. However, one may still speak of continuous versus discrete time control algorithms. We shall focus now on the existing literature on discrete time control and highlight the advantages of continuous time control at the end of this section.

2.1.2 Linear Discrete Time Receding Horizon Control

Model predictive control evolved as a linear control tool and hence we shall refer to its earlier versions as Linear Model Based Predictive Control (LMBPC). The adaptive technique proposed by Clarke, Mohtadi and Tuffs' in their two part paper on Generalised Predictive Control, (Clarke *et al.*, 1987a) and (Clarke *et al.*, 1987b), received the most acceptance (Bitmead *et al.*, 1990). GPC assumes a linear model of a process in discrete time. Thus, any future input, state or output can be expressed as a linear combination of future control increments. This two-part paper introduced GPC as a method superior

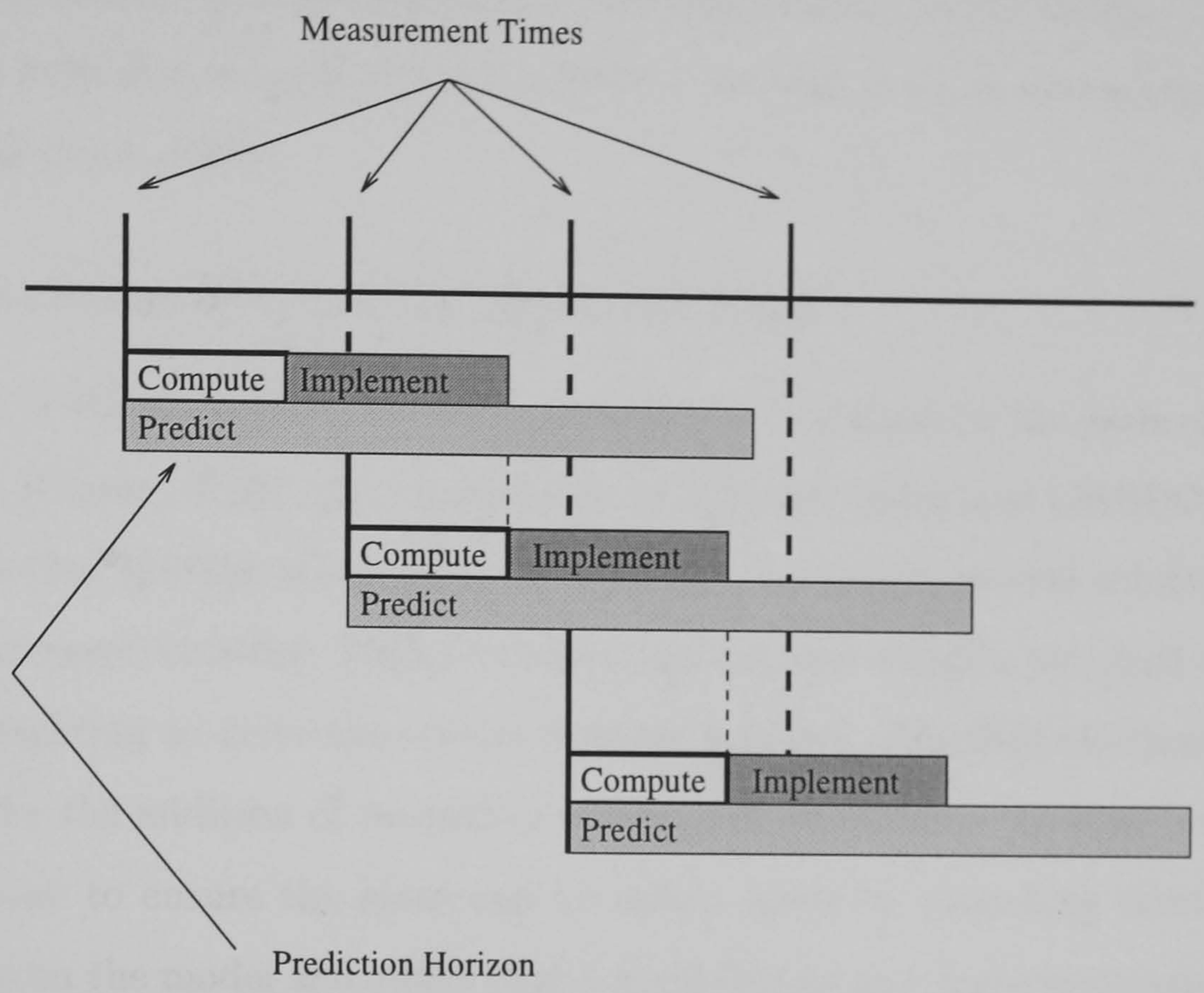


Figure 2.2: Receding Horizon Control Strategy

to generalised minimum variance and pole-placement techniques. Generalised minimum variance controllers suffer from demanding excessive control input in non-minimum phase plants, the result often being intersample rippling. Pole-placement suffers if the plant model is overparameterised, whereas GPC seems unaffected. The "general purpose" aspect of GPC is due to the fact that it is applicable to nonminimum phase plants; open-loop unstable plants or plants with badly-damped poles; plants with variable or unknown dead-time and plants of unknown order. GPC is a receding horizon method which predicts a plant's output over "k" steps based on the assumption of a fixed control horizon beyond which there is a zero control increment. This assumption has the advantages of robustness and simplification of calculations. Future control moves are determined by optimising an objective function to minimise the predicted error subject to operating constraints. The optimisation may be formulated as a quadratic program (QP)

$$J(u,t) = E\{\sum_{j=N_1}^{N_2} \Gamma_j[y_{t+j} - r_{t+j}]^2 + \sum_{j=1}^{N_u} \Lambda_j[\Delta u_{t+j-1}]^2\} \tag{2.1}$$

subject to $\Delta u_{t+i} = 0, i = N_u, ..., N_2$. The calculation of controls is repeated at the beginning of each control horizon, thus making GPC "adaptive" in the sense that using new measurements at each sampling interval ensures that the control implemented at each

t depends on the current state measurement. The minimisation produces Δu_t, Δu_{t-1}, ..., Δu_{t+N_u+1}, but only Δu_t is applied to the system. At time $t+1$ a new minimisation is solved (Bitmead *et al.*, 1990).

2.1.3 Robustness and Stability for Finite Horizons

Most, if not all, of the research in the area of LMBPC is focussed on the issue of stability. As discussed in (Clarke, 1993), the combination of heuristic industrial LMBPC, adaptive control and receding horizon state-space theories has resulted in several solutions to the problem of guaranteed stability. Each of these solutions uses either a terminal constraint or end-point weighting to drive the system towards a target. The finite horizon strategy readily allows for the addition of inequality constraints on actuator or state variables, a very powerful way to ensure the plant can be safely driven to operating limits. Clarke places emphasis on the model and states that a good design captures the dynamics of the plant; provides both the free and forced predictions; is readily understood and verified by plant engineers; is analysable; and, as has been recently recognised, includes a disturbance model (Clarke, 1993).

2.1.4 Robustness and Stability for Infinite Horizons

An alternative stabilising formulation to terminal constraints or endpoint weighting is the infinite horizon formulation. The constrained infinite-horizon model predictive control algorithm is shown in (Balakrishnan *et al.*, 1993) to be globally stabilising for sufficient control moves if and only if the discrete-time linear system is stabilisable and all of its eigenvalues lie in the closed unit disk. For N poles *on* the unit disk, where each pole can be of any multiplicity, the number of control moves must be greater than N to guarantee stability.

A conference paper by (Meadows and Rawlings, 1993) also discusses the infinite horizon formulation. They motivate the use of an infinite horizon by stating the fact that a finite horizon can result in an unstable controller in, for example, a linear continuous time system with a right half-plane zero. The equivalent for discrete time systems is a zero outside the unit disc. By extending the horizon a sufficient length, the receding horizon control becomes stabilising. It is important, however, to note that while formulating the problem with a free endtime may circumvent stability problems, it also may result in the

horizon becoming infinite. Such a formulation is then computationally infeasible.

For linear models, (Meadows and Rawlings, 1993) propose a fixed control horizon with the input constrained to zero beyond the horizon. The objective function is expressed as a finite horizon problem with a quadratic state penalty on the final state. This final state penalty is equal to the value of the objective function on the infinite horizon without any applied control action. A larger quadratic penalty will ensure a stabilising controller. For nonlinear models, it is generally impossible to express the value of an infinite horizon objective in closed form. However, truncation of the infinite series will lose the stability property of the controller. Therefore, a state constraint is added to retain the stability. The infinite horizon objective becomes equivalent to a finite horizon formulation.

2.1.5 Cost Functions

The general LMBPC is formulated with a linear model, linear constraints in the control variables and quadratic objective. The quadratic objective is somewhat restrictive in that it only allows one to minimise the control effort to bring the state to its rest point (0). In a process systems context, the objective is likely to be of a more general nature. One may wish to minimise energy usage or maximise product purity, for example. This introduces the need for more general objective functions which most likely will be nonlinear. Issues such as this lead to the development of nonlinear model based predictive control (NMBPC).

2.1.6 Considering Nonlinear Systems

As pointed out in (Biegler and Rawlings, 1991), the term Linear MBPC is somewhat an abuse of terminology, as the presence of constraints makes LMBPC a nonlinear control problem. There is nothing fundamental in the formulation of linear model-based controllers that restricts the use of nonlinear models. Since most process systems are inherently nonlinear, it seems natural to extend these ideas to nonlinear systems. The main reasons nonlinear control methods have been less widely used in the process industries are:

1. *wealth of linear tools*: The tools of linear control are well-established and well-documented. This coupled with the relative success of such simple control methods

as the proportional-integral-derivative (PID) control have overshadowed the benefits to be gained by applying nonlinear techniques.

2. *availability of nonlinear models*: The rigorous development of nonlinear models from first principles is a time-intensive task. The relative ease with which simple step response models are obtained has resulted in reticence towards development of nonlinear models.

3. *expense of computation*: On-line application of NMBPC has been assumed infeasible due to the fact that computations take far too long to complete in real time.

If regulatory control is the main objective, then linear control may in fact be quite adequate, for the plant will be operating at some steady state and a linearisation of the plant dynamics about this steady state will provide satisfactory results in a neighbourhood of this operating point. However, some plants are severely nonlinear and do not admit to local linearisation. Also, if the plant is subject to frequent changes in operating policy, the step response model will require continual updating. A properly derived nonlinear model will readily adapt to changes in operating conditions. Furthermore, for transient phases such as start-up and shut-down, it may be very important to address the nonlinearity of the process directly in determining the controls for this dynamic problem.

To date there have been essentially three approaches to nonlinear model based predictive control of continuous time systems, as outlined in (Ogunnaike and Ray, 1994):

1. *local linearisation* in which the nonlinear model is linearised about the various steady state operating points.

2. *extended linear MBPC* in which the standard linear MBPC technique is augmented to account for the nonlinearities of the system. The control moves are then perturbations about the controls generated by the linear scheme.

3. *explicit nonlinear MBPC* in which the controller is obtained by direct manipulation of the nonlinear system, either by analytical means or by the solution of a nonlinear program (NLP). Note that for discrete time systems, there is essentially no change in the algorithm due to use of a nonlinear model instead of a linear model as the problem remains solvable as an NLP.

The last of these approaches has received great interest recently. A review paper (Biegler and Rawlings, 1991) outlines recent developments in NMBPC using repeated solution of an NLP to generate the optimal control trajectory. This approach dates back to the work of (Sargent and Sullivan, 1978), among others, who parameterised controls hence translating the control space for continuous time problems from an infinite dimensional to a finite dimensional space. The transformed problem is then readily solved using NLP techniques. The various numerical approaches will be discussed in more detail later in this chapter.

(Kershenbaum *et al.*, 1993) cite several advantages to receding horizon control for the design of nonlinear feedback controllers. Since the control algorithm is developed from a model of the system, a feedback control design can be formulated which takes account of known plant nonlinearities. In comparison with global linearisation techniques, receding horizon control more readily facilitates the incorporation of constraints. Receding horizon control does not require repeated differentiation of the plant model, as is the case with global linearisation, and therefore does not magnify modelling errors by using such differentiations in the controller development.

However, given all of these advantages, one can not assume guaranteed stability. (Kershenbaum *et al.*, 1993) focus on the issue of stabilisation of nonlinear receding horizon controllers. Their work assumes a precise time-varying differential model relating control functions $u(\cdot)$ to state trajectories and perfect measurement of the state. The control variables are subject to constraints

$$u(t) \in \Omega, \text{for } 0 \leq t < \infty \tag{2.2}$$

where $\Omega \subset \mathbb{R}^m$ is a given subset. The control functions are also required to "continuously fit together". The model description must be Lipschitz continuous, with Lipschitz constant independent of t and u, and the origin must be a rest point when the control is set at zero. The cost integrand must be locally Lipschitz continuous and must adequately penalise state deviations from the origin. In fact, the cost integrand must, to a certain extent, dominate the state dynamics in order to penalise such deviations. The minimisation cost of the optimal control problem $P(t, x)$, known as the *value function* $W(t, x)$, must be Lipschitz continuous on bounded subsets of its domain; must possess at least linear growth (due to the dominance of the cost integrand over system dynamics); and must be monotonic along solutions of the closed loop system description.

The work of (Kershenbaum *et al.*, 1993) recognises that this theory is highly idealised. In reality, a precise system description and measurements of all required state variables are seldom available. Also, the computational time is often too great for online implementation. Current research in the area of allowable model mismatch, discrete controller implementation and approximate minimisation is promising (see, for example, Michalska and Mayne (1990)). (Kershenbaum *et al.*, 1993) suggest the need for future work to provide theoretical guidelines relating the choice of cost integrand to closed loop system response and to suggest system transformations to improve robustness of the dynamic model. The latter issue will be discussed in more detail later in this chapter.

The issue of constructing an implementable, stabilising controller for a wide class of nonlinear systems is discussed by (Michalska and Mayne, 1993). They attempt to establish conditions under which receding horizon control stabilises nonlinear, continuous time systems. As exact minimisation and exact satisfaction of the terminal constraint places considerable demands on the on-line optimising controller, Michalska and Mayne relax this constraint to a neighbourhood, say W, of the origin. However, when the terminal constraint is relaxed to $x(T) \in W$, the receding horizon strategy is not stabilising within W. Therefore, within W, a continuous-time, linear feedback controller is used. In the complement of W, a discrete-time feedback controller is employed.

Mayne and Michalska prove the resulting controller to be globally stabilising. The discontinuity at the boundary of W causes no difficulty since the receding horizon strategy steers the state to the boundary of W and the local linear controller takes over from there and steers the state to the origin. The paper gets around the problem of determining feasible solutions to the sequence of different finite horizon optimal control problems (created by the receding horizon algorithm) by *not* requiring a sequence of optimal solutions. Thus, their paper assumes existence of at least one feasible, but not necessarily optimal, solution to the problem. All that is required "is a solution no worse than that of an admissible control-horizon pair obtained from the previous solution" (Michalska and Mayne, 1993). Therefore, once a feasible solution is found for one finite horizon problem (such as the initial $i = 0$ problem), feasible solutions to all subsequent finite horizon problems are easily obtained. In the ideal case of no model error, Mayne and Michalska claim that even "...if subsequent solutions to the finite horizon optimal control problem are *not* improved, the trajectory followed by the (real) system is identical to that obtained from the first

($i = 0$) solution...". This is a direct result of the "dual-mode" controller definition used to deal with the free end time problem.

(Vinter and Michalska, 1993) extend the application of receding horizon control to time-varying, continuous-time, nonlinear systems with state constraints. This paper establishes global stabilisation properties of receding horizon control under suitable conditions. The proposed strategy can be used to obtain feedback controllers for which the corresponding closed-loop trajectories satisfy prespecified state constraints. This introduces an improvement over geometric approaches which are prone to controller saturation. The paper also addresses the issue of robustness by analysing the level of modelling error under which stabilisation is preserved, an issue of interest in this thesis. Due to the complexity of feedback stabilisation of nonlinear time-varying systems, the conditions for stabilisation are necessarily restrictive. The proof by Vinter and Michalska employs nonsmooth Lyapunov functions, avoiding the need to ensure continuous differentiability of the Lyapunov functions considered. The proof requires the assumption that, for each $(t, x) \in [0, \infty) \times \mathbb{R}^n$ an optimal control $\{u^*(s : t, x) | t \leq s \leq t+T\}$ can be found for $P(t, x)$, where $P(t, x)$ is the optimal control problem. Proofs of existence of optimal controls on a finite horizon exist (Fleming and Rishel (1982), Vinter (1987), among others). The mapping $s \mapsto u^*(s; t, x)$ must be continuous from the right at $s = t$, and the mapping $(t, x) \mapsto u^*(t; t, x)$ is continuous on $[0, \infty) \times \mathbb{R}^n$. These hypotheses are necessary to ensure a proper definition and continuity of the receding horizon feedback law. Additional hypotheses are imposed to ensure the existence and uniqueness of solutions to the system equation on any closed interval for arbitrary initial conditions and control function; to ensure that the integrand in the cost function is integrable; to ensure that the value function is locally Lipschitz continuous (the value function is employed as a "Lyapunov function" in the stability analysis of the receding horizon closed-loop system); and to ensure the optimal value function is proper. Two assumptions are required to classify admissible model mismatch. The model-system error must be bounded by

$$\| f(t, x, u) - f^R(t, x, u) \| \leq \gamma(t, x) m(t, x, u) \tag{2.3}$$

for all $t \in \mathbb{R}, x \in \mathbb{R}^n, u \in \mathbb{R}^m$, where $\gamma : [0, \infty) \times \mathbb{R}^n \to [0, \infty)$ is some positive function, f^R is the real system and m is the function integrated in the cost expression. $\mathcal{H}_\infty$ techniques will be outlined later as a viable method of defining the model error bound. The function f^R must be continuous and $f^R(t, \cdot, \cdot)$ must be locally bounded, uniformly with respect to

$t \in [0, \infty)$. When constraints are placed on the state

$$(t, x(t)) \in \mathcal{G} \text{ for all t} \in [0. \infty) \tag{2.4}$$

where

$$\mathcal{G}_i \stackrel{def}{=} \{(t, x) \in \mathbb{R} \times \mathbb{R}^n | \; g_i(t, x) \leq 0\}, \; i \in \underline{n}. \tag{2.5}$$

$$\mathcal{G} \stackrel{def}{=} \bigcap_{i \in \underline{n}} \mathcal{G}_i, \tag{2.6}$$

and $\underline{n} \stackrel{def}{=} \{1, ..., n\}$, Vinter and Michalska employ the additional assumption that the functions $g_i, i \in \underline{n}$, are continuous and locally Lipschitz continuous with respect to x. The origin is an element of the constraint set $\mathcal{G}$. The problem is now a constrained open loop optimal control problem and thus the previous hypotheses must be restated, replacing the unconstrained optimal control problem with the constrained one.

(Rawlings *et al.*, 1994) provide a survey of model predictive control. Their tutorial paper focuses on model predictive control of both constrained linear plants and nonlinear plants. They find that certain questions, such as nominal stability, are not significantly more difficult to address when the plant is nonlinear. It is the implementation of the controller, with the required on-line solution of a nonlinear program, which is significantly more difficult. Their linear formulation is nominally stabilising in the presence of constraints provided inconsistent state constraints are relaxed.

2.1.7 Choice of Horizon

Ideally, one would wish to compute the control trajectory for the indefinite operation of the plant. This would result in an infinite horizon formulation and, hence, an infinite control computation for a nonlinear plant. As previously discussed, infinite horizon formulations have nice stability properties. However, this formulation is neither desirable nor necessary. As the control trajectory is only implemented over a fraction of the prediction horizon, it suffices to choose the horizon long enough such that a change in the horizon length has a negligible effect on the portion of the control which is implemented. Beyond the horizon, we choose the controls as constant at their final value at t_f. This method was studied and successfully applied by (Disli, 1996) to a CO_2 absorption pilot plant. Alternatively,

the horizon length can be posed as an optimisation parameter, such that the optimal control algorithm chooses the optimal horizon over which the objective can be achieved. Of course, this raises issues of stability and continued feasible operation and therefore, the optimal endtime formulation must be posed so as to ensure both stability and feasible operation are maintained.

2.1.8 Conclusions

Much work has been done on stability of finite horizon formulations (Balakrishnan *et al.* (1993), Clarke (1993)); somewhat less on infinite horizon problems (Balakrishnan *et al.* (1993), Meadows and Rawlings (1993)); and some on the application of both formulations to nonlinear, continuous time models (Kershenbaum *et al.* (1993), Michalska and Mayne (1990), Michalska and Mayne (1993), Vinter and Michalska (1993)). The majority of this work has employed discrete time models. However, with the introduction of path inequality constraints the discrete time approach ensures satisfaction of the constraints only at the discretisation points. While computerisation necessitates discretisation, continuous time methods necessarily employ a finer discretisation of the horizon and, hence, satisfaction of the constraints at a greater number of points. This point will be discussed in more detail later in this literature review.

Most of the papers discussing stability begin by assuming the existence of controls. The conditions for existence must be explicitly stated and satisfied before stability can be addressed. If a control exists which achieves the objective, the plant can then be said to be stabilisable. The next section will discuss the general formulation of optimal control problems and give necessary conditions for existence of controls for finite horizon formulations.

2.2 Optimal Control

The key aspect of receding horizon control is the computation of optimal controls based on predictions of the process behaviour. We will now review general optimal control theory for ordinary differential equation (ODE) systems defined over some optimisation horizon. The necessary extensions of these ideas to differential algebraic equation (DAE) systems will be developed later in the thesis.

2.2.1 Statement of Optimal Control Problem

We consider the following ordinary differential system in $\mathbb{R}^n$:

$$\dot{x}(t) = f(t, x(t), u(t)) \tag{2.7}$$

where $x = (x_1, ..., x_n)$ is the vector of states, $u = (u_1, ..., u_m)$ is the vector of controls and $f(t, x, u) = (f_1(t, x, u), ..., f_n(t, x, u))$ is the vector of state equations.

We assume that f is continuous on $\mathbb{R}^{1+n+m}$ and has continuous derivatives with respect to x. The closed set of all admissible controls is defined as

$$u(t) \in \Omega \subset \mathbb{R}^m, \quad t \in [t_0,\ t_f] \tag{2.8}$$

where the control functions $u(\cdot)$ are chosen to be either piecewise constant, piecewise linear or polynomial.

For an arbitrary admissible control $u(t)$ the differential equation may be written as

$$\dot{x} = f(t, x, u(t)) = F(t, x) \tag{2.9}$$

Any absolutely continuous function

$$x(t), \quad t_0 \leq t \leq t_f \tag{2.10}$$

is said to be a solution on the interval $t_0 \leq t \leq t_f$ if it satisfies equation (2.7) for almost all $t \in [t_0, t_f]$ with $x(t_0) = x_0$. By absolute continuity, we may write

$$x(t) = x(t_0) + \int_{t_0}^{t} F(\theta, x(\theta))\, d\theta. \tag{2.11}$$

We define a *performance criterion* for the system,

$$J(x_0, u(\cdot)) = \Psi(t_0, t_f, x(t_0), x(t_f)). \tag{2.12}$$

The optimal control problem is to find, from the class X of all feasible pairs $(x_0, u(\cdot))$, a pair$(x_0, u^*(\cdot))$ which minimises the performance criterion (2.12). This minimising control is called the *optimal control* for a given initial condition.

2.2.2 Equivalent Optimal Control Problem Formulations

Equivalent optimal control problem formulations exist. The *Mayer Problem* takes Ω to be an open subset of $\mathbb{R}^m$ rather than a closed subset, as in the above formulation.

Alternatively, if the performance criterion is defined as

$$J(x_0, u(\cdot)) = \int_{t_0}^{t_f} L(t, x(t), u(t))\ dt, \tag{2.13}$$

the optimisation problem is called a *Lagrange Problem*. Finally, for a performance criterion

$$J(x_0, u(\cdot)) = P^f(x(t_f)) + \int_{t_0}^{t_f} L(t, x(t). u(t))\ dt \tag{2.14}$$

the problem is called a *Bolza Problem*. Note that each of these formulations is generally equivalent. For example, a Lagrange Problem can be formulated as a Mayer Problem by adding another component to the differential equation

$$\dot{x}_{n+1} = L(t, x(t), u(t)) \tag{2.15}$$

with initial condition $x_{n+1}(t_0) = 0$.

We qualify this equivalence using "generally" because the conditions placed on the integrand are different than those on the state dynamics for existence of optimal controls. In particular, the state dynamics are required to be Lipschitz continuous, whereas the integrand is not. We shall discuss these conditions for existence next. All three of these optimisation problems, as well as the optimal control formulation for both the fixed terminal state/free end time and fixed terminal state (or set of states)/fixed end time, are outlined in (Fleming and Rishel, 1982), among others.

2.2.3 Conditions for Existence of Optimal Controls on a Finite Horizon

Before looking for candidate solutions to an optimal control problem, it is logical to examine the problem formulation to determine the conditions under which solutions exist. The basic conditions for the finite horizon continuous time problem are outlined below (Vinter, 1987). The infinite horizon formulation will be analysed in detail in the coming chapter.

For a cost function

$$\min_{u(t)} J(u(t)) = \int_{t_0}^{t_f} L(t, x(t), u(t))\ dt + P(x(t_f)) \tag{2.16}$$

where

$$\begin{gathered} \dot{x}(t) = f(t, x(t), u(t)) \ \ a.e.\ \ t \in [t_0, t_f] \\ x(t_0) = x_0 \in \mathcal{X}_0, \ \ x(t_f) \in \mathcal{X} \\ u(t) \in \Omega, \ \ t \in [t_0, t_f] \end{gathered} \tag{2.17}$$

where $\mathcal{X}_0$ is the set of admissible initial conditions and $\mathcal{X}$ is the set of admissible end conditions. We define the extended velocity function:

$$\tilde{f} = \begin{pmatrix} L(t,x,u) \\ f(t,x,u) \end{pmatrix}, \forall\ (t,x,u) \in [t_0,t_f] \times \mathbb{R}^n \times \Omega \tag{2.18}$$

Suppose that

1. $\tilde{f}(t,x,u)$ is measurable in t and continuous in (x,u)

2. P is continuous

3. Ω is compact

4. $\exists$ a constant k $s.t.$

$$|f(t,x,u) - f(t,y,u)| \leq k|x-y| \ \forall\ x,y \in \mathbb{R}^n,\ u \in \Omega,\ t \in [t_0,t_f]$$

5. $\mathcal{X}_0$ is compact and $\mathcal{X}$ is closed

6. $\exists$ an admissible process $(\bar{x},\bar{u})$

7. $\tilde{f}(t,x,\cdot)$ is convex for all $(t,x) \in [t_0,t_f] \times \mathbb{R}^n$

Then there exists an optimal control. The proof corresponding to these conditions can be found in (Vinter, 1987). A similar proof can be found in (Fleming and Rishel, 1982). Note that, with the exception of condition (7), the above hypotheses are of a not very restrictive nature.

2.2.4 Necessary Conditions for Optimality

Necessary conditions for optimality were first derived by Pontryagin, Boltyanski and Gamkrelidze. Common terminology calls these necessary conditions "Pontryagin's Principle" (Fleming and Rishel, 1982). (Hartl *et al.*, 1995) give an excellent modern survey of the various forms of the necessary and sufficient optimality conditions for deterministic continuous time optimal control problems with state constraints. This paper states various formulations of the maximum principle which may be used as a recipe for determining optimal controls for state constrained problems, but the proposed necessary conditions are not rigorously established.

In order to illustrate the basic development of the necessary conditions. we consider the system (2.16) - (2.17). Assume $u \in \Omega$ corresponds to $g(t,u) \geq 0$. We define the Hamiltonian as

$$H = L(t,x,u) + p^T(t) f(t,x(t),u(t)) + z^T g(t,u(t)) \tag{2.19}$$

where

$$z \begin{cases} \leq 0, & g = 0 \\ = 0, & g > 0. \end{cases} \tag{2.20}$$

The standard necessary conditions for the above problem are as follows:

$$\begin{aligned} \dot{x} &= f(t,x(t),u(t)) \\ \dot{p}^T &= -H_x = -L_x - p^T f_x \end{aligned} \tag{2.21}$$

and the boundary conditions are split as:

$$x(t_0) \text{ given}, \quad p^T(t_f) = P_x(t_f). \tag{2.22}$$

The control u is determined by maximising H with respect to u. Alternatively, we may determine u by the weak maximum principle

$$H_u = 0 \quad \text{or} \quad L_u + p^T f_u + z^T g_u = 0. \tag{2.23}$$

These conditions define a two-point boundary value problem which is a recipe for the optimal solution.

2.2.5 Conclusions

Much work has been done on existence of optimal controls for systems defined over a finite horizon. Of special interest is the free endtime problem, where we must deal with a finite but unknown time horizon. The natural case would be to consider the infinite horizon formulation. A proof of existence of optimal controls for problems of this form will be developed in the next chapter.

The formulation above considers only ODE systems, but this can be easily generalised by applying a reduction technique to translate the DAE system to its ODE equivalent. We shall review reduction techniques in Section 2.5. The primary limitation of the necessary conditions is the absence of state constraints in the problem formulation. While the

linear unconstrained problem has been well-studied in both discrete and continuous time, as has the discrete time linear control constrained problem, state constrained plants are a subject of current research in both linear and nonlinear control, from both the algorithmic and theoretical perspective. Various techniques for dealing with state constraints will be discussed in Section 2.6. We shall first turn to discussion of the various algorithmic methods of dealing with optimal control problems.

2.3 Computational Methods

Most of the computational methods available for solving optimal control problems are based on solving necessary conditions rather than sufficient conditions via a Hamilton-Jacobi approach. Numerically, we must make decisions on how to deal with the problem of determining controls which lie in an infinite dimensional subspace. We outline possible approaches below to the general problem and broaden our discussion to include path inequalities and endpoint equalities in later sections.

2.3.1 Control Parameterisation

(Vassiliadis *et al.*, 1994a) recently reviewed the various approaches to solving optimal control problems and proposed a method based on control parameterisation which can deal with the general problem (2.9)-(2.12) given above. This method converts the infinite-dimensional optimal control problem into a finite-dimensional problem which can be formulated as an NLP. Controls are parameterised as piecewise constant, piecewise linear or piecewise polynomial with or without enforcing continuity at the junctions. The approach operates in two stages. In the first stage, given a control profile, the state and sensitivity equations are integrated using a DAE integrator. The integrations yield function and gradient values which are then passed to an NLP solver which updates the control parameters in the second stage of the algorithm. The control profiles are evaluated using Lagrange polynomials. In (Vassiliadis, 1993a), the time horizon is divided into a number of stages (NS), with each stage subdivided into a number of finite elements (π). The controls are defined as low-order polynomials over each element. For control variable u_j over stage i, we approximate

$$u_j^i(t) = \sum_{k=1}^{M_j} u_{ijk}\phi_k^{f(M_j)}(\tau^i), \quad t \in [t_{i-1},\ t_i] \tag{2.24}$$

of order M_j (degree $< M_j$), where

$$\begin{aligned} \phi_k^{f(M_j)}(\tau) &= 1 \text{ if } M_j = 1 \\ \phi_k^{f(M_j)}(\tau) &= \prod_{\kappa=1,\kappa\neq k}^{M_j} \frac{(\tau - \tau_\kappa)}{(\tau_k - \tau_\kappa)} \text{ if } M_j \geq 2 \end{aligned} \tag{2.25}$$

and τ^i is normalised over time stage i

$$\tau^i = \frac{t - t_{i-1}}{t_i - t_{i-1}} \tag{2.26}$$

where $i = 1, 2, \ldots, NS$; $j = 1, 2, \ldots, \pi$ and $k = 1, 2, \ldots, M_j$. The algorithm continues iteratively updating and applying the controls until an optimal trajectory is determined. The multistage approach is used to allow for discontinuity in the state. That is, it allows for transition from one set of dynamic equations to another as we pass from one stage to the next. This is useful when considering, for example, a batch process in which a reaction takes place from t_0 to t_1, followed by instantaneous mixing with a new feed and distillation of the mixture from t_1 to t_2. Thus, we have two distinct stages, reaction and distillation, with each described by a distinct set of equations.

The existence of state inequalities poses a problem in such an approach, as they may not be directly enforced. They may be dealt with using a penalty function approach (see, for example Vassiliadis (1993b)) or by introducing a barrier function, as will be outlined in this thesis. A modified barrier function may also by employed (see, for example, Vassiliadis (1996) or Polyak (1992)), but the shifting in the logarithmic barrier function allows for violation of the constraint bounds at intermediate solution points, which may be unacceptable due to safety considerations, etc.

The main disadvantage of the control parameterisation method is its tendency to be slow, especially in dealing with path inequalities which must be satisfied at each $t \in [t_0, t_f]$. However, work on parallelisation of the computation of the sensitivities by (Keeping and Pantelides, 1995) has shown the potential for dramatic reductions in computational time. Furthermore, work on parallelising the evaluation of equation residuals and jacobians has been studied by (Paloschi, 1995), among others. Parallelisation algorithms, in combination with ever-increasing computing power, are contributing to the possibility of on-line application of sophisticated optimisation algorithms.

2.3.2 Collocation Methods

The collocation approach (see, for example, Cuthrell and Biegler (1987), Vasantharajan and Biegler (1990), or Renfro *et al.* (1987)) is generally quite similar to the parameterisation approach outlined above. This method parameterises both the state and control, thus transforming the differential equations into a finite set of algebraic equations. The resulting problem is solved as an NLP. The DAE system is integrated implicitly by fitting piecewise polynomials to the system dynamics, forcing equality between the approximation and the system at specified collocation points. Therefore, the coefficients α_j of the polynomial approximation are chosen to satisfy an m^{th} order system of differential equations at M collocation points,

$$z(t) = \sum_{j=1}^{M+m} \alpha_j \Gamma_j(t) \tag{2.27}$$

where Γ_j is an independent polynomial basis function (Tanartkit and Biegler, 1995).

The advantage of this approach is its direct treatment of constraints, eliminating the possibility of infeasible solutions at the collocation points at each iteration. However, for large systems the NLP problem can be prohibitively large. Consider, for example, a distillation column model having 200 variables. For such a system, approximation of the variables with piecewise cubic polynomials over 20 finite intervals would yield 8000 algebraic equations. Control parameterisation does not have the same explosion in system size since the integration and optimisation problems are decoupled. Furthermore, collocation suffers from difficulty dealing with high-index systems. Discretisation destroys the implicit relations obtained by differentiating the original system so "solutions" which pass through the collocation points may not obey these and hence may be nothing like the real solution. Of course, the finer the discretisation, the closer one constrains the curves but the number of points to be safe is much larger than for index one systems. As in all methods involving high-index systems, the free variables in the system must be determined at initialisation. If the index changes over the time horizon, the problem must be re-initialised. For the case of stiff DAE systems, the resulting NLP program can be badly scaled and extremely ill-conditioned. Traditional collocation schemes require a small number of collocation points as high-degree polynomials tend to oscillate. However, collocation on finite elements avoids this problem by employing a two-tiered partitioning scheme.

(Tanartkit and Biegler, 1995) address the high dimensionality problem by decomposing the problem using multiple shooting. Shooting methods are among the most commonly used in solving systems of ordinary differential equations and it is natural to extend these methods to DAE systems. We discuss both single and multiple shooting methods in the next section, as well as briefly addressing the quasilinearisation approach.

2.3.3 Direct Solution of Necessary Conditions

(Pontryagin *et al.*, 1962) employed the calculus of variations to develop the now famous Pontryagin's Maximum Principle. Adjoint based methods, which also make use of the calculus of variations, were among the first used in solving optimal control problems. This variational approach transforms the optimisation problem into a two-point boundary value problem (TPBVP), where the boundaries are defined by the initial and final times of the control horizon. Various algorithmic applications of this approach are outlined in (Bryson and Ho, 1975), with the emphasis on unconstrained problems. In fact, as mentioned in (Tanartkit and Biegler, 1995), traditionally this approach has worked very well for unconstrained systems but exhibits problems when constraints are introduced.

There are basically two approaches: single shooting and multiple shooting. The two approaches are illustrated in Figure 2.3.

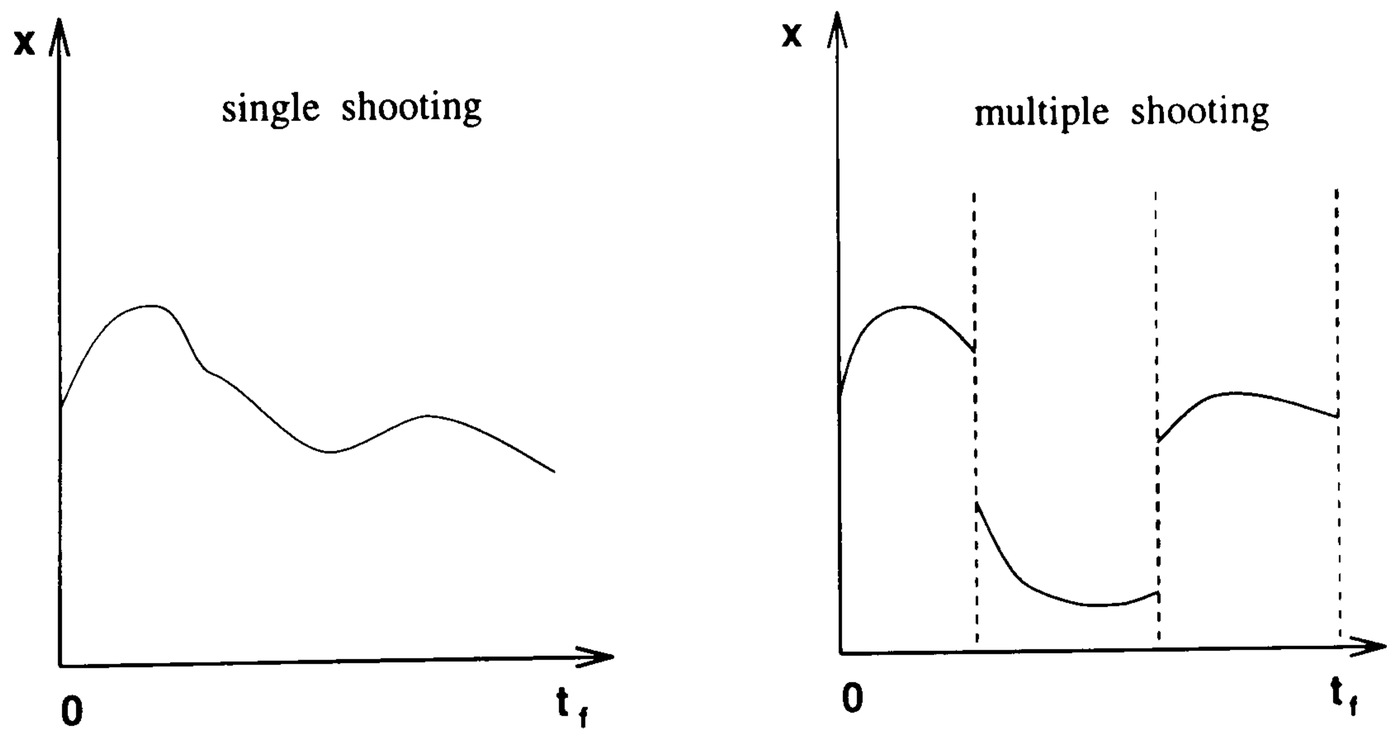


Figure 2.3: Single shooting versus Multiple shooting

2.3.3.1 Single Shooting

In the single shooting approach, the solution to the optimal control problem is sought by sequentially solving the problem in time, starting from guessed initial conditions. Thus, either an initial state trajectory, or an initial control profile, is chosen and a series of Newton iterations are used to update the guess so that the given boundary conditions of the problem (initial conditions and/or final equality constraints) are satisfied. The system equations are explicitly integrated over the full optimisation horizon at each iteration.

2.3.3.2 Multiple Shooting

In multiple shooting, the optimisation horizon is subdivided into smaller intervals. Two approaches are then available. The first determines a consistent set of initial conditions at each interval and integrates the system equations over the smaller intervals (see, for example, Stoer and Bulirsch (1993)). The second approach, and that used by (Tanartkit and Biegler, 1995), is termed simultaneous collocation and implements a local collocation scheme to decompose the system over each subinterval. Finally, a Newton iteration is used to enforce continuity of the state dynamics at interval boundaries.

2.3.3.3 Quasilinearisation

The quasilinearisation approach (see, for example, Miele (1973)) may be adopted in both a single shooting and a multiple shooting approach. This method simply solves a succession of *linear* two-point boundary value problems (TPBVPs) and updates the state and adjoint dynamics at each iteration so as to reduce the error between the approximate linear dynamics and the actual nonlinear system. As successive iterations are made, the residual should decrease.

2.3.3.4 Comparison of Single versus Multiple Shooting

The advantage of single shooting over multiple shooting is the considerably smaller amount of storage space required per iteration. Furthermore, the Jacobian matrices to be inverted are smaller. However, single shooting does not adequately deal with unstable state dynamics in the forward direction. Inherent in applying this method is the assumption that the forward dynamics are stable. Multiple shooting methods integrate over shortened horizons and therefore the cumulative errors are much smaller compared with

those of single shooting. This can have a dramatic effect if the problem is ill-conditioned. However, multiple shooting only reduces the integration horizon and therefore cannot necessarily deal with all unstable systems. The choice of which method to use depends on both the size of the system (storage considerations) and the stability of the system.

2.3.4 Conclusions

Given all of the above considerations, we see the advantages and disadvantages of the various approaches. The advantage of collocation is its direct treatment of constraints, eliminating the possibility of infeasible solutions at the collocation points at each iteration. However, for large systems the NLP problem can be prohibitively large. For control parameterisation, there are two basic approaches. The sensitivity approach involves one forward vector integration per parameter. The adjoint approach involves one backward vector integration for the objective function and one for each endpoint constraint (equality or inequality). In most classical (i.e. aerospace) problems, a fine control discretisation is used and there are few endpoint constraints, heavily favouring the use of adjoints. The alternative approach of solving the equations for the necessary conditions involves only one adjoint system if there are no endpoint equalities but, for shooting methods, many more iterations (i.e. integrations) are required than for control parameterisation. For quasilinearisation, one needs n integrations and, in addition, two backward adjoint and two forward integrations per endpoint equality. This can be more or less than for control parameterisation using adjoints.

In general, integration based schemes are able to handle stiff DAE systems. Furthermore, for constrained systems, the necessarily finer mesh of integration schemes as opposed to the coarser mesh for collocation schemes guarantees satisfaction of the constraints at a greater number of points. We propose a new algorithm, based on the classical idea (see, for example, Bryson and Ho (1975)) of satisfying the necessary conditions of optimality. Of course, the classical necessary conditions must be extended to deal with a DAE system and path inequality constraints, and the theoretical development is given later in the thesis.

State equality constraints at the horizon endpoint present a challenging problem in determining the optimal trajectory via direct solution of the necessary conditions. This

is a result of an underdetermined system at t_f. An explicit analysis which illustrates this point will be provided in Chapter 5. We turn now to discussion of the various methods of determining the appropriate value for the adjoint variable at t_f such that the system is fully determined.

2.4 Dealing with State Endpoint Equality Constraints

In this section we discuss how the various algorithms outlined in Section 2.3 deal with the presence of endpoint equality constraints on the states.

2.4.1 Control Parameterisation

Endpoint equality constraints are naturally handled within the control parameterisation framework, since the evaluation of their value as well as their gradients takes place at a well-defined point. Thus, the constraint is enforced in solving the NLP.

2.4.2 Collocation

Collocation methods easily incorporate endpoint equality constraints. In this approach, the constraints are discretised and enforced at the discretisation points and thus a constraint at t_f will be enforced by solving an NLP.

2.4.3 Direct Solution of Necessary Conditions

Direct solution of the necessary conditions requires more careful thought when dealing with endpoint equality constraints. We consider the system description (2.16)-(2.17), with the associated necessary conditions (2.21)-(2.22).

2.4.3.1 Bryson and Ho (1975) Method

Consider now an endpoint equality constraint in one or more of the states:

$$\phi^f(x(t_f)) = 0. \tag{2.28}$$

In this case, the necessary condition $H_u = 0$ needs additional justification. This is due to the fact that the choice of $u(t)$ is now restricted by the requirement that $\phi^f(x(t_f)) = 0$. We now replace (2.22) with a new set of boundary conditions:

$$x(t_0) \quad \text{given}, \quad p^T(t_f) = P_x(t_f) + \xi^T \phi^f_x(t_f). \tag{2.29}$$

The presence of an additional vector ξ results in an underdetermined system at t_f. The Bryson and Ho (1975) approach to this problem is as follows:

Step 1: Estimate a control trajectory $u(t)$.

Step 2: Integrate the system equations $\dot{x}(t) = f(t, x(t), u(t))$ forward from $x(t_0)$ to generate a state trajectory. Evaluate $\phi^f(x(t_f))$.

Step 3: Integrate the adjoint variable p in two parts, the first (2.30) predicting the changes in the cost due to changes in the control trajectory, the other (2.31) measuring the influence of the control on the endpoint equality. Thus, we have

$$\dot{\tilde{p}} = f_x^T \tilde{p}, \quad \tilde{p}(t_f) = P_x^T(t_f) \tag{2.30}$$

$$\dot{\bar{p}} = f_x^T \bar{p}, \quad \bar{p}(t_f) = \phi_x^{fT}(t_f) \tag{2.31}$$

such that

$$p = \tilde{p} + \bar{p}\xi. \tag{2.32}$$

Step 4: Simultaneously compute the following integrals:

$$\begin{aligned} I_{\phi\phi} &= \int_{t_0}^{t_f} \bar{p}^T f_u W^{-1} f_u^T \bar{p} \; dt \\ I_{\phi P} &= \int_{t_0}^{t_f} \bar{p}^T f_u W^{-1} f_u^T \tilde{p} \; dt, \;\; W > 0. \end{aligned} \tag{2.33}$$

Step 5: Choose $\delta\phi^f$ to initiate an improvement in the next solution towards $\phi^f(x(t_f)) = 0$. For example, choose $\delta\phi^f = -\epsilon\phi^f(x(t_f)), \;\; 0 < \epsilon \leq 1$. Then

$$\xi = -I_{\phi\phi}^{-1}(\delta\phi^f + I_{\phi P}). \tag{2.34}$$

Step 6: Repeat Steps 1 through 4, updating $u(t)$ by

$$\delta u(t) = -W^{-1} f_u^T (\tilde{p}^T + \xi^T \bar{p}^T). \tag{2.35}$$

Thus, we have split the adjoint to measure the effect of δu on δP and $\delta\phi^f$:

$$\delta P = \int_{t_0}^{t_f} \tilde{p}^T f_u \delta u \; dt \tag{2.36}$$

$$\delta\phi^f = \int_{t_0}^{t_f} \bar{p}^T f_u \delta u \; dt. \tag{2.37}$$

We see that since (2.36) and (2.37) are linearised relations, there is no minimum for δP subject to $\delta\phi^f$. This minimum is artificially induced by adding a quadratic penalty function in δu:

$$\delta P^1 = \delta P + \frac{1}{2}\int_{t_0}^{t_f} \delta u^T W \delta u \, dt, \quad W > 0. \tag{2.38}$$

If we adjoin (2.37) to (2.38), equation (2.35) easily follows by minimising over δu.

We shall develop a similar approach in Chapter 5 for more general problems and illustrate its effectiveness using a case study in Chapter 6.

2.4.3.2 Miele's (1973) Gradient-Based Method

(Miele, 1973) proposes a combined gradient-restoration algorithm for dealing with endpoint equalities of the form (2.28), where t_f is either fixed or free. While Miele considers only endpoint equalities of the above form (and ODE systems without inequality constraints), his approach is easily extended to more general systems.

Combined Gradient-Restoration Algorithm

The combined gradient-restoration algorithm is a first-order iterative technique to achieve a reduction in the cost functional P as well as ϕ^f by solving a linear, two-point boundary value problem (TPBVP) at each iteration. To do so, the descent properties $\Delta P < 0$ and $\Delta\phi^f < 0$ must be enforced simultaneously. Thus, we generate variations which simultaneously lower the augmented cost functional as well as the endpoint constraint error.

The technique proposed by Miele is as follows. Assume we have n_e equality constraints ϕ^f. Integrate the system equations (2.21), (2.23) and (2.29) (in linearised form) $n_e + 1$ times using a backward-forward integration scheme, assigning different values to the multiplier ξ^f at each iteration. For example, we may choose multiplier ξ_p^f normal to multipliers ξ_i^f, $i = 1, \ldots, p-1$:

$$\xi_i^f = \begin{bmatrix} \delta_{i1} \\ \delta_{i2} \\ \vdots \\ \delta_{in_e} \end{bmatrix}, \quad i = 1, 2, \ldots, n_e + 1 \tag{2.39}$$

where the Kronecker delta δ_{ij} is such that

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases} \tag{2.40}$$

For each ξ_i^f we generate a trajectory of $\Delta u_i(t)$, $\Delta x_i(t)$, $\Delta \dot{x}_i(t)$ and an associated ϕ_i^f. After completion of $n_e + 1$ integration loops, we search for an appropriate linear combination:

$$\begin{gathered} \Delta u_k(t) = \sum_{i=1}^{n_e+1} k_i \Delta u_i(t), \\ \Delta x_k(t) = \sum_{i=1}^{n_e+1} k_i \Delta x_i(t), \quad \Delta \dot{x}_k(t) = \sum_{i=1}^{n_e+1} k_i \Delta \delta \dot{x}_i(t) \end{gathered} \tag{2.41}$$

such that

$$\phi_0^f + \sum_{i=1}^{n_e+1} k_i (\phi_{\dot{x}_0}^f \Delta \dot{x}_i(t) + f \phi_{x_0}^f \Delta x_i(t) + \phi_{u_0}^f \Delta u_i(t) = 0, \quad \sum_{i=1}^{n_e+1} k_i = 1. \tag{2.42}$$

This in turn generates a value for the multiplier $\xi^f = \sum_{i=1}^{n_e+1} k_i \xi_i^f$.

We form the following family of solutions

$$\begin{aligned} x_{k+1}(t) &= x_k(t) + \beta \Delta x_k(t) \\ u_{k+1}(t) &= u_k(t) + \beta \Delta u_k(t) \\ x_{k+1}(t_0) &= x_k(t_0) + \beta \Delta x_k(t_0), \quad \text{etc.} \end{aligned} \tag{2.43}$$

for some $\beta \in [0, 1)$. We now choose β such that

$$\begin{aligned} |\phi_{k+1}^f| &< |\phi_k^f| \quad \text{when} \quad |\phi_k^f| \geq \phi^* \\ |\phi_{k+1}^f| &< \phi^* \quad \text{when} \quad |\phi_k^f| < \phi^* \end{aligned} \tag{2.44}$$

and

$$\begin{aligned} |P_{k+1}^f| &< |P_k^f| \quad \text{when} \quad |P_k^f| \geq P^* \\ |P_{k+1}^f| &< P^* \quad \text{when} \quad |P_k^f| < P^* \end{aligned} \tag{2.45}$$

for some positive threshold ϕ^* and P^*.

Having computed new values (2.41) which satisfy the linear two-point boundary value problem, as well as an optimum step-size β, we may now generate updated values of the state, control, initial state vector and the endpoint equality ϕ_{k+1}^f with associated multiplier ξ^f. Miele adopts two possible approaches:

1. The varied state $x_{k+1}(t)$, the varied control $u_{k+1}(t)$ and the varied initial state $x_{k+1}(t_0)$ are obtained from (2.43).

2. The varied control $u_{k+1}(t)$ and the varied initial state $x_{k+1}(t_0)$ are obtained from (2.43) and the varied state is obtained by forward integration of the system dynamics $\dot{x} = f(t, x, u)$ subject to the initial conditions, the optimisation parameters and the control trajectory.

One may adopt the second approach and integrate the nonlinear equations directly for the updated control trajectory to generate an exact value for ϕ^f, given the current control trajectory. However, as the dimension of the system equations increases, the first approach is a better computational alternative.

2.4.4 Conclusions

In this section we have outlined the various approaches for dealing with equality constraints in the states at the endpoint. We have shown that such constraints fall very naturally within the control parameterisation and collocation approaches, but some consideration is needed when posing the optimal control problem as a direct solution of the necessary conditions.

The latter two approaches outlined are similar in that both use the adjoint-based approach to generate an update for the multiplier ξ^f. However, the Bryson and Ho approach uses steepest descent to generate an update, whereas Miele uses steepest descent followed by conjugate directions. Effectively, Bryson and Ho integrate $I_{\phi\phi}, I_{\phi P}$, while Miele integrates a partitioned state system, $\delta\tilde{x}, \delta\bar{x}$. Miele's approach requires $n_e + 1$ inner integrations to generate an update for ξ^f for the outer loop, where n_e is the dimension of ϕ^f. This amounts to the integration of $2n_d \times (n_e + 1)$ adjoint and state variables, where n_d is the number of differential states. In contrast, the modified Bryson and Ho approach requires the integration of the vector $\dot{\bar{p}}$, which is of dimension $n_d \times 1$, the matrix $\dot{\bar{p}}$ which is of dimension $n_d \times n_e$ and the matrices $I_{\phi\phi}, I_{\phi P}$ of dimension $n_e \times n_e$ and $n_e \times 1$ respectively. The total number of integrations is then $(n_d + n_e) \times (n_e + 1)$. This can amount to more or less integrations on each major loop, depending on the relative sizes of n_d and n_e. For systems in which the number of states is much larger than the number of endpoint constraints, one would expect the first approach to be faster. Of course, which

method is faster will depend on the size of the system, the number of endpoint constraints and the degree of accuracy of the initial guess.

Both of the latter two approaches are developed in the absence of constraints. While both can easily be extended to deal with control constraints, pure state constraints will pose some difficulties, as the control generated in either update formulation may produce an infeasible state trajectory when implemented. We shall discuss this problem in Chapter 5 and develop alternative solutions.

We will now show that an important advantage of using the classical approach is its direct handling of DAE systems of any index.

2.5 Dealing with High-index DAE Systems

In general, optimal control problem descriptions have been considered in terms of ODE systems. In process systems, algebraic relationships often exist between the differential variables. Furthermore, path inequality constraints introduce further algebraic relationships if they become active along the horizon . The result is a DAE system which may or may not be equivalent to an ODE system. The fact that DAE systems may behave quite differently from ODE systems was recognised only recently. Research into DAE systems over the last ten to twenty years has resulted in a broad literature covering such diverse applications as electric networks, singular perturbation theory and constrained optimal control. (Brenan *et al.*, 1996) present the developments in numerical algorithms for initial value problems in DAEs.

We begin by defining the *index* of a DAE system, in an intuitive sense, as the number of differentiations required to reduce the DAE system to an ODE system. Formally (Brenan *et al.*, 1996):

Definition 2.5.1. *The index of a system*

$$f(t, \dot{x}, x, u, v) = 0 \tag{2.46}$$

is defined as the smallest integer I such that the system:

$$\begin{aligned} f(t, \dot{x}, x, u, v) &= 0 \\ \frac{d^i}{dt^i} f(t, \dot{x}, x, u, v) &= 0, \quad i = 1, 2, ..., I \end{aligned} \tag{2.47}$$

defines $\dot{x}$ *as a locally unique function of* (t, x, u, v).

In general, most of the work done on the above-mentioned activities has assumed a system of index 2 or less. However, many chemical process systems are naturally of higher index and, due to nonlinearities, of varying index over the horizon of interest. Recent work has considered various methods of dealing with higher index problems. The majority of these methods reduce the index through a series of differentiations related to the determination of the index. There are difficulties associated with this approach: the ODE system generated by the reduction has a larger set of solutions than the original DAE; the algebraic relations (constraints) in the DAE become embedded or implicit in the new ODE system and, therefore, the numerical solution may drift away from the algebraic constraint; and the resulting systems may be very large.

We outline below several approaches to index reduction which have appeared in the literature. The final proposed approach is in the spirit of (Jarvis, 1994). Comparisons between our approach and Jarvis' are drawn later in the thesis.

2.5.1 The Gear (1988) Algorithm

(Gear, 1988) adopted the approach of successive differentiations, with appropriate substitutions, to generate an index 0 system. The algebraic equations are differentiated I times, where I is the index of the system, substituting for the differential states using the original differential system. The algebraic equations of the original system plus those developed during the reduction process are reintroduced to yield the following overdetermined system:

$$\begin{aligned} \dot{x} &= f(t, x, y) \\ 0 &= g(t, x, y) \end{aligned} \tag{2.48}$$

Gear removes this overdeterminacy by introducing a vector of variables μ, where the dimension of μ corresponds to the over determinacy

$$\begin{aligned} \dot{x} &= f(t, x, y) + g_x^T \mu \\ 0 &= g(t, x, y). \end{aligned} \tag{2.49}$$

That is, μ is of the same dimension as g. The end result is an index 2 problem since with one differentiation we determine $\mu = 0$ and with a second differentiation we determine $\dot{\mu}$. The disadvantage of this approach, as in others, is that the resulting system size is much greater than that of the original system.

2.5.2 The Campbell (1988) Algorithm

(Campbell, 1988) adopts a method of successive differentiations in which all of the differentiated equations are retained. The result is an underdetermined system in which some of the high-order derivatives are not uniquely determined. Campbell shows that, provided the number of differentiations applied to the system is greater than or equal to the index of the system, the resulting solution is unique in the first order derivatives.

2.5.3 The Bachmann *et al.* (1990) Algorithm

The (Bachmann *et al.*, 1990) algorithm reduces the index by differentiating the algebraic equations and substituting from the original system for any derivatives which arise in the differentiation procedure. Bachmann *et al.* choose not to discard the algebraic equations that have been differentiated, but rather to discard some of the *differential* equations used for the elimination step.

2.5.4 The Chung and Westerberg (1990) Algorithm

The approach taken by (Chung and Westerberg, 1990) is similar to that of (Campbell, 1988) in that all of the equations in higher derivatives are retained, with some of the new variables assigned arbitrary values so that the system may be solved while ensuring the solution of the original system is unique. Chung and Westerberg numerically determine the subset of equations which must be differentiated to reduce the system to index 1. They then symbolically differentiate these equations. The method developed in the paper may be used for both consistent initialisation and stable propagation of DAE systems. However, their method is described in terms of a linear system, and the generalisation to nonlinear systems is not given.

2.5.5 The Mattsson and Söderlind (1993) Algorithm

In the control parameterisation approach, (Feehery and Barton, 1996) consider using the dummy derivative method of (Mattsson and Söderlind, 1993) which permits the derivation of an equivalent index 1 system with the same solution set as the high-index DAE. Parts of the DAE system are analytically differentiated and appended to the original system. The resulting overdetermined system is dealt with by introducing a new algebraic variable for a subset of the derivatives. These algebraic variables are termed *dummy derivatives* and

BIBL. LONDIN. UNIV.

by treating them as such the problem of inconsistency due to discretisation is eliminated. The original problem and that resulting from the reduction procedure are mathematically equivalent as they have identical solution sets in the original (undifferentiated) dependent variables. However, this results in a larger system of equations to be solved by the NLP solver.

2.5.6 The Pantelides *et al.* (1994) Algorithm

(Pantelides *et al.*, 1994) adopt a similar approach to that of (Mattsson and Söderlind, 1993). Using automatic differentiation, (Pantelides *et al.*, 1994) generate the necessary additional equations to reduce the system index. The original system equations appear as a subset of the equations in the index-reduced system and therefore the solution obtained by using existing numerical algorithms to integrate the reduced system satisfies all of the original equations within the integration tolerance.

2.5.7 The Jarvis (1994) Algorithm

(Jarvis, 1994) presents a differentiation-free approach to solving high-index problems. We base our method on the ideas outlined in his thesis. Jarvis' method is based on partitioning the variables and equations in the high index system into two subsets, one of index 0 or 1 which can be solved, given the variables from the second subset. The result is an iterative approach in which the variables which "generate" the high index problem are "guessed", the state trajectory is generated using standard techniques and the algebraic constraints are checked for satisfaction. If these constraints are not satisfied, the procedure is repeated with a new guess. Thus, the system is partitioned into one which may be dealt with using standard integration techniques and another which must be iterated upon. This iteration is easily incorporated into an optimisation iteration.

2.5.8 Conclusions

The key problem in reducing the index of a DAE system is ensuring that the solution of the reduced system is consistent with that of the original system. This means that all methods must avoid the problem of drifting of the algebraic constraints as a result of numerical errors. The key problem is that discretisation destroys the implicit relations obtained by differentiations, so "solutions" which pass through discrete points may not

obey these and hence will be nothing like the real solution. Of course, the finer the discretisation, the closer one constrains the curves, but the number of points to be safe is much larger than for index one systems. Furthermore, no error analysis has been developed which can determine the necessary discretisation.

(Jarvis, 1994) outlined an interesting approach which avoids the standard approach of differentiating the system I times, where I is the index. In this approach, the system is partitioned into an index one subsystem and a subsystem of algebraic constraints. By minimising the integral of the error in the additional algebraic equations and assuming exact solution of the index one DAE system, this method first produces a theoretically equivalent problem. The errors then arise from the discretisation of the solution of the DAE and evaluation of the integral, for which a proper error analysis does exist. This approach will be further discussed and extended later in the thesis.

2.6 Constraint Handling

Traditional closed loop approaches to control result in computationally infeasible problems. Furthermore, the addition of path constraints invalidates these methods. In order to illustrate this, we begin by considering the simple linear, unconstrained problem

$$J = \min_u \int_{t_0}^{t_f} (x^T Q x + u^T R u)\ dt + \frac{1}{2}\, x(t_f)^T S x(t_f) \tag{2.50}$$

subject to

$$\dot{x} = Ax + Bu \tag{2.51}$$

where $Q,\ S \geq 0$ and $R > 0$. We assume for simplicity of exposition that the state is measured. It is easy to see that the cost is dependent on the initial state x_0, and the control trajectory $u(t)$. This linear problem can be solved using a feedback approach, where the feedback law is generated by sequentially solving a Riccati equation of the form

$$-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q, \qquad P(t_f) = S \tag{2.52}$$

and

$$u = -R^{-1}B^T P x \tag{2.53}$$

as the horizon shifts from sampling points. In the receding horizon approach, we only implement the initial control and therefore we may write the optimal control law as

$$u^* = -R^{-1}B^T Px \mid_{t_0} \tag{2.54}$$

where t_0 shifts as the horizon is moved forward.

Note that the addition of the weighted state at t_f to the integral cost in (2.50) is necessary for stability. As mentioned in previous sections, alternative descriptions which provide stability are the infinite horizon formulation and the endpoint equality constrained problem. For a thorough consideration of the use of the endpoint weighting, see (Bitmead *et al.*, 1990), and for the use of endpoint equality constraints, see for example (Kwon and Pearson, 1977).

If we consider nonlinear systems, the closed-loop solution to the unconstrained system requires solution of the Hamilton-Jacobi-Bellman partial differential equation (PDE), a computationally time-consuming equation to solve. Another drawback to the closed-loop approach is its inability to handle constraints. The introduction of constraints invalidates both the Riccati approach and the Hamilton-Jacobi-Bellman solution.

The receding horizon approach, which solves an open-loop problem at every sampling interval, is a practical and computationally effective alternative to formulating a closed loop solution in time and state space. Furthermore, of great importance is the flexible constraint handling capabilities of MBPC schemes. The majority of applications impose hard constraints on the manipulated variables, in addition to the state constraints required to achieve performance objectives (Rawlings *et al.*, 1994). However, some thought must be given when considering inequality constraints. Such constraints pose an interesting problem, as they may be active in certain regions and inactive in others. To date, the most popular method has been the penalty function approach and this and other approaches will be outlined in this section.

2.6.1 Collocation

One method of handling path constraints is that of collocation (see, for example, Cuthrell and Biegler (1987)). In this approach, the system is discretised and constraints are enforced at the discretisation points. However, discretisation cannot guarantee satisfaction of the constraint over the entire time horizon. This is a recurring problem with numerical algorithms.

2.6.2 Conversion to Point Constraint

Several approaches in the literature reformulate path constraints as point constraints by penalising violation of the constraint at any time t along the horizon.

2.6.2.1 Integration of Violation

For $g(t,x) \geq 0$, we may define a new state using a measure, r, of the constraint violation at each t:

$$\begin{aligned} r &= \min[g, 0] \\ \dot{x}_{n+1} &= r^T r, \qquad x_{n+1}(t_0) = 0 \end{aligned} \tag{2.55}$$

and impose the endpoint constraint $x_{n+1}(t_f) = 0$. This method was proposed by (McGill, 1965), (Sage and White, 1977) and (Sullivan, 1977). Alternatively, we may impose each constraint individually

$$\dot{x}_{n+i} = r_i^p, \qquad x_{n+i}(t_0) = 0 \tag{2.56}$$

$$x_{n+i}(t_f) = 0 \tag{2.57}$$

for $i = 1, 2, \ldots, n_i$, where n_i is the number of inequality constraints and the exponent p is either 1 or 2.

The advantage of (2.56)-(2.57) over (2.55) is that more information is supplied to the optimiser by specifying each constraint separately. The major disadvantage of this approach is the discontinuity in r as the constraint switches from active to inactive and vice versa. The application of this idea assumes smoothness of the system, which is not guaranteed. Thus the method is not mathematically rigorous, as it violates the necessary conditions which require smoothness of the system. While the method has been found to work in practice, this is more a result of propagation of numerical errors which effectively allow an ϵ buffer at the constraint boundary. In fact this $\epsilon-$relaxation can be introduced explicitly at the endpoint:

$$\mid x_{n+1}(t_f) \mid \leq \epsilon \tag{2.58}$$

for some small ϵ. This idea was introduced by (Walsh, 1993) and shows favourable numerical results.

2.6.2.2 Vassiliadis' (1993) Hybrid Approach

(Vassiliadis, 1993a) proposes constraints of the form

$$\rho(g(t_i)) \leq \epsilon_i, \qquad i = 0, 1, \ldots, NS \tag{2.59}$$

where NS = number of elements or stages along the time horizon. The constraints are then enforced as interior point constraints at each t_i. The choice of function $\rho(g)$ is divided into 5 possible schemes:

Scheme 1: Bound on Cumulative Error per Element

If $x^*(t)$ denotes a cumulative integral violation defined as in (2.55) or (2.56)-(2.57) , write

$$x^*(t_i) - x^*(t_{i-1}) \leq \epsilon_i, \qquad i = 1, \ldots, NS \tag{2.60}$$

where ϵ_i is chosen as $\epsilon_i = \epsilon h_i/(t_f - t_0)$, where $h_i = t_i - t_{i-1}$ is the length of the element.

Scheme 2: Escalated Tolerance for Path Constraint Value

Alternatively, we may add the interior point constraints defined in (2.60) to derive

$$x^*(t_i) \leq \sum_{k=1}^{i} \epsilon_k \qquad i = 1, 2, \ldots, NS \tag{2.61}$$

Therefore, the accumulated error at each element must be less than some progressively increasing tolerance. Thus, a large error in one element is compensated for by a small error in another.

Scheme 3: Fixed Tolerance for Path Constraint Value

This employs the same formulation as in Scheme 2, but uses a fixed tolerance ϵ_k instead of $\sum_k \epsilon_k$. This scheme is essentially the same as the endpoint equality approach, but provides some information as to when (i.e. in which element) the constraint was violated.

Scheme 4: Direct Discretisation of the Path Constraint

This approach examines the exact value of the constraint at each discretisation point $t_i,\ i = 0, 1, \ldots, NS$. The idea is to incorporate both discretisation points as well as endpoint constraints resulting from the path constraint violation norm to convey as much information to the optimiser as possible.

Scheme 5: Bounding the Integral of the Constraint

This approach imposes

$$\int_{t_{i-1}}^{t_i} g(t)\, dt \geq 0, \qquad i = 1, 2, \ldots, NS \tag{2.62}$$

as well as the endpoint constraint violation term (2.58), thereby bounding the value of the integral of the constraint at interior intervals as well as at the endpoint.

All of the above schemes place emphasis on local violation, global violation or both but, as pointed out by (Vassiliadis, 1993b), they are necessary but not sufficient conditions for the original path constraints to be satisfied throughout the horizon. The essential basis of Vassiliadis' approach is conversion to an endpoint constraint by integration of violations, but this is supplemented by additional constraints to provide extra information as to when the violation occurs.

2.6.2.3 Gritsis' (1990) Bound on Minimum Value of Constraint

(Gritsis, 1990) considers the more general path constraint of the form

$$g(t, x, x', u', v) \geq 0 \tag{2.63}$$

where x' represents algebraic states and u' represents controls. A new function Ψ is defined to be the minimum value of g over the horizon $[t_0, t_f]$,

$$\Psi(x, x', u', v) = \min_{t \in [t_0, t_f]} g(t, x, x', u', v). \tag{2.64}$$

If this minimum is achieved for several $t \in [t_0, t_f]$, the first such t is chosen

$$t_m = \min\{t|\ t \in T_m\} \tag{2.65}$$

where

$$T_m = \{t|\ g(t, x, x', u', v) = \Psi(x, x', u', v)\}. \tag{2.66}$$

This transforms the path constraint to a point constraint at t_m

$$\Psi(x(t_m), x'(t_m), u'(t_m), v) \geq 0 \tag{2.67}$$

Thus if (2.67) is satisfied, so then is (2.63) for every $t \in [t_0, t_f]$. The advantage of this method is that it transforms a path constraint to a point constraint. However, the method does not explicitly account for changes in the time t_m as a result of changes in u and v. As the minimum switches from one t_m to another, a discontinuity arises in the gradient. Therefore, this method also invalidates the necessary conditions due to nonsmoothness in the gradient.

2.6.3 Conversion to Equality Constraint

We now turn to methods which transform inequality constraints to equality constraints.

2.6.3.1 Valentine's (1937) Slack Variable Approach

The slack variable approach known as Valentine's method (Valentine, 1937) introduces a slack variable, s, for inequalities of the form

$$g(t, x) \geq 0\,. \tag{2.68}$$

This allows the inequality constraint (2.68) to be transformed to an equality constraint

$$g(t, x) - \frac{1}{2}s^2(t) = 0 \tag{2.69}$$

The squared term forces g to be greater than or equal to zero. This equality constraint is then appended to the DAE system. As a result of the additional equation, a control becomes a state variable. This results in a DAE system of higher index than the original system, as we have an algebraic constraint on a state. The equation is differentiated repeatedly until an expression is obtained in some control u_j with respect to the system variables x, $u_i (i \neq j)$, v, $s(t)$ and its k derivatives, where k is the number of necessary differentiations. The problem is reformulated to eliminate u_j, set $s^{(k)}(t)$ as a new control variable and $s^{(1)}$, ..., $s^{(k-1)}(t)$ as differential states. Therefore, path constraints are appended to the DAE system as equality constraints and the slack variable becomes a control which ensures satisfaction of the constraints at every iteration.

The difficulty with this approach is that s is not uniquely defined. If s satisfies (2.69), so then does $-s$. Furthermore, at the constraint boundary $s = 0$ the gradient is independent of s and convergence is therefore slow as we approach the boundary. Finally, the repeated differentiations result in a higher dimensional system and, consequently, a greater computational burden.

2.6.3.2 Implicit Constraints

The approach outlined by (Speyer and Bryson, 1968) and others (see Hartl *et al.* (1995) for an excellent review) is based on the necessary conditions for solutions to optimal control problems with a bounded state space. This work requires that when the system enters onto a constrained arc, the arc must have zero values of the state variable constraint

function and all of its time derivatives that do *not* involve the control variable (say $p-1$ of them). Control of the constraint function is then realised by changing the p^{th} time derivative. Thus, for a constraint (2.68), they define the p vector:

$$s(t,x) \triangleq \begin{bmatrix} g(t,x) \\ g^{(1)}(t,x) \\ \vdots \\ g^{(p-1)}(t,x) \end{bmatrix} \tag{2.70}$$

and require $s(t,x) = 0$ as well as

$$c(t,x,u) \triangleq g^p(t,x,u) = 0 \tag{2.71}$$

on a constrained arc. The implicit relation (2.71) defines u as a function of x and t. In modern terms, augmenting the constraint equations to the system equations yields a high-index system which must be differentiated to yield an appropriate function to define the relationship between the current state and the control on a constrained arc. Thus, the dimensionality of the system is increased by the successive differentiations of (2.70). Currently, there is no other approach which successfully deals with pure state constraints when using the necessary conditions as an algorithmic basis. We will discuss this approach in more detail in the next chapter.

Note that these last two approaches generate high-index systems.

2.6.4 Conversion to Unconstrained Problem

An alternative approach reformulates the constrained problem as an unconstrained one by adding either a penalty term or a barrier term to the objective function.

2.6.4.1 Penalty Methods

Consider an inequality constraint of the form (2.63). We may measure the constraint violation at each t as

$$r_i = \min[g_i, 0] \, . \tag{2.72}$$

The constraint violation is then appended to the objective function as

$$\bar{J} = J + K \int_{t_0}^{t_f} r^T r \, dt \tag{2.73}$$

for the case of an integral objective function and the solution is obtained as $K \to \infty$. The drawback of this method is that it frequently results in numerical difficulties due to the non-smoothness of (2.72).

A formulation specific to state constraints, as proposed by (Sage and White, 1977) and (McGill, 1965), is as follows. Consider the constraint equations as in (2.63), extracting the pure state constraints from g and writing these as

$$\underline{h} \leq h(t,x) \leq \overline{h}. \tag{2.74}$$

We may associate a new state with the system which measures the deviation of this constraint from its boundaries as:

$$\dot{\breve{x}} = (h - \underline{h})^2 \underline{H} + (\overline{h} - h)^2 \overline{H}, \qquad \breve{x}(t_0) = 0 \tag{2.75}$$

where

$$\underline{H}(h) = \begin{cases} 0, & \underline{h} < h \\ K_s, & h \leq \underline{h} \end{cases} \tag{2.76}$$

$$\overline{H}(h) = \begin{cases} 0, & h < \overline{h} \\ K_s, & h \geq \overline{h}. \end{cases} \tag{2.77}$$

Therefore, we now have a standard optimal control problem with control and mixed inequality constraints dealt with in the usual way and pure state constraints incorporated as a penalty function, initialising $\breve{x}(t_0) = 0$ and adding $\breve{x}(t_f)$ to the cost function.

2.6.4.2 Barrier Functions and the Interior Point Method

The interior point (IP) method converts inequality constraints of the form (2.63) to equality constraints using a barrier function. One possible method of converting a problem to an IP form is to introduce a slack variable, s, such that

$$g(t,x,x',u',v) - s(t) = 0, \qquad s > 0. \tag{2.78}$$

Unlike Valentine's method, this approach avoids the problem of duplicity in s, as s has a unique value and there are no gradient problems at $s = 0$. The interior point method enforces this equality to some tolerance ϵ. The algebraic equation (2.78) is appended to the DAE system, the slack variable $s(t)$ becomes a state and the augmented system

is fully determined. The slack variable must be initialised at some interior point and is maintained interior by design of the iterations. Its use in solving discrete time linear quadratic regular problems with mixed state and control constraints was discussed by (Wright, 1993). The interior point method is employed in this thesis to handle both mixed and pure control inequality constraints. It will be outlined in more detail in a later chapter.

2.6.5 Conclusions

A number of methods have been proposed and tested in the literature for dealing with inequality constraints. As discretisation is necessary for algorithmic application, it is inevitable that all methods are necessary but not sufficient conditions for the original path constraints to be satisfied for all t. The majority of approaches impose the inequality constraints as some form of penalty to the objective. However, penalty function approaches are prone to numerical difficulties. In this thesis, we shall adopt the interior point method and investigate its behaviour when dealing with inequality constraints of the mixed state/control and pure control form. Pure state constraints present greater difficulties and the special handling of these constraints using the tools outlined above will be discussed in Chapter 5.

2.7 Control in the Presence of Process Uncertainty and Plant/Model Mismatch

2.7.1 Defining Robustness in a Process Systems Context

One of the issues which has always been at the forefront of controller design is the sensitivity of the controlled system to variations in the process parameters. Inevitably, a controller must deal with

- *process upsets* due to disturbances
- *changes in operating conditions* due to changes in product demand or objectives
- *mismatch* between the model dynamics and those of the actual plant, in the case of model-based controllers.

There is always a tradeoff between control system performance and robust stability in control system design. A controller which adequately deals with such variations is termed *robust*. A property, such as stability, is considered *robust* if its existence at one point implies its existence in an open neighbourhood of that point. The key is defining this neighbourhood.

Traditional PID controllers address this delicate balance by determining the controller gain which causes the process to go unstable and then cutting back on this gain by some factor to ensure a reasonable margin of safety. However, the need for more sophisticated controller schemes translates to the need for more sophisticated notions of robustness. Many of the methods of advanced process control involve the use of a detailed process model. Therefore, we transfer our interest from the input/output response of PID schemes to a more sophisticated state space description.

The primary case of interest in this work is robustness to model uncertainty. That is, robustness is measured in terms of the controller's ability to preserve a desired closed-loop response in spite of the imperfection of the plant model. Robustness is therefore defined in terms of an allowable level of model uncertainty. In order for any control algorithm to work satisfactorily in a real environment, the following must be specified (Morari and Doyle, 1986):

- *process model*
- *model uncertainty bounds*
- *types of inputs* (i.e. setpoints and disturbances)
- *performance objectives.*

The goal of this section is to outline a method of optimising both the performance and robustness of a control system. Traditional optimisation methods, like Linear Quadratic Gaussian (LQG) Control, only optimise performance. The discussion which follows outlines $\mathcal{H}_\infty-$ Optimal Control as a viable method of addressing the difficulties associated with modelling errors in unconstrained systems. There is a large literature of techniques for designing robust controllers for linear unconstrained systems. We shall focus our attention on single-input-single-output systems (SISO), as they readily illustrate the important concepts in robust analysis.

2.7.2 Controlling Deviations: 1 and 2−norms versus ∞−norm

Traditionally, minimisation of 1 and 2 norms of the sensitivity function of a SISO system has been used as a means of controlling deviations. (Genceli and Nikolaou. 1993) study the robust stability of DMC controllers for linear plants with 1−norm objective functions and uncertainty in the step response coefficients of the plant model. By robust stability, we refer to a controller's ability to maintain asymptotic stability with zero offset for a plant in the presence of bounded disturbances. (Zheng and Morari, 1993) study robustness using a min-max optimisation of a 1−norm objective function. The disadvantage is that $1-2$ norms keep average deviations small while allowing large peak deviations. The $\infty-$ norm avoids peak excursions from the desired trajectory and results in a linear program which can be solved more efficiently by the simplex method than the 1−norm linear program (Garcia *et al.*, 1989).

2.7.3 $\mathcal{H}_\infty$−Optimisation and Worst Case Design

Kwakernaak's tutorial paper (1993) presents a good introduction to robust control and $\mathcal{H}_\infty-$ optimisation, where $\mathcal{H}_\infty$ is the space of functions on the complex plane that are analytic and bounded in the right-half plane. The problem was originally considered by (Zames, 1981), who minimised the ∞−norm of the sensitivity function of a single-input-single-output linear feedback system. $\mathcal{H}_\infty-$ optimisation of control systems is the minimisation of the peak value of certain closed-loop frequency response functions.

Consider a basic SISO feedback system (Figure 2.4). The plant has transfer function P and the compensator has transfer function C. The sensitivity function is

$$S = \frac{1}{1+PC} \tag{2.79}$$

Zames considered finding a compensator C that makes the closed-loop system stable and

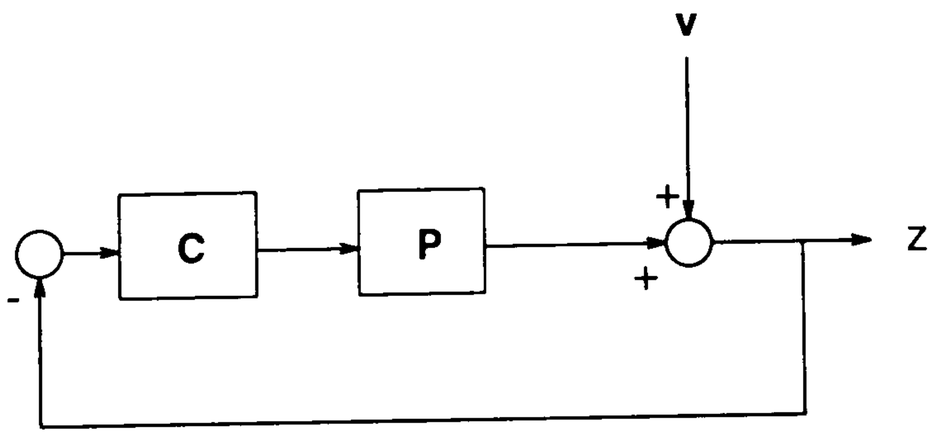


Figure 2.4: SISO Feedback Loop

minimises the peak value of the sensitivity function. For some functions, the peak value may not be assumed for any finite frequency and we therefore use the supremum, or least upper bound,

$$\| S \|_\infty = \sup_{\omega \in \mathbb{R}} |S(j\omega)| \tag{2.80}$$

as the peak value. The use of the $\infty-$norm is justified as follows: if the peak value $\| S \|_\infty$ of the sensitivity function S is small, then the magnitude of S is necessarily small for all frequencies. Therefore, disturbances are uniformly attenuated over all frequencies by using a worst-case optimisation which minimises the effect on the output of the worst disturbance (a harmonic disturbance at the frequency where $|S|$ has its peak value).

In engineering terms, the norm is directly related to the system energy gain for the input with the worst possible frequency distribution. $\mathcal{H}_\infty-$optimisation is concerned with the minimisation of system norms. The worst-case design suggests a differential game theory approach. The controller must be designed using a compensator C that offers the best protection against the worst possible disturbance.

2.7.4 Combining Receding Horizon Control with $\mathcal{H}_\infty-$Optimisation

A suitable goal is to combine receding horizon control with $\mathcal{H}_\infty-$optimisation techniques in a harmonious way. Papers by (Lall and Glover, 1993) and (Tadmor, 1992) explore this idea for linear time varying systems. Lall and Glover extend the work of (Tadmor, 1992), who formulated a receding horizon controller, with each finite horizon optimisation based upon an $\mathcal{H}_\infty-$optimisation. Tadmor's work proved such a controller was stable and satisfied an infinite horizon norm bound.

The controller derived in (Lall and Glover, 1993) is essentially the same as Tadmor's in the state feedback case and differs only in the terminal constraint used. The difficulty arises in the observation feedback case, as it is difficult to naturally formulate the $\mathcal{H}_\infty-$ receding horizon control problem for this situation. For the finite horizon problem, it is usually assumed that the initial state is either completely known or completely unknown at the beginning of each optimisation interval. If it is unknown, it is treated as a disturbance. However, in the receding horizon control problem, we have observations from *before* the optimisation interval. This introduces difficulties in the $\mathcal{H}_\infty-$ controller formulation.

For a given cost function

$$J_\gamma(u,w) = \int_{t_0}^{t_f} \{z(t)^T z(t) - \gamma^2 w(t)^T w(t)\} dt \tag{2.81}$$

with z the error signal and w representing all external inputs including disturbances, sensor noise and commands, Lall and Glover formulate a finite time differential game

$$\inf_{\mu \in M} \sup_{\nu \in N} J_\gamma(\mu, \nu) \tag{2.82}$$

which is a *zero sum* game with μ the *control strategy* for u and ν representing the *external input "strategy"* of w. For this game, u is the minimising player and w is the maximising player. The operator $\mathcal{T}_{zw}$ maps w to z. Thus, the finite interval $\mathcal{H}_\infty$ problem is to find a linear causal control $u(t) = \mu(t, x(t))$ to accommodate the worst case w such that $\| \mathcal{T}_{zw} \|_\infty < \gamma$ and

$$J_\gamma(u,w) \leq -\epsilon \| w \|_2^2 \ \ \forall w \in \mathcal{L}_2[t_0, t_f] \tag{2.83}$$

for some $\epsilon > 0$. If the extremising operators in equation (2.82) are interchangeable, the optimal u and worst case w are called *saddle point strategies*. A saddle point solution $u(t) = \mu^*(t, x(t)), w(t) = \nu^*(t, x(t))$ will satisfy

$$J_\gamma(\mu^*, w) \leq J_\gamma(\mu^*, \nu^*) \leq J_\gamma(u, \nu^*) \ \forall u, w \in \mathcal{L}_2[t_0, tf] \tag{2.84}$$

Pictorially, this is represented by Figures 2.5 and 2.6.

In order to gain both the robustness benefits of the $\mathcal{H}_\infty-$ formulation and the practicality of receding horizon control, Lall and Glover (1993) find $u(t) = \mu^*(t, x(t))$ at time t, where μ^* is the saddle point strategy for player u for

$$\inf_{\mu \in \mathcal{M}_t} \sup_{\nu \in \mathcal{N}_t} \left\{ \int_t^{t+T} \{z(s)^T z(s) - \gamma^2 w(s)^T w(s)\} ds + x(t+T)^T F(t+T) x(t+T) \right\} \tag{2.85}$$

with $\mathcal{M}_t = \{\mu : [t, t+T] \times \mathbb{R}^n \rightarrow \mathbb{R}^{m1}\}$, $\mathcal{N}_t = \{\nu : [t, t+T] \times \mathbb{R}^n \rightarrow \mathbb{R}^{m2}\}$, and $w(t) = \nu^*(t, x(t))$. Therefore, at each time t, a finite horizon $\mathcal{H}_\infty$ problem is solved on the interval $[t, t+T]$ to determine the feedback strategy $u(t) = \mu(t, x(t))$.

In order to find the feedback saddle point strategy for u, an optimal estimate of the state is needed. For the finite horizon time varying $\mathcal{H}_\infty$ problem one can look, at each instant of time t, for the worst possible disturbance $w_{[t_0, t_f]}$ given information about $u_{[t_0, t]}$ and $y_{[t_0, t]}$ (Basar and Bernhard (1991) or Green and Limebeer (1995)). This, in turn,

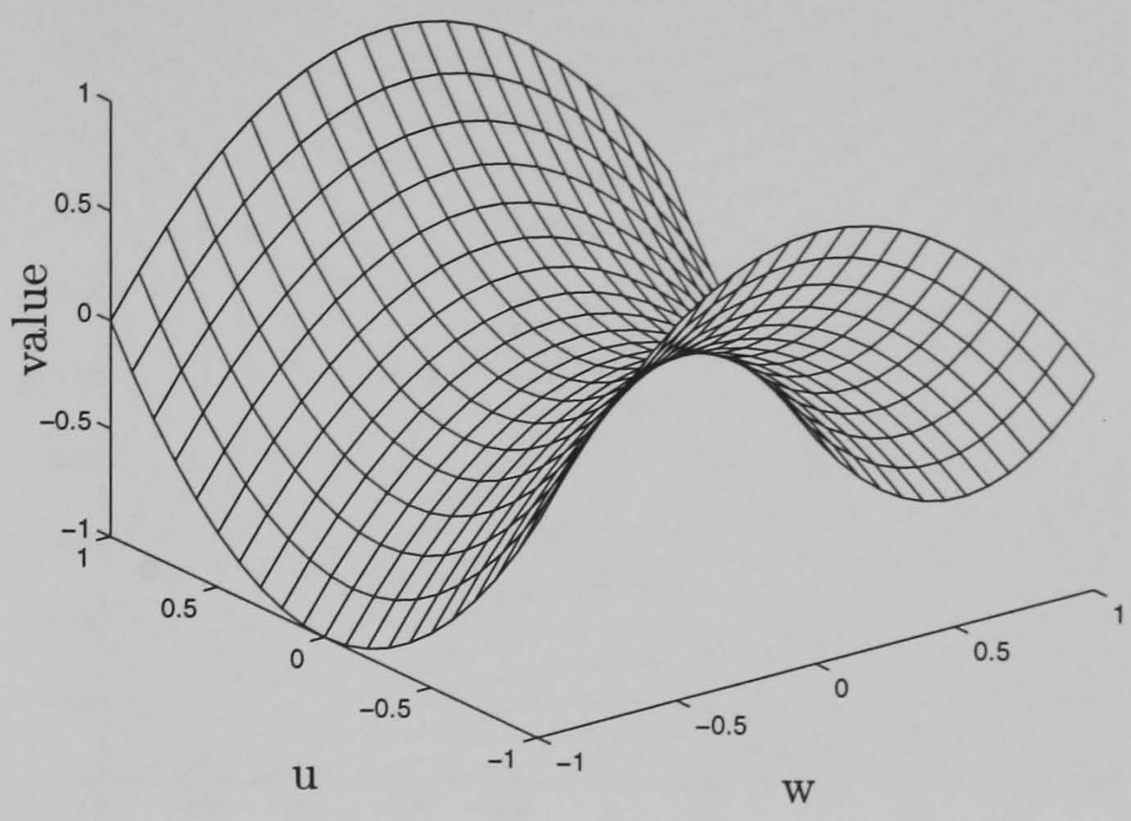


Figure 2.5: Saddle Point

generates the worst possible state trajectory which is used by the feedback law as if it were the actual state. Lall and Glover formulate an alternative which assumes w is trying to maximise $J(0, t_f)$, while u is trying to minimise $J(t_0, t_f)$. This is equivalent, in terms of u, to a moving horizon game in which only the endpoint moves.

To achieve global robustness, the integrand should be convex in $u(t)$ and concave in $w(t)$ so that the solution of (2.84) is a saddle-point. However, if $w(t)$ belongs to a compact set, the saddle-point property is unnecessary.

Obviously, the actual performance is guaranteed to be at least as good as the predicted performance, and the advantage of defining the game in a receding horizon framework is the fact that we are not searching for a feedback law. We take an initial measurement and integrate over time only. This eliminates the difficulties associated with searching for a control over time and state space (which may be very large).

2.7.5 Dealing with Constrained Uncertain Systems

Unfortunately this worst-case approach does not guarantee robustness if inequality constraints are present, for in principle we need to ensure that these are satisfied for all trajectories $x(\cdot)$ generated by the computed $u(\cdot)$, for all possible realisations of $w(\cdot)$. Figure 2.7 illustrates the inability of saddle point solutions to cope with this type of problem. If the uncertain value $w(\cdot)$ deviates from the worst case w^* to $\bar{w}$, we find that the control

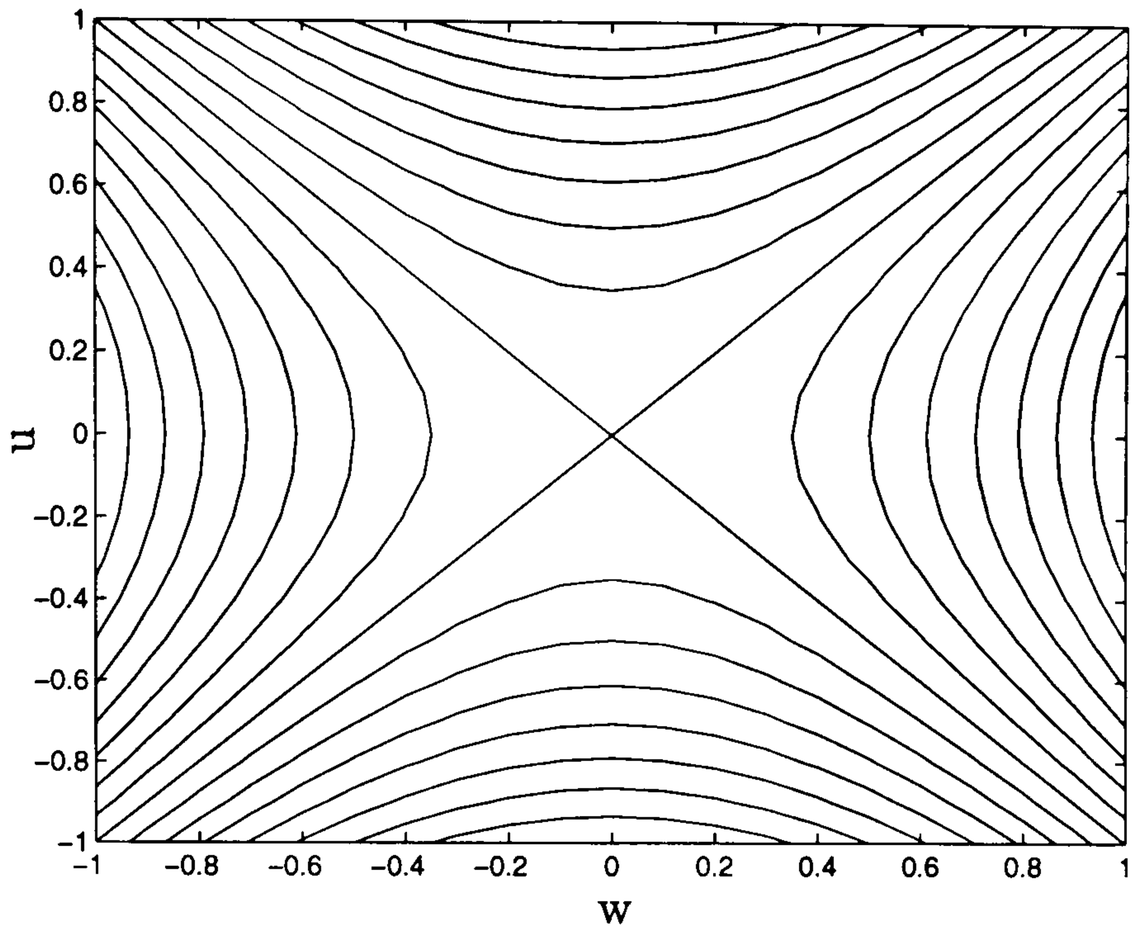


Figure 2.6: Contours of Saddle Point

u^* places us in the infeasible region. While we could convert inequalities using a barrier function and then apply the worst-case approach to the resulting equality constrained problem, it is better to optimise performance for the *most-likely* case (or nominal case), subject to being *feasible* for all cases.

This thesis will propose an alternative method for dealing with uncertainty in constrained systems of a very general nature.

2.7.6 Conclusions

The above outline of $\mathcal{H}_\infty$ theory addresses robustness to disturbances and model mismatch. A combination of finite horizon receding horizon control and $\mathcal{H}_\infty$ robustness techniques was described which has been shown to be capable of dealing with unconstrained uncertain systems. This thesis will contribute to the robustness theory by providing a method for dealing with constrained uncertain systems.

2.8 Concluding Remarks

The field of optimal control has long been an active one, but the discovery of DAEs has opened a broad new region of research. While the behaviour of index zero and one systems is comparable to that of ODE systems, higher index systems are inherently different and

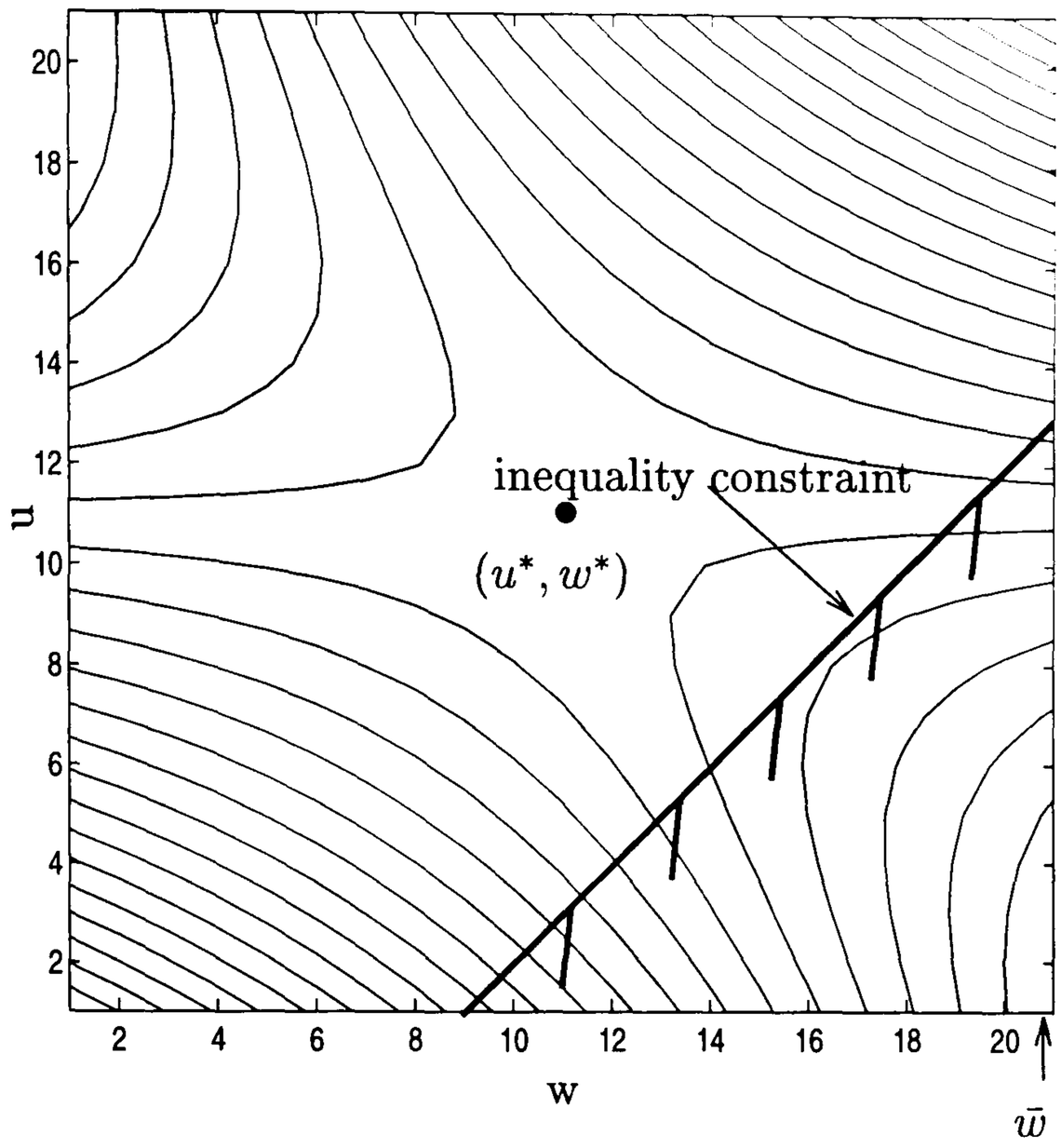


Figure 2.7: Infeasibility of Saddle Point Solution

consideration must be taken of these differences when designing a general purpose optimal control algorithm. Furthermore, the need for robustness in the determination of optimal controls subject to process constraints has been recognised. This is a direct result of the deficiency of both the model and current computing power in dealing with large systems. The literature review highlights the interest areas in each of these topics and the thesis will tackle each of them individually and develop an algorithmic framework which collectively addresses these issues.

CHAPTER 3

Existence of Optimal Controls

The design of feedback controls for nonlinear systems presents a computationally difficult problem, especially when the goal is to achieve online control. As a result, increasing interest is being shown in approaches to nonlinear feedback design, such as receding horizon control, which involve the solution of open loop optimal control problems. Their attractions include their constraint handling capability and the lack of restrictions on the nature of the nonlinear plant for the methods to be applicable. These methods exploit the efficiency of calculating open loop controls while minimising the detrimental effects of open loop policies, such as deviation from the optimal trajectory due to modelling errors and process disturbances. In this chapter we shall explore the necessary conditions one must place on the system dynamics and objective function and control constraints such that optimal controls exist.

3.1 Introduction and General Problem Definition

In order to generate a stabilising state feedback, consider the following idealised design procedure. Denote by x and u the n vector state variable and the m vector control variable respectively. Let Ω denote the set of admissible values of the control variables (i.e. $u \in \Omega$). For example, Ω may be a simple set of box constraints such that

$$\Omega = \{u \in \mathbb{R}^m : a_i \leq u_i \leq b_i \text{ for } i = 1, ..., m\}. \tag{3.1}$$

We assume a dynamic model of the system is available in the form of a first-order vector differential equation,

$$\dot{x}(t) = f(x(t).u(t)). \tag{3.2}$$

This system may be obtained by reducing higher order differential-algebraic equations to a system of first-order equations by various methods described in the literature or by the new method proposed and described in Chapter 8. We may also assume that we have arranged (by changing the origin in the state space) that $x = 0$ corresponds to the desired operating point.

Fix an initial state $\tilde{x}$. Suppose we are able to find an optimal control $u(t)$, $0 \leq t < \infty$, for the optimal control problem $(\tilde{P})$:

$$(\tilde{P}) \qquad \text{Minimise } J := \int_0^\infty \| \bar{x}(t) \|^2 + \bar{\alpha} \| \bar{u}(t) \|^2 \, dt \tag{3.3}$$

subject to:

$$\begin{aligned} \dot{\bar{x}}(t) &= f(t, \bar{x}, \bar{u}) \\ \bar{u}(t) &\in \Omega \\ \bar{x}(0) &= \tilde{x} \end{aligned} \tag{3.4}$$

in which $\bar{\alpha}$ is a parameter chosen appropriately to penalise excessive control action. The class of admissible control functions $u(\cdot)$ is an important issue and we shall discuss this shortly. Write the optimal control $\bar{u}^*_{\tilde{x}}(t)$, $0 \leq t < \infty$, where the subscript $\tilde{x}$ emphasises that the optimal control depends on the initial state for problem $(\tilde{P})$. We choose the feedback law $\bar{u} = F(\tilde{x})$ as the algebraic relationship specifying the control action $\bar{u}$ corresponding to the initial state $\tilde{x}$ to be

$$F(\tilde{x}) := \bar{u}^*_{\tilde{x}}(t)|_{t=0}. \tag{3.5}$$

Thus, given the current state $\tilde{x}$, we solve the open loop optimal control problem $(\tilde{P})$ for initial state $\tilde{x}$ and set $\bar{u}$ to be the initial value of the optimal control. It is this initial control which is implemented in the receding horizon framework until the next state measurement is available. We would expect that the state $x(t)$ of the plant under closed loop control $\bar{u} = F(\tilde{x})$ would converge to the target zero state, since the optimal control problem on which the feedback design is based penalises deviation of the state from the zero state. It can be shown that these expectations are fulfilled if the state trajectory $\bar{x}^*_{\tilde{x}}(t)$ corresponding to initial state $\tilde{x}$ and the optimal open loop control $\bar{u}^*_{\tilde{x}}(t)$ satisfies $\bar{x}^*_{\tilde{x}}(t) \to 0$ as $t \to \infty$ for each initial state $\tilde{x}$, and if there is no plant-model mismatch. Intuitively this is obvious, for if the model is perfect, the predicted output will be exactly

equal to the actual output of the plant and therefore any control which drives the model dynamics to zero will also drive the plant dynamics to the same point.

3.2 Choice of Cost Function in a Process Systems Context

This idealised scheme is perhaps somewhat restrictive. In a process systems context, where plants tend to be open loop stable, one may wish to use alternative choices of cost function J in $(\tilde{P})$ to achieve design objectives besides closed-loop stability, such as speed of response and robust performance. We present here precise conditions on plant models and/or cost functions J (of a kind that might be useful in feedback design) under which $(\tilde{P})$ has an optimal control. Precisely, we seek conditions on the plant model (determined by f and Ω), the cost integrand L and the possible set X_0 of initial states, under which the optimal control problem (P)

$$(P) \qquad \text{Minimise } J := \int_0^\infty L(s, x(s), u(s))\, ds \tag{3.6}$$

subject to:

$$\begin{aligned} \dot{x}(t) &= f(t, x(t), u(t)) \\ u(t) &\in \Omega \\ x(0) &\in X_0 \end{aligned} \tag{3.7}$$

in which $L : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ and $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ are given functions and $\Omega \in \mathbb{R}^m$ and $X_0 \in \mathbb{R}^m$ are given subsets, has a solution.

3.3 Formulating The Infinite Horizon Optimal Control Problem

There is already a substantial literature on problems of optimal control over an infinite horizon, in which optimality conditions are desired and classes of problems are identified for which existence of optimal controls is guaranteed. Much of this is motivated by application to mathematical economics, in which field it is natural to work with notions of optimal controls, for example "overtaking controls", which have infinite costs or costs whose values oscillate as the horizon time tends to $+\infty$ and do not define a limit (see the monograph of (Carlson *et al.*, 1991) for a review of this research and extensive references).

By contrast, the kind of infinite horizon optimal controls relevant to nonlinear feedback design, and therefore of interest in this chapter, have finite costs, and are judged optimal

in the sense that the cost is as small as possible. Another feature of these problems is that the integrand L in the cost $\int_0^\infty L\,dt$ is non-negative, since its role is to penalise excessive state and control excursions. Accordingly, our goal is to give hypotheses on the data for existence of infinite horizon optimal controls, which are taken to have finite cost. We require the cost integrand L to be non-negative, but otherwise place as few restrictions as possible on L, to allow flexibility in the choice of cost functions and therefore in the kind of design specifications achievable through optimisation-based nonlinear feedback design. We permit also explicit time dependence of the plant dynamics and the cost integrand L.

The convergence analysis used here is similar to that originally employed by Roxin (1962), Warga (1972) and others in a finite horizon context. Our "diagonalisation" procedure to handle the infinite horizon has its roots in techniques developed by Halkin (1974) to derive the Maximum Principle for infinite horizon problems. Discrete time analogues of our results on existence of infinite horizon optimal controls were previously supplied by (Keerthi and Gilbert, 1985) and (Meadows and Rawlings, 1993).

Our goal is to give conditions on the data under which the infinite horizon optimal control problem (P) of the introduction has a solution. In order to speak unambiguously of "solutions" to (P) we must first decide on a class of control functions over which an optimal control (together with its associated state trajectory) minimises the cost function. The standard approach to establishing existence of optimal controls is to find a sequence of control functions whose costs converge to the greatest lower bound on the set of costs of admissible controls. The optimal control is then associated with the "limit" of this sequence. A primary consideration in our choice of class of control functions is that the above limit taking procedure should generate a control which is admissible. This is known as the "closure" property. Unfortunately, elementary classes of control functions (piecewise continuously differentiable functions, etc.) do not have this property outside a rather narrow range of optimal control problems.

On the other hand, the class of (Lebesgue) measurable controls does have the desired closure property (under appropriate hypotheses on the functions defining the problem). This is why existence theory for general continuous time optimal control problems is almost universally considered in terms of "measurable" controls. We shall define the term "measurable", as well as provide other useful definitions and theorems, in the next section.

We say that a function $u : [0, \infty) \to \mathbb{R}^m$ is a *control function* if it is measurable and

$$u(t) \in \Omega \text{ for almost every } t \in [0, \infty). \tag{3.8}$$

The standard method of interpreting solutions to the differential equation resulting from adopting the "measurable" control function $u(\cdot)$,

$$\begin{cases} \dot{x}(t) &= f(t, x(t), u(t)) \text{ on } [0, \infty) \\ x(0) &\in X_0, \end{cases} \tag{3.9}$$

is to seek a solution $x(t),\ 0 \leq t < \infty$ which is "locally absolutely continuous". This simply means that x is differentiable almost everywhere, the time derivative function $\dot{x}(t)$ is "integrable" over finite time intervals (it is because we require integrability only over finite time intervals that the functions are referred to as *locally* absolutely continuous) and x can be recovered as the indefinite integral of $\dot{x}$

$$x(t) = x(0) + \int_0^t \dot{x}(s)\, ds, \quad 0 \leq t < \infty. \tag{3.10}$$

The minimisation is then conducted over *admissible control/state pairs* $(u(\cdot), x(\cdot))$ comprising a (measurable) control function $u(t),\ 0 \leq t < \infty$, a locally absolutely continuous n-vector function $x(\cdot)$ which satisfies the dynamical equation (3.9) for almost every t and the constraint on initial states $x(0) \in X_0$. Given any such pair $(x(\cdot), u(\cdot))$ the cost

$$J(x(\cdot), u(\cdot)) := \int_0^\infty L(t, x(t), u(t))\, dt \tag{3.11}$$

is unambiguously defined as a (possibly infinite-valued) Lebesgue integral, since the cost integrand is measurable and non-negative under the hypotheses we shall impose on L.

3.4 Useful Definitions and Concepts

We begin with an introduction to some of the concepts and theorems mentioned above which will be employed in the proof of existence. This section provides an informal review of the main concepts underlying Lebesgue measure and integration. The definitions of null-set, measurable function etc. adopted here are not generally those featuring in the standard treatments (Ash (1972), Royden (1968), etc.) but are an extension of our intuitive understanding of the ideas involved.

To begin with, we define a null-set $\mathcal{O}$ of the real line $\mathbb{R} = (-\infty, \infty)$. Intuitively, a null-set is a set of zero length (or zero "Lebesgue measure"). The difficulty lies in defining

the length of a possibly very complicated set. This problem is overcome by saying that for $\mathcal{O}$ to be a null-set it must be contained in a set (a countable union of open intervals) to which a measure (of length) can be assigned and for which this measure can be made arbitrarily small. The term *countable* arises by considering a one-to-one correspondence between the elements in the set and the natural numbers $\mathbb{N}$, or a finite subset of $\mathbb{N}$. Thus, if we can associate each element of a set A with a natural number, we say the set is *countable*. More precisely, if this correspondence is with the whole of $\mathbb{N}$, we say the set is *countably infinite*. If the correspondence is with a finite subset of $\mathbb{N}$, we say the set is *countably finite*. To be precise in our definition of a null–set:

Definition 3.4.1 (Null–set). *A subset $\mathcal{O} \subset \mathbb{R}$ is a null-set if for any $\epsilon > 0$ there exists a countable family $\{(a_i, b_i)\}_{i=1}^{\infty}$ of open intervals in $\mathbb{R}$ such that $\mathcal{O} \subset \bigcup_i (a_i, b_i)$ and $\sum_i |a_i - b_i| < \epsilon$.*

Given this simple definition of the null-set, we can now attach meaning to a (Lebesgue) measurable function:

Definition 3.4.2 (Lebesgue Measurable). *A function $f : \mathbb{R} \to \mathbb{R}$ is (Lebesgue) measurable if there exists a sequence of continuous functions $\{f_i\}$ and a null-set $\mathcal{O} \subset \mathbb{R}$ such that*

$$f_i(t) \to f(t), \quad \text{for all } t \in \mathbb{R} \backslash \mathcal{O}$$

Alternatively we may write, $f : [a, b] \to \mathbb{R}$ measurable in case for all real α and β, the set $\{t | t \in [a, b] \text{ and } \alpha < f(t) < \beta\}$ is measurable in $\mathbb{R}$. This bound on $f(t)$ is illustrated in Figure 3.2 and will be useful in defining the Lebesgue integral.

If a property holds for all t excluding possibly a null-set, we say that the property holds "almost everywhere" or "for almost every" t (a.e. for short). A measurable function $f : [a, b] \to \mathbb{R}$ with domain some interval $[a, b]$ is simply the restriction of some measurable function on $(-\infty, \infty)$ to the interval $[a, b]$. A measurable vector-valued function $f : [a, b] \to \mathbb{R}^k$ is a function whose components $f^{(1)}, ..., f^{(k)}$ are measurable in the above sense.

Fix an interval I. For any non-negative, measurable function $f : I \to \mathbb{R}$ there is a procedure for defining the "Lebesgue" integral, $\int_I f(t)dt$, which may be infinite. The Lebesgue integral is an extension of the Riemann integral. They coincide for continuous

$f's$, but the Lebesgue integral is defined for a far larger class of integrands, for which the Riemann sums may possibly not converge. For completeness, we recall the definition of the Riemann integral (depicted in Figure 3.1).

Definition 3.4.3 (Riemann Integrable). *Let $f(x)$ be a bounded real-valued function defined on an interval $[a, b]$, such that*

$$a = \epsilon_0 < \epsilon_1 < \cdots < \epsilon_n = b$$

is a subdivision of the interval. We define the following sums

$$S = \sum_{i=1}^{n} (\epsilon_i - \epsilon_{i-1}) M_i$$

and

$$s = \sum_{i=1}^{n} (\epsilon_i - \epsilon_{i-1}) m_i$$

where

$$M_i = \sup_{\epsilon_{i-1} < x < \epsilon_i} f(x), \quad m_i = \inf_{\epsilon_{i-1} < x < \epsilon_i} f(x).$$

The upper and lower Riemann integrals are then

$$R_u = \inf S$$

$$R_l = \sup s.$$

where the inf *and* sup *operators are taken over all possible partitions of the interval $[a, b]$. If $R_u = R_l$, we say that f is Riemann integrable.*

As an example of the limitations of this integral, consider the following example:

$$f(x) = \begin{cases} 0, & x \text{ irrational} \\ 1, & x \text{ rational.} \end{cases} \tag{3.12}$$

For $x \in [a, b]$, $R_u = b - a$ and $R_l = 0$. Therefore, $R_u \neq R_l$ and the function is not Riemann integrable. However, we see that $f(x) = 1$ on a measurable set and zero elsewhere. We wish to be able to define this integral in terms of the measure of the set over which $f(x) = 1$. Below we adopt a method of defining the Lebesgue integral which parallels that of the Riemann integral as much as possible. We depict the Lebesgue integral in Figure 3.2 to illustrate the definition given below.

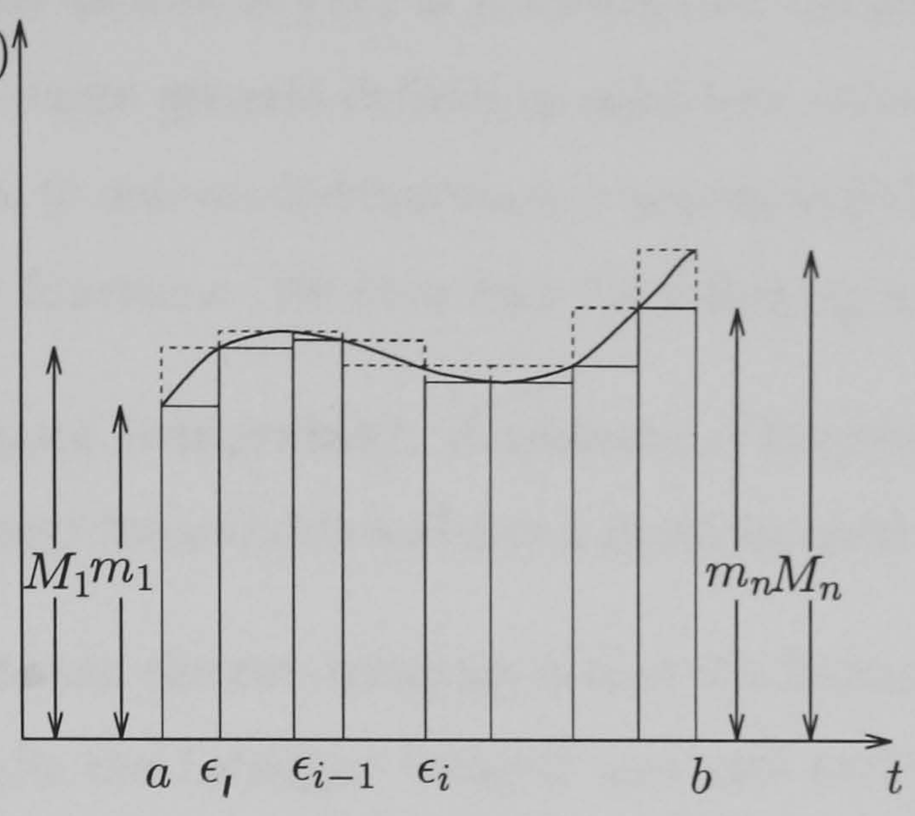


Figure 3.1: Riemann Integral

Definition 3.4.4 (Lebesgue Integrable). *Let $f(x)$ be bounded and measurable on the interval $[a, b]$. Suppose that α and β are any two real numbers such that $\alpha < f(x) < \beta$. Divide the range α to β into n subintervals such that*

$$\alpha = r_0 < r_1 < ... < r_n = \beta.$$

Let $E_k,\ k = 1, 2, ..., n$ be the set of all x in $[a, b]$ such that $r_{k-1} \leq f(x) < r_k$:

$$E_k = \{x : r_{k-1} \leq f(x) < r_k\},\ \ k = 1, 2, ..., n.$$

Since $f(x)$ is measurable, these sets are measurable and disjoint. We define the following sums:

$$S_i = \sum_{k=1}^{n} r_k m(E_k)$$

$$s_i = \sum_{k=1}^{n} r_{k-1} m(E_k)$$

where i denotes a particular partition of the range and $m(E_n)$ is the measure of a set E_n. By varying the partition, as well the number of subintervals n, we obtain a set of values for S_i and s_i. We denote these sets as

$$S = \{S_i\} \ \text{ and } \ s = \{s_i\}.$$

The upper and lower Lebesgue integrals are then

$$L_u = \inf S$$

$$L_l = \sup s.$$

If $L_u = L_l$, we say that $f(x)$ is Lebesgue integrable on $[a, b]$.

This definition then tells us that if $f(x)$ is bounded and measurable on $[a, b]$, then f is Lebesgue integrable. A more general definition considers unbounded functions. The extension of the above ideas to unbounded functions is accomplished by applying a limiting procedure to non-negative functions. We then find the following more general definition:

Definition 3.4.5 (Lebesgue Integrable). *A measurable function $f : I \to \mathbb{R}$ is (Lebesgue) integrable if $|f|$ is (Lebesgue) measurable and has a finite integral.*

The main difference between the two integrals is that the Riemann integral uses intervals and their lengths, while the Lebesgue integral uses sets and their measures. Thus, the Lebesgue integral is applicable in more general cases. The above example is therefore Lebesgue integrable, and the value of this integral is $b - a$. Every Riemann integrable function is Lebesgue integrable. Obviously, the converse is not true.

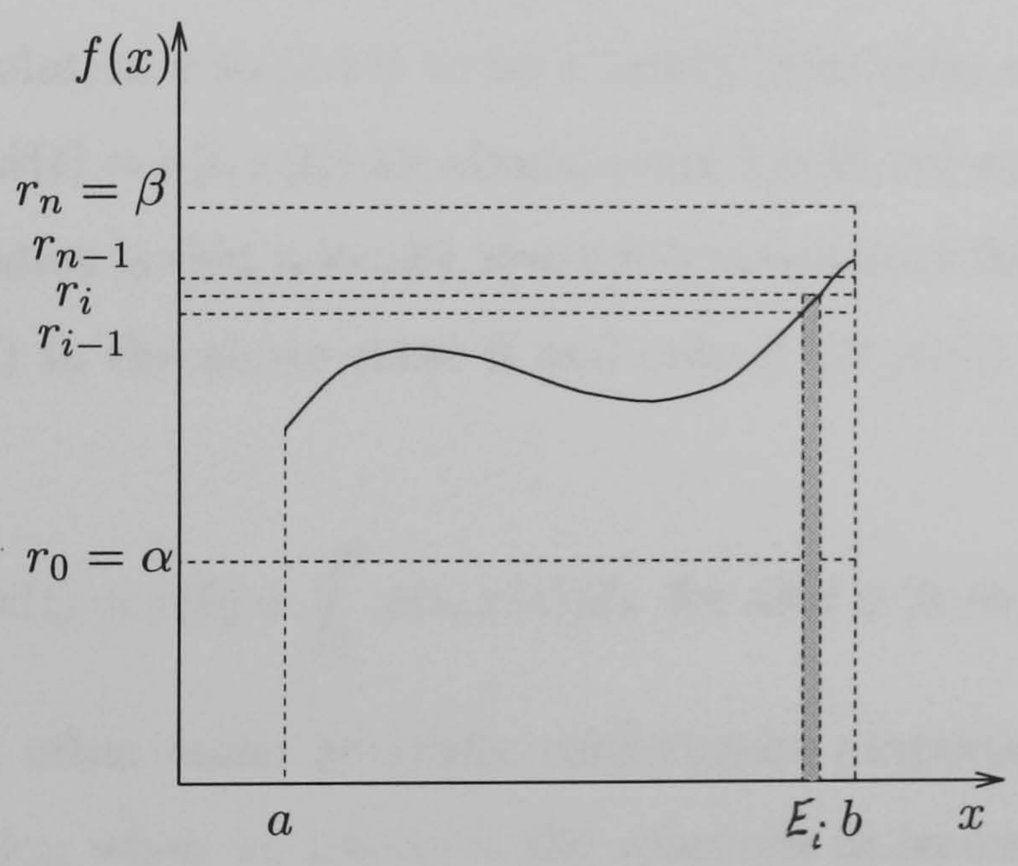


Figure 3.2: Lebesgue Integral

The integral of a vector valued function is simply defined to be the vector of integrals of the components of the function. An integrable vector valued function is one whose components are integrable.

We turn now to the differential equation

$$\begin{cases} \dot{x}(t) &= \phi(t, x(t)) \\ x(0) &= x_0 \end{cases} \tag{3.13}$$

where $\phi : [0, \infty) \times \mathbb{R}^n \to \mathbb{R}^n$ and $x_0 \in \mathbb{R}^n$. Assume that $\phi(\cdot, x)$ is measurable for fixed x and $\phi(t, \cdot)$ is continuous for fixed t. As previously discussed, a natural choice of class

of functions in which to seek solutions to this differential system is the class of *locally absolutely continuous* functions:

Definition 3.4.6. *The class of locally absolutely continuous n-vector valued functions on* $[0, \infty)$ *comprises functions* x *which are differentiable at all points (excluding possibly a null-set) and for which the restriction of* $\dot{x}(\cdot)$ *to any bounded interval* $[0, T]$ *is integrable, and which are expressible as the integral of the derivative, i.e.*

$$x(t) = x(0) + \int_0^t \dot{x}(s)ds \quad \textit{for all } t \in [0, \infty). \tag{3.14}$$

We may set the integrand $\dot{x}$ to any arbitrary value on the null-set on which the derivative is not defined. The choice of value is irrelevant since adjusting the integrand on a null-set leaves the integral unaltered. The qualifier "local" indicates that only the restrictions of $\dot{x}$ to *bounded* intervals $[0, T]$ are required to be integrable.

We now define a solution x to (3.13) to be a locally absolutely continuous function on $[0, \infty)$ which satisfies $\dot{x}(t) = \phi(t, x(t))$ for almost every t in $[0, \infty)$ and for which $x(0) = x_0$. An important observation is that a locally absolutely continuous function $x : [0, \infty) \to \mathbb{R}^n$ is a solution to (3.13) in the above sense if and only if $\phi(t, x(t))$ is integrable on finite intervals and

$$x(t) = x(0) + \int_0^t \phi(s, x(s))ds \quad \text{for all } t \in [0, \infty). \tag{3.15}$$

This is because it is often easier to study convergence properties of the solutions to the differential equation when we interpret the solutions in terms of the above integral equation.

We now consider further general results from real analysis (see, for example, Royden (1968)). We begin with some "warm-up" definitions before stating the remaining crucial definitions and theorems for the proof:

$\| x \|$ denotes the Euclidean norm of the m-vector x, $\| x \| := (\sum x_i^2)^{\frac{1}{2}}$. We write $\mathbb{B}^m$ for the open unit ball in $\mathbb{R}^m$, namely $\mathbb{B}^m = \{x \in \mathbb{R}^m : \| x \| < 1\}$.

Recalling our intuitive definition of a set, we consider the following:

Definition 3.4.7 (σ-algebra). *(Royden, 1968)*

An algebra A of sets is called a σ*-algebra if every union of a countable collection of sets in A is again in A.*

The intersection of any collection of closed sets is closed and the union of any finite collection of closed sets is closed. However, the union of a *countable* collection of closed sets is not necessarily closed. This is obvious if we consider the set of rational numbers, which is not closed, although the set is the union of a countable collection of closed sets, each of which contains exactly one number. Thus, to characterise $\sigma-$algebras of sets which contain all of the closed sets, we must consider more general sets than simply open and closed sets. This leads to the following useful definition.

Definition 3.4.8 (Borel set). *(Royden, 1968)*

The collection B of Borel sets is the smallest σ-algebra which contains all of the open sets.

Armed with these definitions, we may now consider the following definition and theorem.

Definition 3.4.9 ($L \times B$ measurable). *(Clarke, 1983)*

Consider a function $F(t, x(t), \dot{x}(t))$, $t \in [a, b], x \in \mathbb{R}^n$. $L \times B$ denotes the σ-algebra of subsets of $[a, b] \times \mathbb{R}^{2n}$ generated by product sets $M \times N$, where M is a Lebesgue measurable subset of $[a, b]$ and N is a Borel subset of $\mathbb{R}^{2n}$.

Theorem 3.4.1 (Measurable Selection). *(Clarke, 1983), Thm. 3.1.1*

Consider a multifunction $F : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$, a mapping from $\mathbb{R}^m$ to the subsets of $\mathbb{R}^n$. Let F be measurable, closed and nonempty on $S \subset \mathbb{R}^m$. Then there exists a measurable function $\gamma : S \to \mathbb{R}^n$ such that $\gamma(x)$ belongs to $F(x)$ for all $x \in S$.

This theorem will be necessary to assert the existence of a control/trajectory pair which satisfies the system equations.

Theorem 3.4.2 (Compactness of Trajectories). *(Clarke, 1983). Thm. 3.1.7*

Let F be $L \times B$ measurable, and let $\{x_j\}$ be a sequence of trajectories on $[a, b]$ satisfying

i. $x_j(t) \in X(t)$ and $|\dot{x}(t)| \leq \phi(t)$ for almost all $t \in [a, b]$.

ii. $\dot{x}_j(t) \in F(t, x_j(t) + y_j(t)) + r_j(t)\mathbb{B}$ for $t \in A_j$, where $\{y_j\}, \{r_j\}$ are sequences of measurable functions on $[a, b]$ which converge uniformly to 0, and where $\{A_j\}$ is a sequence of measurable subsets of $[a, b]$ such that measure $(A_j) \to (b - a)$.

iii. The sequence $\{x_j(a)\}$ is bounded.

Then there is a subsequence of $\{x_j\}$ which converges uniformly to a trajectory x which is a trajectory for F.

This theorem then allows us to conclude that solutions to a differential inclusion are preserved under limit taking. This result will be important when we extract a convergent sequence of trajectories over $[0, \infty)$ using the diagonalisation procedure.

The distance of a point $x \in \mathbb{R}^m$ from a closed set A in $\mathbb{R}^m$, written $d_A(x)$, is defined to be

$$d_A(x) = \min_{y \in A} \| x - y \| . \tag{3.16}$$

Definition 3.4.10 (Hausdorff Distance). *Given two sets $A, B \subset \mathbb{R}^m$ the Hausdorff distance between them is*

$$d(A, B) = \max\{\max_{y \in A} d_B(y), \max_{y \in B} d_A(y)\}. \tag{3.17}$$

This is illustrated in Figure 3.3.

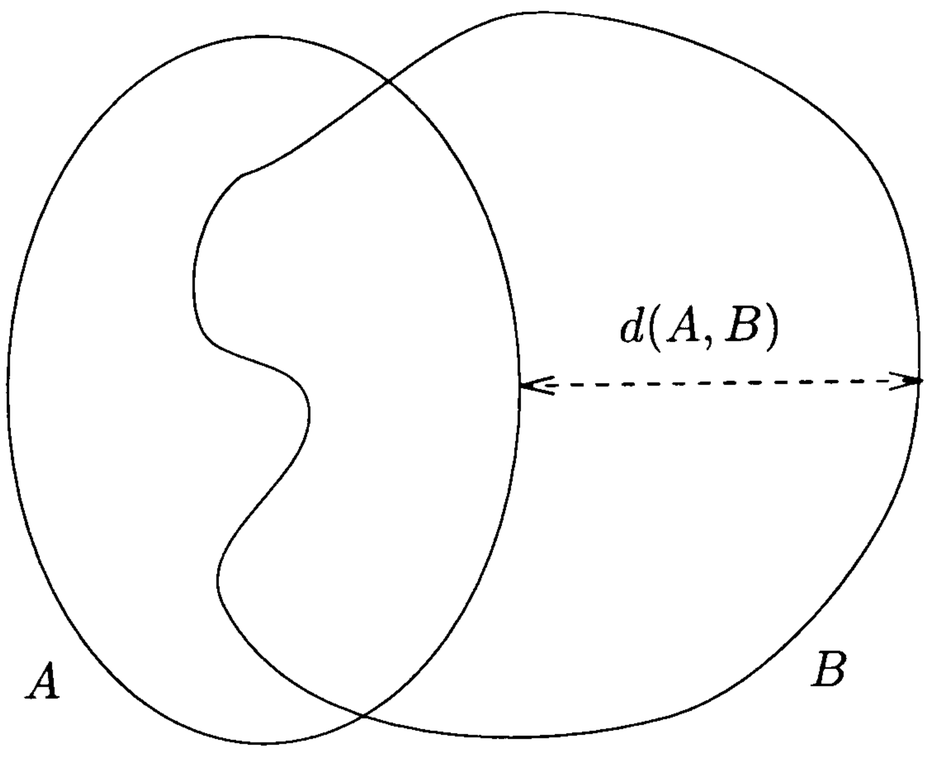


Figure 3.3: Hausdorff Distance

Definition 3.4.11 (Hausdorff Continuous). *Take $D \subset \mathbb{R}^k$ and a multifunction $G : D \rightrightarrows \mathbb{R}^m$, where G is a mapping from points in D to* subsets *of $\mathbb{R}^m$, such that $G(x)$ is a closed set for each $x \in D$. We say G is (Hausdorff) continuous on D if for any sequence $\{x_i\}$ in D with $x_i \to x$, $d(G(x_i), G(x)) \to 0$ as $i \to \infty$ (Figure 3.4).*

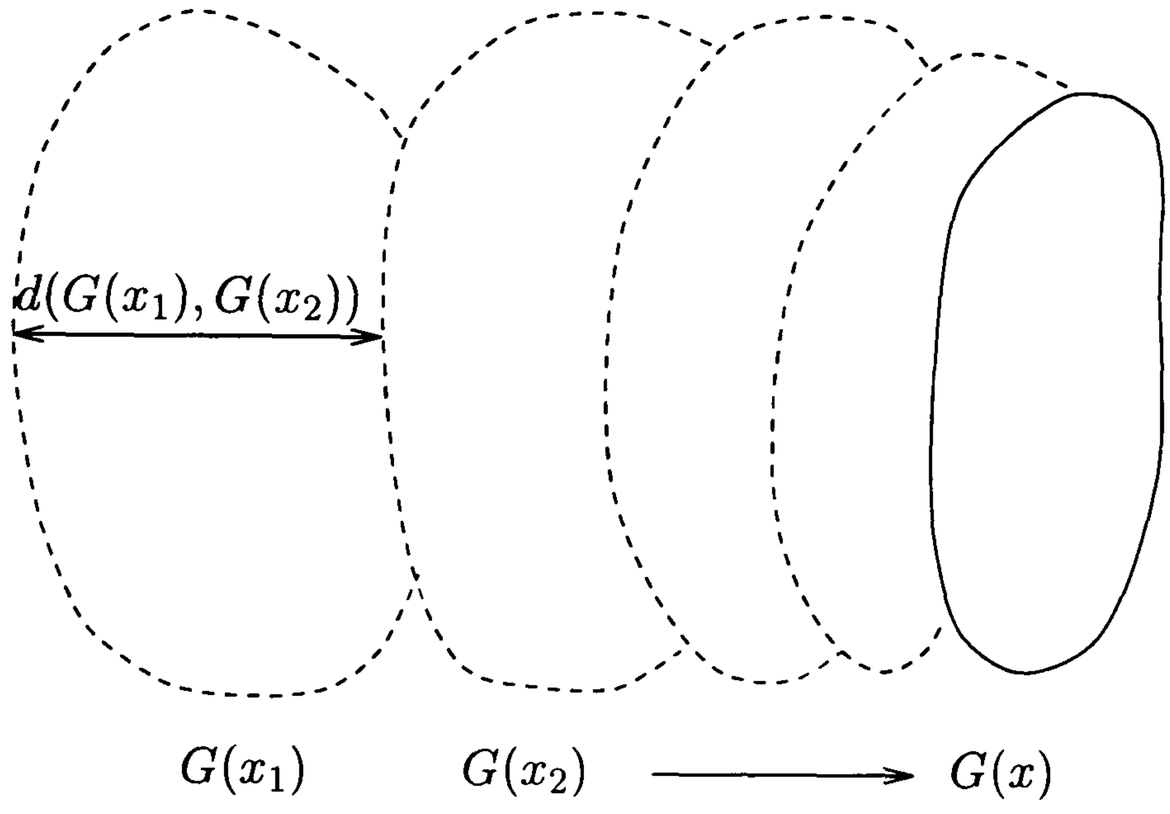


Figure 3.4: Hausdorff Continuity

Lemma 3.4.1 (Fatou's Lemma). *(Kingman and Taylor, 1966)*

If $\{f_n\}$ is a sequence of measurable functions which is bounded below by an integrable function, then

$$\int \lim_{n\to\infty} \inf f_n d\mu \leq \lim_{n\to\infty} \inf \int f_n d\mu. \tag{3.18}$$

Given all of the above definitions and theorems, we may now construct a proof of existence of optimal controls defined on an infinite horizon.

3.5 Conditions for the Existence of Infinite Horizon Optimal Controls

We begin by stating the theorem we wish to prove:

Theorem 3.5.1. *Assume that the data for problem (P) satisfies the hypotheses:*

$(H1)$: *$f(t,x,u)$ is measurable in t for fixed (x,u) and continuous in u for fixed (t,x).*

$(H2)$: *There exists a locally integrable function $c(\cdot)$ such that*

$$|f(t,x,u)| \leq c(t)(1+|x|) \text{ a.e. } t \in [0,\infty), \quad x \in \mathbb{R}^n \text{ and } u \in \Omega$$

$(H3)$: *There exists a locally integrable function $k(\cdot)$ such that*

$$|f(t,x,u) - f(t,y,u)| \leq k(t)|x-y| \text{ a.e. } t \in [0,\infty), \quad x,y \in \mathbb{R}^n \text{ and } u \in \Omega$$

$(H4)$: *$L(t,x,u) \geq 0$ for all (t,x,u). $L(\cdot,x,u)$ is measurable on $[0,\infty)$ for all (x,u). $L(t,\cdot,\cdot)$ is continuous for almost every $t \in [0,\infty)$. L is bounded on bounded sets.*

$(H5)$: *The "extended velocity set"*

$$\mathcal{S}_{t,x} := \{(v,\gamma) \in \mathbb{R}^n \times \mathbb{R}:\ v = f(t,x,u),\ \gamma \geq L(t,x,u)\}$$

is convex for all x, for almost every $t \in [0,\infty)$.

$(H6)$: *X_0 is compact, Ω is compact*

Then (P) has a minimising control function for each $x(0) \in X_0$.

3.5.1 Comments on Existence Conditions

(a) The linear growth and global Lipschitz continuity conditions in $(H2)$ and $(H3)$ are simply stated hypotheses ensuring that given any initial state x_0 and control $u(t),\ 0 \leq t < \infty$, a state trajectory $x(t),\ 0 \leq t < \infty$ corresponding to $u(\cdot)$ and satisfying $x(0) = x_0$ may be developed over all time, and further that sequences of state trajectories considered in the proof of the existence theorem are bounded. They could be substituted by less restrictive (but more complicated) conditions having this role where, for example, $(H3)$ is required to hold only for $x's$ and $y's$ in some suitably large bounded set. On the other hand, the conditions on L (hypotheses $(H4)$) are unrestrictive to allow flexibility in applications to optimisation-based nonlinear controller design.

(b) The hypothesis $(H5)$ (convexity of the extended velocity set) is to ensure "optimal" controls are not "chattering controls" (controls which are limits in some sense of rapidly oscillating controls), which are not admissible in our framework.

As a simple example (Vinter, 1987), consider the case where we have $\Omega = \{-1\} \bigcup \{1\}$, quadratic cost function $\int_0^1 x^2(t)\,dt$ and state equation $f(t,x,u) = u(t)$. This system does not satisfy the convexity hypothesis $(H5)$. The control strategy is shown in Figure 3.5 with corresponding state trajectory shown in Figure 3.6.

The limit as $i \to \infty$ is $x(\cdot) = 0$. This implies $u(\cdot) = 0 \notin \Omega$. This non-convex problem fails to have a solution as the limit of rapidly switching controls does not exist in a conventional sense. Therefore, $(H5)$ is necessary to eliminate the possibility of such a situation arising.

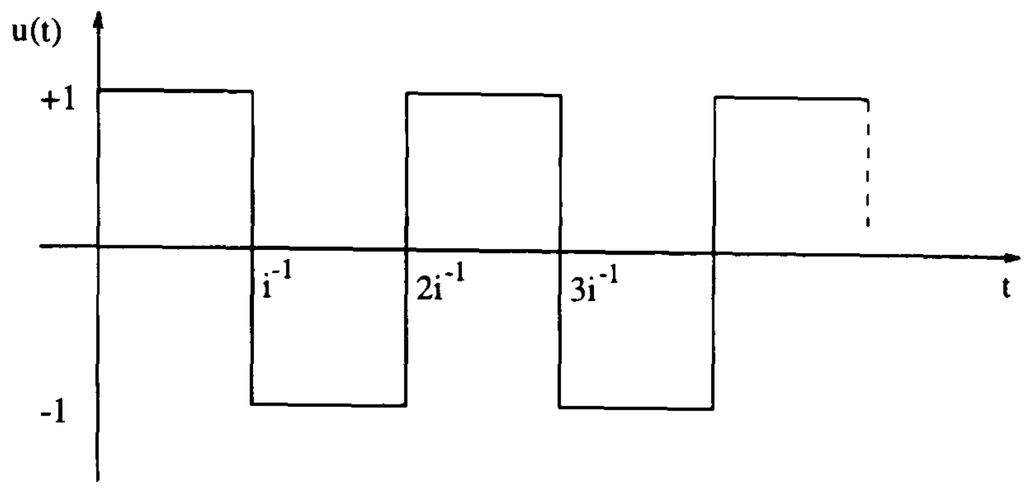


Figure 3.5: Graph of $u_i(\cdot)$, $i = 1, 2, ...$

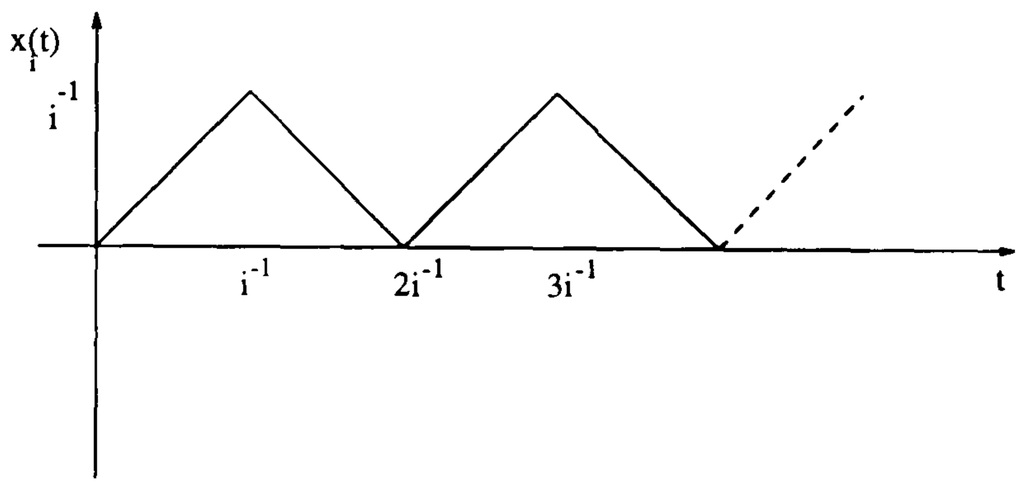


Figure 3.6: Graph of $x_i(\cdot)$, $i = 1, 2, ...$

(*c*) The omission of a hypothesis

"there exists an admissible control/state pair with finite cost"

(3.19)

is just a matter of semantics, to the extent that if all admissible control/state pairs have only infinite cost, then we regard any admissible control/state pair (and at least one exists) as optimal, since no other control/state pair has lower cost. Thus the assertions of the theorem are automatically valid if (3.19) is violated. Of course this is an uninteresting case; the theorem is of use only if f and L satisfy supplementary conditions (such as there exists a control/state pair $(u(\cdot), x(\cdot))$ and a time $T \geq 0$ such that $L(t, x(t), u(t)) = 0$ for $t \geq T$) ensuring validity of (3.19).

3.5.2 Proof of Theorem 3.5.1

We note at the outset that admissible control/state pairs exist. Indeed, for any $x_0 \in X_0$ and measurable function $\bar{u}\colon [0, \infty) \to \Omega$ it is known that a unique solution $\bar{x}(\cdot)$ to:

$$\dot{x} = f(t, x(t), u(t)). \quad x(0) = x_0 \tag{3.20}$$

can be developed over the entire infinite interval $[0, \infty)$ in view of assumptions $(H1)$ and $(H2)$ (see (Sontag, 1990)). Then $(\bar{x}(\cdot), \bar{u}(\cdot))$ is one such pair.

We may assume without loss of generality that an admissible control state pair exists for which the cost is finite (otherwise any admissible pair achieves the infimum cost (∞)). In this case, there exists a "minimising sequence" of finite cost admissible control/state pairs $\{(x_i(\cdot), u_i(\cdot))\}$ satisfying

$$\lim_i J(x_i(\cdot), u_i(\cdot)) = \inf\{(P)\} \quad < \infty \tag{3.21}$$

in which $\inf\{(P)\}$ denotes the infimum cost for (P).

Consider a strictly increasing sequence of positive times $\{T_0 = 0, T_1, ..., T_j, ...\}$ such that $T_j \to \infty$ as $j \to \infty$. Define

$$z_i(t) := \int_0^t L(s, x_i(s), u_i(s))\, ds. \tag{3.22}$$

for $i = 1, 2, ...$

Fix j. Under the hypotheses there exist $r(\cdot) \in L^1[0, T_j]$, $K > 0$ and $\epsilon > 0$ such that, if we define the multifunction $F\colon [0, T_j] \times \mathbb{R}^n \to \mathbb{R}^{n+1}$ to be

$$F(t, x) := \{(f(t, x, u), \gamma + L(t, x, u))\colon\ \gamma \geq 0,\ u \in \Omega\} \bigcap r(t)\ \mathbb{B}^{n+1} \tag{3.23}$$

then the following assertions are valid:

- $(x_i(t), z_i(t)) \in K\ \mathbb{B}^{n+1}$, for all $t \in [0, T_j]$ and all i
- $F(t, x)$ is convex, compact valued and non-empty for $(t, x) \in [0, T_j] \times (K + \epsilon)\ \mathbb{B}^n$
- $F(t, x) \subset r(t)\mathbb{B}^{n+1}$, for all $(t, x) \in [0, T_j] \times (K + \epsilon)\ \mathbb{B}^n$
- $F(t, \cdot)$ is (Hausdorff) continuous on $(K + \epsilon)\ \mathbb{B}^{n+1}$
- $$\begin{cases} (\dot{x}_i(t), \dot{z}_i(t)) \in F(t, x_i(t)) \quad a.e. \quad t \in [0, T_j] \\ x_i(0) \in X_0,\ z_i(0) = 0 \end{cases}$$

The above properties imply

$$\begin{cases} \{(\dot{x}_i(0), \dot{z}_i(0))\}_{i=1}^{\infty} \text{ is a bounded sequence} \\ \parallel (\dot{x}_i(t), \dot{z}_i(t)) \parallel \leq\ r(t) \text{ a.e. } t \in [0, T_j]. \end{cases} \tag{3.24}$$

Now define $\tilde{F}(t,(x,z)) := F(t,x)$ and $X(t) := K\mathbb{B}^{n+1}$. $\tilde{F}(t,(x_i(t),z_i(t)))$ is continuous on $X(t) + \epsilon\mathbb{B}^{n+1}$. These relationships (along with certain measurability conditions on $\tilde{F}(\cdot,(x,z))$ which are also satisfied) are a special case of hypotheses under which solutions to the differential inclusion $(\dot{x},\dot{z}) \in \tilde{F}(t,(x,z))$ are known to be preserved under limit taking (Theorem 3.4.2). This "compactness of trajectories" theorem tells us that, along a subsequence :

$$x_i^j(t) \to \bar{x}^j(t), \quad z_i^j(t) \to \bar{z}^j(t) \quad \text{uniformly on } [0,T_j] \tag{3.25}$$

for some $(\bar{x}^j(\cdot),\bar{z}^j(\cdot)) \in (AC([0,T_j];\ \mathbb{R}^{n+1})$[1] satisfying

$$\begin{cases} (\dot{\bar{x}}^j(t),\dot{\bar{z}}^j(t)) \in \tilde{F}(t,(\bar{x}^j(t),\bar{z}^j(t))) \\ \bar{x}^j(0) \in X_0, \quad \bar{z}^j(0) = 0. \end{cases} \tag{3.26}$$

An appropriate measurable selection theorem (e.g. Theorem 3.4.1) may now be invoked. This asserts the existence of a control function $\bar{u}^j : [0,T_j] \to \Omega$ and a measurable function $\gamma^j : [0,T_j] \to [0,\infty)$ such that

$$\dot{\bar{x}}^j(t) = f(t,\bar{x}^j(t),\bar{u}^j(t)) \ \text{ a.e. on } [0,T_j] \tag{3.27}$$

and

$$\begin{aligned} \bar{z}^j(T_k) - \bar{z}^j(T_{k-1}) &= \int_{T_{k-1}}^{T_k} \gamma^j(t)dt + \int_{T_{k-1}}^{T_k} L(t,\bar{x}^j(t),\bar{u}^j(t))\ dt \\ &\geq \int_{T_{k-1}}^{T_k} L(t,\bar{x}^j(t),\bar{u}^j(t))\ dt \text{ for } \ k = 1,...,j. \end{aligned}$$

By (3.22) and (3.25), we have

$$\lim_{i\to\infty} \int_{T_{k-1}}^{T_k} L(t,\dot{x_i^j}(t),\dot{u_i^j}(t))\ dt \geq \int_{T_{k-1}}^{T_k} L(t,\bar{x}^j(t),\bar{u}^j(t))\ dt \tag{3.28}$$

(along the subsequence), for $k = 1,...,j$.

We now extract a subsequence S of the functions $\{u_i : [0,\infty) \to \Omega,\ x_i : [0,\infty) \to \mathbb{R}^n\}$ by means of a diagonalisation procedure. Specifically, we extract to begin with a subsequence such that, for $j = 1$,

$$x_i^j(t) \to \bar{x}^j(t) \ \text{ uniformly on } [0,T_j] \tag{3.29}$$

[1] We use the notation $AC([0,T];\mathbb{R}^n)$ to denote an n-vector valued function which is absolutely continuous over the interval $[0,T]$.

and

$$\lim_{i\to\infty}\int_{T_{k-1}}^{T_k} L(t,x_i^j(t),u_i^j(t))\,dt \geq \int_{T_{k-1}}^{T_k} L(t,\bar{x}^j(t),\bar{u}^j(t))\,dt \tag{3.30}$$

for $k = 1,...,j$. Choose the first element of S to be the first element in this subsequence.

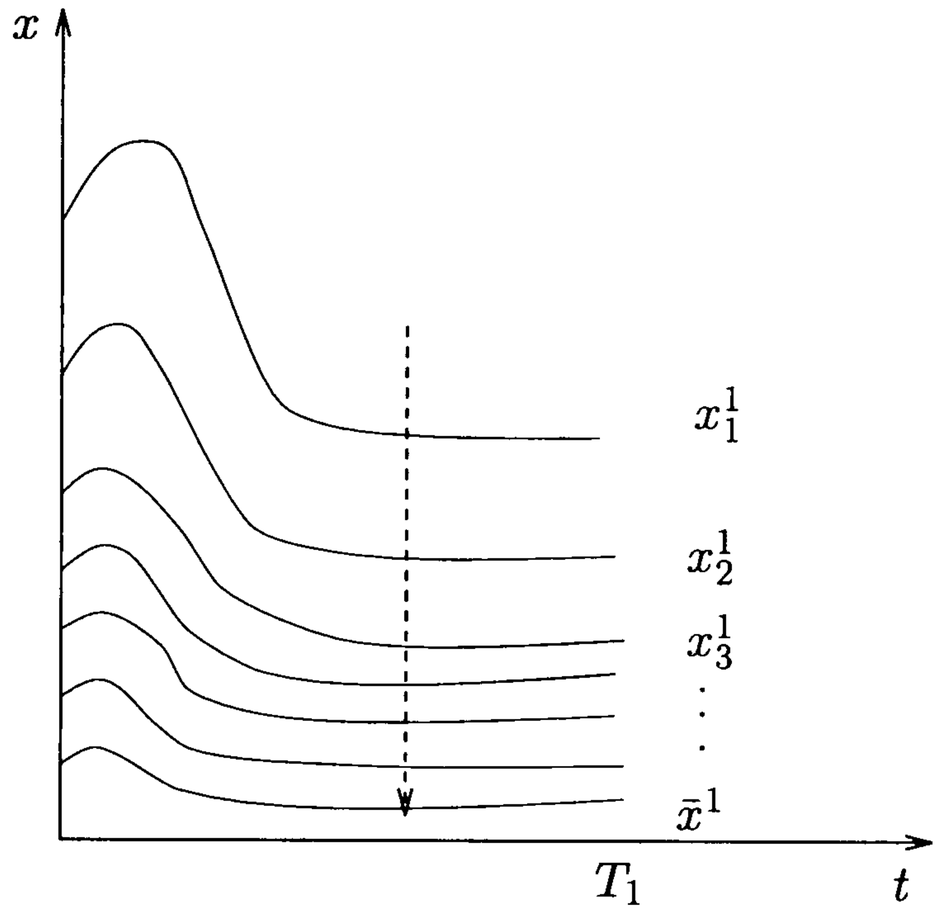


Figure 3.7: Diagonalisation Procedure: S_1 on $[0,T_1]$

Set $j = 2$ and extract a further subsequence such that (3.29) and (3.30) are again satisfied for this larger j, and choose the second element in S to be the second element in this most recent subsequence S_j. We proceed in this way for $j = 3,4,...$ thereby generating the subsequence S_n as follows

$$\left.\begin{array}{rl} S_1: & x_1^1,\, x_2^1,\, x_3^1,\ldots \text{ s.t. } x_i^1 \;\to\; \bar{x}^1 \text{ on } [0,T_1] \\ S_2: & x_1^2,\, x_2^2,\, x_3^2,\ldots \text{ s.t. } x_i^2 \;\to\; \bar{x}^2 \text{ on } [0,T_2] \\ & \vdots \\ S_n: & x_1^n,\, x_2^n,\, x_3^n,\ldots \text{ s.t. } x_i^n \;\to\; \bar{x}^n \text{ on } [0,T_n] \end{array}\right\} \tag{3.31}$$

where $n \to \infty$ and $S_n \subset S_{n-1} \subset \cdots \subset S_2 \subset S_1$. The extracted diagonal elements of each subsequence form the new minimising subsequence S,

$$S = \{\tilde{x}_i\} = \{x_1^1,\, x_2^2,\, x_3^3,\, \ldots\, ,x_n^n\, ,\, \ldots\} \tag{3.32}$$

Now, if we define

$$\bar{x}(t) := \bar{x}^j(t), \text{ for } t \in [T_{j-1}, T_j], \ j = 1, 2. \ldots \tag{3.33}$$

and

$$\bar{u}(t) := \bar{u}^j(t), \text{ for } t \in [T_{j-1}, T_j], \ j = 1, 2, \ldots, \tag{3.34}$$

then

$$\tilde{x}_i \ \rightarrow \ \bar{x} \text{ for all t } \epsilon \, [0, \infty). \tag{3.35}$$

Therefore $(\bar{x}(\cdot), \bar{u}(\cdot))$ is a control function and corresponding state trajectory which satisfies the initial state constraint and

$$\lim_i I_i^j \geq I^j \ \ (\geq 0) \qquad\qquad \text{for } j = 1, 2, \ldots \tag{3.36}$$

where

$$I_i^j := \int_{T_{j-1}}^{T_j} L(t, x_i(t), u_i(t)) \, dt \ \text{ and } I^j := \int_{T_{j-1}}^{T_j} L(t, \bar{x}(t), \bar{u}(t)) \, dt. \tag{3.37}$$

Now consider any family of convergent sequences of non-negative numbers $\{a_i^j\}_{i=1}^{\infty}$ parameterised by $j = 1, 2, \ldots$ and a "limiting" sequence of nonnegative numbers $\{\bar{a}^j\}_{j=1}^{\infty}$ such that

$$\lim_{i \to \infty} a_i^j \geq \bar{a}^j \tag{3.38}$$

for each j. It follows from Fatou's Lemma that

$$\lim_{i \to \infty} (\sum_{j=1}^{\infty} a_i^j) \ \geq \ \sum_{j=1}^{\infty} \bar{a}^j. \tag{3.39}$$

Identifying a_i^j with I_i^j and $\bar{a}^j$ with I^j we conclude that

$$\inf\{(P)\} = \lim_{i \to \infty} \sum_{j=!}^{\infty} I_i^j \ \geq \ \sum_{j=!}^{\infty} I^j = J(\bar{x}(\cdot), \bar{u}(\cdot)). \tag{3.40}$$

Since $(\bar{x}(\cdot), \bar{u}(\cdot))$ is admissible this implies that $(\bar{x}(\cdot), \bar{u}(\cdot))$ is a minimiser. Existence of a minimiser therefore has been proved.

■

3.6 Conclusions

A proof of existence of optimal controls for deterministic continuous time infinite horizon systems has been presented. This work is the continuous time analogue of the discrete time work of (Keerthi and Gilbert, 1985) and (Meadows and Rawlings, 1993). The conditions on the data are motivated by process systems applications where, unlike applications in mathematical economics where notions of "overtaking controls" allow infinite or oscillating costs, the goal is to attain some minimum finite cost. The system description is of a general nature, permitting nonlinear time-varying dynamics as well as a time-varying objective function. This allows flexibility in the choice of objective function in order to achieve common design goals such as speed of response and minimum cost.

Further extensions of this work will need to consider the effect of plant/model mismatch on the stabilising property of the idealised control. This requires definition of a permissible level of mismatch and/or a fixed time (i.e. sampling time) over which the optimal control is applicable before which a new set of optimal controls must be calculated (i.e. a receding horizon framework of application).

The most important deficiency in this work is the absence of state constraints. The incorporation of state constraints in the problem makes the determination of conditions for existence of optimal controls a much more difficult problem. In the next chapters we address the problem of dealing with state constraints within an algorithmic framework.

CHAPTER 4

Necessary Conditions for Optimal Controls

There are several approaches, as discussed in the literature review, to translate the system description into a recipe for generating optimal controls. One such approach is the direct solution of necessary conditions. This chapter will outline the basic formulation of this thesis, which solves an exact maximisation of the Hamiltonian at each time t. The key contribution is the development of the appropriate necessary conditions for constrained DAE systems of a general nature. The basic formulation reduces the optimal control problem to a two point boundary value problem.

The Hamiltonian description of this thesis is unique in its handling of mixed control/state and pure control inequality constraints. The approach outlined uses a logarithmic barrier function to inhibit deviations from the constrained feasible region. This will form the basis for development of new necessary conditions for constrained DAE systems.

4.1 Problem Definition

4.1.1 Original System Definition

Consider the following optimal control problem for a general DAE system, with state and control inequality and equality constraints on the path and endpoints:

$$\min_{u(\cdot),v} P(t_f, \dot{x}(t_f), x(t_f), u(t_f), v) \tag{4.1}$$

subject to

$$\begin{array}{rcccl} & & f(t,\dot{x}(t),x(t),u(t),v) & = & 0 \\ \underline{g} & \leq & g(t,\dot{x}(t),x(t),u(t),v) & \leq & \overline{g} \\ \\ & & \phi^0(t_0,\dot{x}(t_0),x(t_0),u(t_0),v) & = & 0 \\ \underline{\psi}^0 & \leq & \psi^0(t_0,\dot{x}(t_0),x(t_0),u(t_0),v) & \leq & \overline{\psi}^0 \\ \\ & & \phi^f(t_f,\dot{x}(t_f),x(t_f),u(t_f),v) & = & 0 \\ \underline{\psi}^f & \leq & \psi^f(t_f,\dot{x}(t_f),x(t_f),u(t_f),v) & \leq & \overline{\psi}^f \\ \\ \underline{v} & \leq & V(v) & \leq & \overline{v} \end{array} \tag{4.2}$$

for $t \in [t_0, t_f]$, where the superscript 0 indicates equations enforced at initial time t_0 and the superscript f indicates equations enforced at the final time t_f. We use v to denote time-invariant parameters and $u(\cdot)$ to denote "controls". Note that the minimisation may be performed over $x(t_0)$, $\dot{x}(t_0)$ as well as $u(\cdot)$, v if the initial state is not fixed. We omit these terms only to prevent bulkiness under the "min". All functions are assumed measurable in t and twice continuously differentiable with respect to all arguments. From this point onwards, we shall remove explicit time dependence in the equations. Assume all controls and states to be time dependent.

Note that objective functions of the form

$$P = \int_{t_0}^{t_f} L(t,\dot{x},x,u,v)\ dt \tag{4.3}$$

fit easily within this framework by introducing a new state

$$\dot{x}_{n+1} = L(t,\dot{x},x,u,v), \qquad x_{n+1}(t_0) = 0. \tag{4.4}$$

The objective function then becomes

$$P(t_f,\dot{x}(t_f),x(t_f),u(t_f),v) = x_{n+1}(t_f) \tag{4.5}$$

The optimal control problem is to choose an admissible set of initial states, a control trajectory and parameters (which may also include the final time in a free endtime formulation) to minimise the objective P subject to the equality and inequality constraints in the system description (4.2).

We note that in the DAE system $f(t, \dot{x}, x, u, v) = 0$, x represents the true state variables, whose derivatives also appear in the equations, while "algebraic variables" x' are included in the "control" vector u. We further assume that the problem is well-posed, so that it has a solution such that the optimal control is finite. This means that either the objective function is convex or the system is constrained such that the optimum is finite. Thus, there exists a level set of the objective function such that its intersection with the feasible region is compact. Furthermore, we assume there is a partition $u = [u', x']$ such that $[f_{\dot{x}},\ f_{x'}]$ is nonsingular along feasible trajectories. This of course implies that the DAE system has index one, but we give in a later chapter a technique for transforming a problem involving a DAE system of arbitrary index to one satisfying these conditions.

The proposed new algorithm is based on the classical idea (see, for example, Bryson and Ho (1975)) of satisfying the necessary conditions of optimality. Of course, the classical necessary conditions must be extended to deal with a DAE system and path inequality constraints, and the theoretical development will be outlined shortly. In essence this is achieved by using barrier functions to deal with inequalities, which provides a natural framework for solving the resulting equality-constrained optimisation problem by interior point methods.

4.1.2 Modified System Definition

We use a logarithmic barrier function to inhibit deviations from the constrained feasible regions. The result is the following augmented cost function:

$$\begin{aligned}\min_{u(\cdot),v} \hat{P} = P + x_p(t_f) - \\ \nu^0 \sum_i \{\ln(\psi_i^0 - \underline{\psi}_i^0) + \ln(\overline{\psi}_i^0 - \psi_i^0)\} - \nu^f \sum_i \{\ln(\psi_i^f - \underline{\psi}_i^f) + \ln(\overline{\psi}_i^f - \psi_i^f)\} - \\ \mu^0 \sum_i \{\ln(g_i^0 - \underline{g}_i^0) + \ln(\overline{g}_i^0 - g_i^0)\} - \mu^f \sum_i \{\ln(g_i^f - \underline{g}_i^f) + \ln(\overline{g}_i^f - g_i^f)\} - \\ \rho \sum_k \{\ln(V_k - \underline{v}_k) + \ln(\overline{v}_k - V_k)\}\end{aligned} \tag{4.6}$$

where

$$f_p = \dot{x}_p + \mu(t) \sum_i \{\ln(g_i(t) - \underline{g}_i(t)) + \ln(\overline{g}_i(t) - g_i(t))\} = 0, \quad x_p(t_0) = 0. \tag{4.7}$$

We will see shortly that the logarithmic barrier terms contribute nothing to the cost at the optimum, as the interior point method forces the multipliers $\nu^0, \nu^f, \mu^0, \mu^f$ and ρ to converge to zero as the inequalities converge to their feasible values.

4.2 Development of Adjoint System

Our assumptions ensure that there is at least one feasible solution and that for any feasible x_0 there is a unique solution for each (u', v) pair. We introduce an adjoint system to provide complementary information to the state dynamics. Our assumptions also ensure a unique solution of the adjoint system $\bar{p}(t)$ for each $\bar{p}(t_f)$.

We develop the adjoint system by introducing exact differential equations and integrating factors to solve a differential-algebraic system of equations. Consider a general system of equations

$$F(t, \dot{X}, X) = 0. \tag{4.8}$$

We linearise about some admissible nominal trajectory $[\dot{\bar{X}} \quad \bar{X}]$ and evaluate at $[\dot{\bar{X}} + \dot{X} \quad \bar{X} + X]$ to yield

$$F(t, \dot{\bar{X}}, \bar{X}) + \frac{\partial F}{\partial \dot{X}}\dot{X} + \frac{\partial F}{\partial X}X = 0 \tag{4.9}$$

and simplify to obtain

$$F_{\dot{X}}\, dX + F_X X\, dt = 0. \tag{4.10}$$

Introducing an integrating factor $\tilde{y}$, we write

$$\tilde{y}^T F_{\dot{X}}\, dX + \tilde{y}^T F_X X\, dt = 0. \tag{4.11}$$

Equation (4.11) is an exact differential if

$$\frac{\partial}{\partial t}(\tilde{y}^T F_{\dot{X}}) = \frac{\partial}{\partial X}(\tilde{y}^T F_X X) \tag{4.12}$$

and therefore

$$-\frac{d}{dt}(\tilde{y}^T F_{\dot{X}}) + \tilde{y}^T F_X = 0. \tag{4.13}$$

This may be written in the standard adjoint notation as

$$\begin{aligned} \dot{\bar{p}} &= F_X^T \tilde{y} \\ \bar{p} &= F_{\dot{X}}^T \tilde{y}. \end{aligned} \tag{4.14}$$

Therefore, if we consider the following primary functions:

$$F = \begin{bmatrix} f_p \\ f \end{bmatrix} \quad \dot{X} = \begin{bmatrix} \dot{x}_p \\ \dot{x} \\ \dot{x}' \end{bmatrix} \qquad \tilde{y} = \begin{bmatrix} y_p \\ y \end{bmatrix} \tag{4.15}$$

$$\overline{p} = \begin{bmatrix} p_p \\ p \\ \tilde{p} \end{bmatrix} \quad X = \begin{bmatrix} x_p \\ x \\ x' \end{bmatrix}$$

and define the adjoint system as in equation (4.14), we obtain

$$\begin{bmatrix} \dot{p}_p \\ \dot{p} \\ \dot{\tilde{p}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ \mu g_x^T(\underline{G}^{-1} - \overline{G}^{-1}) & f_x^T \\ \mu g_{x'}^T(\underline{G}^{-1} - \overline{G}^{-1}) & f_{x'}^T \end{bmatrix} \begin{bmatrix} y_p \\ y \end{bmatrix} \tag{4.16}$$

and

$$\begin{bmatrix} p_p \\ p \\ \tilde{p} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \mu g_{\dot{x}}^T(\underline{G}^{-1} - \overline{G}^{-1}) & f_{\dot{x}}^T \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_p \\ y \end{bmatrix} \tag{4.17}$$

where $\underline{G} = diag[g_i - \underline{g}_i]$ and $\overline{G} = diag[\overline{g}_i - g_i]$. We expand this out and simplify to yield

$$\begin{aligned} \dot{p}_p &= 0 \\ \dot{p} &= \mu g_x^T(\underline{G}^{-1} - \overline{G}^{-1})y_p + f_x^T y \\ \dot{\tilde{p}} &= \mu g_{x'}^T(\underline{G}^{-1} - \overline{G}^{-1})y_p + f_{x'}^T y \end{aligned} \tag{4.18}$$

and

$$\begin{aligned} p_p &= y_p \\ p &= \mu g_{\dot{x}}^T(\underline{G}^{-1} - \overline{G}^{-1})y_p + f_{\dot{x}}^T y \\ \tilde{p} &= 0. \end{aligned} \tag{4.19}$$

The simplifications yield $\tilde{p}(t) = 0$ and therefore $\dot{\tilde{p}}(t) = 0$. Along (t_0, t_f) we take $\underline{z} = \mu\underline{G}^{-1}y_p$, $\overline{z} = \mu\overline{G}^{-1}y_p$ and $z = \underline{z} - \overline{z}$. Then, at each $t \in (t_0, t_f)$ we are left with the

following adjoint system definition:

$$
\begin{aligned}
f &= 0 \\
\dot{p} &= f_x^T y + g_x^T z \\
p &= f_{\dot{x}}^T y + g_{\dot{x}}^T z \\
0 &= f_{x'}^T y + g_{x'}^T z
\end{aligned}
\tag{4.20}
$$

$$
(g_i - \underline{g}_i)\underline{z}_i \;=\; \mu\, y_p, \;\; (\overline{g}_i - g_i)\overline{z}_i \;=\; \mu\, y_p, \;\; z_i = \underline{z}_i - \overline{z}_i, \;\; \forall i
$$

Therefore, we have the original state dynamics, assumed stable along $[t_0, t_f]$, and the complementary adjoint dynamics which are stable along $[t_f, t_0]$.

4.3 Variational Analysis

We see that the system is a two-point boundary value problem due to ϕ^0, ψ^0, ϕ^f and ψ^f. We shall see that the formulation will naturally lead to the initialisation of the states at t_0, if they are not fixed by the problem, and adjoint variables at t_f. To derive the necessary transversality conditions to satisfy the constraints at t_0 and t_f, we adopt a variational approach. Consider a base trajectory $(\dot{x}, x, u, v)$ and a perturbed trajectory $(\dot{\overline{x}}, \overline{x}, \overline{u}, \overline{v})$, where $\overline{x} = x + \delta x$. Then, by Taylor's Theorem, we can write first variations of the cost (4.1):

$$
\Delta P = P_{\dot{x}}\delta\dot{x} + P_x\delta x + P_u\delta u + P_v\delta v + \epsilon_P \tag{4.21}
$$

where $P_{\dot{x}}$, P_x, P_u and P_v are evaluated at the base values. If the perturbations $\delta\dot{x}$, δx, δu and δv from ΔP were all independent, we could simply set the coefficients of the perturbations in (4.21) to zero to yield the necessary conditions of optimality. However, the system constraints (4.2) introduce dependence between the perturbations and therefore we must take these constraints into consideration.

4.3.1 Augmented Cost Functional

Consider first variations of the augmented cost (4.6):

$$
\begin{aligned}
\Delta\hat{P} = P_{\dot{x}}\delta\dot{x} + P_x\delta x + P_u\delta u + P_v\delta v + \delta x_p^{\mathbf{f}} - \nu^0((\underline{\Psi}^0)^{-1} - (\overline{\Psi}^0)^{-1})(\psi_{\dot{x}}^0\delta\dot{x} + \psi_x^0\delta x + \\
\psi_u^0\delta u + \psi_v^0\delta v)^0 - \nu^f((\underline{\Psi}^f)^{-1} - (\overline{\Psi}^f)^{-1})(\psi_{\dot{x}}^f\delta\dot{x} + \psi_x^f\delta x + \psi_u^f\delta u + \psi_v^f\delta v)^f - \\
\mu^0(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + g_u\delta u + g_v\delta v)^0 - \mu^f(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + \\
g_u\delta u + g_v\delta v)^f - \rho(\underline{V}^{-1} - \overline{V}^{-1})^f V_v\delta v.
\end{aligned}
\tag{4.22}
$$

where $\underline{\Psi} = diag[\psi_i - \underline{\psi_i}]$, $\overline{\Psi} = diag[\overline{\psi_i} - \psi_i]$, $\underline{G} = diag[g_i - \underline{g_i}]$, $\overline{G} = diag[\overline{g_i} - g_i]$, $\underline{V} = diag[V(v_i) - \underline{v_i}]$ and $\overline{V} = diag[\overline{v_i} - V(v_i)]$.

Variations of the system equations fall into two categories. Along the horizon (t_0, t_f), we consider variations in $\dot{x}, x$ and v for fixed $\bar{u}$:

$$\begin{aligned} \bar{f}_p &= \tilde{f}_p + \delta\dot{x}_p + \mu e^T(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + g_v\delta v) + \epsilon_{f_p} = 0 \\ \bar{f} &= \tilde{f} + f_{\dot{x}}\delta\dot{x} + f_x\delta x + f_v\delta v + \epsilon_f = 0 \end{aligned} \tag{4.23}$$

where ϵ represents the higher order terms and $\tilde{}$ above a function indicates evaluation at the trajectory $(\dot{x}, x, \overline{u}, v)$.

For the endpoints t_0 and t_f, we consider variations in $\dot{x}, x, v$ and u:

$$\begin{aligned} \bar{f}_p &= \tilde{f}_p + \delta\dot{x}_p + \mu e^T(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + g_u\delta u + g_v\delta v) + \epsilon_{f_p} = 0 \\ \bar{f} &= \tilde{f} + f_{\dot{x}}\delta\dot{x} + f_x\delta x + f_u\delta u + f_v\delta v + \epsilon_f = 0 \\ \overline{\phi}^0 &= \tilde{\phi}^0 + \phi^0_{\dot{x}}\delta\dot{x} + \phi^0_x\delta x + \phi^0_u\delta u + \phi^0_v\delta v + \epsilon^0_\phi = 0 \\ \overline{\phi}^f &= \tilde{\phi}^f + \phi^f_{\dot{x}}\delta\dot{x} + \phi^f_x\delta x + \phi^f_u\delta u + \phi^f_v\delta v + \epsilon^f_\phi = 0 \end{aligned} \tag{4.24}$$

where, again, ϵ represents the higher order terms and $\tilde{}$ above a function indicates evaluation at the trajectory $(\dot{x}, x, u, v)$.

Strictly, we need to prove that these higher order terms are negligible. This requires showing that $\| \delta\dot{x} \|$, $\| \delta x \|$ are $O(\| \delta u' \| + \| \delta v \| + \| \delta x(t_0) \|)$. In the proof, we shall employ the Implicit Function Theorem, Mean Value Theorem and Gronwall's Lemma and therefore include the definitions below for completeness.

Definition 4.3.1. *(Ortega and Rheinboldt, 1970) A mapping $F : D \subset \mathbb{R}^n \to \mathbb{R}^m$ is Gateaux-differentiable (or G-differentiable) at an interior point x of D if there exists a linear operator $A \in L(\mathbb{R}^n, \mathbb{R}^m)$ such that, for any $h \in \mathbb{R}^n$,*

$$\lim_{t \to 0} \frac{1}{t} \| F(x + th) - Fx - tAh \| = 0.$$

Definition 4.3.2. *(Ortega and Rheinboldt, 1970) A mapping $F : D \subset \mathbb{R}^n \to \mathbb{R}^m$ is Frechet-differentiable (or F-differentiable) at an interior point x of D if there exists a linear operator $A \in L(\mathbb{R}^n, \mathbb{R}^m)$ such that, for any $h \in \mathbb{R}^n$.*

$$\lim_{h \to 0} \frac{1}{\| h \|} \| F(x + h) - Fx - Ah \| = 0.$$

Theorem 4.3.1. *Mean Value Theorem (First Order) (Ortega and Rheinboldt, 1970)*

Assume that $F : D \subset \mathbb{R}^n \to \mathbb{R}^m$ is G-differentiable on a convex set $D_0 \subset D$. Then, for any $x,\ y \in D_0$

$$\| F(y) - F(x) \| \leq \sup_{0 \leq t \leq 1} \| F'(x + t(y - x)) \| \, \| x - y \|$$

Theorem 4.3.2. *Implicit Function Theorem (Ortega and Rheinboldt, 1970)*

Suppose that $F : D \subset \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^n$ is continuous on an open neighbourhood $D_0 \subset D$ of a point (x^0, y^0) for which $F(x^0, y^0) = 0$. Assume that F_x exists in a neighbourhood of (x^0, y^0) and is continuous at (x^0, y^0), and that $F_x(x^0, y^0)$ is nonsingular. Then there exist open neighbourhoods $S_1 \subset \mathbb{R}^n$ and $S_2 \subset \mathbb{R}^p$ of x^0 and y^0 respectively, such that for any $y \in \overline{S}_2$, the equation $F(x, y) = 0$ has a unique solution $x = Hy \in \overline{S}_1$ and the mapping $H : S_2 \to \mathbb{R}^n$ is continuous. Moreover, if F_y exists at (x^0, y^0), then H is $F-$differentiable at y^0 and

$$H'(y^0) = -[F_x(x^0, y^0)]^{-1} F_y(x^0, y^0).$$

Lemma 4.3.1. *Gronwall's Lemma (Cronin, 1980)*

If $u(\tau),\ v(\tau)$ and $w(\tau)$ are non-negative functions for $t_0 \leq \tau \leq t$, and if

$$u(t) \leq w(t) + \int_{t_0}^{t} u(\tau) v(\tau)\, d\tau$$

then the following inequality is true

$$u(t) \leq w(t) \exp \int_{t_0}^{t} v(\tau)\, d\tau.$$

4.3.2 Proof that higher order terms in variational expansion are negligible

Consider, for example, the linearised system equations in (4.24). We write $\bar{f} = f(t, \dot{\bar{x}}, \bar{x}, \bar{u}, \bar{v})$ and $\tilde{f} = f(t, \dot{x}, x, u, v)$. Given $\tilde{f}$ and $f_{\dot{x},x'}$ nonsingular at $(\dot{x}, x, u, v)$ and continuous on a neighbourhood, then by the Implicit Function Theorem, there is a unique solution for $(\dot{\bar{x}}, \bar{x}, \bar{u}, \bar{v})$ in this neighbourhood and we may write

$$\left\| \begin{bmatrix} \delta\dot{x} \\ \delta x' \end{bmatrix} \right\| \leq \| (f_{\dot{x},x'})^{-1} \| \{ \| f_x \| \, \| \delta x \| + \| f_{u'} \| \, \| \delta u' \| + \| f_v \| \, \| \delta v \| + \| \bar{f} - \tilde{f} \| \}. \tag{4.25}$$

Applying the Mean Value Theorem, we get

$$\left\| \begin{bmatrix} \delta\dot{x} \\ \delta x' \end{bmatrix} \right\| \leq K(\| \delta x \| + \| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|) \tag{4.26}$$

where $K = \max_{t\in[t_0,t]}\{\| (f_{\dot{x},x'})^{-1} \| (\| f_x \|, \| f_{u'} \|, \| f_v \|, 1)\}$. Therefore, we may write

$$\| \delta\dot{x}(t) \| \leq K(\| \delta x \| + \| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|) \tag{4.27}$$

and

$$\| \delta x(t) \| \leq \| \delta x_0 \| + \int_{t_0}^{t} K(\| \delta x \| + \| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|)\, dt. \tag{4.28}$$

From Gronwall's Lemma we have

$$\begin{aligned} \| \delta x(t) \| &\leq \{\| \delta x_0 \| + \int_{t_0}^{t} K\, (\| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|)\, dt\}\ \exp(\int_{t_0}^{t} K\, dt) \\ &\leq M(t)\exp(K(t-t_0)). \end{aligned} \tag{4.29}$$

where

$$M(t) = \| \delta x_0 \| + K \sup_{t\in[t_0,t]} \{\| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|\}(t - t_0). \tag{4.30}$$

Therefore, for some K', we may bound $\| \delta\dot{x} \|$, $\| \delta x \|$, $\| \delta x' \|$ by

$$K'(\| \delta x_0 \| + \sup_{t\in[t_0,t]} (\| \delta u' \| + \| \delta v \| + \| \bar{f} - \tilde{f} \|)) \tag{4.31}$$

provided $t_f - t_0$ is bounded. Assuming $\bar{f} \to \tilde{f}$ as we approach the optimum, we then have

$$\| \delta x(t) \| = O(\| \delta x_0 \| + \sup_{t\in[t_0,t_f]} \| \delta u' \| + \| \delta v \|) \tag{4.32}$$

and

$$\| \delta\dot{x}(t) \| = O(\| \delta x_0 \| + \sup_{t\in[t_0,t_f]} \| \delta u' \| + \| \delta v \|). \tag{4.33}$$

Again consider the linearised state equation in (4.24),

$$\epsilon_f = f(t, \dot{\bar{x}}, \bar{x}, \bar{u}, \bar{v}) - f(t, \dot{x}, x, u, v) - f_{\dot{x}}\delta\dot{x} - f_x\delta x - f_u\delta u - f_v\delta v \tag{4.34}$$

and therefore, given $\mathcal{X} = \{\dot{x}, x, u, v\}$, we apply the Mean Value Theorem to obtain the following bound on the norm of ϵ_f,

$$\begin{aligned} \| \epsilon_f \| \leq \{ & \| f_{\dot{x}}(t, \dot{\tilde{x}}, x, u, v) - f_{\dot{x}}(t, \dot{x}, x, u, v) \| \| \delta\dot{x} \| + \\ & \| f_x(t, \dot{x}, \tilde{x}, u, v) - f_x(t, \dot{x}, x, u, v) \| \| \delta x \| + \\ & \| f_{u'}(t, \dot{x}, x, \tilde{u}, v) - f_{u'}(t, \dot{x}, x, u, v) \| \| \delta u' \| + \\ & \| f_v(t, \dot{x}, x, u, \tilde{v}) - f_v(t, \dot{x}, x, u, v) \| \| \delta v \|\} \\ \leq & K(\| \delta\dot{x} \| + \| \delta x \| + \| \delta u' \| + \| \delta v \|) \end{aligned} \tag{4.35}$$

where

$$K = \max \max_{\tilde{x} \in [x, \bar{x}]} \{ \| \Delta f_{\dot{x}} \|, \| \Delta f_x \|, \| \Delta f_u \|, \| \Delta f_v \| \} \tag{4.36}$$

and $\| \Delta f_{\dot{x}} \| = \| f_{\dot{x}}(t, \tilde{\dot{x}}, x, u, v) - f_{\dot{x}}(t, \dot{x}, x, u, v) \|$, etc. and all norms are maximum norms over $[t_0, t]$. Finally, using (4.32) and (4.33), we conclude

$$\| \epsilon_f \| = O(\| \delta x_0 \| + \sup_{t \in [t_0, t_f]} \| \delta u' \| + \| \delta v \|). \tag{4.37}$$

■

Similar proofs follow for the remaining equations in (4.24).

4.3.3 Integration of DAE system

We may then integrate this linear DAE system with the help of the adjoint system (4.20). We then have[1]

$$\begin{aligned} \frac{d}{dt}(p^T \delta x + p_p \delta x_p) &= \dot{p}^T \delta x + p^T \delta \dot{x} + \dot{p}_p \delta x_p + p_p \delta \dot{x}_p \\ &= (y^T f_x + z^T g_x)\delta x + (y^T f_{\dot{x}} + z^T g_{\dot{x}})\delta \dot{x} - \\ &\quad p_p(\tilde{f}_p + z^T(g_{\dot{x}} \delta \dot{x} + g_x \delta x + g_v \delta v) + \epsilon_{f_p}) \\ &= y^T(f_{\dot{x}} \delta \dot{x} + f_x \delta x) - z^T g_v \delta v - p_p \tilde{f}_p - p_p \epsilon_{f_p}. \end{aligned} \tag{4.38}$$

Simplifying this expression using the variations from (4.23) yields

$$\frac{d}{dt}(p^T \delta x + p_p \delta x_p) = -y^T(\tilde{f} + f_v \delta v + \epsilon_f) - z^T g_v \delta v - p_p \tilde{f}_p - p_p \epsilon_{f_p}. \tag{4.39}$$

Integrating, we obtain:

$$\begin{aligned} (p_p \delta x_p + p^T \delta x)^f - (p_p \delta x_p + p^T \delta x)^0 = -\int_{t_0}^{t_f} \{ y^T(\tilde{f} + f_v \delta v + \epsilon_f) + \\ p_p \tilde{f}_p + z^T g_v \delta v + p_p \epsilon_{f_p} \} dt. \end{aligned} \tag{4.40}$$

The vanishing of the first variation of the augmented cost function for every admissible system of perturbations occurs if the coefficients of the dependent perturbations vanish (see, for example, Pontryagin *et al.* (1962)). Therefore, by introducing appropriate multipliers, we adjoin the perturbed system equations (4.24) and integrated DAE system

[1] The derivation of the last line of 4.38 makes use of the fact (to be shown later) that $p_p = 1$.

(4.40) to the perturbed augmented cost function and obtain,

$$\begin{aligned}
\Delta\hat{P}^* = {} & P_{\dot{x}}\delta\dot{x} + P_x\delta x + P_u\delta u + P_v\delta v + \delta x_p - \nu^0((\underline{\Psi}^0)^{-1} - (\overline{\Psi}^0)^{-1})(\psi^0_{\dot{x}}\delta\dot{x} + \psi^0_x\delta x + \\
& \psi^0_u\delta u + \psi^0_v\delta v) - \nu^f((\underline{\Psi}^f)^{-1} - (\overline{\Psi}^f)^{-1})(\psi^f_{\dot{x}}\delta\dot{x} + \psi^f_x\delta x + \psi^f_u\delta u + \psi^f_v\delta v) - \\
& \mu^0(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + g_u\delta u + g_v\delta v)^0 - \mu^f(\underline{G}^{-1} - \overline{G}^{-1})(g_{\dot{x}}\delta\dot{x} + g_x\delta x + \\
& g_u\delta u + g_v\delta v)^f - \rho(\underline{V}^{-1} - \overline{V}^{-1})^f V_v\delta v \\
& + \eta^{0T}\{f_{\dot{x}}\delta\dot{x} + f_x\delta x + f_u\delta u + f_v\delta v + \epsilon^0_f\}^0 + \eta^{fT}\{f_{\dot{x}}\delta\dot{x} + f_x\delta x \\
& + f_u\delta u + f_v\delta v + \epsilon^f_f\}^f + \xi^{0T}\{\phi^0_{\dot{x}}\delta\dot{x} + \phi^0_x\delta x + \phi^0_u\delta u + \phi^0_v\delta v + \epsilon^0_\phi\} \\
& + \xi^{fT}\{\phi^f_{\dot{x}}\delta\dot{x} + \phi^f_x\delta x + \phi^f_u\delta u + \phi^f_v\delta v + \epsilon^f_\phi\} \\
& + (p_p\delta x_p + p^T\delta x)^0 - (p_p\delta x_p + p^T\delta x)^f - \int_{t_0}^{t_f}\{y^T(\tilde{f} + f_v\delta v + \epsilon_f) \\
& + p_p\tilde{f}_p + z^T g_v\delta v + p_p\epsilon_{f_p}\}dt.
\end{aligned} \tag{4.41}$$

4.3.4 Transversality Conditions

The contribution to $\Delta\hat{P}^*$ of the perturbations δx_p, $\delta\dot{x}$, δx, δu, δv vanishes if we add common terms in these perturbations and set the resulting coefficients to zero. Again, we write $\underline{z} = \mu\underline{G}^{-1}y_p$, $\overline{z} = \mu\overline{G}^{-1}y_p$ and treat the multipliers on ψ^0, ψ^f, V in a similar fashion. This yields the following problem formulation:

Find $\{\dot{x}(t), x(t), u(t)\}$, $t \in [t_0, t_f]$ satisfying (4.2) as well as the transversality conditions:

4.3.4.1 Initial Transversality Conditions

At $t = t_0$:

$$\begin{aligned}
f^T_x\eta^0 + g^T_x\zeta^0 + \phi^T_x\xi^0 + \psi^T_x\omega^0 + p^0 &= 0 \\
f^T_{\dot{x}}\eta^0 + g^T_{\dot{x}}\zeta^0 + \phi^T_{\dot{x}}\xi^0 + \psi^T_{\dot{x}}\omega^0 &= 0 \\
f^T_u\eta^0 + g^T_u\zeta^0 + \phi^T_u\xi^0 + \psi^T_u\omega^0 &= 0 \\
(\psi^0_i - \underline{\psi}^0_i)\underline{\omega}^0_i = \nu^0,\ (\overline{\psi}^0_i - \psi^0_i)\overline{\omega}^0_i = \nu^0,\ \omega^0_i &= \overline{\omega}^0_i - \underline{\omega}^0_i \\
(g^0_i - \underline{g}^0_i)\underline{\zeta}^0_i = \mu^0,\ (\overline{g}^0_i - g^0_i)\overline{\zeta}^0_i = \mu^0,\ \zeta^0_i &= \overline{\zeta}^0_i - \underline{\zeta}^0_i \\
(v^0_i - \underline{v}^0_i)\underline{\lambda}^0_i = \rho\ ,\ (\overline{v}^0_i - v^0_i)\overline{\lambda}^0_i = \rho\ ,\ \lambda^0_i &= \overline{\lambda}^0_i - \underline{\lambda}^0_i \\
\phi^0 = 0,\ f = 0 &
\end{aligned} \tag{4.42}$$

$$(f_v^T\eta^0 + g_v^T\zeta^0 + \phi_v^T\xi^0 + \psi_v^T\omega^0 + V_v^T\lambda^0) + (f_v^T\eta^f + g_v^T\zeta^f + \phi_v^T\xi^f + \psi_v^T\omega^f + P_v^T)$$
$$= \int_{t_0}^{t_f} (f_v^T y + g_v^T z)\, dt.$$

4.3.4.2 Endpoint Transversality Conditions

At $t = t_f$:

$$\begin{aligned}
1 - p_p &= 0 \\
f_x^T\eta^f + g_x^T\zeta^f + \phi_x^T\xi^f + \psi_x^T\omega^f + P_x^T - p^f &= 0 \\
f_{\dot{x}}^T\eta^f + g_{\dot{x}}^T\zeta^f + \phi_{\dot{x}}^T\xi^f + \psi_{\dot{x}}^T\omega^f + P_{\dot{x}}^T &= 0 \\
f_u^T\eta^f + g_u^T\zeta^f + \phi_u^T\xi^f + \psi_u^T\omega^f + P_u^T &= 0 \qquad (4.43) \\
(\psi_i^f - \underline{\psi}_i^f)\underline{\omega}_i^f = \nu^f,\ (\overline{\psi}_i^f - \psi_i^f)\overline{\omega}_i^f = \nu^f,\ \omega_i^f &= \overline{\omega}_i^f - \underline{\omega}_i^f \\
(g_i^f - \underline{g}_i^f)\underline{\zeta}_i^f = \mu^f,\ (\overline{g}_i^f - g_i^f)\overline{\zeta}_i^f = \mu^f,\ \zeta_i^f &= \overline{\zeta}_i^f - \underline{\zeta}_i^f \\
\phi^f = 0,\ f = 0.
\end{aligned}$$

4.3.5 Path Conditions

After eliminating the terms associated with the dependent perturbations, we are then left with the following more modest cost function:

$$\Delta\hat{P}^* = -\int_{t_0}^{t_f} \{p_p\tilde{f}_p + y^T\tilde{f}\}dt + \epsilon \qquad (4.44)$$

where, again, ϵ represents the negligible higher order terms. Here ν^0, ν^f and $\mu(t)$ are barrier function parameters and $p(t)$ is the adjoint function, while y, z, η, ξ, ω and λ are appropriate multipliers. From this equation we can see that to minimise $\hat{P}^*$ we must choose $\delta u'$ to maximise $p_p\tilde{f}_p + y^T\tilde{f} = \tilde{f}_p + y^T\tilde{f}$ for fixed $\delta\dot{x}, \delta x, \delta x'$. This corresponds to

$$u' = \arg\max\{y^T f + \dot{x}_p + \mu\sum_i \{\ln(g_i - \underline{g}_i) + \ln(\overline{g}_i - g_i)\}\} \qquad (4.45)$$

since $p_p = 1$ from (4.18), (4.19) and (4.43). Therefore, we solve

$$f_{u'}^T y + g_{u'}^T z = 0 \qquad (4.46)$$

for $z = \underline{z} - \overline{z} = \mu(\underline{G}^{-1} - \overline{G}^{-1})$. Note that, in view of the maximisation, the Hessian matrix of (4.45) must be non-positive definite at the optimum with respect to u'. In fact one can show that for $\mu > 0$ sufficiently small, the Hessian matrix is negative definite.

Combining (4.46) with (4.20), we obtain:

$$\begin{aligned} f &= 0 \\ \dot{p} &= f_x^T y + g_x^T z \\ p &= f_{\dot{x}}^T y + g_{\dot{x}}^T z \\ 0 &= f_u^T y + g_u^T z \end{aligned} \tag{4.47}$$

$$(g_i - \underline{g}_i)\underline{z}_i \;=\; \mu \quad , \quad (\overline{g}_i - g_i)\overline{z}_i \;=\; \mu \quad , \quad z_i = \underline{z}_i - \overline{z}_i, \;\; \forall i$$

which must be satisfied at each point along the trajectory. Thus, (4.42), (4.43) and (4.47) together define the necessary optimality conditions for (4.1) and (4.2).

4.3.6 Necessary Conditions in the Limit

In the limit as $\mu \to 0$, we have the following necessary conditions:

$$\begin{aligned} f &= 0 \\ \dot{p} &= f_x^T y + g_x^T z \\ p &= f_{\dot{x}}^T y + g_{\dot{x}}^T z \\ 0 &= f_u^T y + g_u^T z \end{aligned} \tag{4.48}$$

$$(g_i - \underline{g}_i)\underline{z}_i \;=\; 0, \;\; (\overline{g}_i - g_i)\overline{z}_i \;=\; 0, \;\; \underline{z}_i, \; \overline{z}_i \geq 0$$

$$z_i = \underline{z}_i - \overline{z}_i, \;\; \forall i$$

which must be satisfied at each point along the optimal trajectory. If we consider an ODE system in the absence of inequality constraints at t_0, we find that these necessary conditions reduce to those stated in Informal Theorem 4.1 in (Hartl *et al.*, 1995). Hartl *et al.* include jump conditions on the adjoint variable to correspond with activation of pure state constraints. We shall outline an appropriate means of characterising the adjoint in this case in the next chapter.

It is important to note here the current gap in theory with respect to the multipliers associated with constraints. As pointed out in (Hartl *et al.*, 1995), proofs of existence for optimal control problems involving constraints, in particular state constraints, are generally formal in the sense that the existence of the multiplier z associated with a state constraint is assumed and not rigorously established. The difficulty in translating the necessary conditions to an algorithm is establishing a method of determining the precise values of these multipliers.

4.4 Conclusions

We have developed new necessary conditions for nonlinear DAE systems constrained by equality constraints at the initial time and final time as well as by general inequality constraints at the endpoints and along the path. The necessary conditions were developed using a logarithmic barrier function as a means of penalising deviations from the constrained, feasible region. Thus, we have contributed an innovative and practical extension of necessary conditions to constrained DAE systems of a general nature using barrier functions.

The derivation presented in this chapter assumed the problem to be normal. If we were to consider the more general case where the multiplier on the objective P was π, we would have $\pi > 0$ if $\mu > 0$, provided the equality constraints at the endpoints are linearly independent. In the limit as $\mu \to 0$, we would still have $\pi > 0$ if constraint qualifications like (2.7) in (Hartl *et al.*, 1995) hold for both the endpoint equalities and inequalities. This constraint qualification essentially requires that the gradients with respect to the states of the equality constraints and active inequality constraints must be linearly independent. An argument based on the work of (Fiacco and McCormick, 1968) can be used to assert existence of the multipliers in the limit as $\mu \to 0$. Furthermore, in contrast to the informal theorems given in Hartl (1995), our adjoints $p(\cdot)$ are absolutely continuous over $[t_0, t_f]$.

The resulting system is a two point boundary value problem, characterised by very general initial and end specifications. Various methods can be employed to solve this optimal control problem. In the next chapter, we outline the basic formulation of this thesis, which solves an exact maximisation of the Hamiltonian at each time t, iterating on the control.

CHAPTER 5

Algorithm Development

There are many methods which can be used to solve the two point boundary value problem derived in the previous chapter. Among the possibilities, all of which were discussed in some detail in Chapter 2, are collocation; control parameterisation; multiple shooting and a forward-backward approach which iterates either on the adjoint variable $p(t)$ or on the control $u(t)$. We shall adopt the latter approach, analogous to Bryson and Ho (1975), adopting a Newton step rather than steepest descent. We choose to iterate on the control variable, as an initial guess of the control allows for more intuition than the adjoint variable.

The Hamiltonian description of this thesis is unique in its handling of inequality constraints. The approach outlined in this chapter converts these inequalities to equalities by introducing a slack variable which must be strictly interior to the constraint space. Interiority is enforced by using a logarithmic barrier function, as described in the previous chapter. Thus, we augment the cost function with a logarithmic penalty term and transform the original inequality constrained optimal control problem to a sequence of equality constrained optimal control problems. The numerical routine then enforces equality between the constraint function and the slack variable by using an interior point routine. The interior point method will be described in detail.

State equality constraints at the horizon endpoint as well as state inequality constraints along the horizon present challenging problems in determining the optimal trajectory. Means of effectively dealing with both of these cases are discussed in this chapter.

The algorithm deals with index one DAE systems which are assumed to be deterministic descriptions of the system dynamics. Extensions to high index DAEs and uncertain

systems will be discussed in detail in later chapters.

5.1 Introduction of Slack Variables

We transform our inequality constraints g, ψ^0, ψ^f and V to equality constraints by introducing the slack variables s, σ^0, σ^f and r respectively. This generates the following system of equations, modified from (4.2) and (4.6):

$$\min_{u(\cdot),v,s} \hat{P}(t_f, \dot{x}(t_f), x(t_f), u(t_f), v) \tag{5.1}$$

subject to

$$\begin{aligned} f(t, \dot{x}(t), x(t), u(t), v) &= 0 \\ g(t, \dot{x}(t), x(t), u(t), v) - s &= 0 \\ \\ \phi^0(t_0, \dot{x}(t_0), x(t_0), u(t_0), v) &= 0 \\ \psi^0(t_0, \dot{x}(t_0), x(t_0), u(t_0), v) - \sigma^0 &= 0 \\ \\ \phi^f(t_f, \dot{x}(t_f), x(t_f), u(t_f), v) &= 0 \\ \psi^f(t_f, \dot{x}(t_f), x(t_f), u(t_f), v) - \sigma^f &= 0 \\ \\ V(v) - r &= 0 \end{aligned} \tag{5.2}$$

for $t \in [t_0, t_f]$, where $\underline{g} < s < \overline{g}$, $\underline{\psi}^0 < \sigma^0 < \overline{\psi}^0$, $\underline{\psi}^f < \sigma^f < \overline{\psi}^f$, $\underline{v} < r < \overline{v}$ and $\hat{P}$ is as in equation (4.6) of Chapter 4, substituting the slack variables for the constraint values:

$$\begin{aligned} \min_{u(\cdot),v,s} \hat{P} = P + x_p(t_f) - \\ \nu^0 \sum_i \{\ln(\sigma_i^0 - \underline{\psi}_i^0) + \ln(\overline{\psi}_i^0 - \sigma_i^0)\} - \nu^f \sum_i \{\ln(\sigma_i^f - \underline{\psi}_i^f) + \ln(\overline{\psi}_i^f - \sigma_i^f)\} - \\ \mu^0 \sum_i \{\ln(s_i^0 - \underline{g}_i^0) + \ln(\overline{g}_i^0 - s_i^0)\} - \mu^f \sum_i \{\ln(s_i^f - \underline{g}_i^f) + \ln(\overline{g}_i^f - s_i^f)\} - \\ \rho \sum_k \{\ln(r_k - \underline{v}_k) + \ln(\overline{v}_k - r_k)\} \end{aligned} \tag{5.3}$$

The slack variables are required to be strictly interior to the constraint space such that when the equalities introduced in the modified system (5.2) are satisfied. so then are the inequalities of the original system (4.2).

Therefore, we now have the following logarithmic barrier function:

$$f_p = \dot{x}_p + \mu(t) \sum_i \{\ln(s_i(t) - \underline{g}_i(t)) + \ln(\overline{g}_i(t) - s_i(t))\} = 0, \; x_p(t_0) = 0. \tag{5.4}$$

If we consider the following modified primary functions (see Section 4.2):

$$F = \begin{bmatrix} f_p \\ f \\ g_p \end{bmatrix}, \quad \dot{X} = \begin{bmatrix} \dot{x}_p \\ \dot{x} \\ \dot{x}' \\ \dot{s} \end{bmatrix}, \qquad \tilde{y} = \begin{bmatrix} y_p \\ y \\ z \end{bmatrix} \tag{5.5}$$

$$\overline{p} = \begin{bmatrix} p_p \\ p \\ \tilde{p} \\ p_s \end{bmatrix}, \quad X = \begin{bmatrix} x_p \\ x \\ x' \\ s \end{bmatrix}$$

where $g_p = g(t, \dot{x}, x, u, v) - s(t)$, and define the adjoint system as in equation (4.14), we obtain

$$\begin{bmatrix} \dot{p}_p \\ \dot{p} \\ \dot{\tilde{p}} \\ \dot{p}_s \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & f_x^T & g_x^T \\ 0 & f_{x'}^T & g_{x'}^T \\ \mu(\underline{S}^{-1} - \overline{S}^{-1}) & 0 & -I \end{bmatrix} \begin{bmatrix} y_p \\ y \\ z \end{bmatrix} \tag{5.6}$$

and

$$\begin{bmatrix} p_p \\ p \\ \tilde{p} \\ p_s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & f_{\dot{x}}^T & g_{\dot{x}}^T \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_p \\ y \\ z \end{bmatrix} \tag{5.7}$$

where $\underline{S} = diag[s_i - \underline{g}_i]$ and $\overline{S} = diag[\overline{g}_i - s_i]$. We expand this out and simplify to yield

$$\begin{aligned} \dot{p}_p &= 0 \\ \dot{p} &= f_x^T y + g_x^T z \\ \dot{\tilde{p}} &= f_{x'}^T y + g_{x'}^T z \\ \dot{p}_s &= \mu(\underline{S}^{-1} - \overline{S}^{-1}) y_p - z \end{aligned} \tag{5.8}$$

and

$$\begin{aligned} p_p &= y_p \\ p &= f_{\dot{x}}^T y + g_{\dot{x}}^T z \\ \tilde{p} &= 0 \\ p_s &= 0. \end{aligned} \tag{5.9}$$

The simplifications yield $\tilde{p}(t) = 0$ and therefore $\dot{\tilde{p}}(t) = 0$. Furthermore, $p_s = 0$ and therefore $z = \mu(\underline{S}^{-1} - \overline{S}^{-1})y_p$. At each $t \in (t_0, t_f)$ we are left with the following adjoint system definition:

$$f = 0,\ g - s = 0$$

$$\dot{p} = f_x^T y + g_x^T z$$

$$p = f_{\dot{x}}^T y + g_{\dot{x}}^T z \tag{5.10}$$

$$0 = f_{x'}^T y + g_{x'}^T z$$

$$(s_i - \underline{g}_i)\underline{z}_i \ = \ \mu\, y_p,\ \ (\overline{g}_i - s_i)\overline{z}_i \ = \ \mu\, y_p,\ \ z_i = \underline{z}_i - \overline{z}_i,\ \ \forall i$$

Therefore, we have the original state dynamics, assumed stable along $[t_0, t_f]$, and the complementary adjoint dynamics which are stable along $[t_f, t_0]$.

5.1.1 Modified Transversality Conditions

The presence of slack variables results in slight modifications to the transversality and path conditions of Chapter 4 and yields the following problem formulation:

Find $\{\dot{x}(t), x(t), u(t)\},\ t \in [t_0, t_f]$ satisfying (5.2) as well as the transversality conditions:

5.1.1.1 Modified Initial Transversality Conditions

At $t = t_0$:

$$\begin{aligned}
f_x^T \eta^0 + g_x^T \zeta^0 + \phi_x^T \xi^0 + \psi_x^T \omega^0 + p^0 &= 0 \\
f_{\dot{x}}^T \eta^0 + g_{\dot{x}}^T \zeta^0 + \phi_{\dot{x}}^T \xi^0 + \psi_{\dot{x}}^T \omega^0 &= 0 \\
f_u^T \eta^0 + g_u^T \zeta^0 + \phi_u^T \xi^0 + \psi_u^T \omega^0 &= 0 \qquad (5.11a) \\
(\sigma_i^0 - \underline{\psi}_i^0)\underline{\omega}_i^0 \ = \ \nu^0,\ \ (\overline{\psi}_i^0 - \sigma_i^0)\overline{\omega}_i^0 \ = \ \nu^0,\ \ \omega_i^0 &= \overline{\omega}_i^0 - \underline{\omega}_i^0 \\
(s_i^0 - \underline{g}_i^0)\underline{\zeta}_i^0 \ = \mu^0,\ \ (\overline{g}_i^0 - s_i^0)\overline{\zeta}_i^0 \ = \mu^0,\ \ \zeta_i^0 &= \overline{\zeta}_i^0 - \underline{\zeta}_i^0 \\
(r_i^0 - \underline{v}_i^0)\underline{\lambda}_i^0 \ = \rho\ ,\ \ (\overline{v}_i^0 - r_i^0)\overline{\lambda}_i^0 \ = \rho\ ,\ \ \lambda_i^0 &= \overline{\lambda}_i^0 - \underline{\lambda}_i^0 \\
\phi^0 = 0,\ \ \psi^0 = \sigma^0,\ \ f = 0,\ \ g = s,\ \ v &= r
\end{aligned}$$

$$(f_v^T\eta^0 + g_v^T\zeta^0 + \phi_v^T\xi^0 + \psi_v^T\omega^0 + V_v^T\lambda) + (f_v^T\eta^f + g_v^T\zeta^f + \phi_v^T\xi^f + \psi_v^T\omega^f + P_v^T) = \int_{t_0}^{t_f}(f_v^T y + g_v^T z)\,dt. \tag{5.11b}$$

5.1.1.2 Modified Endpoint Transversality Conditions

At $t = t_f$:

$$\begin{aligned}
1 - p_p &= 0 \\
f_x^T\eta^f + g_x^T\zeta^f + \phi_x^T\xi^f + \psi_x^T\omega^f + P_x^T - p^f &= 0 \\
f_{\dot{x}}^T\eta^f + g_{\dot{x}}^T\zeta^f + \phi_{\dot{x}}^T\xi^f + \psi_{\dot{x}}^T\omega^f + P_{\dot{x}}^T &= 0 \\
f_u^T\eta^f + g_u^T\zeta^f + \phi_u^T\xi^f + \psi_u^T\omega^f + P_u^T &= 0 \qquad (5.12) \\
(\sigma_i^f - \underline{\psi}_i^f)\underline{\omega}_i^f = \nu^f,\ (\overline{\psi}_i^f - \sigma_i^f)\overline{\omega}_i^f = \nu^f,\ \omega_i^f &= \overline{\omega}_i^f - \underline{\omega}_i^f \\
(s_i^f - \underline{g}_i^f)\underline{\zeta}_i^f = \boldsymbol{\mu}^f,\ (\overline{g}_i^f - s_i^f)\overline{\zeta}_i^f = \boldsymbol{\mu}^f,\ \zeta_i^f &= \overline{\zeta}_i^f - \underline{\zeta}_i^f \\
\phi^f = 0,\ \psi^f = \sigma^f,\ f = 0,\ g = s.
\end{aligned}$$

5.1.2 Modified Path Conditions

As before, we eliminate the terms associated with the dependent perturbations, leaving the following more modest cost function:

$$\Delta\hat{P}^* = -\int_{t_0}^{t_f}\{p_p\tilde{f}_p + y^T\tilde{f}\}dt + \epsilon \tag{5.13}$$

where, again, ϵ represents the negligible higher order terms. Here s, σ^0, σ^f, r are slack variables, ν^0, ν^f and $\mu(t)$ are barrier function parameters and $p(t)$ is the adjoint function, while y, z, η, ξ, ω and λ are appropriate multipliers. Thus, to minimise $\hat{P}^*$ we must choose $\delta u'$ to maximise $p_p\tilde{f}_p + y^T\tilde{f} = \tilde{f}_p + y^T\tilde{f}$ for fixed $\delta\dot{x}, \delta x, \delta x'$. This corresponds to

$$(u', s) = \arg\max\{y^T f + \dot{x}_p + \mu\sum_i\{\ln(s_i - \underline{g}_i) + \ln(\overline{g}_i - s_i)\}|\ g_i = s_i\} \tag{5.14}$$

since $p_p = 1$ from (5.8), (5.9) and (5.12). Therefore, we solve

$$f_{u'}^T y + g_{u'}^T z = 0. \tag{5.15}$$

The corresponding necessary conditions then become:

$$f = 0,\ g - s = 0$$

$$\dot{p} = f_x^T y + g_x^T z$$

$$p = f_{\dot{x}}^T y + g_{\dot{x}}^T z \tag{5.16}$$

$$0 = f_u^T y + g_u^T z$$

$$(s_i - \underline{g}_i)\underline{z}_i \ = \ \mu,\ \ (\overline{g}_i - s_i)\overline{z}_i \ = \ \mu,\ \ z_i = \underline{z}_i - \overline{z}_i.\ \ \forall i.$$

5.2 The Basic Algorithm

The basic algorithm has the form:

1. Given $u'(t), t \in (t_0, t_f)$ and v, set $p(t_0) = 0$.

2. Determine initial conditions $\dot{x}(t_0), x(t_0)$ and $u(t_0)$ from (5.11a).

3. Integrate $f(t, \dot{x}, x, u, v) = 0$ from t_0 to t_f, obtaining $(\dot{x}, x, x')$.

4. Determine final condition $p(t_f)$ from (5.12).

5. Integrate (5.16) from t_f to t_0, obtaining $(u, s, p, \dot{x})$ for fixed (x, v).

6. Update v from (5.11b).

7. Repeat from step 2 to convergence.

The integration of the DAE systems in steps 3 and 5 may be accomplished using DASOLV (Jarvis and Pantelides, 1992). DASOLV is a general-purpose package for the solution of large, sparse, mixed systems of differential and algebraic equations. It uses the backwards differentiation formulae (BDF) of orders 1 to 5, consisting of a predictor and a corrector step. The $k-$step BDF method on the n^{th} integration step can be represented as:

$$\dot{x}_n = \frac{1}{h_n \beta_k'}(\alpha_k' x_n + \gamma_{k,n}') \tag{5.17}$$

where

$$\gamma_{k,n}' = \sum_{j=0}^{k-1} \alpha_j' x_{n+j-k}. \tag{5.18}$$

When this is substituted into the DAE, the equation becomes

$$f(t, \frac{1}{h_n \beta_k'}(\alpha_k' x_n + \gamma_{k,n}'), x_n, u_n, v) = 0 \tag{5.19}$$

on step n. This transforms the DAE system into a set of algebraic equations which may be solved using a modified Newton method. The predictor step obtains values for the variables at the new position using an extrapolation of the polynomial interpolating past values of the variables. The corrector phase iteratively solves the set of nonlinear algebraic equations (5.19) using a modified Newton method. The difference between the predictor step and the corrector step is used to estimate the local truncation error on each step. DASOLV offers the option of estimating the local error or the normal error (Jarvis and Pantelides, 1991), with the latter being a more appropriate measure of error for systems with fast transients.

Note that this basic algorithm assumes that changes in the state dynamics at step 3 as a result of implementing the new control are within an allowable level with respect to the constraints present at each t. This point will be discussed in more detail in Section 5.9. Furthermore, the presence of equality constraints at t_f results in (5.12) being underdetermined and, hence, one can not immediately solve for $p(t_f)$. This will be discussed in Section 5.8.

5.3 Inequality Constraints and The Interior Point Method

We use an interior point algorithm to solve the subproblems in steps 2, 4 and 5, which essentially uses the damped Newton method, also reducing the barrier parameter at each iteration. The interior point iteration computes the step:

$$\mathcal{X}_{k+1} = \mathcal{X}_k + \alpha_k \Delta \mathcal{X}_k \tag{5.20}$$

where $\mathcal{X}_k$ depends on the system of equations we are solving and $\Delta\mathcal{X}_k$ is obtained by a linearisation of the system at $\mathcal{X}_k$. The parameter $\alpha_k \in (0, 1]$, is determined such that we reduce the norm of the residuals of the system while satisfying the interiority condition. Imbedding this interior point iteration within DASOLV requires moderate changes in the corrector phase. The corrector computes a modified Newton step such that the residuals of the DAE system are within an acceptable tolerance. The interior point iteration damps each corrector step to maintain interiority.

5.3.1 Determining $\Delta\mathcal{X}_k$

In order to determine our descent direction at each $t \in (t_0, t_f)$, we linearise the system (5.16) about $(\dot{x}, p, u, s, y, z)$ for fixed x, recalling $u = [u', x']$. From the previous section we know that DASOLV parameterises $\dot{p}$ as a function of p. This yields:

$$\begin{bmatrix} f_x^T \\ f_{\dot{x}}^T \\ f_u^T \end{bmatrix} y + \begin{bmatrix} g_x^T \\ g_{\dot{x}}^T \\ g_u^T \end{bmatrix} z - \begin{bmatrix} \frac{1}{h\beta'}(\alpha' p + \gamma') \\ p \\ 0 \end{bmatrix} + \begin{bmatrix} f_x^T \\ f_{\dot{x}}^T \\ f_u^T \end{bmatrix} \Delta y + \begin{bmatrix} g_x^T \\ g_{\dot{x}}^T \\ g_u^T \end{bmatrix} \Delta z$$
$$- \begin{bmatrix} \frac{\alpha'}{h\beta'} I \\ I \\ 0 \end{bmatrix} \Delta p + H_{\mathcal{K}\Upsilon} \begin{bmatrix} \Delta\dot{x} \\ \Delta u \end{bmatrix} = 0 \tag{5.21}$$

where $H_{\mathcal{K}\Upsilon} = \sum_i y^i f^i_{\mathcal{K}\Upsilon} + \sum_i z^i g^i_{\mathcal{K}\Upsilon}$, $\Upsilon = (\dot{x}, u)$, $\mathcal{K} = (x, \Upsilon)$ and

$$f + f_{\dot{x}}\Delta\dot{x} + f_u \Delta u = 0$$

$$g - s + g_{\dot{x}}\Delta\dot{x} + g_u\Delta u - \Delta s = 0$$

$$\underline{S}\ \underline{z} - \gamma\mu e + \underline{Z}\Delta s + \underline{S}\Delta\underline{z} = 0$$

$$\overline{S}\ \overline{z} - \gamma\mu e - \overline{Z}\Delta s + \overline{S}\Delta\overline{z} = 0. \tag{5.22}$$

Note that μ has been multiplied by an additional parameter γ in the equations relating the slack variable s to its associated multiplier z. This parameter is a target reduction parameter for μ and is associated with the convergence of the interior point method. It will be discussed in Section 5.3.3.

Solving (5.22) for Δs and $\Delta z = \Delta\underline{z} - \Delta\overline{z}$ and substituting into (5.21) yields the following square system on (t_0, t_f):

$$\begin{bmatrix} H_{\mathcal{K}\Upsilon} & -I_p & f_{\mathcal{K}}^T \\ f_\Upsilon & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\Upsilon \\ \Delta p \\ \Delta y \end{bmatrix} = - \begin{bmatrix} r \\ 0 \end{bmatrix} \tag{5.23}$$

where

$$\Upsilon = (\dot{x}, u),\ \mathcal{K} = (x, \Upsilon),\ \overline{p}^T = [\frac{1}{h\beta'}(\alpha' p + \gamma'), p, 0]$$

$$I_p^T = [\frac{\alpha'}{h\beta'} I, I, 0]$$

$$r = f_{\mathcal{K}}^T y + g_{\mathcal{K}}^T z - \overline{p} - g_{\mathcal{K}}^T\{\underline{S}^{-1}\underline{r} - \overline{S}^{-1}\overline{r} + (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})(g - s)\}$$

$$H_{\mathcal{K}\Upsilon} = \sum_j y^j f^j_{\mathcal{K}\Upsilon} + \sum_j z^j g^j_{\mathcal{K}\Upsilon} - g_{\mathcal{K}}^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) g_\Upsilon$$

$$H_{\Upsilon\Upsilon} = \sum_j y^j f^j_{\Upsilon\Upsilon} + \sum_j z^j g^j_{\Upsilon\Upsilon} - g_{\Upsilon}^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) g_\Upsilon \tag{5.24}$$

$$\underline{r} = \underline{S}\ \underline{z} - \gamma\mu e,\ \overline{r} = \overline{S}\overline{z} - \gamma\mu e$$

$$\underline{S} = diag[s_i - \underline{g}_i],\ \overline{S} = diag[\overline{g}_i - s_i]$$

$$\underline{Z} = diag[\underline{z}_i],\ \overline{Z} = diag[\overline{z}_i]$$

together with

$$\Delta s = g - s + g_\Upsilon \Delta\Upsilon$$

$$\Delta\underline{z} = -\underline{S}^{-1}(\underline{r} + \underline{Z}\Delta s),\ \Delta\overline{z} = -\overline{S}^{-1}(\overline{r} - \overline{Z}\Delta s),\ \Delta z = \Delta\underline{z} - \Delta\overline{z}. \tag{5.25}$$

We require that f_Υ has full rank and $H_{\Upsilon\Upsilon}$ is negative definite. This ensures that the coefficient matrix in (5.23) is nonsingular for h sufficiently small, where h is the time step. Hence we can solve (5.23) for $\Delta\Upsilon$, Δp, Δy and obtain Δs, $\Delta\underline{z}$, $\Delta\overline{z}$, Δz from (5.25). Thus

$$\Delta\mathcal{X}_k = (\Delta\dot{x}_k,\ \Delta p_k,\ \Delta u_k,\ \Delta y_k,\ \Delta s_k,\ \Delta\underline{z}_k,\ \Delta\overline{z}_k) \tag{5.26}$$

is uniquely defined by (5.23) and (5.25). If the coefficient matrix in (5.23) is singular, we set

$$H_{\mathcal{K}\Upsilon} = \begin{bmatrix} -\rho I - g_x^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) g_\Upsilon \\ -g_\Upsilon^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) g_\Upsilon \end{bmatrix} \tag{5.27}$$

for some small $\rho > 0$.

We pause here to justify our claim that f_Υ full rank and $H_{\Upsilon\Upsilon}$ negative definite ensure that the coefficient matrix is nonsingular.

Lemma 5.3.1. *Given equation (5.23). Write*

$$A = \begin{bmatrix} H_{\mathcal{K}\Upsilon} & -I_p & f_{\mathcal{K}}^T \\ f_\Upsilon & 0 & 0 \end{bmatrix}. \tag{5.28}$$

If f_Υ has full rank and $H_{\Upsilon\Upsilon}$ is negative definite, then A is nonsingular for sufficiently small h, where h is the time step.

Proof:

If A is singular, it has a non-zero eigenvector satisfying

$$\begin{aligned} H_{x\Upsilon}\Delta\Upsilon - \frac{\alpha'}{h\beta'}\Delta p + f_x^T\Delta y &= 0 \\ H_{\dot{x}\Upsilon}\Delta\Upsilon - \Delta p + f_{\dot{x}}^T\Delta y &= 0 \\ H_{u\Upsilon}\Delta\Upsilon + f_u^T\Delta y &= 0 \\ f_\Upsilon\Delta\Upsilon &= 0. \end{aligned} \tag{5.29}$$

Eliminating Δp for small h, the correction terms are then $O(h, \| \Delta\Upsilon, \Delta y \|)$:

$$\begin{aligned} H_{\Upsilon\Upsilon}\Delta\Upsilon + f_\Upsilon^T\Delta y &= O(h, \| \Delta\Upsilon, \Delta y \|) \\ f_\Upsilon\Delta\Upsilon &= 0. \end{aligned} \tag{5.30}$$

Since $f_\Upsilon\Delta\Upsilon = 0$, we have $\Delta\Upsilon^T f_\Upsilon^T\Delta y = (f_\Upsilon\Delta\Upsilon)^T\Delta y = 0$. Therefore

$$\Delta\Upsilon^T H_{\Upsilon\Upsilon}\Delta\Upsilon + \Delta\Upsilon^T f_\Upsilon^T\Delta y = \Delta\Upsilon^T H_{\Upsilon\Upsilon}\Delta\Upsilon = O(h, \| \Delta\Upsilon \|^2). \tag{5.31}$$

But,

$$H_{\Upsilon\Upsilon} < 0 \Rightarrow \Delta\Upsilon^T H_{\Upsilon\Upsilon}\Delta\Upsilon = \Omega(\| \Delta\Upsilon \|^2). \tag{5.32}$$

This is a contradiction and, therefore, A is nonsingular for sufficiently small h.

■

At $t = t_0$, we must solve for the initial state if it is undetermined. We linearise the system equations (5.11) about $(\dot{x}, x, u, v)$ and substitute for Δs, $\Delta\sigma$ and the corresponding multipliers $\Delta\zeta$, $\Delta\omega$, as in the previous case.

The interior point iteration for (5.11) is:

$$\begin{bmatrix} H^0_{\Upsilon\Upsilon} & F_\Upsilon^T \\ F_\Upsilon & 0 \end{bmatrix} \begin{bmatrix} \Delta\Upsilon \\ \Delta\vartheta \end{bmatrix} = - \begin{bmatrix} r_0 \\ 0 \end{bmatrix} \tag{5.33}$$

where

$$F = (f, \phi),\ \Upsilon = (\dot{x}, x, u, v),\ \vartheta = (\eta, \xi),\ \overline{p} = [0, p, 0, \tilde{p}]'$$

$$\tilde{p} = (f_v^T \eta + g_v^T \zeta + \phi_v^T \xi + \psi_v^T \omega + P_v^T)^f - \int_{t_0}^{t_f} (f_v^T y + g_v^T z)\, dt$$

$$r_0 = F_\Upsilon^T \vartheta + g_\Upsilon^T \zeta + \psi_\Upsilon^T \omega + \overline{p} + g_\Upsilon^T \{\underline{S}^{-1}\underline{r} - \overline{S}^{-1}\overline{r} + (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})(g - s)\} +$$
$$\psi_\Upsilon^T \{\underline{\Sigma}^{-1}\underline{r}_\sigma - \overline{\Sigma}^{-1}\overline{r}_\sigma + (\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})(\psi - \sigma)\} +$$
$$V_\Upsilon^T \{\underline{R}^{-1}\underline{r}_r - \overline{R}^{-1}\overline{r}_r + (\underline{R}^{-1}\underline{\Lambda} + \overline{R}^{-1}\overline{\Lambda})(V - r)\}$$

$$\overset{o}{H}_{\Upsilon\Upsilon} = \textstyle\sum_j \vartheta^j F_{\Upsilon\Upsilon}^j + \sum_j \zeta^j g_{\Upsilon\Upsilon}^j + \sum^j \omega^j \psi_{\Upsilon\Upsilon}^j + g_\Upsilon^T (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) g_\Upsilon +$$
$$\psi_\Upsilon^T (\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})\psi_\Upsilon + V_\Upsilon^T (\underline{R}^{-1}\underline{\Lambda} + \overline{R}^{-1}\overline{\Lambda}) V_\Upsilon$$

$$\underline{r} = \underline{S}\underline{\zeta} - \gamma\mu e,\ \overline{r} = \overline{S}\ \overline{\zeta} - \gamma\mu e$$

$$\underline{S} = diag[s_i - \underline{g}_i],\ \overline{S} = diag[\overline{g}_i - s_i] \tag{5.34}$$

$$\underline{Z} = diag[\underline{\zeta}_i],\ \overline{Z} = diag[\overline{\zeta}_i]$$

$$\underline{r}_\sigma = \underline{\Sigma\omega} - \gamma\mu e,\ \overline{r}_\sigma = \overline{\Sigma}\overline{\omega} - \gamma\mu e$$

$$\underline{\Sigma} = diag[\sigma_i - \underline{\psi}_i],\ \overline{\Sigma} = diag[\overline{\psi}_i - \sigma_i]$$

$$\underline{\Omega} = diag[\underline{\omega}_i],\ \overline{\Omega} = diag[\overline{\omega}_i]$$

$$\underline{r}_r = \underline{R\lambda} - \gamma\mu e,\ \overline{r}_r = \overline{R\lambda} - \gamma\mu e$$

$$\underline{R} = diag[r_i - \underline{v}_i],\ \overline{R} = diag[\overline{v}_i - r_i]$$

$$\underline{\Lambda} = diag[\underline{\lambda}_i],\ \overline{\Lambda} = diag[\overline{\lambda}_i]$$

together with

$$\Delta s = g - s + g_\Upsilon \Delta\Upsilon$$

$$\Delta\underline{\zeta} = -\underline{S}^{-1}(\underline{r} + \underline{Z}\Delta s),\ \Delta\overline{\zeta} = -\overline{S}^{-1}(\overline{r} - \overline{Z}\Delta s),\ \Delta\zeta = \Delta\overline{\zeta} - \Delta\underline{\zeta}$$

$$\Delta\sigma = \psi - \sigma + \psi_\Upsilon \Delta\Upsilon$$

$$\Delta\underline{\omega} = -\underline{\Sigma}^{-1}(\underline{r}_\sigma + \underline{\Omega}\Delta\sigma),\ \Delta\overline{\omega} = -\overline{\Sigma}^{-1}(\overline{r}_\sigma - \overline{\Omega}\Delta\sigma),\ \Delta\omega = \Delta\overline{\omega} - \Delta\underline{\omega} \tag{5.35}$$

$$\Delta r = V - r + V_\Upsilon \Delta\Upsilon$$

$$\Delta\underline{\lambda} = -\underline{R}^{-1}(\underline{r_r} + \underline{\Lambda}\Delta r),\ \Delta\overline{\lambda} = -\overline{R}^{-1}(\overline{r_r} - \overline{\Lambda}\Delta r),\ \Delta\lambda = \Delta\overline{\lambda} - \Delta\underline{\lambda}.$$

We require that F_Υ has full rank and that $\overset{c}{H}_{\Upsilon\Upsilon}$ is positive definite. This ensures that the coefficient matrix in (5.33) is nonsingular (a proof similar to that for Lemma 5.3.1 shows this to be true). We can thus solve (5.33) for $\Delta\Upsilon$, $\Delta\vartheta$ and obtain Δs, $\Delta\underline{\zeta}$, $\Delta\overline{\zeta}$, $\Delta\zeta$,

$\Delta\sigma$, $\Delta\underline{w}$, $\Delta\overline{w}$, Δw, Δr, $\Delta\underline{\lambda}$, $\Delta\overline{\lambda}$, $\Delta\lambda$ from (5.35). Therefore

$$\Delta\mathcal{X}_k = (\Delta\dot{x}_k, \Delta u_k, \Delta v_k, \Delta\eta_k, \Delta\xi_k, \Delta s_k, \Delta\underline{\zeta}_k, \Delta\overline{\zeta}_k, \Delta\sigma_k, \Delta\underline{\omega}_k, \Delta\overline{\omega}_k, \Delta r, \Delta\underline{\lambda}_k, \Delta\overline{\lambda}_k) \tag{5.36}$$

is uniquely defined by (5.33) and (5.35). If the coefficient matrix in (5.33) is singular, we set

$$\overset{o}{H}_{\Upsilon\Upsilon} = g_\Upsilon^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})g_\Upsilon + \psi_\Upsilon^T(\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})\psi_\Upsilon + V_\Upsilon^T(\underline{R}^{-1}\underline{\Lambda} + \overline{R}^{-1}\overline{\Lambda})V_\Upsilon. \tag{5.37}$$

Finally, the system of equations for the endpoint transversality conditions (5.12) are:

$$\begin{bmatrix} \overset{f}{H}_{\Upsilon\Upsilon} & F_\Upsilon^T \\ F_\Upsilon & 0 \end{bmatrix} \begin{bmatrix} \Delta\Upsilon \\ \Delta\vartheta \end{bmatrix} = - \begin{bmatrix} r_f \\ 0 \end{bmatrix} \tag{5.38}$$

where

$$F = (f, \phi),\ \Upsilon = (\dot{x}, u),\ \vartheta = [\eta, \xi]',\ \overline{p} = [0, p]'$$

$$r_f = F_\Upsilon^T\vartheta + g_\Upsilon^T\zeta + \psi_\Upsilon^T\omega + \overline{p} + g_\Upsilon^T\{\underline{S}^{-1}\underline{r} - \overline{S}^{-1}\overline{r} + (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})(g - s)\} +$$

$$\psi_\Upsilon^T\{\underline{\Sigma}^{-1}\underline{r}_\sigma - \overline{\Sigma}^{-1}\overline{r}_\sigma + (\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})(\psi - \sigma)\}$$

$$\overset{f}{H}_{\Upsilon\Upsilon} = \textstyle\sum_j \vartheta^j F^j_{\Upsilon\Upsilon} + \sum_j \zeta^j g^j_{\Upsilon\Upsilon} + \sum^j \omega^j \psi^j_{\Upsilon\Upsilon} + g_\Upsilon^T(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})g_\Upsilon +$$

$$\psi_\Upsilon^T(\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})\psi_\Upsilon$$

$$\underline{r} = \underline{S}\underline{\zeta} - \gamma\mu e,\ \overline{r} = \overline{S\zeta} - \gamma\mu e$$

$$\underline{S} = diag[s_i - \underline{g}_i],\ \overline{S} = diag[\overline{g}_i - s_i] \tag{5.39}$$

$$\underline{Z} = diag[\underline{\zeta}_i],\ \overline{Z} = diag[\overline{\zeta}_i]$$

$$\underline{r}_\sigma = \underline{\Sigma\omega} - \gamma\mu e,\ \overline{r}_\sigma = \overline{\Sigma}\overline{\omega} - \gamma\mu e$$

$$\underline{\Sigma} = diag[\sigma_i - \underline{\psi}_i],\ \overline{\Sigma} = diag[\overline{\psi}_i - \sigma_i]$$

$$\underline{\Omega} = diag[\underline{\omega}_i],\ \overline{\Omega} = diag[\overline{\omega}_i]$$

together with

$$\Delta s = g - s + g_\Upsilon\Delta\Upsilon$$

$$\Delta\underline{\zeta} = -\underline{S}^{-1}(\underline{r} + \underline{Z}\Delta s),\ \Delta\overline{\zeta} = -\overline{S}^{-1}(\overline{r} - \overline{Z}\Delta s),\ \Delta\zeta = \Delta\overline{\zeta} - \Delta\underline{\zeta}$$

$$\Delta\sigma = \psi - \sigma + \psi_\Upsilon\Delta\Upsilon \tag{5.40}$$

$$\Delta\underline{\omega} = -\underline{\Sigma}^{-1}(\underline{r_\sigma} + \underline{\Sigma}\Delta\sigma),\ \Delta\overline{\omega} = -\overline{\Sigma}^{-1}(\overline{r_\sigma} - \overline{W}\Delta s),\ \Delta\omega = \Delta\overline{\omega} - \Delta\underline{\omega}.$$

We require that F_Υ has full rank and that $H^f_{\Upsilon\Upsilon}$ is positive definite. This ensures that the coefficient matrix in (5.38) is nonsingular (a proof similar to that for Lemma (5.3.1) shows this to be true). Therefore we can solve (5.38) for $\Delta\Upsilon$. $\Delta\nu$ and obtain Δs, $\Delta\underline{\zeta}$, $\Delta\overline{\zeta}$, $\Delta\zeta$, $\Delta\sigma$, $\Delta\underline{w}$, $\Delta\overline{w}$, Δw from (5.40). Thus

$$\Delta\mathcal{X}_k = (\Delta\dot{x}_k, \Delta u_k, \Delta\nu_k, \Delta\xi_k, \Delta s_k, \Delta\underline{\zeta}_k, \Delta\overline{\zeta}_k, \Delta\sigma_k, \Delta\underline{\omega}_k, \Delta\overline{\omega}_k) \tag{5.41}$$

is uniquely defined by (5.38) and (5.40). If the coefficient matrix in (5.38) is singular, we set

$$H^f_{\Upsilon\Upsilon} = g^T_\Upsilon(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})g_\Upsilon + \psi^T_\Upsilon(\underline{\Sigma}^{-1}\underline{\Omega} + \overline{\Sigma}^{-1}\overline{\Omega})\psi_\Upsilon. \tag{5.42}$$

5.3.2 Determining the Damping Factor α_k

Having determined a Newton step, we must then dampen this step to reduce the residuals and maintain interiority. Hence α_k in (5.20) is chosen to minimise:

$$\mu_{k+1} = \{\underline{z}^T_{k+1}(s_{k+1} - \underline{g}) + \overline{z}^T_{k+1}(\overline{g} - s_{k+1})\}/2n_i,\ s \in \mathbb{R}^{n_i} \tag{5.43}$$

subject to:

$$\begin{gathered}\underline{S}_{k+1}\underline{z}_{k+1} \geq \beta_{k+1}\mu_{k+1}e,\ \overline{S}_{k+1}\overline{z}_{k+1} \geq \beta_{k+1}\mu_{k+1}e \\ \mu_{k+1}/\mu_k \geq \beta_{k+1} \parallel \tilde{r}_{k+1} \parallel_\infty / \parallel \tilde{r}_k \parallel_\infty\end{gathered} \tag{5.44}$$

where $\tilde{r} = (f^T_\Upsilon y + g^T_\Upsilon z - \overline{p},\ \ g - s,\ \ f)$. The choice of parameters β_k, γ_k and other details of the interior point method will be discussed in the next section.

Using (5.20) and (5.22), we may write (5.43) and (5.44) as :

$$\begin{aligned}\mu_{k+1} &= \mu_k(1-\alpha_k) + \alpha_k\ \gamma_k\ \mu_k + \alpha^2_k\{\Delta\underline{z}^T_k\Delta\underline{s}_k + \Delta\overline{z}^T_k\Delta\overline{s}_k\}/2n_i \\ \underline{z}^i_{k+1}\underline{s}^i_{k+1} &= \underline{z}^i_k\underline{s}^i_k(1-\alpha_k) + \alpha_k\ \gamma_k\ \mu_k + \alpha^2_k\Delta\underline{z}^i_k\Delta\underline{s}^i_k \\ \overline{z}^i_{k+1}\overline{s}^i_{k+1} &= \overline{z}^i_k\overline{s}^i_k(1-\alpha_k) + \alpha_k\ \gamma_k\ \mu_k + \alpha^2_k\Delta\overline{z}^i_k\Delta\overline{s}^i_k.\end{aligned} \tag{5.45}$$

We wish to minimise μ_{k+1} with respect to α_k and therefore we simply apply the calculus of variations to our expansion of μ_{k+1} and obtain

$$\bar{\alpha}_k = \frac{n_i\mu_k(1-\gamma_k)}{\Delta\overline{z}^T_k\Delta\overline{s}_k + \Delta\underline{z}^T_k\Delta\underline{s}_k}, \tag{5.46}$$

provided the denominator is greater than zero.

We are constrained to take a step along $\Delta\mathcal{X}_k$ no larger than $\Delta\mathcal{X}_k$. Thus, we must ensure that a minimising $\tilde{\alpha}_k$ is less than or equal to 1. Therefore, we take:

$$\tilde{\alpha}_k = \min\{\bar{\alpha}_k, 1\}. \tag{5.47}$$

However, with $\tilde{\alpha}_k$ there is no guarantee that the inequalities in equation (5.44) will be satisfied. We therefore use the Armijo rule to determine the final value of this damping parameter as $\alpha_k = \theta^k \tilde{\alpha}_k$, where $\theta \in (0,1)$ and k is the smallest integer such that both (5.43) and (5.44) are satisfied. In most cases this method was sufficient for determining the optimal controls and optimisation parameters. However, a more exact method of determining α_k was derived for improved convergence rates.

Consider the quadratics generated from (5.44) by using the expansions in (5.45). For the first two inequalities of (5.44) we have

$$\begin{aligned} a_2^i \alpha_k^2 + a_1^i \alpha_k + a_0^i &\geq 0 \\ b_2^i \alpha_k^2 + b_1^i \alpha_k + b_0^i &\geq 0 \end{aligned} \tag{5.48}$$

where

$$\begin{aligned} a_0^i &= \overline{z}_k^i \overline{s}_k^i - \beta_{k+1}\mu_k \geq 0 \\ a_1^i &= \beta_{k+1}\mu_k(1-\gamma_k) + \gamma_k\mu_k - \overline{z}_k^i \overline{s}_k^i \\ &= \gamma_k\mu_k(1-\beta_{k+1}) - a_0^i \\ a_2^i &= \Delta\overline{z}_k^i \Delta\overline{s}_k^i - \beta_{k+1}\{\Delta\underline{z}_k^T \Delta\underline{s}_k + \Delta\overline{z}_k^T \Delta\overline{s}_k\}/n_i \end{aligned} \tag{5.49}$$

and

$$\begin{aligned} b_0^i &= \underline{z}_k^i \underline{s}_k^i - \beta_{k+1}\mu_k \geq 0 \\ b_1^i &= \beta_{k+1}\mu_k(1-\gamma_k) + \gamma_k\mu_k - \underline{z}_k^i \underline{s}_k^i \\ &= \gamma_k\mu_k(1-\beta_{k+1}) - b_0^i \\ b_2^i &= \Delta\underline{z}_k^i \Delta\underline{s}_k^i - \beta_{k+1}\{\Delta\underline{z}_k^T \Delta\underline{s}_k + \Delta\overline{z}_k^T \Delta\overline{s}_k\}/n_i. \end{aligned} \tag{5.50}$$

In order to solve the *third* inequality in (5.44), we must approximate the residual r_{k+1} using Newton's method:

$$\begin{aligned} r + r_{\mathcal{X}}\Delta\mathcal{X} &= 0 \\ r_{k+1} &= r_k + \alpha_k r_{\mathcal{X}}\Delta\mathcal{X} + \tfrac{1}{2}\alpha_k^2 \Delta\mathcal{X}^T r_{\mathcal{X}\mathcal{X}}\Delta\mathcal{X} + O(\| \alpha_k\Delta\mathcal{X} \|^2) \\ &= r_k(1-\alpha_k) + \tfrac{1}{2}\alpha_k^2 \Delta\mathcal{X}^T r_{\mathcal{X}\mathcal{X}}\Delta\mathcal{X} + O(\| \alpha_k\Delta\mathcal{X} \|^2). \end{aligned} \tag{5.51}$$

This yields two quadratics:

$$\begin{aligned} c_2^i \alpha_k^2 + c_1^i \alpha_k + c_0^i &\geq 0 \\ d_2^i \alpha_k^2 + d_1^i \alpha_k + d_0^i &\geq 0 \end{aligned} \tag{5.52}$$

where

$$\begin{aligned} c_0^i &= \overline{\beta}_{k+1}\mu_k\epsilon_0 - r_k \geq 0 \\ c_1^i &= \overline{\beta}_{k+1}\mu_k\epsilon_0(\gamma_k - 1) + r_k = \overline{\beta}_{k+1}\mu_k\epsilon_0\gamma_k - c_0^i \\ c_2^i &= \overline{\beta}_{k+1}\epsilon_0\{\Delta\underline{z}_k^T\Delta\underline{s}_k + \Delta\overline{z}_k^T\Delta\overline{s}_k\}/2n_i - \tfrac{1}{2}\Delta\mathcal{X}^T r_{\mathcal{X}\mathcal{X}}\Delta\mathcal{X} \end{aligned} \tag{5.53}$$

and

$$\begin{aligned} d_0^i &= \overline{\beta}_{k+1}\mu_k\epsilon_0 + r_k \geq 0 \\ d_1^i &= \overline{\beta}_{k+1}\mu_k\epsilon_0(\gamma_k - 1) - r_k = \overline{\beta}_{k+1}\mu_k\epsilon_0\gamma_k - d_0^i \\ d_2^i &= \overline{\beta}_{k+1}\epsilon_0\{\Delta\underline{z}_k^T\Delta\underline{s}_k + \Delta\overline{z}_k^T\Delta\overline{s}_k\}/2n_i + \tfrac{1}{2}\Delta\mathcal{X}^T r_{\mathcal{X}\mathcal{X}}\Delta\mathcal{X} \end{aligned} \tag{5.54}$$

for $\overline{\beta}_{k+1} = 1/\beta_{k+1}$ and $\epsilon_0 = r_k/\mu_k$.

Note that from (5.53) and (5.54):

$$\begin{aligned} c_0^i + c_1^i\alpha_k + c_2^i\alpha_k^2 &= c_0^i(1-\alpha_k) + \overline{\beta}_k\mu_k\epsilon_0\gamma_k\alpha_k + c_2^i\alpha_k^2 \geq 0 \\ d_0^i + d_1^i\alpha_k + d_2^i\alpha_k^2 &= d_0^i(1-\alpha_k) + \overline{\beta}_k\mu_k\epsilon_0\gamma_k\alpha_k + d_2^i\alpha_k^2 \geq 0 \end{aligned} \tag{5.55}$$

Hence $c_0^i + c_1^i\alpha_k + c_2^i\alpha_k^2 < 0$ only if

$$\frac{1}{2}\Delta\mathcal{X}^T r_{\mathcal{X}\mathcal{X}}\Delta\mathcal{X} > \overline{\beta}_{k+1}\epsilon_0\{\Delta\underline{z}_k^T\Delta\underline{s}_k + \Delta\overline{z}_k^T\Delta\overline{s}_k\}/2n_i \tag{5.56}$$

and in this case

$$d_0^i + d_1^i\alpha_k + d_2^i\alpha_k^2 \geq 0, \quad \text{all} \quad \alpha_k \in [0,1]. \tag{5.57}$$

Hence it is only necessary to solve one quadratic.

Each of these quadratics $f(\alpha_k)$ will generate curves in the form of the 4 cases shown in Figure (5.1).

$$\begin{aligned} &\text{Case 1:} \qquad a_1^i > 0,\ a_2^i > 0 \\ &\text{Case 2:} \qquad a_1^i < 0,\ a_2^i > 0 \\ &\text{Case 3:} \qquad a_1^i > 0,\ a_2^i < 0 \\ &\text{Case 4:} \qquad a_1^i < 0,\ a_2^i < 0. \end{aligned} \tag{5.58}$$

This also follows for quadratics b, c, d.

It is important to note that for all of the above quadratics, $f(0) \geq 0$ as a result of the interior point conditions imposed on the previous iterate ($a_0^i,\ b_0^i,\ c_0^i,\ d_0^i \geq 0$). We therefore must determine the zeros $\bar{\alpha}_k$ of each of the quadratics, choose

$$\tilde{\alpha}_k = \max\{\bar{\alpha}_k \geq 0\ s.t.\ 0 \leq \alpha^* \leq \bar{\alpha}_k,\ f(\alpha^*) \geq 0\} \tag{5.59}$$

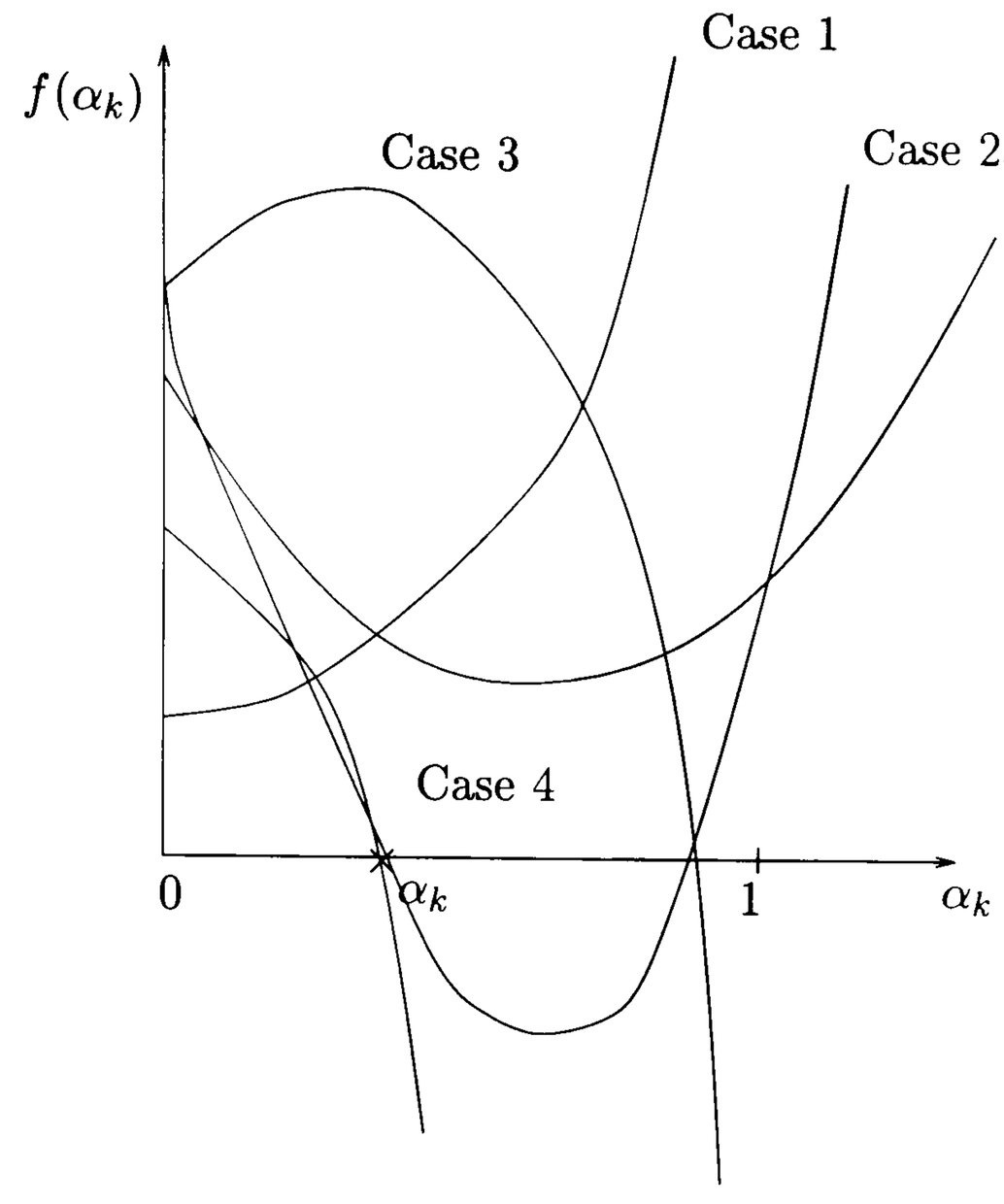


Figure 5.1: Determining α_k

for each individual quadratic and, finally, take

$$\alpha_k = \min\{\tilde{\alpha}_k^a,\ \tilde{\alpha}_k^b,\ \tilde{\alpha}_k^c,\ \tilde{\alpha}_k^d,\ 1\}. \tag{5.60}$$

Therefore, α_k satisfies inequalities (5.44). Thus, the largest $\alpha_k \in [0,1]$ satisfying

$$a_0 + a_1\alpha_k + a_2\alpha_k^2 \geq 0, \quad a_0 \geq 0 \tag{5.61}$$

is given by

$$\begin{array}{lrcll} \text{If} & a_1 & \geq & 0,\ a_2 \geq 0 & \text{then } \alpha_k = 1 \quad \text{(Case 1)} \\ \text{If} & 4a_0\ a_2 & \geq & a_1^2 & \text{then } \alpha_k = 1 \quad \text{(Case 2)} \\ \text{Otherwise} & \alpha_k & = & 2a_0/(\sqrt{a_1^2 - 4a_0a_2} - a_1) & \text{if } a_0 > 0 \\ & & = & -a_1/\ a_2 & \text{if } a_0 = 0. \quad \text{(Cases 2a, 3 or 4)} \end{array} \tag{5.62}$$

5.3.3 Interior Point Algorithm

Consider the interior point algorithm as implemented on (t_0, t_f) for equations (5.16), linearised to equations (5.23) - (5.25). For more details, see (Sargent, 1994 revised 1995).

5.3.3.1 Initialisation

Given an initial point $\mathcal{X}_0 = (x_0,\ u_0,\ y_0,\ s_0,\ \underline{z}_0,\ \overline{z}^0)$ such that s_0 is strictly interior to the constraint and $\| \mathcal{X}_0 \|_\infty < C$ for some chosen constant bound C. Compute $\mu_0 = (\underline{z}_0^T \underline{s}_0 + \overline{z}_0^T \overline{s}_0)/2n_i$, where n_i is the number of path inequalities and $\underline{s}_0 = s_0 - \underline{g},\ \overline{s}_0 = \overline{g} - s_0$. Choose $\beta_1 \in (0, \min\{1,\ \underline{z}_0^i \underline{s}_0^i/\mu,\ \overline{z}_0^i \overline{s}_0^i/\mu,\ i = 1, 2, ..., n_i\})$, $\beta \in (0, \beta_1)$. $\gamma \in (0,1)$. $\rho \in (0.1)$. Set $\gamma_0 = 0, \delta_1 = 1, k = 0$.

5.3.3.2 General Major Iteration

1. Compute

$$\begin{aligned} r_k &= \begin{bmatrix} f_x^T \\ f_{\dot{x}}^T \\ f_u^T \end{bmatrix} y + \begin{bmatrix} g_x^T \\ g_{\dot{x}}^T \\ g_u^T \end{bmatrix} z - \begin{bmatrix} \dot{p} \\ p \\ 0 \end{bmatrix} \\ \tilde{r}_k &= g - s_k \end{aligned} \tag{5.63}$$

2. If $\max \{\| r_k \|_\infty, \| \tilde{r}_k \|_\infty\} \leq \epsilon$ then STOP (solution found).

3. Solve (5.23) and (5.25) for $\Delta \mathcal{X}_k$.

4. Compute α_k to minimise μ_{k+1} subject to (5.44) (see section 5.3.2) and compute $\mathcal{X}_{k+1}$ from (5.20) and $\mu_{k+1} = (\underline{z}_{k+1}^T \underline{s}_{k+1} + \overline{z}_{k+1}^T \overline{s}_{k+1})/2n_i$.

5. If $\| \mathcal{X}_{k+1} \|_\infty > C$ then STOP (failure).

6. i. If $\mu_{k+1} \leq \rho \mu_k$ or $\gamma_k = \gamma$, then set $\gamma_{k+1} = 0,\ \delta_{k+2} = \delta_{k+1}(\mu_{k+1}/\mu_k)^{\frac{1}{2}}$.

   Otherwise, $\gamma_{k+1} = \gamma,\ \delta_{k+2} = \delta_{k+1}$.

   ii. Compute $\beta_{k+2} = \beta + \delta_{k+2}(\beta_1 - \beta)$.

7. Set $k = k + 1$ and return to step 1.

5.3.4 Implementation Details

In order to implement the interior point method, we must first choose an initial point $\mathcal{X}_0$. The choice of initial point is dependent on the type of inequality constraints in the problem. If the inequality constraints are simply bounds on the controls, we choose u_0 as some value strictly within the constraint boundaries. The initial states and adjoint variables are then determined by applying u_0 and the slacks s_0 are completely determined

as $s_0 = u_0$. The multipliers $\underline{z}_0$, $\overline{z}_0$ are chosen such that $\mu_0 = 1$ and we choose $y_0 = 0$. The key factor is that the slacks must be chosen to satisfy interiority conditions so that $\underline{S}, \overline{S}$ are positive diagonal matrices.

The case in which the inequality constraints are functions of the states is less straightforward, as a feasible u_0 may generate an infeasible initial state trajectory. For most problems, it is relatively straight forward to determine an initial feasible trajectory. For process systems examples, we may choose as a first guess the control trajectory which maintains the system at a known feasible steady state. Whether or not the initial trajectory is feasible, we initialise the associated slack variables as

$$s_0 = \begin{cases} g_0, & \underline{g} < g_0 < \overline{g} \\ \overline{g} - \epsilon, & g_0 = \overline{g} \\ \underline{g} + \epsilon, & g_0 = \underline{g} \end{cases} \tag{5.64}$$

depending on where any constraint saturation occurs relative to the constraint bounds. The factor of key importance is ensuring that the initial value of the slack variable is strictly interior to the constraint.

5.3.5 Convergence of the Interior Point Method

We now consider the convergence of the interior point algorithm as applied at each t over (t_0, t_f). We assume that any changes in the state in the forward integration of the system are such that the starting point for each step is feasible. Furthermore, we assume that Δp is negligible for small h. To avoid ill conditioning of the coefficient matrices in (5.23), (5.38) and (5.33), we employ a scaling procedure. Note that the nature of the discussion which follows is not completely rigorous and as such we refer to the ideas as an "Informal Theorem".

Informal Theorem 5.3.1. *Given a system of equations $f(t, \dot{x}, x, x', u', v) = 0$ such that $\begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}$ is nonsingular and inequalities $\underline{g} \leq g(t, \dot{x}, x, x', u', v) \leq \overline{g}$ such that $\overline{H}_k = \sum_j y_k^j f_{\Upsilon\Upsilon}^j + \sum_j z_k^j g_{\Upsilon\Upsilon}^j$ is non-positive definite and $g_{\dot{x},u}^T(\underline{S}_k^{-1}\underline{Z}_k + \overline{S}_k^{-1}\overline{Z}_k)g_{\dot{x},u}$ is positive definite. Then the interior point algorithm terminates for any $\epsilon > 0$ in a finite (polynomial) number of iterations.*

Discussion of Informal Theorem 5.3.1:

Assuming Δp is negligible for small h, consider the following scaled version of (5.23) defined along (t_0, t_f):

$$\begin{bmatrix} G_k^{-\frac{1}{2}} H_k G_k^{-\frac{1}{2}} & G_k^{-\frac{1}{2}} f_\Upsilon^T \\ f_\Upsilon G_k^{-\frac{1}{2}} & 0 \end{bmatrix} \begin{bmatrix} G_k^{\frac{1}{2}} \Delta\Upsilon_k \\ \Delta y_k \end{bmatrix} = - \begin{bmatrix} G_k^{-\frac{1}{2}} \hat{r}_k \\ 0 \end{bmatrix} \tag{5.65}$$

where

$$\begin{gathered} \Upsilon_k = (\dot{x}_k, u_k),\ \overline{p}^T = [p^T, 0] \\ \hat{r}_k = f_\Upsilon^T y_k + g_\Upsilon^T z_k - \overline{p}_k - g_\Upsilon^T \{\underline{S}_k^{-1} \underline{r}_k - \overline{S}_k^{-1} \overline{r}_k + (\underline{S}_k^{-1} \underline{Z}_k + \overline{S}_k^{-1} \overline{Z}_k)(g - s_k)\} \\ H_k = \overline{H}_k - G_k,\ \ \overline{H}_k = \textstyle\sum_j y_k^j f_{\Upsilon\Upsilon}^j + \sum_j z_k^j g_{\Upsilon\Upsilon}^j \\ G_k = g_\Upsilon^T (\underline{S}_k^{-1} \underline{Z}_k + \overline{S}_k^{-1} \overline{Z}_k) g_\Upsilon \\ \underline{r}_k = \underline{S}_k \underline{z}_k - \gamma_k \mu_k e,\ \overline{r}_k = \overline{S}_k \overline{z}_k - \gamma_k \mu_k e \\ \underline{S}_k = diag[s_k^i - \underline{g}^i],\ \overline{S}_k = diag[\overline{g}^i - s_k^i] \\ \underline{Z}_k = diag[\underline{z}_k^i],\ \overline{Z}_k = diag[\overline{z}_k^i] \end{gathered} \tag{5.66}$$

together with

$$\begin{gathered} \Delta s_k = g - s_k + g_\Upsilon \Delta\Upsilon_k \\ \Delta \underline{z}_k = -\underline{S}_k^{-1} (\underline{r}_k + \underline{Z}_k \Delta s_k),\ \Delta \overline{z}_k = -\overline{S}_k^{-1} (\overline{r}_k - \overline{Z}_k \Delta s_k). \end{gathered} \tag{5.67}$$

In order to prove convergence, we must show that:

- the steps $\Delta\Upsilon_k$, Δy_k are well-defined (i.e. (5.65) has a solution), uniformly bounded and satisfy the interior point conditions
- the iteration sequence eventually satisfies the termination conditions, and
- the termination conditions imply the desired result.

We have

$$\begin{aligned} \overline{H}_k &\leq 0 \\ G_k &> 0 \end{aligned} \tag{5.68}$$

by our system assumptions, as well as f_Υ full rank. By Lemma 5.3.1 we know that the coefficient matrix of (5.65) is nonsingular. Thus, (5.65) has a solution.

We may write (5.65) as

$$\begin{bmatrix} G_k^{\frac{1}{2}} \Delta\Upsilon_k \\ \Delta y_k \end{bmatrix} = - \begin{bmatrix} G_k^{-\frac{1}{2}} H_k G_k^{-\frac{1}{2}} & G_k^{-\frac{1}{2}} f_\Upsilon^T \\ f_\Upsilon G_k^{-\frac{1}{2}} & 0 \end{bmatrix}^{-1} \begin{bmatrix} G_k^{-\frac{1}{2}} \hat{r}_k \\ 0 \end{bmatrix}$$
$$= - \begin{bmatrix} \tilde{P}_k \tilde{H}_k^{-1} & \tilde{H}_k^{-1} \tilde{A}_k^T \tilde{G}_k^{-1} \\ (\tilde{H}_k^{-1} \tilde{A}_k^T \tilde{G}_k^{-1})^T & -(\tilde{A}_k \tilde{H}_k \tilde{A}_k^T)^{-1} \end{bmatrix} \begin{bmatrix} G_k^{-\frac{1}{2}} \hat{r}_k \\ 0 \end{bmatrix} \quad (5.69)$$

where $\tilde{H}_k = G_k^{-\frac{1}{2}} H_k G_k^{-\frac{1}{2}}$, $\tilde{A}_k = f_\Upsilon G_k^{-\frac{1}{2}}$, $\tilde{G}_k = \tilde{A}_k \tilde{H}_k^{-1} \tilde{A}_k^T$ and $\tilde{P}_k = I - \tilde{H}_k^{-1} \tilde{A}_k^T \tilde{G}_k^{-1} \tilde{A}_k$. $\tilde{P}_k$ is a projection matrix, so $\| \tilde{P}_k \| = 1$. Below we use n generically to correspond to the appropriate dimensions of the vector whose norm we are considering.

We have

$$\tilde{H}_k = G_k^{-\frac{1}{2}} H_k G_k^{-\frac{1}{2}} = G_k^{-\frac{1}{2}} (\bar{H}_k - G_k) G_k^{-\frac{1}{2}} = G_k^{-\frac{1}{2}} \bar{H}_k G_k^{-\frac{1}{2}} - I < 0. \quad (5.70)$$

Write $P_k = -G_k^{-\frac{1}{2}} \bar{H}_k G_k^{-\frac{1}{2}} \geq 0$. For any non-zero vector v, we have

$$\| v \| \| \tilde{H}_k v \| \geq v^T (-\tilde{H}_k) v = v^T (I + P_k) v \geq v^T v. \quad (5.71)$$

Hence, $\tilde{H}_k$ is nonsingular and $\| \tilde{H}_k v \| \geq \| v \|$. Then

$$\| v \| = \| \tilde{H}_k \tilde{H}_k^{-1} v \| \geq \| \tilde{H}_k^{-1} v \|, \quad \text{for all } v \quad (5.72)$$

and therefore $\| \tilde{H}_k^{-1} \| \leq 1$.

Given an interior point $\mathcal{X}_k = (\dot{x}_k, u_k, y_k, s_k, \underline{z}_k, \overline{z}_k)$, assume $\| \mathcal{X}_k \|_\infty \leq C < \infty$. Using the interior point inequality in (5.44), we have

$$\begin{aligned} G_k &= g_\Upsilon^T [(\underline{S}_k \underline{Z}_k)^{-1} \underline{Z}_k^2 + (\overline{S}_k \overline{Z}_k)^{-1} \overline{Z}_k^2] g_\Upsilon \\ \therefore G_k^{\frac{1}{2}} &\leq \overline{C} (\beta_k \mu_k)^{-\frac{1}{2}}. \end{aligned} \quad (5.73)$$

Similarly,

$$\begin{aligned} G_k &= g_\Upsilon^T [\underline{S}_k^{-2} (\underline{S}_k \underline{Z}_k) + \overline{S}_k^{-2} (\overline{S}_k \overline{Z}_k)] g_\Upsilon \\ \therefore G_k^{-\frac{1}{2}} &\leq \overline{C} (\beta_k \mu_k)^{-\frac{1}{2}}. \end{aligned} \quad (5.74)$$

Define $\hat{G}_k = \underline{S}_k^{-1} \underline{Z}_k + \overline{S}_k^{-1} \overline{Z}_k$. We may write

$$\hat{G}_k^{-\frac{1}{2}} \underline{S}_k^{-1} = (\underline{S}_k^{-1} \underline{Z}_k + \overline{S}_k^{-1} \overline{Z}_k)^{-\frac{1}{2}} \underline{S}_k^{-1} \leq (\underline{S}_k^{-1} \underline{Z}_k)^{-\frac{1}{2}} \underline{S}_k^{-1} = (\underline{S}_k \underline{Z}_k)^{\frac{1}{2}} \leq \overline{C} (\beta_k \mu_k)^{-\frac{1}{2}} \quad (5.75)$$

for some $\overline{C} < \infty$ independent of n and k.

From (5.44) we have

$$\begin{aligned} \| \hat{r}_k \| &= O(n^{\frac{1}{2}}\mu_k), \qquad \| \tilde{r}_k \| = O(n^{\frac{1}{2}}\mu_k). \\ \| \underline{r}_k \| &= O(n^{\frac{1}{2}}\mu_k), \qquad \| \overline{r}_k \| = O(n^{\frac{1}{2}}\mu_k). \end{aligned} \tag{5.76}$$

Thus for the residual vector in (5.65) we have,

$$\| G_k^{-\frac{1}{2}} \hat{r}_k \| = O(\sqrt{n\mu_k}). \tag{5.77}$$

Furthermore,

$$\| \tilde{A}_k{}^T \| = O(\mu_k^{-\frac{1}{2}}) \tag{5.78}$$

and therefore

$$\begin{aligned} \| G_k^{\frac{1}{2}} \Delta\Upsilon_k \| &= O(\sqrt{n\mu_k}), \ \| \Delta y_k \| = O(n^{\frac{1}{2}}), \\ \| \Delta\Upsilon_k \| &= \| G_k^{-\frac{1}{2}} G_k^{\frac{1}{2}} \Delta\Upsilon_k \| \leq \| G_k^{-\frac{1}{2}} \| \, \| G_k^{\frac{1}{2}} \Delta\Upsilon_k \| = O(n^{\frac{1}{2}}). \end{aligned} \tag{5.79}$$

From (5.67),

$$\begin{aligned} \Delta\underline{z}_k &= -(\underline{S}_k\underline{Z}_k)^{-1}\underline{Z}_k\underline{r}_k - (\underline{S}_k\underline{Z}_k)^{-1}\underline{Z}_k^2\Delta s_k \\ \Delta\overline{z}_k &= -(\overline{S}_k\overline{Z}_k)^{-1}\overline{Z}_k\overline{r}_k + (\overline{S}_k\overline{Z}_k)^{-1}\overline{Z}_k^2\Delta s_k \end{aligned} \tag{5.80}$$

and it follows that

$$\| \Delta s_k \| = O(n^{\frac{1}{2}}\mu_k), \ \| \Delta\underline{z}_k \| = O(n^{\frac{1}{2}}), \ \| \Delta\overline{z}_k \| = O(n^{\frac{1}{2}}). \tag{5.81}$$

Furthermore, from (5.67), (5.75), (5.76) and (5.81), we have $\| \hat{G}_k^{-\frac{1}{2}} \Delta\underline{z}_k \| = O(\sqrt{n\mu_k})$ and $\| \hat{G}_k^{\frac{1}{2}} \Delta s_k \| = O(\sqrt{n\mu_k})$. Therefore, $\| \Delta\underline{z}_k^T \Delta\underline{s}_k \| = \| \Delta\underline{z}_k^T \hat{G}_k^{-\frac{1}{2}} \hat{G}_k^{\frac{1}{2}} \Delta\underline{s}_k \| = O(n\mu_k)$ and $\| \Delta\overline{z}_k^T \Delta\overline{s}_k \| = O(n\mu_k)$.

The next key part of the "proof" is to show that α_k is bounded away from zero. This ensures that the step calculated in (5.65) is nonzero and satisfies the interior point conditions (5.44). Consider the quadratics (5.48) and (5.52) generated by expanding the inequalities (5.44). We require

$$\begin{aligned} a_2^i\alpha_k^2 + a_1^i\alpha_k + a_0^i &\geq 0 \\ \text{i.e.} \qquad a_2^i\alpha_k^2 + a_0^i(1-\alpha_k) + \gamma_k\mu_k(1-\beta_{k+1})\alpha_k &\geq 0 \end{aligned} \tag{5.82}$$

From the previous iteration, we have $a_0^i \geq 0$. Consider the limiting case $a_0^i = 0$. Therefore, for $\alpha_k \in [0,1]$, either $a_2^i \geq 0$ or $\alpha_k \leq -\gamma_k\mu_k(1-\beta_{k+1})/a_2^i$. But $|a_2^i| = O(\mu_k)$ and

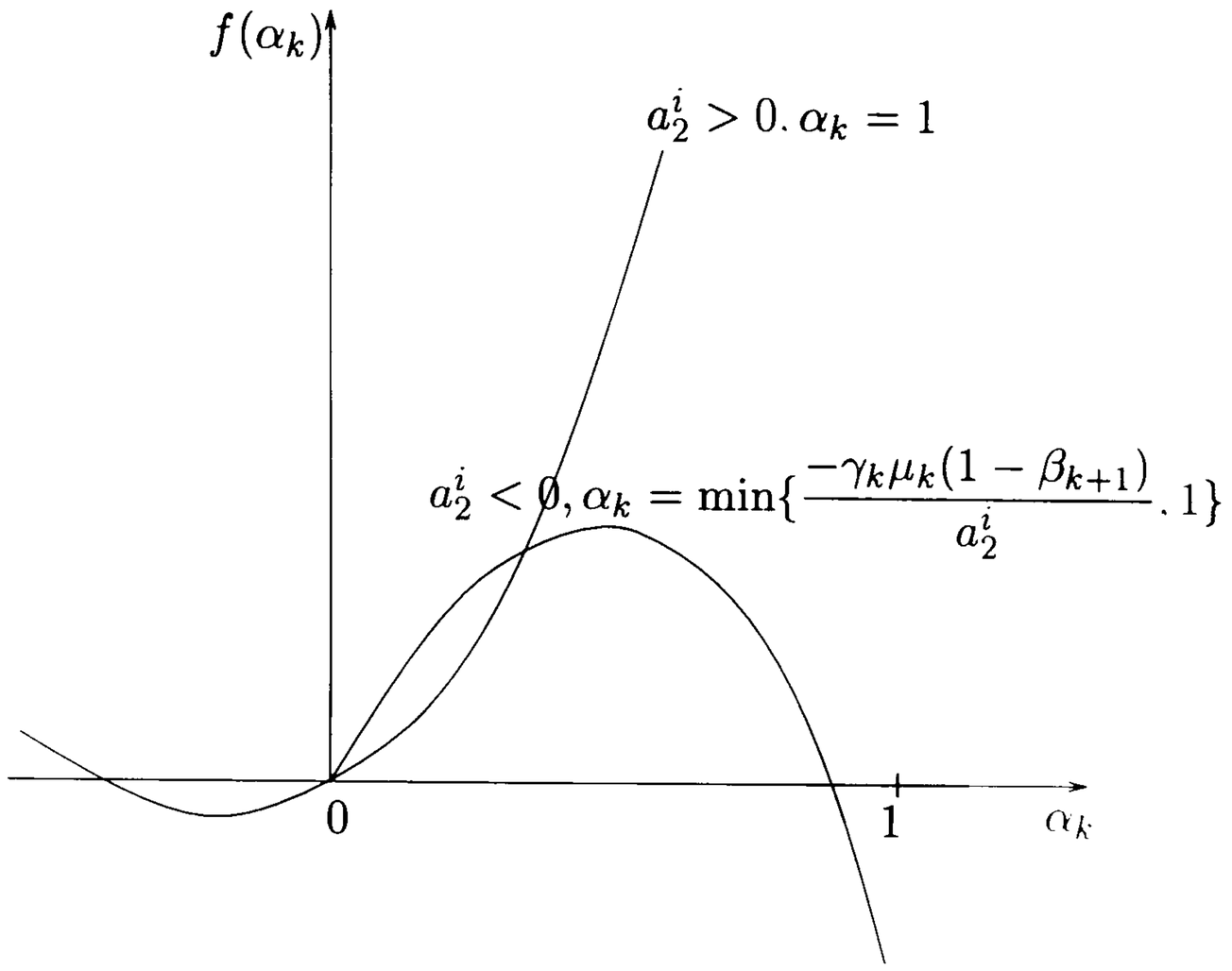


Figure 5.2: Demonstrating $\alpha_k > 0$

$\gamma_k = \gamma > 0$. Hence the largest α_k is bounded below, away from zero. To illustrate this point pictorially, we present the two cases in Figure 5.2.

Now, from (5.44), its quadratic expansion (5.45), and (5.46), if $\gamma_k = \gamma$, the limiting α_k to satisfy (5.44) is $\Omega(n^{-1})$.

We now must show that the algorithm satisfies the termination conditions and terminates within $O(n \log \epsilon / n_i)$ iterations. Since $\underline{z}_k^i \underline{s}_k^i \leq \underline{z}_k^T \underline{s}_k$, $\overline{z}_k^i \overline{s}_k^i \leq \overline{z}_k^T \overline{s}_k$ for all i, it follows from (5.44) that the termination test:

$$\max\{\| r_k \|_\infty, \| \tilde{r}_k \|_\infty\} \leq \epsilon \tag{5.83}$$

is satisfied if

$$\begin{aligned} \underline{z}_k^T \underline{s}_k + \overline{z}_k^T \overline{s}_k &\leq \min\{\epsilon, \beta_k (\underline{z}_0^T \underline{s}_0 + \overline{z}_0^T \overline{s}_0)\epsilon / \| \breve{r}_0 \|_\infty\} \\ i.e. \quad \mu_k &\leq \min\{\epsilon / n_i, \epsilon \beta_k \mu_0 / \| \breve{r}_0 \|_\infty\} \end{aligned} \tag{5.84}$$

where $\| \breve{r}_0 \|_\infty = \max\{\| r_0 \|_\infty, \| \tilde{r}_0 \|_\infty\}$. This implies $\mu_k / \mu_0 = O(\epsilon / n_i)$ on termination.

We have, from (5.45) and (5.46),

$$\mu_{k+1} \leq \mu_k (1 - \alpha_k (1 - \gamma_k)/2). \tag{5.85}$$

Thus, $\{\mu_k\}$ is decreased by a fixed fraction at least every alternate step due to Step 6 in the algorithm, provided $\| \mathcal{X}_k \|_\infty \leq C$. Therefore, eventually we satisfy the termination

test in Step 2, unless termination occurs earlier at Step 5 due to $\| \mathcal{X} \|_\infty > C$. Thus, in order to ensure the termination conditions are satisfied, we must show that $\| \mathcal{X}_k \|_\infty$ remains uniformly bounded.

Up to this point we have shown that (5.65) has a solution and hence the steps are well-defined. Because elements of $\underline{S}$, $\overline{S}$, $\underline{Z}$ and $\overline{Z}$ tend to zero, it is not possible to prove that the inverse of the coefficient matrix is uniformly bounded as $k \to \infty$. However, we were able to show that the solution $\Delta\mathcal{X}_k$ remains uniformly bounded. We then were able to show that the damping parameter α_k remains bounded away from zero at each step. Finally, we were able to conclude that $\mu_k \to 0$. All of the above results rest on the assumption that $\| \mathcal{X}_k \|_\infty \leq C < \infty$.

The difficult portion of the proof then is to show that $\| \mathcal{X}_k \|_\infty$ is uniformly bounded. The standard technique for proving this makes use of the negative-definiteness of H and a regularity condition. Since, in our case, H varies, we must use descent conditions to ensure that the iterates $(\dot{x}, x, u, v)$ remain in a compact set. Since $\{\mu_k\}$ is monotonically decreasing and we terminate at $\mu_k \geq \epsilon$, it may be sufficient to show that $(\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z}) \geq \rho I$ for some fixed $\rho > 0$ and $\mu_k \geq \epsilon$. Furthermore, if we assume f_Υ, g_Υ uniformly full row rank over a compact set $\{\dot{x}, x, u, v\}$, we can then deduce that the inverse of the coefficient matrix is uniformly bounded for the finite sequence and, hence, that the multipliers are uniformly bounded. Thus, this last point in the informal proof requires further investigation to move to the formal stage and we include this in our future directions.

■

Similar "informal" proofs would follow for the interior point iterations at t_0 and t_f.

5.4 Overall Convergence of Algorithm

Starting with a trajectory $\mathcal{X}_k = (\dot{x}_k, x_k, u_k, v_k)$, Steps 4, 5, 2, 3 generate a new trajectory $\mathcal{X}_k + \Delta\mathcal{X}_k$, and to ensure convergence of this outer iteration we need to ensure that it achieves an overall reduction in the residuals of the modified necessary conditions for the reduced $\mu_k(\cdot)$ trajectory generated in Step 5. Although (5.44) ensures such a reduction in the backward integration, this is essentially based on a linearisation of $f = 0$, and the effects of nonlinearity in the following forward integration need to be taken into account.

However, the nonlinear effects in the *scaled* trajectory $\mathcal{X}_k + \Delta\mathcal{X}_k$ can be made as small

as desired by choosing α_k sufficiently small, so we can choose α_k, for example by the Armijo rule, so that the residual $\tilde{r}_k = \max_{t\in[t_0,t_f]}\{\| f_\Upsilon^T y_k + g_\Upsilon^T z_k - \bar{p},\ g - s_k,\ f \|_\infty\}$ is reduced by at least a factor $(1 - \delta\alpha_k)$ for some fixed $\delta \in (0, 1)$.

Alternatively, we can re-integrate $f = 0$ for scaled values $x_0 + \alpha\Delta x_0$, $u_k(\cdot) + \alpha_k \Delta u(\cdot)_k$, again choosing α_k to reduce $\tilde{r}_k$. This has the advantage that each trajectory satisfies the system equations $f = 0$ (and, hence, f can be dropped in the definition of $\tilde{r}_k$).

A standard argument can be used to show that α_k is bounded away from zero, and hence that $\|\tilde{r}_k\|$ decreases strictly monotonically. It is possible to become trapped at a local non-zero minimum of $\|\tilde{r}_k\|$.

Of course, it must be possible to complete each cycle of the basic algorithm, and this follows from our basic assumptions (Chapter 4) provided that the trajectory remains in the interior of the feasible region. This is ensured by choosing α_k to satisfy (5.44) on the backward integration, and we similarly need to check the interior point conditions $\underline{S}_k\ \underline{z}_k \geq \beta_{k+1}\mu_k e$, $\overline{S}_k\ \overline{z}_k \geq \beta_{k+1}\mu_k e$ at each step on the forward integration. If any of these are violated, we immediately re-start the forward integration with a reduced value of α_k.

This strategy represents a straightforward extension of the standard strategy for the damped Newton method for solution of nonlinear algebraic equations (see, for example, (Moré and Cosnard, 1979)). However, as they point out, it does not provide an absolute guarantee of convergence, and the scaling usually significantly slows down the rate of convergence. They therefore recommend nominal use of the standard Newton iteration, including only mild protection against excessively large steps, rather than forcing residual reduction at each iteration. In light of this advice, we have not therefore implemented the above strategy in our preliminary testing of the algorithm.

However, special problems do arise with the basic algorithm when the path constraints depend strongly on the state variables, and these will be considered in Section 5.9.

5.5 Singular Solutions

We comment here on the relative benefit of the above formulation in dealing with singular solutions. In singular problems, the Hamiltonian is independent of the control along a segment of the optimal trajectory and therefore cannot be maximised with respect to the control. However, the computational effect is such that the singular control does not arise

until the solution becomes exact. The essential point is that the necessary conditions apply only along the optimal trajectory, while the computation also uses properties in the neighbourhood of the trajectory. Thus we iterate towards the solution, with the algorithm essentially unaware that the solution it is approaching is a singular one. In effect, the Hamiltonian is dependent on the control indirectly via the computation of the multiplier z.

5.6 Free Endtime Problems

To deal with free or minimum time problems, we extend the time-invariant parameter vector v to include τ, where our time horizon has been transformed as follows:

$$t = t_0 + \tau t', \quad \text{where } t \in [t_0, t_f], \quad t' \in [0,1], \quad \tau = t_f - t_0 > 0 \tag{5.86}$$

Write $v' = [v, \ \tau]$. We define $f'(t', \dot{x}', x, u, v') = f(t, \dot{x}, x, u, v)$. Taking the partial derivative with respect to τ, we obtain:

$$\begin{aligned} f'_\tau &= f_t t' - \tau^{-1} f_{\dot{x}} \dot{x} \\ &= \tau^{-1}(f_t(t - t_0) - f_{\dot{x}}\dot{x}). \end{aligned} \tag{5.87}$$

The same procedure may be applied to g, ψ, and ϕ. Now consider our endpoint transversality equations with respect to our new time-invariant parameter τ. This yields:

$$\begin{aligned} &(f_\tau^T \eta^0 + g_\tau^T \zeta^0 + \phi_\tau^T \xi^0 + \psi_\tau^T \omega^0 + V_\tau^T \lambda^0) + (f_\tau^T \eta^f + g_\tau^T \zeta^f + \phi_\tau^T \xi^f + \psi_\tau^T \omega^f + V_\tau^T \lambda^f + P_\tau^T) \\ &\quad = \int_0^1 \{y^T f'_\tau + z^T g'_\tau\}^T \ dt' \\ &\quad = \tau^{-1} \int_0^1 \{(t - t_0)(f_t^T y + g_t^T z) - \dot{x}^T (f_{\dot{x}}^T y + g_{\dot{x}}^T z)\} \ dt' \\ &\quad = \tau^{-2} \int_{t_0}^{t_f} \{(t - t_0)(f_t^T y + g_t^T z) - \dot{x}^T p\} \ dt. \end{aligned} \tag{5.88}$$

At $t = t_f$, $t' = 1$ and $f_\tau = f_t - \tau^{-1} f_{\dot{x}} \dot{x}$. Similarly, at $t = t_0$, $t' = 0$ and $f_\tau = -\tau^{-1} f_{\dot{x}} \dot{x}$. Therefore, using the endpoint transversality equations with respect to $\dot{x}$ and transforming the integration horizon to $dt = \tau dt'$, we obtain:

$$(f_t^T \eta^f + g_t^T \zeta^f + \phi_t^T \xi^f + \psi_t^T \omega^f + P_t^T) = \tau^{-1} \int_0^1 \{(t - t_0)(f_t^T y + g_t^T z) - \dot{x}^T p\} \ dt'. \tag{5.89}$$

We then may easily deal with free endtime or minimum time problems by normalising our time horizon and introducing a new time-invariant parameter to our system.

If τ is in fact bounded, the situation is slightly different. For this case, the partial derivative of V with respect to τ is $V_\tau = 1$ at $t' = 1$ and therefore the transversality condition becomes:

$$(f_t^T \eta^f + g_t^T \zeta^f + \phi_t^T \xi^f + \psi_t^T \omega^f + P_t^T + \lambda^f) = \tau^{-1} \int_0^1 \{(t - t_0)(f_t^T y + g_t^T z) - \dot{x}^T p\} \; dt'. \tag{5.90}$$

5.7 Parameter Optimisation

It is worth noting that the parameter optimisation problem, outlined in the previous section for the special case of the free endtime problem, can also be dealt with by considering these parameters as states. Thus

$$\begin{aligned} \dot{x} &= 0 \\ x(0) &= v \end{aligned} \tag{5.91}$$

In this way, the problem becomes one of initialising the states at optimal values.

5.8 Dealing with Equality Constraints at the Endpoints

The subproblems in steps 2 and 4 of the basic algorithm of Section 5.2 can be solved similarly, but we note that with equality constraints $\phi^f(x) = 0$, system (5.12) is not sufficient to determine ξ^f, and hence $p(t_f)$, while the values $x(t_f)$ generated in step 3 will not in general satisfy these constraints. To avoid this difficulty, we explore two options. The first option splits the adjoint variable into two parts measuring the individual sensitivities of the cost and endpoint equality constraints respectively to variations in the control trajectory and/or parameter values and is in the flavour of ideas proposed by (Bryson and Ho, 1975). The second option uses a modification of the combined gradient-restoration techniques of (Miele, 1973) to determine the optimal control trajectory and/or time-invariant design parameters by an appropriate linear combination.

5.8.1 Partition of the Adjoint Equations

Consider the initialisation of $p(t_f)$. The endpoint transversality equations yield:

$$p(t_f) = P_x^T + f_x^T \eta^f + g_x^T \zeta^f + \psi_x^T \omega^f + \phi_x^T \xi^f. \tag{5.92}$$

We assume our constraints to be divisible into those dependent on $(t, \dot{x}, u, v)$ and those dependent on $x(t_f)$:

$$\phi = \left[\tilde{\phi}(t, \dot{x}, u, v) \quad \bar{\phi}(x).\right] \tag{5.93}$$

The difficulty lies in the determination of $\xi^f = \left[\tilde{\xi}^f \quad \bar{\xi}^f\right]^T$. In the derivation which follows, we treat the parameters as time-invariant states for simplicity of exposition. Consider splitting the adjoint variable into two parts as follows (see (Bryson and Ho, 1975) for a similar derivation for ODEs in the absence of inequality constraints, as outlined in the literature review):

$$\begin{aligned} \tilde{p}(t_f) &= P_x^T + f_x^T \eta^f + g_x^T \zeta^f + \psi_x^T \omega^f + \tilde{\phi}_x^T \tilde{\xi}^f \\ \bar{p}(t_f) &= \bar{\phi}_x^T \end{aligned} \tag{5.94}$$

such that $p(t_f) = \tilde{p}(t_f) + \bar{p}(t_f)\bar{\xi}^f$ and

$$\begin{aligned} \dot{\tilde{p}} &= f_x^T \tilde{y} + g_x^T z \\ \dot{\bar{p}} &= f_x^T \bar{y}. \end{aligned} \tag{5.95}$$

We have from our adjoint system definition (5.16):

$$\begin{bmatrix} p \\ 0 \end{bmatrix} = \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix} y + \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} z \tag{5.96}$$

and therefore we may write

$$\begin{aligned} y &= \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix}^{-1} \left(\begin{bmatrix} p \\ 0 \end{bmatrix} - \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} z \right) \\ &= \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix}^{-1} \left(\begin{bmatrix} \tilde{p} \\ 0 \end{bmatrix} + \begin{bmatrix} \bar{p} \\ 0 \end{bmatrix} \bar{\xi}^f - \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} z \right). \end{aligned} \tag{5.97}$$

We may write this multiplier in two parts such that $y = \tilde{y} + \bar{y}\bar{\xi}^f$ and

$$\begin{aligned} \tilde{y} &= \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix}^{-1} \left(\begin{bmatrix} \tilde{p} \\ 0 \end{bmatrix} - \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} z \right) \\ \bar{y} &= \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix}^{-1} \begin{bmatrix} \bar{p} \\ 0 \end{bmatrix}. \end{aligned} \tag{5.98}$$

We now linearise $\bar{\phi} = 0$:

$$\begin{aligned} \bar{\phi} + \bar{\phi}_x \delta x^f &= 0 \\ \text{i.e. } \bar{\phi} + (\bar{p}^f)^T \delta x^f &= 0. \end{aligned} \tag{5.99}$$

Hence, linearising $f = 0$, we have

$$f_{\dot{x}} \delta \dot{x} + f_x \delta x + f_u \delta u \approx 0 \tag{5.100}$$

and

$$\begin{aligned} \frac{d}{dt} \bar{p}^T \delta x &= \dot{\bar{p}}^T \delta x + \bar{p}^T \delta \dot{x} \\ &= \bar{y}^T (f_{\dot{x}} \delta \dot{x} + f_x \delta x) \\ &= -\bar{y}^T f_u \delta u. \end{aligned} \tag{5.101}$$

Therefore

$$\bar{\phi} + (\bar{p}^0)^T \delta x^0 - \int_{t_0}^{t_f} \bar{y}^T f_u \delta u = 0. \tag{5.102}$$

We obtain $\delta u = \delta \tilde{u} + \delta \bar{u} \bar{\xi}^f$ by solving (5.23), where again we can write $\Upsilon = \tilde{\Upsilon} + \bar{\Upsilon} \bar{\xi}^f$ $p = \tilde{p} + \bar{p} \bar{\xi}^f$ and $y = \tilde{y} + \bar{y} \bar{\xi}^f$ to obtain:

$$\begin{bmatrix} H & -I_p & f_{\mathcal{K}}^T \\ f_\Upsilon & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\tilde{\Upsilon}, \Delta\bar{\Upsilon} \\ \Delta\tilde{p}, \Delta\bar{p} \\ \Delta\tilde{y}, \Delta\bar{y} \end{bmatrix} = - \begin{bmatrix} \tilde{r}, \bar{r} \\ 0, 0 \end{bmatrix} \tag{5.103}$$

where the partitioning of the residuals and the elements of the matrix follow as a result of the partitioning of the variables. Thus

$$\tilde{H} = \sum_i \tilde{y}^i f^i_{\Upsilon\Upsilon} + \sum_i z^i g^i_{\Upsilon\Upsilon}. \tag{5.104}$$

We note that since H is a function of y and z, we must choose to evaluate $y = \tilde{y} + \bar{y} \bar{\xi}^f$ at the current value of $\bar{\xi}^f$.

Therefore, we have

$$\bar{\phi} + (\bar{p}^0)^T \delta x^0 - \int_{t_0}^{t_f} \bar{y}^T f_u (\delta \tilde{u} + \delta \bar{u} \bar{\xi}^f) dt = 0. \tag{5.105}$$

Write

$$\begin{aligned} I_{\phi\phi} &= \int_{t_0}^{t_f} \bar{y}^T f_u \delta \bar{u} \, dt \\ I_{\phi P} &= \int_{t_0}^{t_f} \bar{y}^T f_u \delta \tilde{u} \, dt \end{aligned} \tag{5.106}$$

and therefore

$$\bar{\xi}^f = I_{\phi\phi}^{-1}(\bar{\phi} + (\bar{p}^0)^T \delta x^0 - I_{\phi P}). \tag{5.107}$$

As we approach the optimum, $\delta\bar{\phi} \to 0$, $\delta x^0 \to 0$ and $\bar{\xi}^f \to -I_{oo}^{-1} I_{oP}$. If the problem is a minimum time problem, equation (5.99) becomes

$$\bar{\phi} + \bar{\phi}_x \delta x^f + \bar{\phi}_t \delta t^f = 0 \tag{5.108}$$

which carries through to yield

$$\bar{\phi} + \bar{\phi}_t \delta t^f + (\bar{p}^0)^T \delta x^0 - \int_{t_0}^{t_f} \bar{y}^T f_u (\delta\tilde{u} + \delta\bar{u}\bar{\xi}^f) dt = 0. \tag{5.109}$$

The presence of the barrier function yields a more natural result in comparison with Bryson and Ho's approach. Due to linearisation (see Chapter 2), Bryson and Ho are forced to artificially amend the variational cost by appending a quadratic term with weighting W. Their approach requires a somewhat arbitrary choice of $W > 0$ which ensures an appropriate reduction in $\bar{\phi}$ at each iteration. They require $\delta\bar{\phi} = -\epsilon\bar{\phi}$, where $\epsilon \in (0, 1]$. Thus, it may be necessary to modify the choice of W to ensure the assumed and actual $\delta\bar{\phi}$ are of the same order of magnitude after the first iteration of the algorithm.

However, this makes no use of second order information in f, g or the curvature of the barrier function. In contrast, our approach determines $\delta\bar{u}$ and $\delta\tilde{u}$ via equation (5.103) and therefore does incorporate second order information. Our W is uniquely defined by the interior point parameters. We refer the reader to (Bryson and Ho, 1975) for more details.

5.8.2 Miele's Combined Gradient-Restoration Scheme

As an alternative to the above formulation, we may employ Miele's scheme, as outlined in Section 2.4.3.2. This approach chooses a set of basis vectors for $\{\xi_i\}_{i=1}^{\bar{n}_e+1}$, where the basis is of dimension $i = 1, ..., \bar{n}_e + 1$; generates the appropriate perturbed adjoint trajectory $\Delta p_i(t)$ and corresponding $\Delta u_i(t)$ for each ξ_i. The corresponding $\Delta x_i(t)$ may then be generated either by applying $\Delta u_i(t)$ to the original nonlinear system or to the linearised system. We may then generate a linear combination such that

$$\phi_0^f + \sum_{i=1}^{\bar{n}_e+1} k_i(\phi_{\dot{x}_0}^f \Delta\dot{x}_i(t) + f\phi_{x_0}^f \Delta x_i(t) + \phi_{u_0}^f \Delta u_i(t) + \phi_{v_0}^f \Delta v_i) = 0 \tag{5.110}$$

for some set $\{k_i | \quad k_i \leq 1\}$. Additional scaling may be required to ensure that we also generate a decrease in P^f, the objective at t_f.

This approach is easily accommodated in the basic algorithm of Section 5.2. We simply incorporate the additional integrations required for the linear combination in a looping procedure, centred around step 4. Thus, we choose an basis $\{\xi_i\}_{i=1}^{\bar{n}_e+1}$ for ξ of dimension $\bar{n}_e + 1$. For each ξ_i, we determine $p(t_f)$ in step 4, generate a new control in step 5 and a corresponding state in step 2. At the end of our first $\bar{n}_e + 1$ integrations, we determine the appropriate linear combination. This yields a new value for ξ. We therefore begin the next loop with this value of ξ and proceed through the next set of $\bar{n}_e + 1$ integrations. After each set i of integrations, we may replace the corresponding ξ_i with the latest update, thereby moving us closer to the correct value.

It is important to note that Miele's approach works best in the limiting case as $\mu \to 0$. This is a direct consequence of the fact that for $\mu_1, \mu_2 > 0, \quad \mu_1 \neq \mu_2$, the same state will generate a slightly different control. Thus, the linear combination will most likely result in a smaller reduction in ϕ^f for larger μ.

5.8.3 Conclusions

We have presented two equally viable solutions for dealing with endpoint equality constraints. The first approach provides an easier method of determining ξ^f, as the update of ξ^f follows very easily from integration of the partitioned adjoint system. This approach requires $(n_d + \bar{n}_e) \times (\bar{n}_e + 1)$ integrations per iteration, where n_d is the number of differential states in the system. The second approach requires $2n_d \times (\bar{n}_e + 1)$ integrations per iteration to generate an update for the multiplier ξ^f. For a small number of states, the difference per integration will be insignificant. However, as the number of states increases, the second approach becomes significantly more computationally intensive. Which approach is better will then depend on the number of endpoint equality constraints, the number of states and how accurately the control trajectory is initialised.

5.9 Infeasibility of Inequality Path Constraints

The solution procedure outlined in Section 5.2 computes the optimal control at each t by solving the necessary conditions evaluated at the previously generated state trajectory. The presence of state equality constraints at the endpoint presents difficulties and the

previously outlined approaches work well in the absence of path constraints, as will be demonstrated for a simple case study in Chapter 6.

However, in the presence of path constraints, one has the further complication that the constraint must be satisfied for all t. One can therefore imagine a great conflict between optimising the objective, as well as maintaining feasibility with respect to the path constraint. Furthermore, there may be the added requirement of achieving an endpoint constraint on a state.

The true difficulty in the basic approach rests on the assumption that the control trajectory generated for a fixed feasible state trajectory will in turn generate another feasible state trajectory upon implementation. This will not, in general, be true. The method of local scaling of the control has no effect, as one cannot guarantee that the local control generated at the previous iteration will generate a feasible step. Experiments involving rescaling of the entire trajectory, as discussed in Section 5.4, produced very slow convergence. Based on this, we propose the following solution to this problem, which makes use of active set strategies. This approach provides the most effective solution.

5.9.1 Method of Solution using Active Sets

One method of dealing with state-dependent constraints, which may be easily employed using the basic algorithm above, is as follows:

Assume the state dynamics to be stable in the forward direction. Consider a subset of the inequality constraints g which are state-dependent:

$$\underline{\bar{g}} \leq \bar{g}(t, \dot{x}, x, u, v) \leq \bar{\bar{g}}. \tag{5.111}$$

We therefore have a partition of g such that $g = [\tilde{g} \;\; \bar{g}]$. When a state-dependent constraint is binding, we may then consider the modified system equations (considered here for the case when the upper constraint is binding):

$$\tilde{f} = \begin{cases} f(t, \dot{x}, x, u, v) &= 0 \\ \bar{g}(t, \dot{x}, x, u, v) - \bar{\bar{g}} &= 0. \end{cases} \tag{5.112}$$

In order to determine the appropriate control to maintain feasibility, it may be necessary to differentiate all or part of the system I times. where I is the index of system (5.112). This will generate a corresponding modified adjoint system (we re-use $\tilde{f}$ to represent the

index-reduced version of (5.112)):

$$\begin{aligned}
\tilde{f} = 0,\ \tilde{g} - s = 0 \\
\dot{p} = \tilde{f}_x^T y + \tilde{g}_x^T z \\
p = \tilde{f}_{\dot{x}}^T y + \tilde{g}_{\dot{x}}^T z \\
0 = \tilde{f}_u^T y + \tilde{g}_u^T z
\end{aligned} \tag{5.113}$$

$$(s_i - \underline{\tilde{g}}_i)\underline{z}_i \ = \ \mu,\ \ (\overline{\tilde{g}}_i - s_i)\overline{z}_i \ = \ \mu,\ \ z_i = \underline{z}_i - \overline{z}_i,\ \ \forall i$$

which must be satisfied at each point along the constrained trajectory. We may incorporate this modified system into the algorithm in two ways, outlined below.

5.9.1.1 Modified Basic Algorithm - I

The first option maintains the flavour of the basic algorithm, integrating the state dynamics in the forward direction and the adjoint dynamics in the backward direction. The modified algorithm (without endpoint equality constraints) has the following form:

1. Given $u'(t), t \in (t_0, t_f)$ and v, set $p(t_0) = 0$.

2. Determine initial conditions $\dot{x}(t_0), x(t_0)$ and $u(t_0)$ from (5.11a).

3. Integrate $f(t, \dot{x}, x, u, v) = 0$ from t_0 to t_f, obtaining $(\dot{x}, x, x')$ for the current $u'(t)$. If at any step from t to $t + \Delta t$, $\overline{g}(t, \dot{x}, x, u, v)$ becomes binding, re-initialise the system at the point of activation using the modified system dynamics $\tilde{f}$ and determine the corresponding control. At the next step, attempt to use the current $u'(t)$, as computed on the backward integration.

4. Determine the final condition $p(t_f)$ from (5.12).

5. Integrate (5.16) from t_f to t_0, obtaining $(u, s, p, \dot{x})$ for fixed $x(t), v$.

6. Update v from (5.11b).

7. Repeat from step 2 to convergence.

5.9.1.2 Modified Basic Algorithm - II

As an alternative, we consider the following modification of the basic algorithm (without endpoint equality constraints):

1. Given v and $p(t_0)$.

2. Determine initial conditions $\dot{x}(t_0), x(t_0)$ and $u(t_0)$ from (5.11a).

3. Integrate (5.16) from t_0 to t_f, obtaining $(\dot{x}, x, \dot{p}, p, u, s)$. If at any step from t to $t + \Delta t$, $\overline{g}(t, \dot{x}, x, u, v)$ becomes binding, switch to the modified system dynamics $\tilde{f}$ and corresponding modified adjoint system (5.113).

4. Determine final condition $p(t_f)$ from (5.12).

5. Integrate (5.16) from t_f to t_0, obtaining $(\dot{x}, x, \dot{p}, p, u, s)$ for fixed v. If at any t, $\overline{g}(t, \dot{x}, x, u, v)$ becomes binding, switch to the modified system dynamics $\tilde{f}$ and modified adjoint system (5.113).

6. Update v from (5.11b).

7. Repeat from step 2 to convergence.

Detection of activation of the state constraint is easily implemented using the IMPDSC subroutine in DASOLV. DASOLV handles implicit discontinuities using a discontinuity locking mechanism (see Figure 5.3, Jarvis (1992)). At the beginning of each time step in the integration, the active branch of a discontinuous equation system is determined. This branch is then "locked" in for the duration of the step. Once a successful step has been taken, DASOLV checks to see whether or not the system is still within the same active branch. If it is not, a bisection search coupled with an interpolation procedure is used to determine the position at which the system change occurs (C in Figure 5.3). The integration then restarts from point C with the equation locked into the new active branch over the next time step, finishing at D.

In the presence of endpoint equality constraints in the state, we adopt a variation of the Miele approach outlined in Chapter 2, Section 2.4.3.2. Thus, at Step 4, we determine $p(t_f)$ for the required end conditions on $x(t_f)$. We imbed the algorithm in a loop in order to generate linear combinations, as required by the Miele approach. Thus, starting from Step 4, we loop $\bar{n}_e + 1$ times through Steps 2 - 5, initialising $p^i(t_f)$ at each t_f choosing a different ξ_i from a basis of multipliers $\{\xi_i\}_{i=1}^{\bar{n}_e+1}$. We then generate a linear combination which reduces both the equality constraint residual and cost based on the previous values. We repeat this looping procedure until we reach a solution.

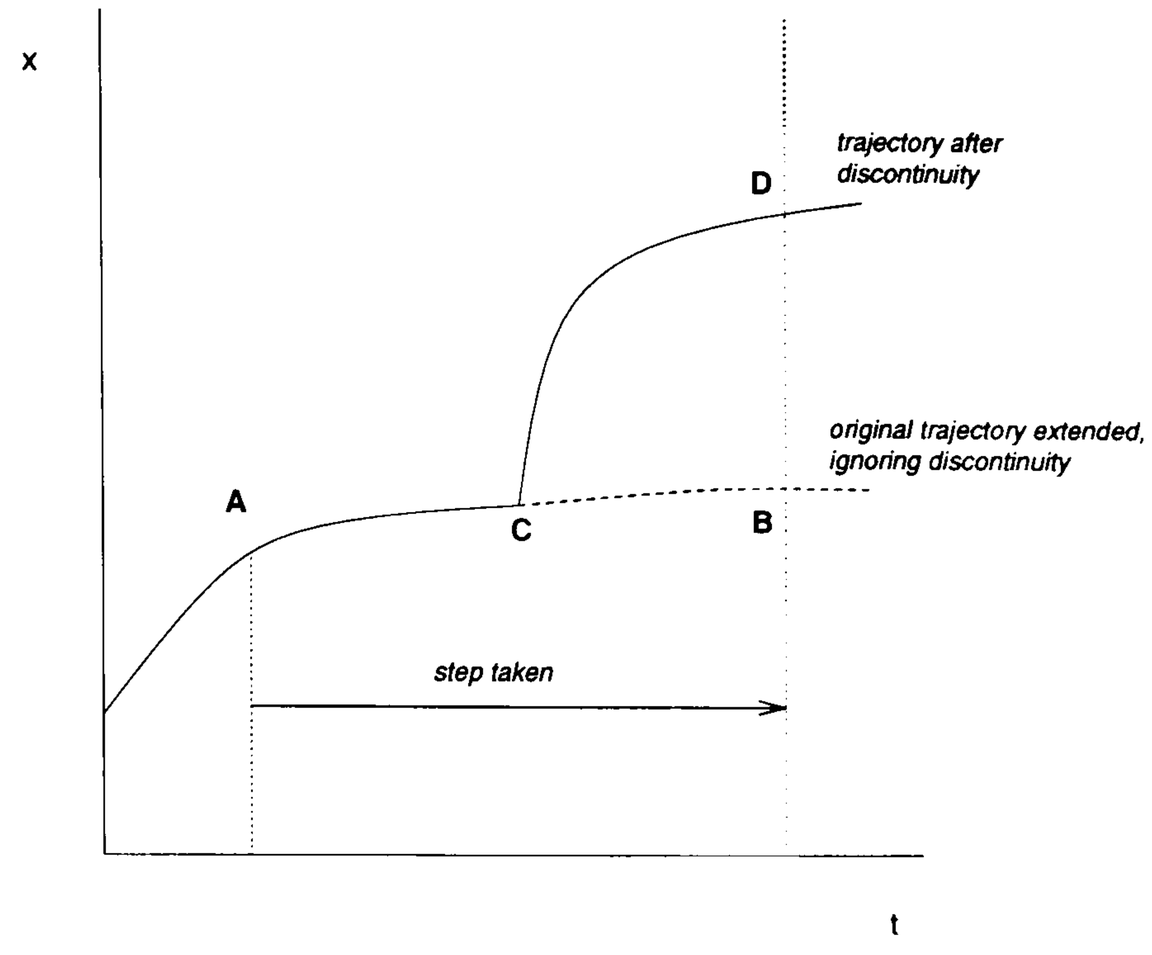


Figure 5.3: DASOLV: "Locking" of an implicit discontinuity

5.10 An Illustrative Example

We shall now consider a simple example to illustrate the application of the theory developed in Chapter 4 and extended in this chapter. The problem we shall consider is as follows:

Car Problem

$$\max_u x_1 \tag{5.114}$$

subject to

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= u(t) \\ -1 \leq & \quad u \quad \leq 1 \\ x_2 &\leq 0.4 \end{aligned} \tag{5.115}$$

where $0 \leq t \leq 1$, $x(0) = [0 \quad 0]$ and $x_2(1) = 0$. Therefore, we wish to maximise the distance travelled in a fixed time, subject to constraints on the acceleration and velocity, starting and finishing at rest.

5.10.1 Direct Application of the Necessary Conditions

In this section we demonstrate the suitability of the necessary conditions (5.16).

Clearly, because of (5.114), the optimal policy will be to make x_1 as large as possible, subject to the velocity constraint $x_2(t) \leq 0.4$ and the boundary condition $x_2(1) = 0$. This generates the following optimal policy,

$$u = \begin{cases} 1, & t \in [0, 0.4) \\ 0, & t \in [0.4, 0.6] \\ -1, & t \in (0.6, 1], \end{cases} \quad x_2 = \begin{cases} t, & t \in [0, 0.4) \\ 0.4, & t \in [0.4, 0.6] \\ 1 - t, & t \in (0.6, 1]. \end{cases} \tag{5.116}$$

We shall now determine the multipliers and adjoint variables and verify the necessary conditions. From (4.48) we have

$$\begin{aligned} \dot{p}_1 &= 0 \\ \dot{p}_2 &= -y_1 + z_2 \\ p_1 &= y_1 \\ p_2 &= y_2 \\ -y_2 + z_1 &= 0 \\ \underline{z}_1(u+1) &= 0, \quad \underline{z}_1 \geq 0 \\ \overline{z}_1(1-u) &= 0, \quad \overline{z}_1 \geq 0 \\ z_1 &= \underline{z}_1 - \overline{z}_1 \\ \overline{z}_2(0.4 - x_2) &= 0, \quad \overline{z}_2 \geq 0 \\ z_2 &= -\overline{z}_2 \end{aligned} \tag{5.117}$$

The boundary conditions for $p(t_f)$ are determined by (4.43) as $\nu^f \to 0$:

$$\begin{aligned} p(t_f) &= P_x^T + \phi_x^T \xi^f \\ \therefore p(t_f) &= \begin{bmatrix} -1 \\ \xi^f \end{bmatrix}. \end{aligned} \tag{5.118}$$

Consider $t = 0^+$. Since $\dot{p}_1 = 0$ and $p_1(t_f) = -1$, we have $p_1 = y_1 = -1$. Furthermore, $x_2(0) < 0.4$ and therefore $\overline{z}_2 = 0$. Thus, $\dot{p}_2 = -y_1 = 1$. Suppose $p_2(0) \geq 0$. Then $p_2(0^+) > 0$ and hence $z_1 > 0$, $\underline{z}_1 > 0$, $u = -1$. This implies x_2 decreases and, since $\dot{p}_2 = 1$, there can be no further switch, implying $x_2(1) < 0$, a contradiction.

Hence $p_2(0) < 0$ and $p_2(0^+) < 0$. Hence $z_1 < 0$, $\overline{z}_1 > 0$, $u = +1$. This continues until either $p_2 = 0$ or $x_2 = 0.4$.

Suppose $p_2(\underline{t}) = 0$, $x_2(\underline{t}) < 0.4$. As above, this implies a switch to $u = -1$, which continues to $t = 1$. But then $x_2(1) = \underline{t} - (1 - \underline{t}) = 0$, whence $\underline{t} = \frac{1}{2}$, $x_2(\underline{t}) = \frac{1}{2}$, a contradiction. Hence, $x_2(\underline{t}) = 0.4$, $\underline{t} = 0.4$ and $p_2(\underline{t}) \leq 0$. We must switch to $u \leq 0$ at $\underline{t}$ to maintain $x_2 \leq 0.4$, so $\overline{z}_1 = 0$, $z_1 = \underline{z}_1 \geq 0$, $p_2 = z_1 \geq 0$. Thus, $p_2(\underline{t}) = 0$.

As above, $\dot{p}_2(\underline{t}) > 0$ implies $u = -1$, $t > \underline{t}$ and again $\underline{t} = \frac{1}{2}$, a contradiction. Similarly, $\dot{p}_2(\underline{t}) < 0$ implies $p_2(\underline{t}^+) < 0$, $z_1 < 0$, $\overline{z}_1 > 0$, $u = 1$. This violates $x_2 \leq 0.4$, so we must have $\dot{p}_2(\underline{t}) = 0$, $\overline{z}_2 = 1$, $x_2 = 0.4$. Then $p_2(\underline{t}^+) = 0$, so $z_1 = 0$, implying $\underline{z}_1 = \overline{z}_1 = 0$, so $-1 < u < 1$. This cannot continue until $t = 1$. Therefore, as above, we must finally switch to $u = -1$, $t \geq \underline{t}$, choosing $\underline{t}$ so that $x_2 = 0$ at $t = 1$ (i.e. $\underline{t} = 0.6$).

This therefore shows that the necessary conditions determine a unique trajectory $x(\cdot)$ and unique adjoints and multipliers, summarised below:

$$y_1 = p_1 = -1, \quad \forall t, \quad y_2 = p_2 = \begin{cases} t - 0.4, & t \in [0.0.4) \\ 0, & t \in [0.4, 0.6] \\ t - 0.6, & t \in (0.6, 1] \end{cases}$$

$$z_1 = \begin{cases} t - 0.4, & t \in [0, 0.4) \\ 0, & t \in [0.4, 0.6] \\ t - 0.6, & t \in (0.6, 1], \end{cases} \qquad z_2 = \begin{cases} 0, & t \in [0.0.4) \\ 1, & t \in [0.4, 0.6] \\ 0, & t \in (0.6, 1]. \end{cases} \tag{5.119}$$

We now turn to examining the adjoint variables in greater detail. Continuity of p requires $p_2(t_f) = \xi^f = 0.4$. We are now left to show that the proposed method of Section 5.8.1 for dealing with endpoint equality constraints yields the correct value for ξ^f.

We split the adjoint variable p into two parts:

$$\begin{aligned} \dot{\tilde{p}} &= f_x^T \tilde{y} + g_x^T z = \begin{bmatrix} 0 \\ -\tilde{y}_1 + z_2 \end{bmatrix} \\ \dot{\bar{p}} &= f_x^T \bar{y} = \begin{bmatrix} 0 \\ -\bar{y}_1 \end{bmatrix}. \end{aligned} \tag{5.120}$$

where $y = \tilde{y} + \bar{y}\xi^f$, $p = \tilde{p} + \bar{p}\xi^f$ and

$$\begin{aligned} \tilde{y} &= \tilde{p} \\ \bar{y} &= \bar{p}. \end{aligned} \tag{5.121}$$

We have $\bar{p}(t_f) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and therefore $\bar{p}(t) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Similarly, $\tilde{p}(t_f) = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$ and therefore

$$\tilde{p}_2 = \tilde{y}_2 = \begin{cases} t - 0.8, & t \in [0, 0.4) \\ -0.4, & t \in [0.4, 0.6] \\ t - 1, & t \in (0.6, 1]. \end{cases} \tag{5.122}$$

In order to generate $\delta u' = \delta\tilde{u}' + \delta\bar{u}'\bar{\xi}^f$, we look at the linearisation of (5.16) about $(\dot{x}, x', u', y, z, s)$. Linearising $f = 0$ yields:

$$\begin{bmatrix} \delta\dot{x} \\ \delta x' \end{bmatrix} = - \begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}^{-1} f_{u'}\delta u'. \tag{5.123}$$

Linearising the subsystem:

$$\begin{aligned} f_{\dot{x}}^T y + g_{\dot{x}}^T z - p &= 0 \\ f_{x'}^T y + g_{x'}^T z &= 0 \end{aligned} \tag{5.124}$$

gives the following equation relating $\delta u'$ to δy:

$$\delta y = - \begin{bmatrix} f_{\dot{x}} \\ f_{x'} \end{bmatrix}^{-1} \left(\begin{bmatrix} r_{\dot{x}} \\ r_{x'} \end{bmatrix} + M\delta u' \right) \tag{5.125}$$

where

$$\begin{aligned} \begin{bmatrix} r_{\dot{x}} \\ r_{x'} \end{bmatrix} &= \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix} y + \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} (z - z^*) - \begin{bmatrix} p \\ 0 \end{bmatrix} \\ M &= -H_{\kappa\kappa} \begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}^{-1} f_{u'} + H_{\kappa u'} - \\ \begin{bmatrix} g_{\dot{x}}^T \\ g_{x'}^T \end{bmatrix} (\underline{S}^{-1}\underline{Z} &+ \overline{S}^{-1}\overline{Z})(g_{u'} - \begin{bmatrix} g_{\dot{x}} & g_{x'} \end{bmatrix} \begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}^{-1} f_{u'}) \\ z^* &= \underline{S}^{-1}\underline{r} - \overline{S}^{-1}\overline{r} + (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})\,(g - s) \end{aligned} \tag{5.126}$$

and $\kappa = [\dot{x},\ x']$, $H_{\kappa\kappa} = \sum_i y^i f^i_{\kappa\kappa} + \sum_i z^i g^i_{\kappa\kappa}$ for some κ.

Substituting this expression into the linearisation of $f_{u'}^T y + g_{u'}^T z = 0$, we obtain:

$$\delta u' = -W^{-1} \left[f_{u'}^T \left(y - \begin{bmatrix} f_{\dot{x}}^T \\ f_{x'}^T \end{bmatrix}^{-1} \begin{bmatrix} r_{\dot{x}} \\ r_{x'} \end{bmatrix} \right) + g_{u'}^T (z - z^*) \right] \tag{5.127}$$

where

$$W = H_{u'u'} - \begin{bmatrix} H_{u'\dot{x}} & H_{u'x'} \end{bmatrix} \begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}^{-1} f_{u'} + f_{u'}^T M - g_{u'}^T (\underline{S}^{-1}\underline{Z} + \overline{S}^{-1}\overline{Z})(g_{u'} - \begin{bmatrix} g_{\dot{x}} & g_{x'} \end{bmatrix} \begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}^{-1} f_{u'}). \quad (5.128)$$

In the limit as $\mu \to 0$, $\underline{r} \to 0$, $\overline{r} \to 0$, $g \to s$, we have

$$\delta u' = -W^{-1}(f_{u'}^T p + g_{u'}^T z). \quad (5.129)$$

We partition $\delta u' = \delta \tilde{u}' + \delta \bar{u}' \bar{\xi}^f$ such that

$$\begin{aligned} \delta \tilde{u}' &= -W^{-1}(f_{u'}^T \tilde{p} + g_{u'}^T z) = -W^{-1}(-\tilde{p}_2 + z_1) \\ \delta \bar{u}' &= -W^{-1} f_{u'}^T \bar{p} = W^{-1} \bar{p}_2. \end{aligned} \quad (5.130)$$

Simple substitutions yield

$$\dot{I}_{\phi P} = W^{-1}(-\tilde{p}_2 + z_1), \quad \dot{I}_{\phi\phi} = -W^{-1} \quad (5.131)$$

and, since $I_{\phi P}(0) = I_{\phi\phi}(0) = 0$,

$$I_{\phi P} = 0.4W^{-1}, \quad I_{\phi\phi} = -W^{-1}. \quad (5.132)$$

Finally,

$$\begin{aligned} \xi^f &= I_{\phi\phi}^{-1}\{-I_{\phi P}\} \\ &= -W\{-0.4W^{-1}\} \to 0.4. \end{aligned} \quad (5.133)$$

Therefore, we see that the necessary optimality conditions formulated in this thesis are consistent with the intuitive optimal solution.

5.10.2 An Alternative Method of Handling State Inequality Constraints

An apparently simple solution for the car problem is to replace the state constraint with the following point constraint (formulated as a new state):

$$\dot{x}_3 = (x_2 - 0.4)^2 H, \qquad x_3(t_0) = x_3(t_f) = 0 \quad (5.134)$$

where

$$H = \begin{cases} 0, & x_2 < 0.4 \\ K_s, & x_2 \geq 0.4. \end{cases} \quad (5.135)$$

Therefore, we now have a standard optimal control problem without state inequality constraints. We now apply the necessary conditions (5.16) to this modified problem and find:

$$\begin{aligned} \dot{p}_1 &= 0 \\ \dot{p}_2 &= -y_1 - 2(x_2 - 0.4)Hy_3 \\ \dot{p}_3 &= 0 \end{aligned} \tag{5.136}$$

Thus, we have $p_1(t) = -1, \;\; p_3(t) = \xi_2$. This implies $y_1(t) = -1$ and $y_3(t) = \xi_2$. From the previous section, we expect $\dot{p}_2 = -y_1 + k$, where $k = 0$ when the constraint is inactive and $k = -1$ when the state constraint activates. It is easy to see that the same result will hold for the inactive constraint when using the point constraint approach. However, once the constraint is active, there is no way of guaranteeing $2(x_2 - 0.4)K_s y_3 = 1$. Consider the following. We remove the assumption of normality in (5.12) and replace the unit multiplier on the cost function with π. This gives us $y_1(t) = p_1(t) = -\pi$. If $x_2 > 0.4$ over any interval, (5.134) gives $x_3^f > 0$. Therefore, we must have $x_2 \leq 0.4$ for all t, and

$$\begin{aligned} \dot{p}_2 &= -y_1 = \pi \\ \therefore p_2 &= \pi(t-1) + \xi_1. \end{aligned} \tag{5.137}$$

For $\pi = 1$:

$$\begin{aligned} p_2 < 0 \;\text{ if }\; t < 1 - \xi_2 \;\; &\Rightarrow \;\; u = +1 \\ p_2 > 0 \;\text{ if }\; t > 1 - \xi_2 \;\; &\Rightarrow \;\; u = -1. \end{aligned} \tag{5.138}$$

In order to satisfy $x_2^f = 0$, we require $\xi_2 = \frac{1}{2}$ (i.e. a switch at $t = \frac{1}{2}$). However, this violates the constraint $x_2 \leq 0.4$ and therefore presents a contradiction. Hence, we must have $\pi = 0$. The numerical solution with $\pi = 1$ cannot converge to the correct solution. This non-normality associated with using the above transformation for active state constraints is discussed in (Hartl *et al.*, 1995).

If we now formulate the problem using a penalty function approach, we have

$$\min_u -x_1 + x_3. \tag{5.139}$$

Thus, we reformulate the problem such that we minimise $x_3(t_f)$ as opposed to enforcing $x_3(t_f) = 0$. For this case, we have $p_1(t) = -1, \quad p_3(t) = 1$. Therefore, we require $2(x_2 - 0.4)K_s = 1$. If we formulate the penalty as $\dot{x}_3 = (x_2 - 0.4)H$, we then require $K_s = 1$. This demonstrates the non-arbitrary nature of the penalty parameter K_s. The

analysis above is applicable in this case, as the system equations on $t \in (t_0, t_f)$ remain the same, the only change being that $y_3 = p_3 = p_3^f = \pi K_s$.

Therefore, it has been shown that neither a point constraint nor a penalty function approach provides a suitable means of transforming the path constrained problem into a sensible unconstrained problem. The above analysis demonstrates the non-normal nature of such transformations. The transformation from a state constrained problem to an unconstrained problem results in the loss of essential information.

5.11 Conclusions

A new algorithm is presented for model-based optimal control. The basic algorithm uses a new approach for computing optimal controls, designed to apply to DAE systems of index one and to allow for general constraints along the trajectory and at the initial and end points. Inequality constraints are transformed to equality constraints by introducing a slack variable. Deviations from the feasible region are prevented via a logarithmic barrier function.

The chapter discusses alternative formulations for dealing with endpoint equality constraints in the states. The first option splits the adjoint into two parts, one measuring the effect of changes in the control and optimisation parameters on the cost and the other measuring their effect on the endpoint equality. The second option requires the system to be integrated $\bar{n}_e + 1$ times for each outer iteration to generate an update for the multipliers. These two methods will be compared in the next chapter using a case study.

Path constraints present a challenging problem. Two basic approaches were proposed for dealing with path constraints. For both approaches, one generates an index one system by appropriate differentiations and substitutions of the system equations and the algebraic equation associated with the path constraint. This may be easily incorporated into the basic algorithm, employing the automatic differentiation routine of Pantelides *et al.* (1994), or determined *a priori*. Thus, we carry two systems of equations, one being the original system and the second being the modified system for when the constraint is binding. The first approach recognises that due to stability restrictions, we integrate the state dynamics in the forward direction and the adjoint dynamics in the backward direction for fixed state. We implement the control generated on the backward pass as we

integrate the state dynamics forward, modifying this control if the trajectory results in the transition from an infeasible to a feasible region of the state space. Thus at every time t, we attempt to implement the calculated optimal control. If this results in an infeasible step, we instead use the control determined by the modified system. The second approach integrates the state and adjoint dynamics in parallel in both the forward and backward direction. Again, the dynamics switch from the original to the modified system when the constraint becomes active. This second method relies on stability of both the state and adjoint equations in both the forward and backward direction.

Many chemical process systems are of index greater than one. Furthermore, mismatch between the true plant and the model of the plant, as well as plant disturbances, eliminate robustness characteristics in this approach. In the coming chapters we will address both of these issues and propose methods to deal both with uncertainty and with high-index systems.

CHAPTER 6

Numerical Validation

Optimal control problems involving differential-algebraic systems with inequality constraints on controls, the path and the endpoint, and equality constraints at the endpoints arise frequently in dynamic systems. The algorithm developed in previous chapters provides a means of dealing with such problems. This chapter will outline several such systems and illustrate the effectiveness of the proposed algorithm in determining the optimal control trajectory and/or optimal parameters of the system.

The case studies presented have been solved using various tolerance levels for both the integration and optimisation phases of the algorithm. The integration tolerance corresponds to the tolerance employed by the integrator to control the local error of the trajectory at each integration step. The optimisation tolerance controls the satisfaction of equality constraints and provides a gauge for the objective function variance from iteration to iteration. Since the state trajectory provides a basis for computing the new control trajectory, the integration tolerance is necessarily tighter than the optimisation tolerance. In practice, testing constraint violations and objective function variance provides an earlier termination point than the residuals of the necessary conditions.

All results are reported for a SUN SPARC 5 workstation. Where possible, results are compared with alternative algorithms, such as DAEOPT (Vassiliadis, 1993a). The algorithm developed in this thesis necessarily uses a finer parameterisation of the problem than either DAEOPT (Vassiliadis, 1993a) or the collocation approach of (Cuthrell and Biegler, 1987), (Renfro *et al.*, 1987) or (Vasantharajan and Biegler, 1990), and hence may obtain a better solution than these methods for a given optimisation tolerance.

6.1 Consecutive First Order Reactions

We begin by considering the problem of two first order reactions occurring in series (Vassiliadis, 1993b). The objective is to maximise the concentration of the intermediate product, x_2, by manipulating the reaction temperature u.

$$\max_u P(x_{t_f}) = x_2(t_f) \tag{6.1}$$

subject to

$$\begin{aligned} \dot{x}_1(t) &= -k_1 x_1(t) \\ \dot{x}_2(t) &= k_1 x_1(t) - k_2 x_2(t) \\ k_1 &= 65.50 \exp(-5027.7/u) \\ k_2 &= 1970.0 \exp(-8044.3/u) \end{aligned} \tag{6.2}$$

with $x(0) = [1 \;\; 0]$, $0 \leq t \leq 25$ and $600.0 \leq u(t) \leq 1000.0$.

As in (Vassiliadis, 1993b), the control profile was initialised at a constant temperature of $800K$. This reaction scheme is characterised by a steep optimal temperature profile, thereby making the determination of the control difficult. For an integration tolerance of 10^{-6} and an optimisation tolerance of 10^{-3}, the algorithm produced a solution of 0.5412 in 13 iterations and 292.5 CPU seconds. The corresponding temperature profile is shown in Figure 6.1. The optimal objective of 0.54190 was achieved in 20 iterations and 380.6 CPU seconds, using an integration tolerance of 10^{-8} and an optimisation tolerance of 10^{-5}. The temperature profile is shown in Figure 6.2. The corresponding concentration profile is shown in Figure 6.3 and the convergence rates are illustrated in Figure 6.4.[1] The comparison with (Vassiliadis, 1993b)[2] presents some interesting conclusions. Vassiliadis is able to achieve the latter optimisation tolerance in 92.4 CPU seconds, with an objective of 0.541957. However, his best result of 0.54201 is achieved in 3337.5 CPU seconds, with a total of 181 sensitivity integrations and 203 state integrations. We are able to achieve a total of 240 integrations, 2 integrations forward and 10 integrations backward per iteration, in less than 15% of this time. Thus, our cost per integration is significantly lower than the control parameterisation approach proposed in (Vassiliadis, 1993b), even though our parameterisation is much finer than his.

---

[1] In this case, and in all other examples, the error corresponds to the log norm of the deviation of the objective from the optimum.

[2] All results reported in this chapter from (Vassiliadis, 1993b) are for a SUN SPARC 2

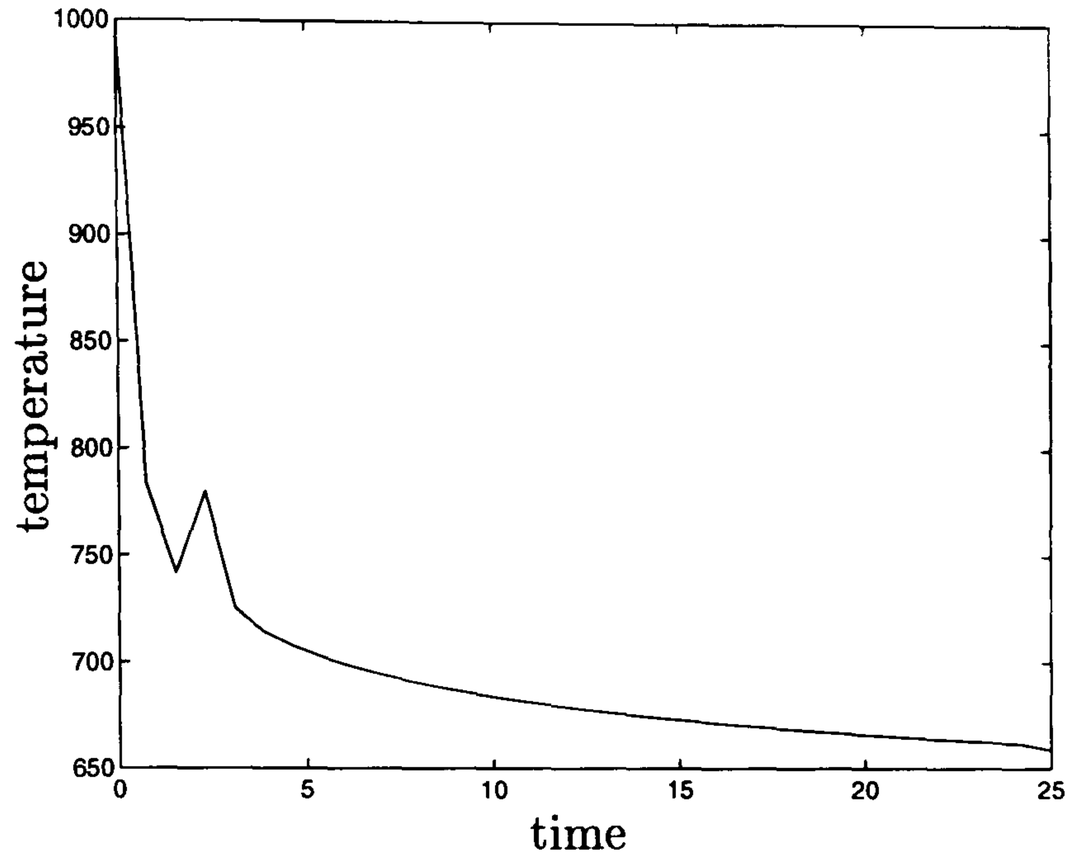


Figure 6.1: Consecutive Reactor: Temperature Profile 1

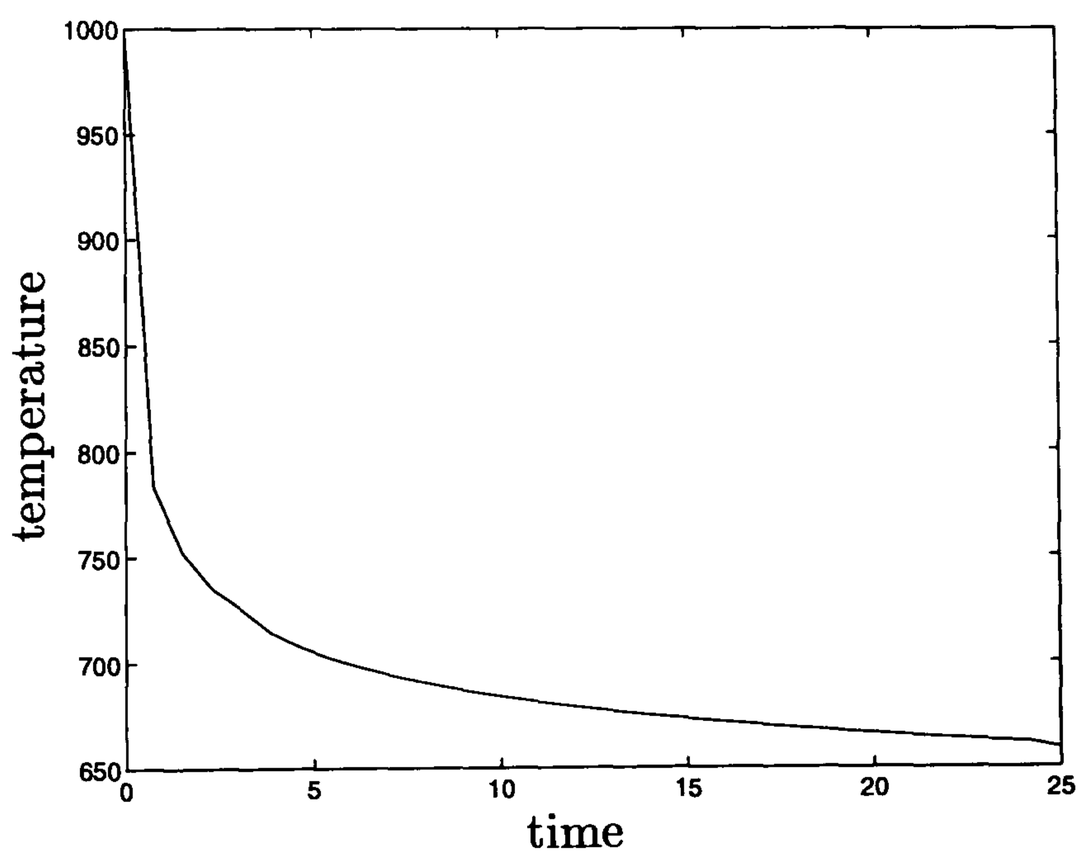


Figure 6.2: Consecutive Reactor: Temperature Profile 2

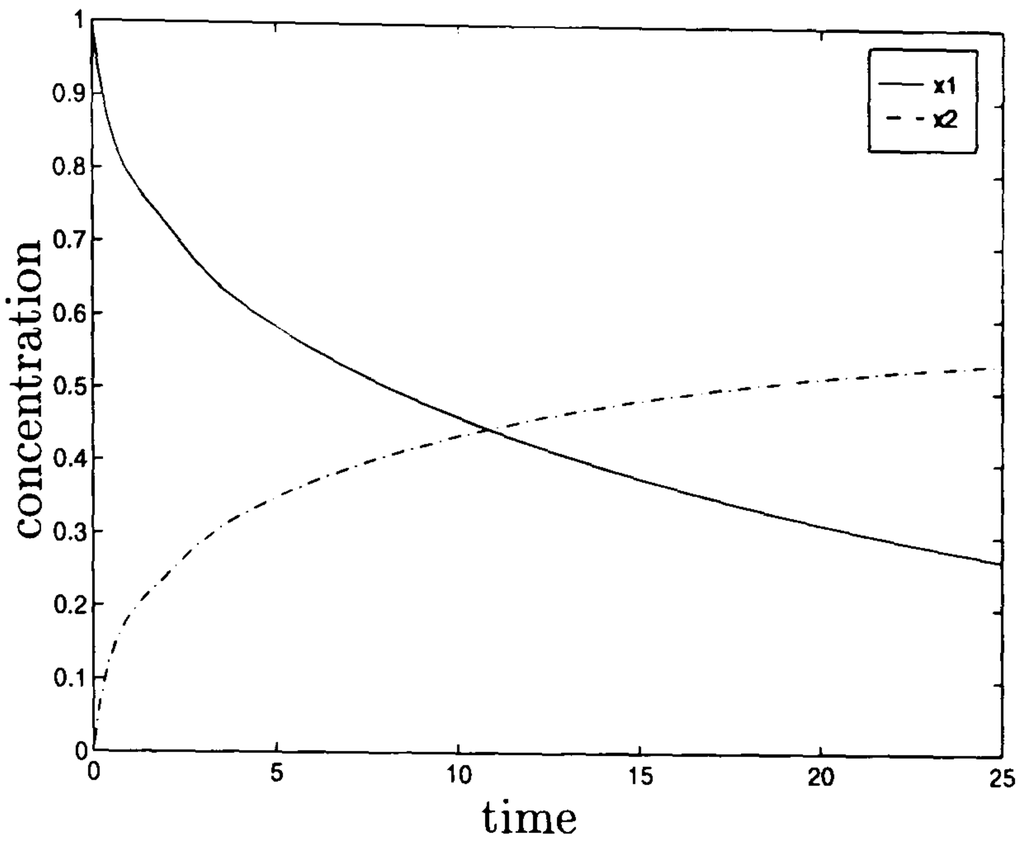


Figure 6.3: Consecutive Reactor: Concentration Profiles

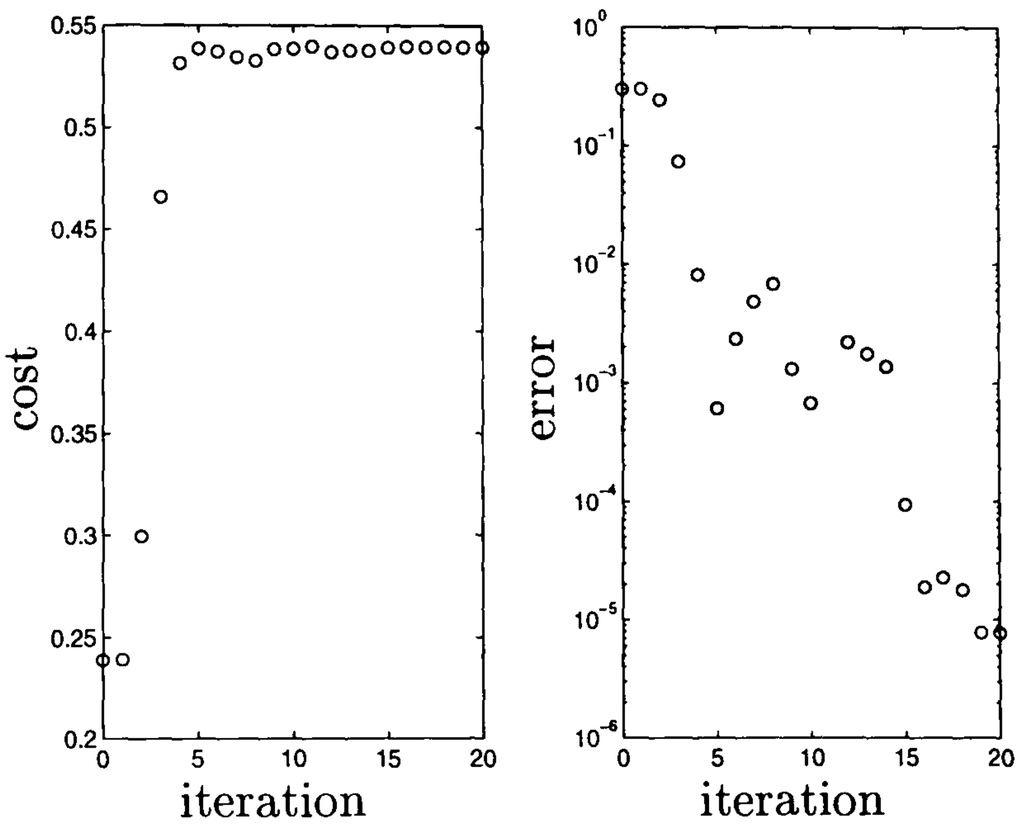


Figure 6.4: Consecutive Reactor: Cost and Error Profile

6.2 Batch Reactor with Control Saturation

We consider the batch reactor problem of (Ray, 1981), as considered by (Logsdon and Biegler, 1992)[3] and (Vassiliadis, 1993b), among others.

$$\max_{u} P(x_{t_f}) = x_2(t_f) \tag{6.3}$$

[3]The results quoted in (Logsdon and Biegler, 1992) are for a Vax 6320

subject to

$$\begin{aligned} \dot{x}_1(t) &= -(u + u^2/2)x_1 \\ \dot{x}_2(t) &= ux_1 \end{aligned} \tag{6.4}$$

with $x(0) = [1 \;\; 0]$, $0 \leq t \leq 1$ and $0.0 \leq u(t) \leq 5.0$.

We choose a constant initial control profile of 1.0. Notice that the control saturates at its upper bound as we approach the end of the horizon. An optimisation tolerance of 10^{-3} is achieved in 8 iterations and 376.1 CPU seconds, with an objective value of 0.573. Using an integration tolerance of 10^{-6} and an optimisation tolerance of 10^{-4}, a maximum concentration of x_2 of 0.5735 was achieved within 14 major iterations, with an overall CPU time of 418.2 seconds. Approximately 75% of this time is spent on the first two iterations, generating a good update from the initial guess. The optimal control profile is shown in Figure 6.5 and the optimal concentrations in Figure 6.6, for the latter optimisation tolerance. Figure 6.7 illustrates convergence. Thus, the objective result compares favourably with the results reported by (Logsdon and Biegler, 1992) and (Vassiliadis, 1993b), which varied between 0.5691 and 0.57354. This upper value, reported by (Vassiliadis, 1993b), was for an extremely tight optimisation tolerance of 10^{-10}. The computational times reported by (Vassiliadis, 1993b) are in the range of $32.0 - 1145$ CPU seconds. In order to achieve a value of 0.57354, DAEOPT required 1145 CPU seconds, with 108 quadratic programming solutions and 136 line search function evaluations.

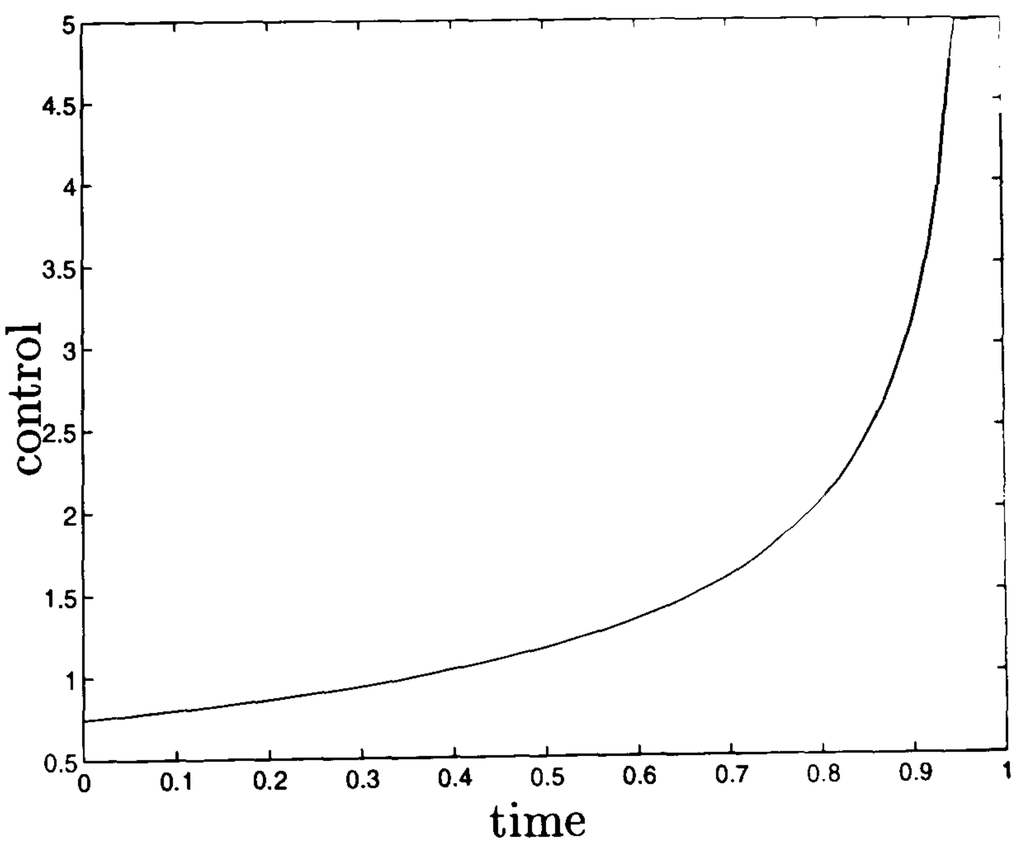


Figure 6.5: Batch Reactor: Control Profile

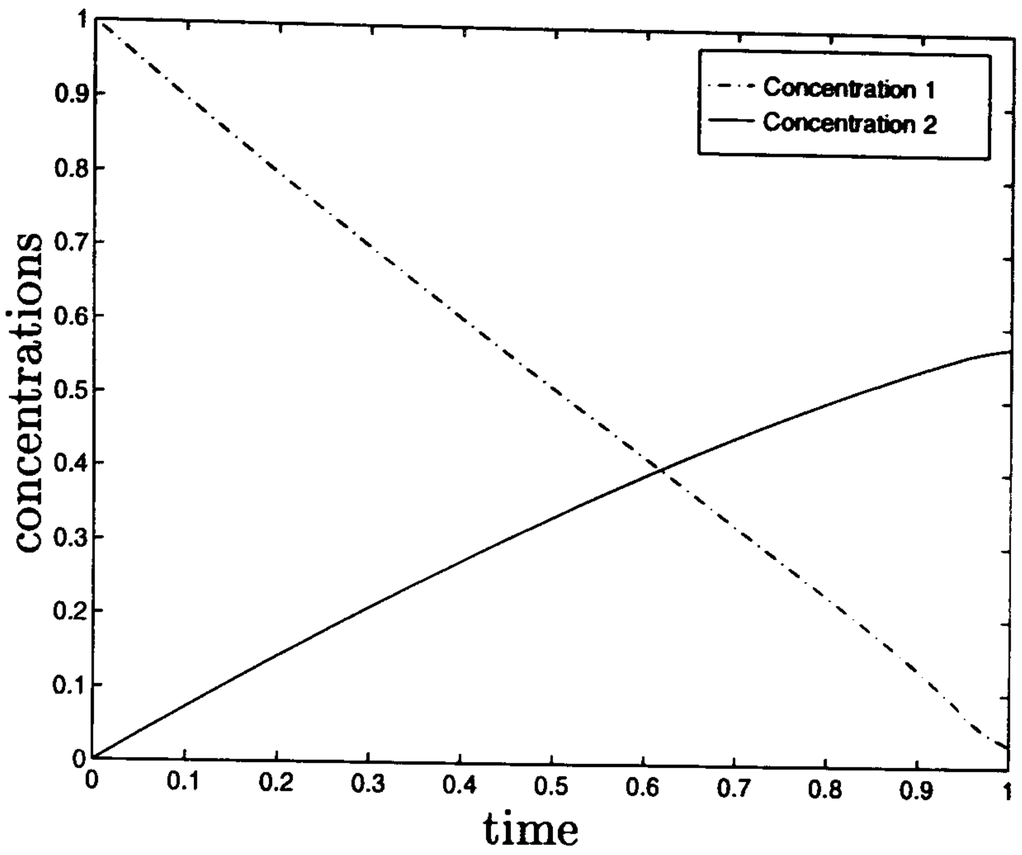


Figure 6.6: Batch Reactor: Concentration Profiles

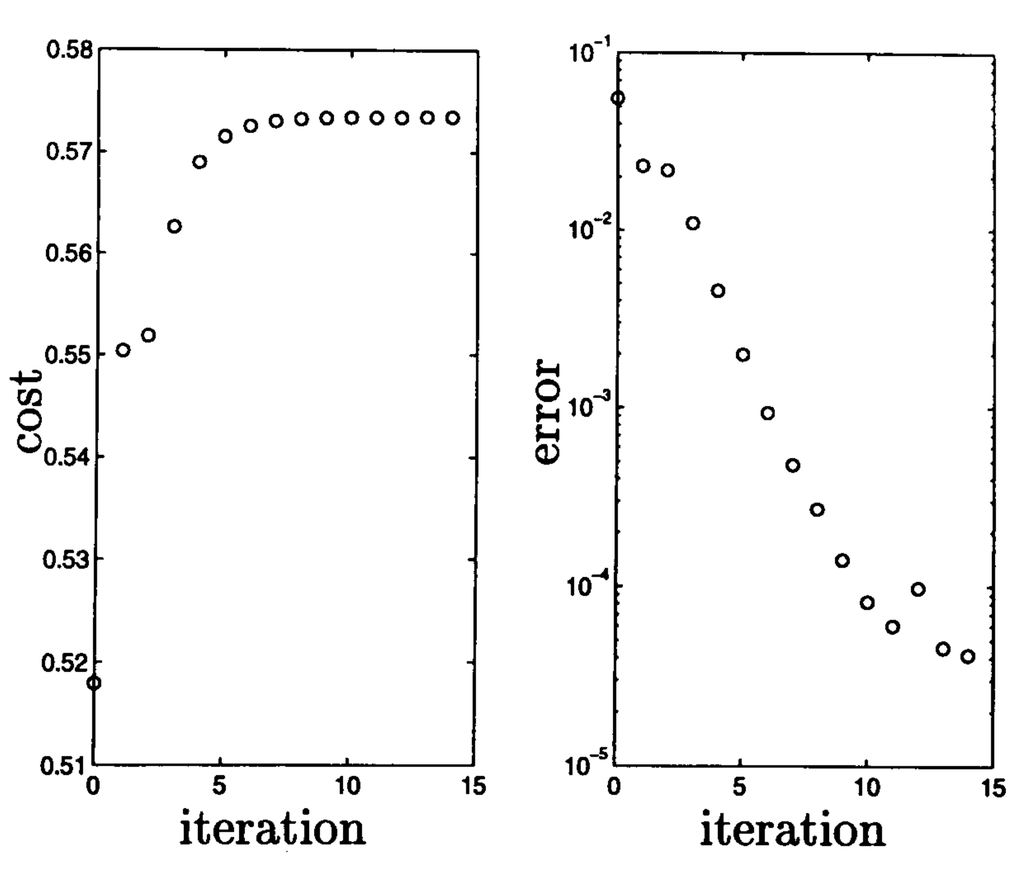


Figure 6.7: Batch Reactor: Cost and Error Profiles

6.3 Catalyst Mixing Problem

This problem determines the optimal mixing policy of two catalysts along the length of a tubular reactor. The mixing ratio of the catalysts is the control variable and its optimal profile has a singular arc, making this a difficult problem to solve. The presence of slack

variables in our approach allows us to iterate with $|g - s| \to 0$ as we converge towards the optimal solution.

The problem was first posed by (Gunn and Thomas, 1965) and has been solved by (Logsdon, 1990), using orthogonal collocation on finite elements, and by (Vassiliadis, 1993b) using the control parameterisation technique. The problem is described as follows:

$$\max_{u(\cdot)} = x_3(1.0) \tag{6.5}$$

subject to

$$\begin{aligned} \dot{x}_1(t) &= u(10.0x_2 - x_1) \\ \dot{x}_2(t) &= u(x_1 - 10.0x_2) - (1.0 - u)x_2 \\ x_3 &= 1.0 - x_1 - x_2 \end{aligned} \tag{6.6}$$

with $x_1(0) = 1.0, x_2(0) = 0.0$ and $0.0 \leq u(t) \leq 1.0$ on $t \in [0, 1]$.

The optimal mixing policy involves two switches. In the control parametrisation approach, the controls can be restricted to piecewise constant, eliminating the possibility of a smooth optimal trajectory. In our approach, we allow more flexibility in the determination of the control trajectory and, hence, end up with a smooth approximation of the actual singular trajectory. We choose a uniform initial mixing ratio of 0.170. While the singular arc is not precisely resolved, the optimal cost calculated by our algorithm is extremely close to the values determined in the above references. Our result of 0.0480 is favourable to the best objective value of 0.0480557 reported by (Vassiliadis, 1993b) using an optimisation tolerance of 10^{-7}. The optimal value in this thesis was obtained for an integration tolerance of 10^{-6} and an optimisation tolerance of 10^{-4}. The CPU time required by Vassiliadis was 14.4 seconds. In this case, we required 4393.4 CPU seconds, and 45 iterations to achieve the desired tolerance. This is a significant difference in computational time, but again it is a direct result of the freedom in generating a smooth trajectory. An optimisation tolerance of 10^{-3} would be achieved in 14 iterations, with a CPU time of 1366.8 seconds. The optimal mixing ratio and corresponding state profiles are shown in Figures 6.8 and 6.9 respectively for the tightest optimisation tolerance. The convergence is illustrated in Figure 6.10.

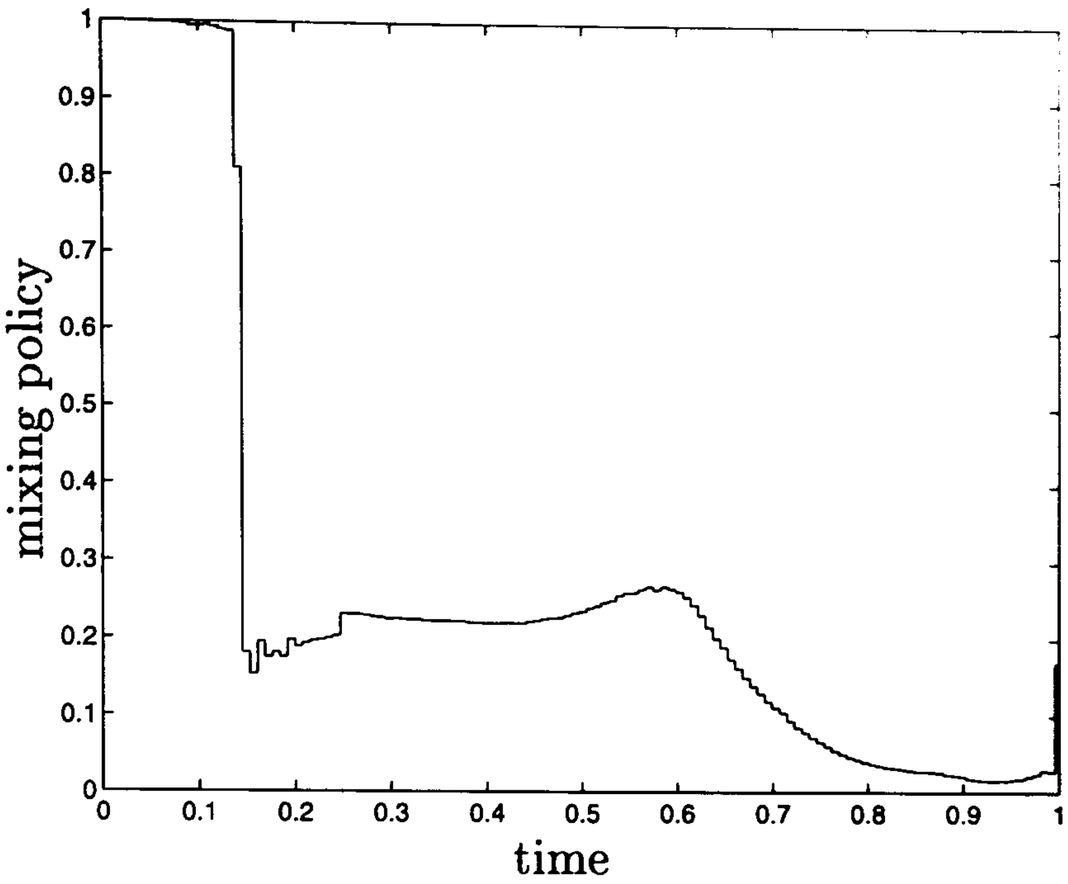


Figure 6.8: Catalyst Mixing: Control Profiles

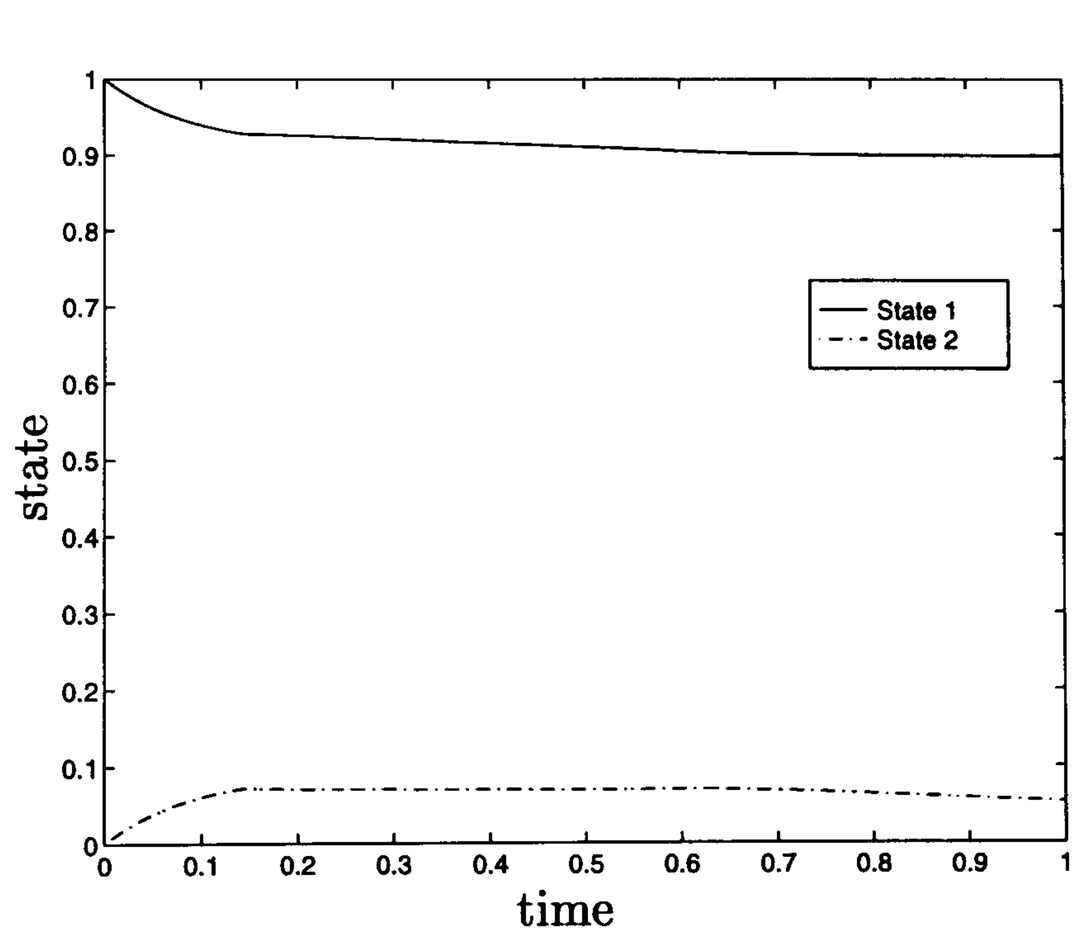


Figure 6.9: Catalyst Mixing: State Profiles

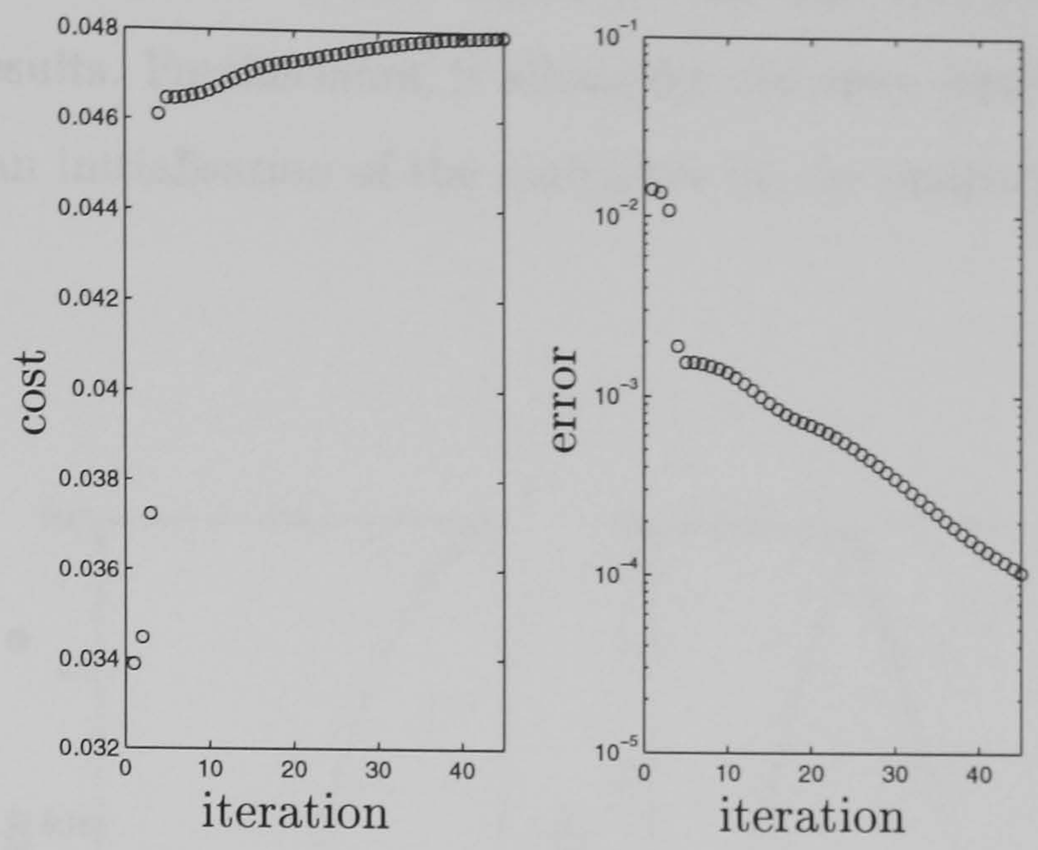


Figure 6.10: Catalyst Mixing: Cost and Error Profiles

6.4 Car Problem

We will now explore a variation of the famous car problem (see, for example, Vassiliadis (1993b) and Tanartkit and Biegler (1995)[4]). We wish to maximise the distance travelled over a fixed time horizon, such that the car starts and finishes at rest.

$$\min_u P(x_{t_f}) = -x_1(t_f) \tag{6.7}$$

subject to

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= u(t) \end{aligned} \tag{6.8}$$

with $x(0) = [0 \quad 0]$, $x_2(t_f) = 0$ and $-1.0 \leq u(t) \leq 1.0$.

Therefore x_1 represents the distance travelled by the vehicle, x_2 is the speed of the vehicle and u represents acceleration. We initialise at a constant acceleration of $u = 0$.

For comparative purposes, we consider both the partitioning approach, developed in Chapter 5, Section 5.8.1, as well as the modified Miele approach, introduced in Chapter 2 and modified in Chapter 5, in dealing with the endpoint constraint. Using the former, the algorithm converges to the optimum of 0.25 in 6 iterations, with a total time of 51.76 CPU seconds with an integration tolerance of 10^{-6} and an optimisation tolerance of 10^{-3}. The latter method yields convergence in 6 major iterations, with a total CPU time of 54.04

[4] All results reported from (Tanartkit and Biegler, 1995) are for a DEC station 5000.

seconds using the same tolerance levels. Thus, we see that the partitioning approach yields slightly faster results. Furthermore, it allows for the more intuitive initialisation of acceleration rather than initialisation of the multiplier on the endpoint constraint.

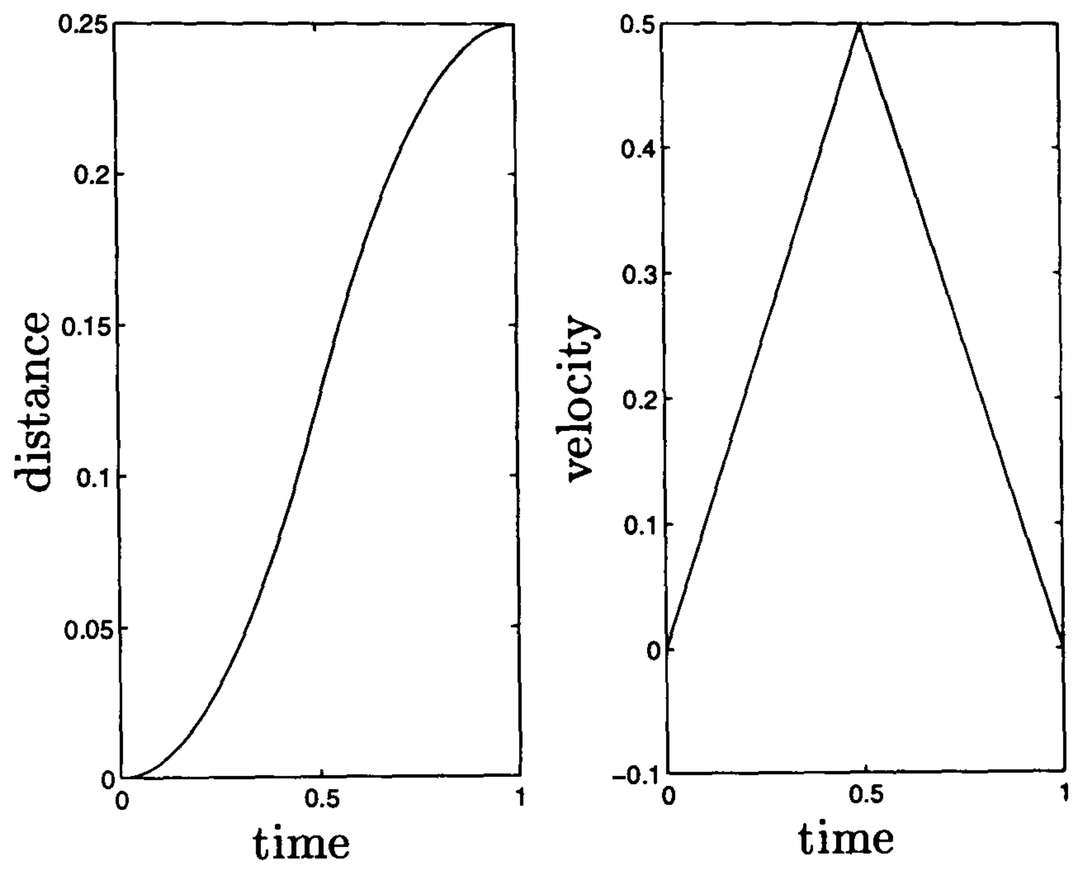


Figure 6.11: Car: Distance and Velocity Profiles

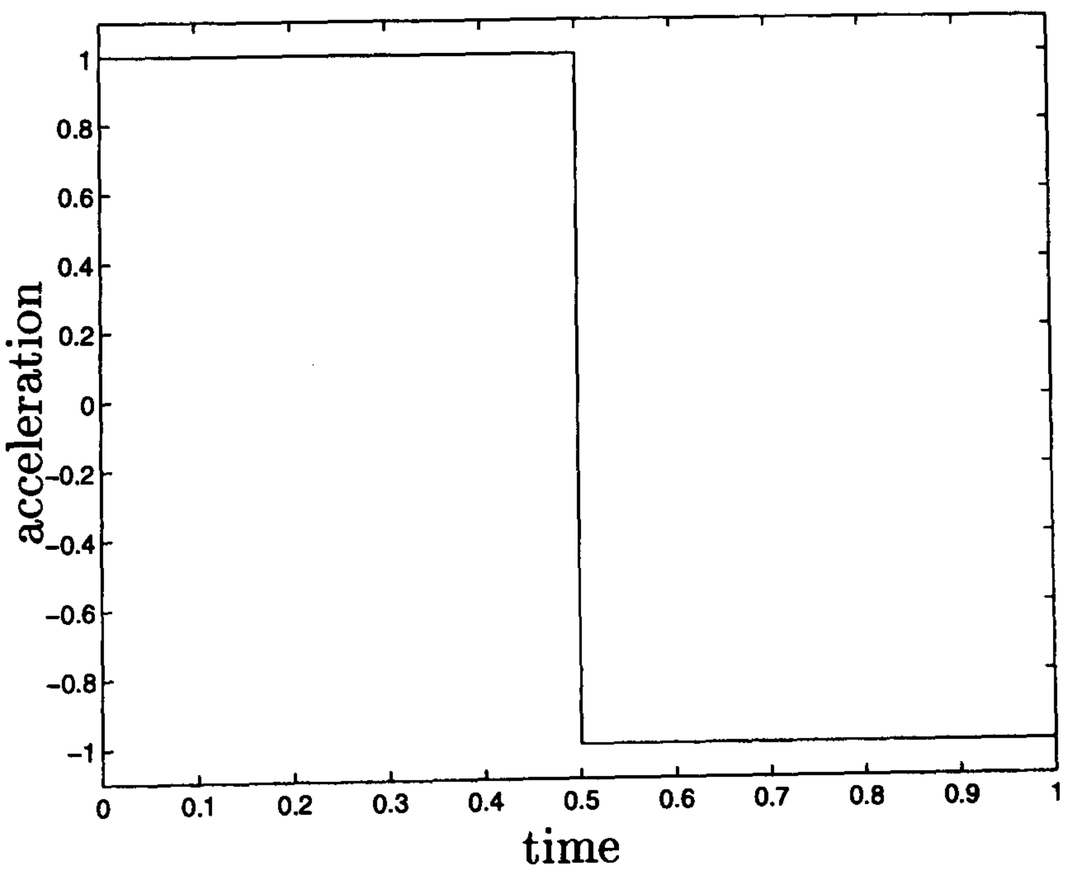


Figure 6.12: Car: Acceleration Profile

6.5 Minimum Time Car Problem

We now consider the exact problem posed by (Vassiliadis, 1993b) and (Tanartkit and Biegler, 1995). In this problem we wish to minimise the time required to cover a fixed distance, starting and finishing at rest.

$$\min_u P = t_f \tag{6.9}$$

subject to

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= u(t) \end{aligned} \tag{6.10}$$

with $x(0) = [0 \quad 0]$, $x_1(t_f) = 300$, $x_2(t_f) = 0$ and $-2.0 \leq u(t) \leq 1.0$.

We initialise the control profile at a constant value of $u(t) = 0$. We choose an initial endtime of 25. This value is obtained by considering the minimum time required to reach a distance of 300 units, without any constraint imposed on the end velocity. Furthermore, we impose a constraint on our time horizon to ensure a nonsingular matrix for our initial transversality conditions. Therefore, we impose $0 \leq t_f \leq 60$. We see that the optimum value is well away from the constraint boundaries and therefore the constraint has no effect on the solution. For these initial values, we achieve convergence in 6 iterations, for an integration tolerance of 10^{-6} and an optimisation tolerance of 10^{-3}. This requires a total of 69.2 CPU seconds. The method employed for dealing with the endpoint constraints is the method of partitioning.

(Tanartkit and Biegler, 1995) use a bilevel strategy, with an outer problem of determining the finite element lengths and an inner problem of determining the control and state variables once the element sizes are fixed. They cite CPU times of 14.5 seconds for 5 outer iterations using 3 collocation points and 2 finite elements, and 31.1 seconds for 4 outer iterations with 3 collocation points and 3 finite elements. (Vassiliadis, 1993b) solves this problem in 2.5 CPU seconds, but again makes use of prior knowledge of the analytical solution by initialising the problem with 3 piecewise constant elements.

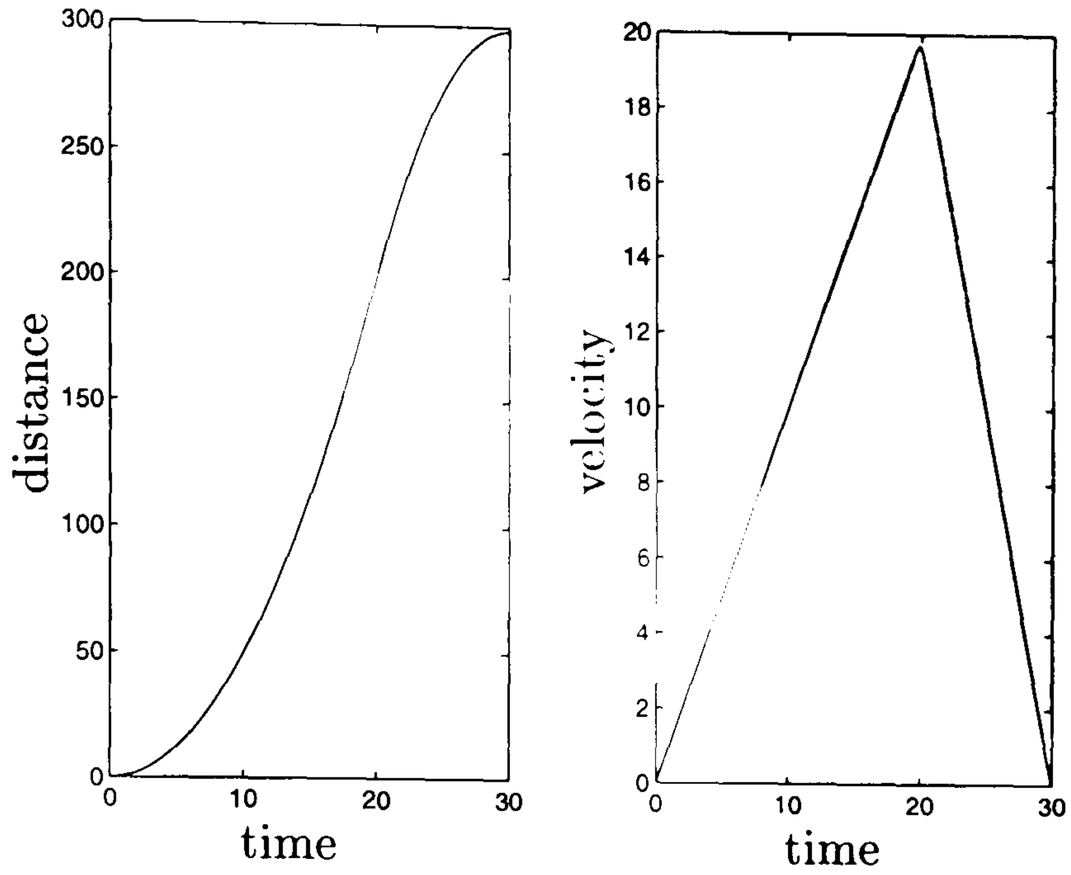


Figure 6.13: Minimum Time Car: Distance and Velocity Profiles

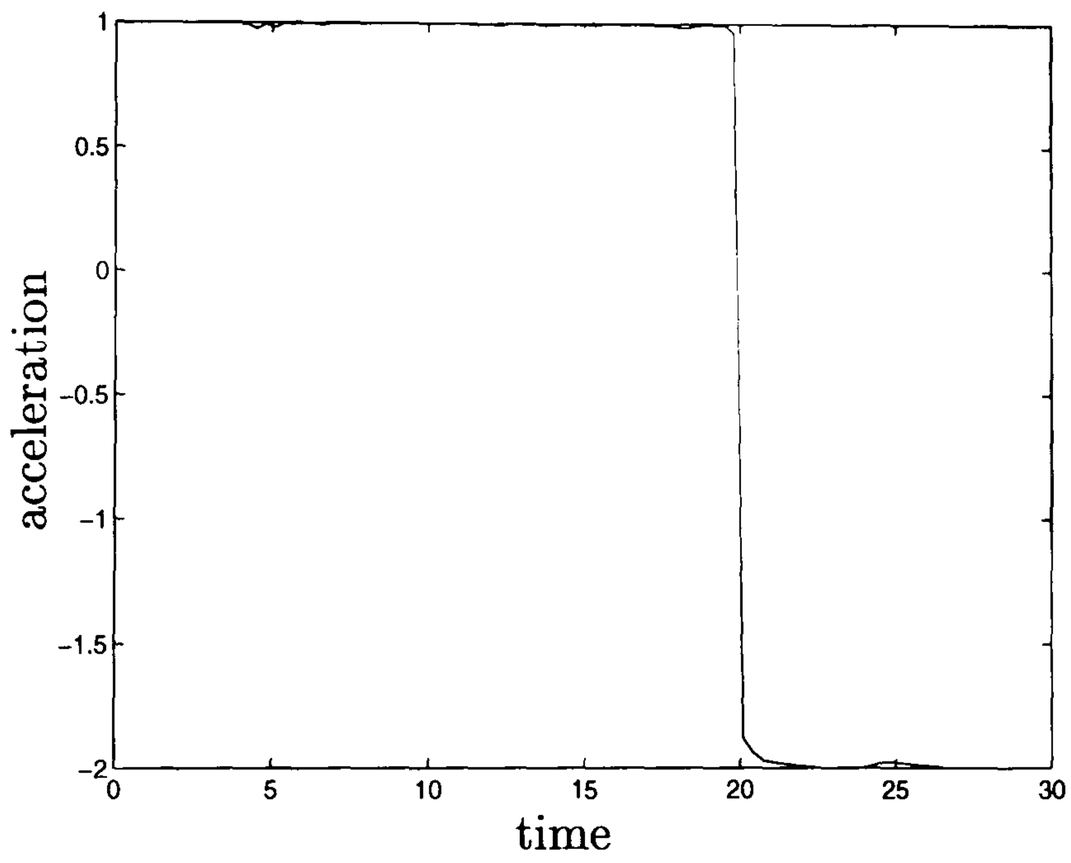


Figure 6.14: Minimum Time Car: Acceleration Profile

6.6 State Constrained Car Problem

We now complicate the original car problem further by introducing a path inequality constraint of the form:

$$0 \leq x_2(t) \leq 0.4 \tag{6.11}$$

This serves to introduce further switches in the control profile when the constraint

boundary is hit at $t = 0.4$ and when we are in a position to move from this constraint at $t = 0.6$. The results are intuitive but nonetheless difficult for standard optimal control algorithms to deal with in an efficient manner.

While this problem has an intuitively obvious solution, it is a difficult one to achieve numerically, as both a state inequality constraint and an endpoint equality constraint are present, both acting on the same variable (velocity). (Vassiliadis, 1993b) alludes to the difficulty of solving a similar problem, stating that convergence from an arbitrary starting point was difficult using DAEOPT. He initialises the problem at the optimal solution for the unconstrained case. Furthermore, he initialises the number of elements at 3, thereby utilising some advanced knowledge of the switching structure of the constrained optimum.

In our approach, the initialisation made use of the initial and final values of the adjoint variable corresponding to the unconstrained problem and therefore relied less on prior information about the constrained profile than Vassiliadis' initialisation. For this example, the algorithm converges to the optimal control trajectory in one major iteration using the active set approach developed in Chapter 5, Section 5.9.1. One major iteration comprises a forward and backward integration of both the state and adjoint dynamics, followed by a forward integration to generate the appropriate state dynamics and, finally, a forward integration of the appropriate linear combination of the $n_e + 1 = 2$ trajectories. This is achieved in an overall CPU time of 66.8 seconds for an integration tolerance of 10^{-5} and an optimisation tolerance of 10^{-3}. If we initialise μ at a small value of 10^{-2} the computational time drops to 1.25. This compares favourably with (Vassiliadis, 1993b), who achieved results in times of $9.2 - 125.3$ CPU seconds for a similar problem, depending on his method of discretising the state constraint. The corresponding tolerance levels were not given in (Vassiliadis, 1993b). (Tanartkit and Biegler, 1995) report results in the range of $1.9 - 61.3$ CPU seconds for the same problem as studied by Vassiliadis. However, the low range value again uses knowledge of the analytical solution by initialising with three optimally placed elements. The high range value is obtained within a framework which accommodates the element placement via bilevel programming. In this case, the trajectory is initialised at a constant acceleration within the control constraint boundaries.

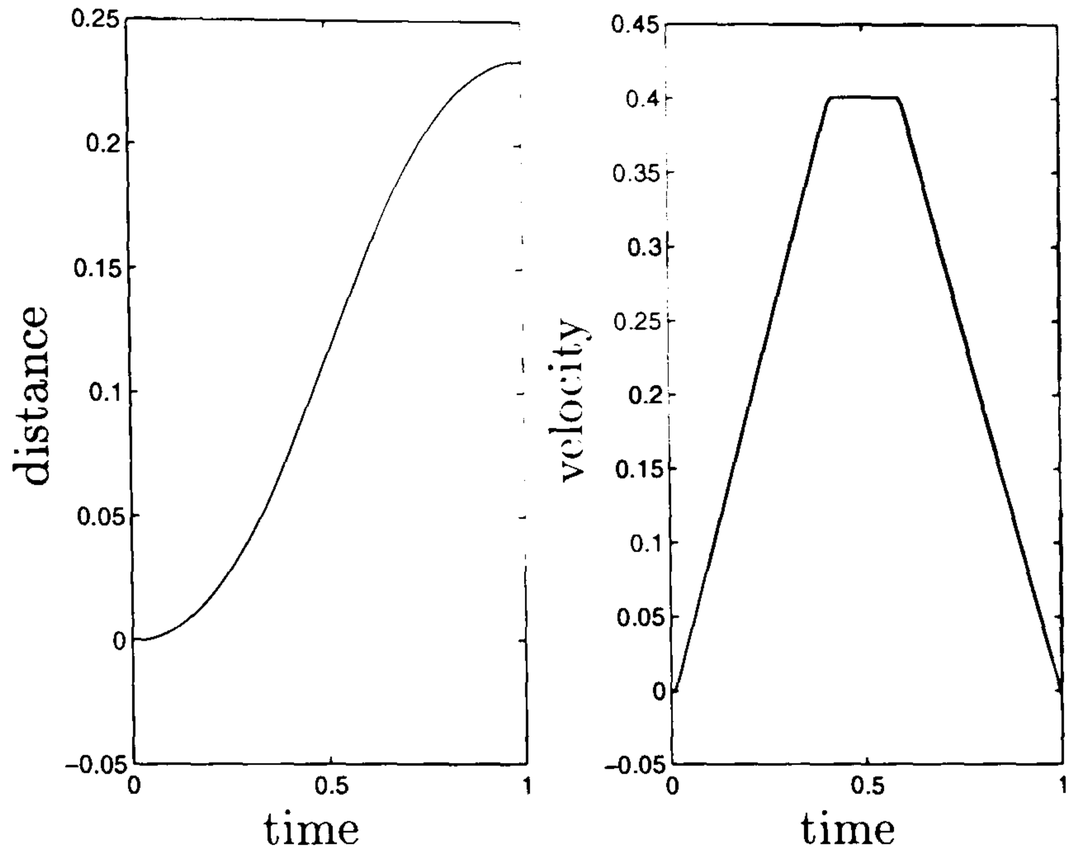


Figure 6.15: Constrained Car: Distance and Velocity Profiles

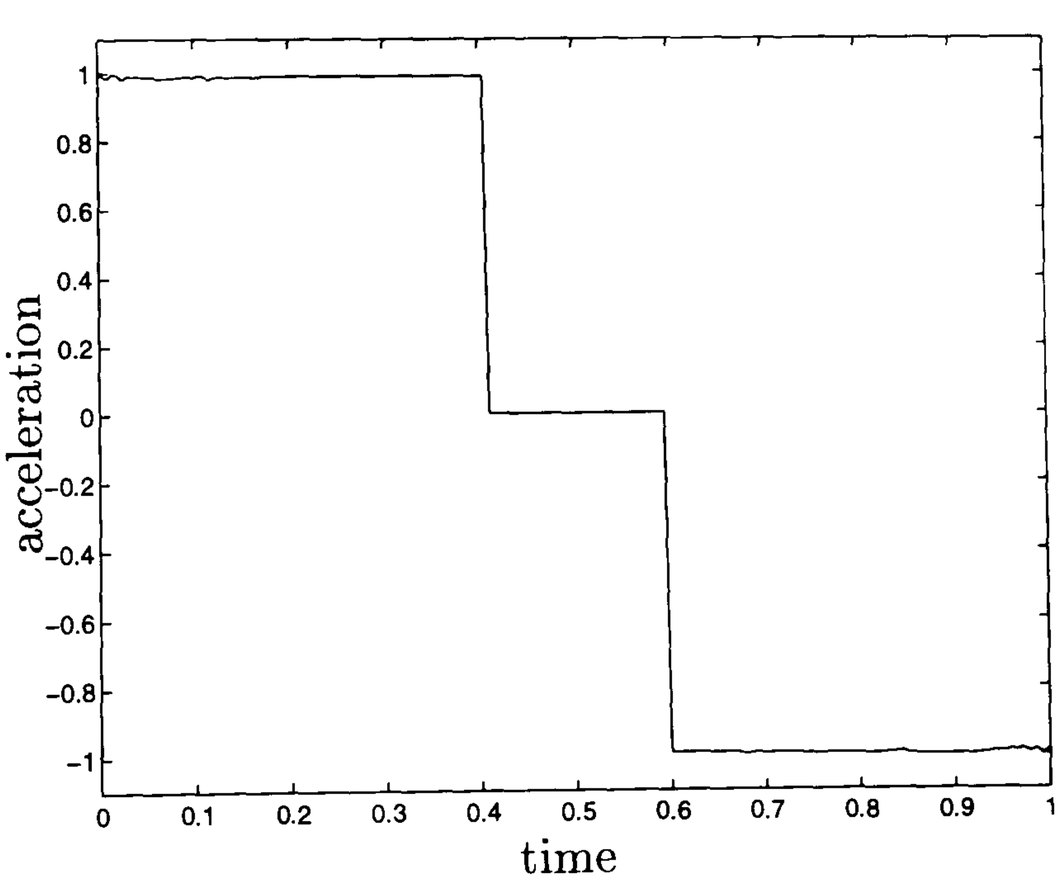


Figure 6.16: Constrained Car: Acceleration Profile

6.7 Conclusions

Several interesting case studies have been explored in this chapter. They demonstrate the applicability and reliability of the proposed approach in a range of examples. All of the problems were small in size, but served to illustrate its effectiveness in dealing with problems involving DAE systems; control constraints; endpoint equality constraints; state constraints; nonlinear systems; singular trajectories; and parameter optimisation.

The routines used to run these test cases were developed in Matlab, with interfaces to the Fortran routine DASOLV. The results show that in most cases the computational time is of the same order of magnitude as DAEOPT's results. In the case of the catalyst mixing problem, the results are significantly longer. This is a direct result of the flexibility in the determination of the control trajectory. However, it is important to note that the code developed is merely a piece of research software, with many opportunities for improvement of its efficiency. The sole purpose of this thesis was not the development of an efficient algorithm, but rather the development of the theory and the testing of this theory. The case studies demonstrate the suitability of the approach.

The state constrained car problem illustrated the suitability of the active set strategy developed in the previous chapter. In this case, the stability of the state and adjoint dynamics was exploited and the second modified version of the basic algorithm was employed. Convergence to the optimum was rapid when the problem was initialised at the unconstrained optimum.

Of the two proposed methods for dealing with endpoint equality constraints, the first method of partitioning the adjoint system was found to be better. This is a direct result of the fact that, in general, there will be less endpoint constraints than state equations. Hence the first approach requires less work. Both approaches require initial control profiles and, perhaps, time horizons, which are intuitive variables for estimation purposes. However, as demonstrated by the state constrained car problem, one can achieve very fast convergence using the Miele approach if one has a good initial estimate for the multipliers. In this case, the adjoint trajectory corresponding to the unconstrained problem provided an excellent initial estimate of the multipliers corresponding to the constrained problem and convergence was rapid.

The above case studies and optimal trajectories assume the models to be accurate descriptions of the system dynamics. The next chapter will broaden the scope of this

work by addressing the implications of uncertainty in the process dynamics and initial states and proposing a new methodology for incorporating robustness considerations in optimal control calculations for constrained DAE systems.

CHAPTER 7

Robust Control

7.1 Dealing with Uncertainty

As pointed out in the introduction, inevitably there will be mismatch between the model we use to choose our controls and a fully descriptive model of the plant. This will be a result of both modelling and computational restrictions. The presence of disturbances also introduces a factor of uncertainty in our system description. Therefore, controls must be robust, giving reasonable performance in the face of variations of the actual conditions from the predictions.

7.1.1 Differential Game Theory Approach for Unconstrained Systems

If there are no inequality constraints, one approach is to combine the practical receding horizon control methodology with the robust characteristics of $\mathcal{H}_\infty$ differential game theory techniques (see, for example, (Lall and Glover,1993) and (Kwon *et al.*, 1983)). The strategy is to solve a fixed or free endtime $\mathcal{H}_\infty$ game from each sampling point to a horizon that moves or "recedes" with time.

The general linear theory formulates the objective function as (see (Green and Limebeer, 1995), for more details):

$$\min_{u(\cdot)} \max_{w(\cdot)} \{J(u(t), w(t)) = \int_{t_0}^{t_f} z(t)^T z(t) - \gamma^2 w(t)^T w(t)\ dt\} \tag{7.1}$$

where $z(t)$ defines some desired performance objective and $w(t)$ is a disturbance, or more generally an uncontrollable input. We can in fact also include a vector of time-independent

parameters v to allow for model uncertainty, and use a general nonlinear cost function

$$\min_{u(\cdot)} \max_{v,w(\cdot)} \{J(u(t), w(t), v) = \int_{t_0}^{t_f} L(t, \dot{x}(t), x(t), u(t), w(t), v)\ dt\} \tag{7.2}$$

To achieve global robustness, the integrand should be convex in $u(t)$ and concave in $w(t)$, v so that the solution of (7.2) is a saddle-point. However, if v, $w(t)$ belong to compact sets, the saddle-point property is unnecessary.

The actual performance is guaranteed to be at least as good as the predicted performance, and the advantage of defining the game in a receding horizon framework is the fact that we are not searching for a feedback law. We take an initial measurement and integrate over time only. This eliminates the difficulties associated with searching for a control over time and state space (which may be very large). One can deal with a general system like (4.2), with (7.2) replacing (4.1), using the interior point approach.

7.1.2 New Modified Inequality Approach for Constrained Systems

However, the worst case situation may not arise very often and then this strategy is unduly cautious. It would obviously be better to choose controls which are optimal for the most likely situation (characterised, for example, by the nominal model), but for which performance does not deteriorate "too badly" as conditions deviate from this. However, this involves a more careful characterisation of risks and uncertainties and leads to a substantially more difficult problem.

Often process systems behaviour is dominated by constraints and a viable alternative is to assume the constraints are formulated so that any feasible operation is in fact reasonable. We can then choose controls to optimise the performance for the nominal model subject to these constraints. Of course we still have uncertainty to deal with and must ensure that we can continue operation from any feasible state, allowing for this uncertainty.

However in the receding horizon context we in fact implement the computed control only over one sample interval before re-determining the state, and hence only need this strong robustness property over this interval. Since the interval will usually be small, we can afford to be conservative and use a relatively simple bound to secure this property, and hence obtain a practical algorithm. It is important to note that in order for this approach to be applicable, we must assume the system to be controllable in the sense that the constraint does not become active at the end of the sampling interval.

Probably the simplest model is to assume:

$$\begin{aligned} \| x(t_0) - \bar{x}(t_0) \| &\leq K_x. \\ \| v - \bar{v} \| &\leq K_v. \\ \| w(t) - \bar{w}(t) \| &\leq K_w. \end{aligned} \tag{7.3}$$

for $t \in [t_0, t_f]$ where $\bar{x}(t_0)$ is the estimated initial state, $\bar{v}, \bar{w}$ are nominal ("most likely") values, and K_x, K_v, K_w are fixed constants.

We recall the partition $u = [u', x']$, where u' represents the true controls and x' the algebraic variables. We wish to ensure that within a neighbourhood of the nominal value $(\dot{\bar{x}}, \bar{x}, \bar{u}, \bar{w}, \bar{v})$, we do not violate the state inequality constraints

$$\underline{g} \leq g(t, \dot{x}, x, w, u, v) \leq \bar{g}. \tag{7.4}$$

We linearise the state equations about the nominal trajectory to a first order approximation

$$\bar{f} + f_{\dot{x}}(\dot{x} - \dot{\bar{x}}) + f_x(x - \bar{x}) + f_{x'}(x' - \bar{x}') + f_w(w - \bar{w}) + f_v(v - \bar{v}) = 0. \tag{7.5}$$

Since $\bar{f} = 0$, we may write this as

$$\begin{bmatrix} \dot{x} - \dot{\bar{x}} \\ x' - \bar{x}' \end{bmatrix} + (f_{\dot{x},x'})^{-1}\{f_w(w - \bar{w}) + f_v(v - \bar{v}) + f_x(x - \bar{x})\} = 0. \tag{7.6}$$

Taking norms, we obtain

$$\begin{aligned} \left\| \begin{bmatrix} \dot{x} - \dot{\bar{x}} \\ x' - \bar{x}' \end{bmatrix} \right\| &\leq \| (f_{\dot{x},x'})^{-1} \| \; \| \{f_w(w - \bar{w}) + f_v(v - \bar{v}) + f_x(x - \bar{x})\} \| \\ &\leq \| (f_{\dot{x},x'})^{-1} \| (\| f_w \| K_w + \| f_v \| K_v + \| f_x \| \; \| x - \bar{x} \|). \end{aligned} \tag{7.7}$$

We may then write

$$\| \dot{x}(t) - \dot{\bar{x}}(t) \| \leq \| (f_{\dot{x},x'})^{-1} \| (\| f_w \| K_w + \| f_v \| K_v + \| f_x \| \; \| x(t) - \bar{x}(t) \|) \tag{7.8}$$

and

$$\| x'(t) - \bar{x}'(t) \| \leq \| (f_{\dot{x},x'})^{-1} \| (\| f_w \| K_w + \| f_v \| K_v + \| f_x \| \; \| x(t) - \bar{x}(t) \|). \tag{7.9}$$

Using (7.3) and (7.8), we find

$$\| x(t) - \bar{x}(t) \| \leq K_x + \int_{t_0}^{t} \| (f_{\dot{x},x'})^{-1} \| (\| f_w \| K_w + \| f_v \| K_v)\, dt + \int_{t_0}^{t} \| (f_{\dot{x},x'})^{-1} \| \| f_x \| \| x(t) - \bar{x}(t) \|\ dt \tag{7.10}$$

Then, for a given t and $u'(t) = \bar{u}'(t)$, we may apply Gronwall's Lemma (Lemma 4.3.1).

Applying this to (7.10), we find

$$\| x(t) - \bar{x}(t) \| \leq M(t)\ \exp\{\| (f_{\dot{x},x'})^{-1} \| \| f_x \| (t - t_0)\} \tag{7.11}$$

where

$$M(t) = K_x + \| (f_{\dot{x},x'})^{-1} \| (\| f_w \| K_w + \| f_v \| K_v)\ (t - t_0) \tag{7.12}$$

and the norms are maximum norms over $[t_0, t]$ and we have assumed that $f(t), g(t)$ depend on v, $w(t)$. Of course the norms can also be approximated.

We then solve the optimal control problem using the nominal values $\bar{x}(t_0), \bar{v}, \bar{w}(\cdot)$, but with modified inequality constraints,

$$\underline{g}^i \leq g^i \pm \Delta^i \leq \overline{g}^i \tag{7.13}$$

i.e.

$$\underline{g}^i + \Delta^i \leq g^i \leq \overline{g}^i - \Delta^i \tag{7.14}$$

where

$$\Delta^i = \| g^i_{\dot{x}} \| \| \dot{x} - \dot{\bar{x}} \| + \| g^i_x \| \| x - \bar{x} \| + \| g^i_{x'} \| \| x' - \bar{x}' \| + K_w \| g^i_w \| + K_v \| g^i_v \|, \tag{7.15}$$

$i = 1, \ldots, n_i$ and $\| \dot{x} - \dot{\bar{x}} \|$, $\| x' - \bar{x}' \|$ and $\| x - \bar{x} \|$ are given by (7.8), (7.9) and (7.11).

This yields the following modified formulation of the inequality constraints

$$\underline{g}^i \leq g^i \pm \Delta^i \leq \overline{g}^i, \quad t \in [t_0, t_0 + \tau] \tag{7.16}$$

$$\underline{g}^i \leq g^i \pm K_x \| g^i_x \| \leq \overline{g}^i, \quad t \in (t_0 + \tau, t_f]$$

where τ is the sampling interval (Figure 7.1).

Satisfaction of these inequalities over the horizon guarantees satisfaction within a neighbourhood of the nominal trajectory. This neighbourhood is defined by (7.3). More sophisticated approximations could be used, to obtain less conservative controls at a cost of more computation, but this would only be justified for relatively large sampling intervals. The above analysis assumes the problem remains well-posed and has a solution subject to the above modified inequality constraints.

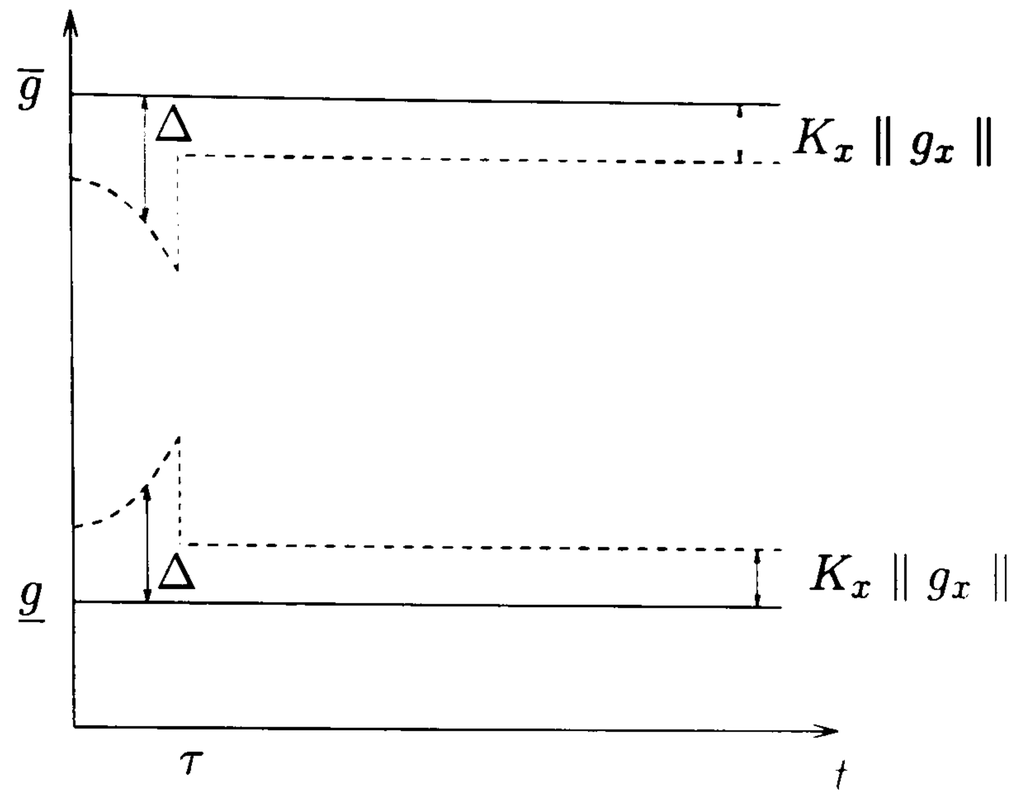


Figure 7.1: Modified Inequality Bounds

7.2 An Illustrative Example

In order to illustrate this approach to robustness, we reconsider the intuitively simple constrained car problem of the previous chapter. We include uncertainty in the initial state by setting $K_x = 0.1$. This uncertainty propagates through the state trajectory and takes effect when the state constraint activates, by shifting the constraint boundary by a margin of 0.1. The resulting profiles are shown in Figures 7.2 and 7.3.

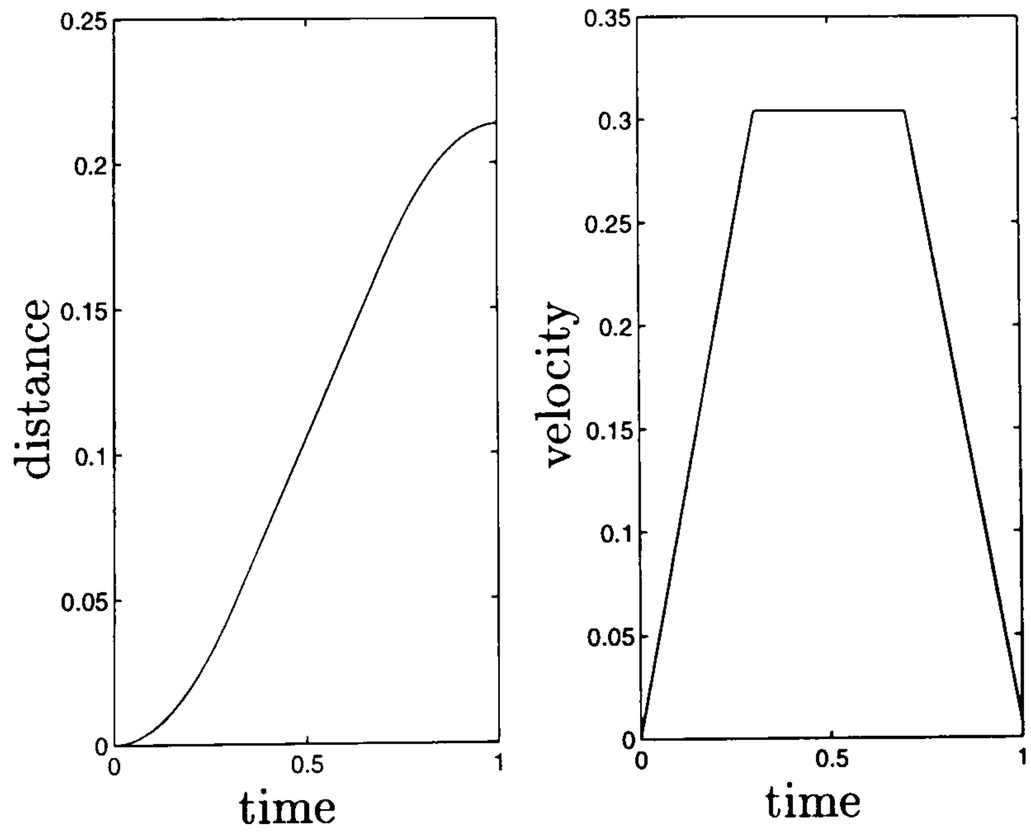


Figure 7.2: Constrained Car: Distance and Velocity Profiles with Robustness Considerations

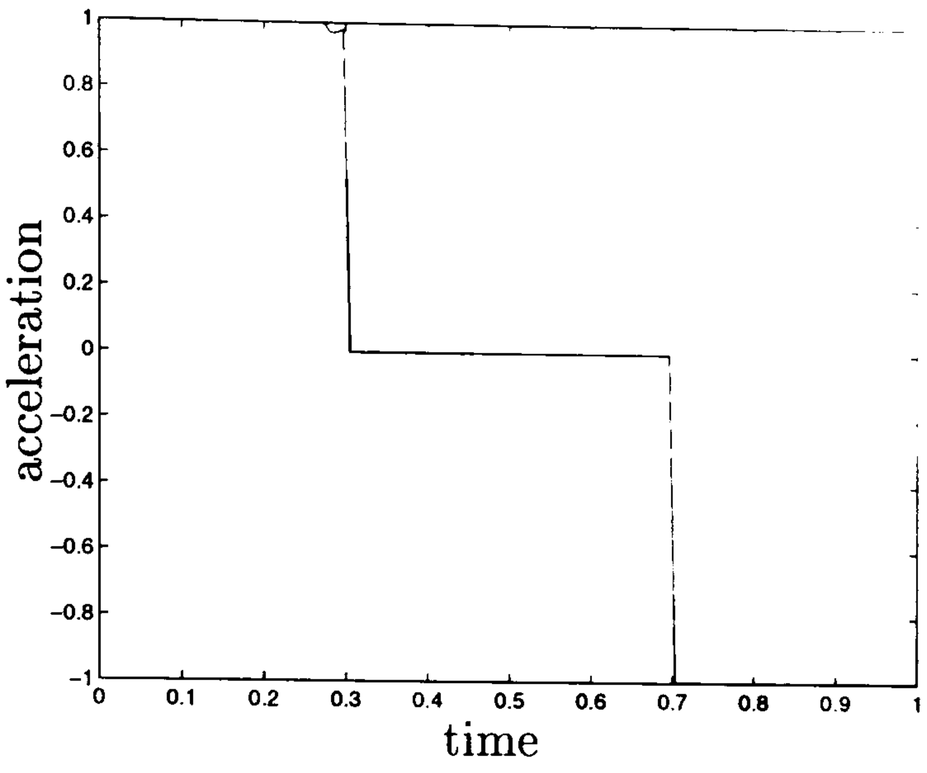


Figure 7.3: Constrained Car: Acceleration Profile with Robustness Considerations

This simple example therefore illustrates the effectiveness of the above approach in extrapolating the effect of initial uncertainty along the trajectory.

7.3 Conclusions

We have presented a simple, useful method of dealing with model and parametric uncertainty. This method exploits the receding horizon framework to incorporate a strong robustness property over the sampling period and a less conservative modified constraint for the remainder of the horizon so as to ensure strict satisfaction of the inequality constraints between measurements. The conservativeness of these robust controls depends on the values of the parameters K_x, K_v and K_w chosen by the user. These should be chosen to reflect the reliability of the model and the state measurement and the implications of violating the constraints. The less the user's faith in the available model, the larger the value of these parameters. Furthermore, the stronger the need to strictly satisfy the constraints, the larger the parameters. The method is simple to implement and provides intuitive results in cases such as the state constrained car example of this chapter.

CHAPTER 8

An Algorithm for High Index DAE Systems

High index DAE systems are normally solved by reducing them to an equivalent index-one system by successive differentiations and eliminations, and (Pantelides *et al.*, 1994) show how this can be achieved numerically, using automatic differentiation, within an optimal control algorithm. In general this implies differentiation of the controls, with consequent implications for their smoothness. Here we describe a simpler technique requiring no such differentiations. The method is based on numerically *partitioning* the high-index system into two subsystems. The first may be easily dealt with using a standard integrator for index 1 systems and the second is dealt with iteratively using an optimal control formulation. (Jarvis, 1994) presented a similar approach using structural partitioning.

8.1 Partitioning of High Index DAE Systems

We assume the vector u has not been partitioned into $[u'\ x']$. In optimal control, by definition we can choose $u'(t)$ at each t subject to the constraints, and we do not generally know *a priori* the partition $u = [u'\ x']$. The simulation or integration case, discussed in (Jarvis, 1994), necessarily assumes a partition for controlled systems and is a special case which easily fits within the more general description below.

We consider the system:

$$f(t, \kappa) = 0 \tag{8.1}$$

where $\kappa = (\dot{x}, x, u)$, $t \in \mathbb{R}$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^{m+q}$ and $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{m+q} \to \mathbb{R}^{n+m}$. The sum $m + q$ indicates the existence of m algebraic variables and q controls.

We generate the Jacobian matrix $[f_{\dot{x}}\ f_u]$ in order to solve

$$\begin{bmatrix} f_{\dot{x}} & f_u \end{bmatrix} \begin{bmatrix} \delta\dot{x} \\ \delta u \end{bmatrix} = -f. \tag{8.2}$$

We note that $[f_{\dot{x}}\ f_u]$ is non-square. We require a partitioning of the system (8.1) as follows:

$$x = \begin{bmatrix} \tilde{x} \\ \hat{x} \end{bmatrix},\ \tilde{x} \in \mathbb{R}^{n_1},\ \hat{x} \in \mathbb{R}^{n-n_1}, \qquad u = \begin{bmatrix} \tilde{x}' \\ u' \end{bmatrix},\ \tilde{x}' \in \mathbb{R}^{m_1},\ u' \in \mathbb{R}^{m-q-m_1}$$

$$f = \begin{bmatrix} \tilde{f} \\ \hat{f} \end{bmatrix}, \qquad \tilde{f} \in \mathbb{R}^{n_1+m_1},\ \hat{f} \in \mathbb{R}^{n+m-n_1-m_1} \tag{8.3}$$

such that $[\tilde{f}_{\dot{\tilde{x}}}\ \tilde{f}_{\tilde{x}'}]$ is square and nonsingular with rank $(n_1 + m_1)$ equal to the rank of $[f_{\dot{x}}\ f_u]$.

8.2 Numerical Partitioning

The partitioning may be determined numerically by forming the factorisation as follows:

$$L\ R\ [f_{\dot{x}}\ f_u]\ C = \begin{bmatrix} \tilde{U} & U \\ 0 & 0 \end{bmatrix} \tag{8.4}$$

where L is lower-triangular with unit diagonal and corresponds to the cumulative steps of Gaussian elimination, R is a row permutation matrix and C is a column permutation matrix. $\tilde{U}$ is a $(n_1 + m_1) \times (n_1 + m_1)$ nonsingular upper triangular matrix and U is a $(n_1 + m_1) \times (n + m + q - n_1 - m_1)$ nonzero matrix. Therefore, $\tilde{f}$ is the first $n_1 + m_1$ rows of $R\ f$ and $[\tilde{x}\ \ \tilde{x}']$ is determined from the first $n_1 + m_1$ elements of $C^T \begin{bmatrix} x \\ u \end{bmatrix}$.

The partition $[\tilde{U},\ U]$ induces the partition $\kappa = [(\dot{\tilde{x}}, \tilde{x}'), (\dot{\hat{x}}, u')]$ and $[\tilde{U}, U] = [\tilde{U}, \hat{U}, U']$, where $\tilde{U}$ corresponds to the Jacobian of $\tilde{f}$ with respect to $(\dot{\tilde{x}}, \tilde{x}')$, $\hat{U}$ corresponds to the Jacobian of $\tilde{f}$ with respect to $\dot{\hat{x}}$ and U' corresponds to the Jacobian of $\tilde{f}$ with respect to u'. In order to determine $\dot{\hat{x}}$, we define a new control $\dot{\hat{x}} = \hat{u}$ and append this equation to the system equations. This yields the following partitioned Jacobian of the augmented system,

$$J = \left[\begin{array}{c|cc} \tilde{U} & \hat{U} & U' \\ 0 & I & 0 \\ \hline 0 & 0 & 0 \end{array}\right]. \tag{8.5}$$

This shows that we can solve for $(\dot{\tilde{x}}, \tilde{x}', \dot{\hat{x}})$, given $(\hat{u}, u')$. and integrate $\dot{\hat{x}} = \hat{u}$ to obtain $\hat{x}$. This demonstrates the clear division between those variables which are obtained by integration, $(\tilde{x}, \hat{x})$, those determined to satisfy the DAE ($\dot{\tilde{x}}, \tilde{x}', \dot{\hat{x}}$) and those which are determined by optimisation, $(\hat{u}, u')$. We note that the presence of the identity matrix in (8.5) is a result of appending the equations $\dot{\hat{x}} = \hat{u}$ to the $\tilde{f}$ system of equations. This allows us to extract the state trajectory $\hat{x}$ for a given "control" $\hat{u}$. The null rows in (8.4) correspond to the additional set of equations $\hat{f}(t, x) = 0$. If in fact $\hat{f}$ was additionally dependent on an element of u', we could then solve for this element, which thus becomes an algebraic variable. Therefore, $\hat{f}$ is only dependent on t and x.

In forming the factorisation, we first choose only $\dot{x}_i$ as pivot variables, and once suitable pivots (above a given threshold) are exhausted, others are chosen from the remaining variables. The pivots are chosen in an ordered fashion, yielding a unique factorisation which is the best conditioned subsystem of equations, $\tilde{f}$ of dimension $n_1 + m_1$, with the remaining equations being $\hat{f}$. This is an important element of the algorithm, for often the case will arise when the choice of "control" variable is not necessarily unique. Consider the following system

$$\begin{aligned} \dot{x}_1 &= f_1(t, x_1, x_2, x_3) \\ 0 &= f_2(t, x_1, x_2, x_3). \end{aligned} \tag{8.6}$$

In order for this system to be index 1, either x_2 or x_3 must be specified as a control variable. A simple reduction procedure would not give an indication of which is better. if either. However, ordering the pivots ensures that the factorisation is unique.

Below we consider a simple example to clarify these ideas.

8.2.1 Example of partitioning

Consider the following system

$$\begin{aligned} f_1 &= \dot{x}_1(t) - x_1 - x_2 - u_2 = 0 \\ f_2 &= \dot{x}_2(t) - x_1 + u_1 - u_3 = 0 \\ f_3 &= x_1 + x_2 = 0 \\ f_4 &= 2x_1 + x_2 + u_4 = 0. \end{aligned} \tag{8.7}$$

For this system we find

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad \tilde{U} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}. \tag{8.8}$$

Therefore, the rank of the jacobian is 3. Rf indicates that $\tilde{f} = [f_1 \ f_2 \ f_4]$, and from $C^T \begin{bmatrix} x \\ u \end{bmatrix}$ we find that $\tilde{x} = [x_1 \ x_2]$, $\tilde{x}' = u_4$ and $u' = [u_1 \ u_2 \ u_3]$.

8.3 Formulating the Augmented Optimal Control Problem

To deal with the equations $\hat{f}$, we may define:

$$\dot{x}_r = \frac{1}{2}\hat{f}^T\hat{f} \quad \text{or} \quad \dot{x}_r = |\hat{f}|, \qquad x_r(t_0) = 0 \tag{8.9}$$

and impose the endpoint constraint $x_r(t_f) = 0$. We would expect the 2-norm representation to behave better than the absolute value because of the smoothness of the former compared with the latter. Note that (Jarvis, 1994) used $x_r(t_f)$ as an objective function and obtained good results for several examples. In this work, the 2-norm representation was found to yield very jerky control profiles and poor objective functions. We shall compare the two using an example later in this chapter.

Therefore, we have augmented the original optimal control formulation with a new endpoint equality constraint, associated with the partition of the original system into an index 1 system and algebraic constraints. We partition the system numerically by applying Gaussian elimination to yield (8.4). We may then use a standard integrator to determine $(x, \tilde{x}')$ from the index 1 subset of the original system.

8.4 Numerical Partitioning versus Structural Partitioning

It is worth commenting on the relative advantages and disadvantages of using a numerical approach rather than a structural approach to partitioning the system. We begin with a few definitions from (Jarvis, 1994):

Definition 8.4.1. *A* **structural matrix** *is a matrix which contains two basic entries:*

- *a* **zero** *entry, represented by* **0**
- *a* **non-zero** *entry, represented by* **X**

Therefore, for each entry $a_{ij} \neq 0$ we place an **X**. Thus the matrix

$$A = \begin{bmatrix} x^2 & 0 & x-y \\ xy & y^2 & 0 \\ 0 & x+y & 0 \end{bmatrix} \tag{8.10}$$

has structural matrix

$$A_s = \begin{bmatrix} X & 0 & X \\ X & X & 0 \\ 0 & X & 0 \end{bmatrix}. \tag{8.11}$$

Definition 8.4.2. *We say an $n \times n$ matrix is* **structurally singular** *if any subset of $k \in [1, n]$ rows contains non-zero entries in fewer than k distinct columns (i.e. if A_s has any zero columns).*

The advantage of structural matrices is that performing operations on a structural matrix is more efficient than manipulating the matrix itself. It allows for exploitation of any sparsity in the Jacobian of the system. However, structural singularity is sufficient but not necessary for numerical singularity. Therefore, a numerical approach

will detect singularities a structural approach would miss. Structural singularity is a qualitative property, whereas numerical singularity is quantitative. Consider the following simple example as an illustration:

$$\begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 2u \\ 0 & 1 & 0 \\ 0 & 0 & u+1 \end{bmatrix} \tag{8.12}$$

Structurally, the matrix is nonsingular, but if $u = -1$ we have a numerically singular Jacobian which must be dealt with accordingly. Therefore, the rank determined by the structural approach is an upper bound on the true rank of $\begin{bmatrix} f_{\dot{x}} & f_{x'} \end{bmatrix}$. The numerical approach has the advantage of detecting changes in the index of the system over the time horizon.

We note that nonsingularity of A where $Ax = b$ is not enough for a numerical solution. To ensure that the solution x is bounded, we require $\| b \|$ and $\| A^{-1} \|$ bounded. The latter is ensured by using a non-zero pivot threshold. Partial or full pivoting (see, for example, Golub and Van Loan (1983), pp. 66-69) ensures $\| A^{-1} \|$ is as small as possible for a given threshold. In partial pivoting, row interchanges are introduced in the elimination process. Full pivoting introduces both row and column interchanges and hence is a more costly alternative.

Determining the numerical index of the system introduces the problem of arbitrarily small pivots which would be zero were it not for rounding errors in the factorisation procedure. This requires the specification of some pivot threshold below which we take the pivot to be zero. This threshold may be difficult to determine in practice. In this algorithm, we are not concerned with determining the explicit index of the system. Hence, choosing an unduly large threshold in generating (8.4) merely results in a larger number of "zero" rows and hence a larger set of equations $\hat{f}$. This is no great disadvantage in our proposed algorithm, so long as the true residual of $\hat{f}$ is computed using the current values of $\dot{x}, u$.

It is therefore apparent that there are advantages and disadvantages to both the structural and numerical approach. However, the numerical approach provides a stronger characterisation of the partition. This chapter's goal is to provide an exploration of the numerical approach for comparison with the structural approach of (Jarvis, 1994).

8.5 An Illustrative Example

8.5.1 Parallel Reaction Scheme

We consider a parallel reaction scheme in a tank of fixed unit volume. The goal is to maximise the output of species 2 subject to a cost penalty on the feed of pure species 1:

$$\max_F \int_0^1 Lx_1 - F\alpha \, dt \tag{8.13}$$

subject to

$$\begin{aligned} f_1 &= \dot{x}_1 - F + Lx_1 + x_1 + x_1^2 = 0 \\ f_2 &= \dot{x}_2 + Lx_2 - x_1 = 0 \\ f_3 &= \dot{x}_3 + Lx_3 - x_1^2 = 0 \\ f_4 &= x_1 + 0.9x_2 + 0.8x_3 - 1 = 0. \end{aligned} \tag{8.14}$$

where

F	feed flowrate
L	exit liquid flowrate
x_1, x_2, x_3	molar concentration
α	cost weighting on feed

and $x_1(0) = 1$, $x_2(0) = x_3(0) = 0$, $\alpha = 0.1$, $0 \leq F,\ L \leq 10$. In view of the algebraic constraint $f_4 = 0$, the system has index two. The integral objective function was converted to an endpoint objective by defining an additional state variable. Equation (8.9) was used to deal with $f_4 = 0$, thereby generating a new state, $x_r(\cdot)$.

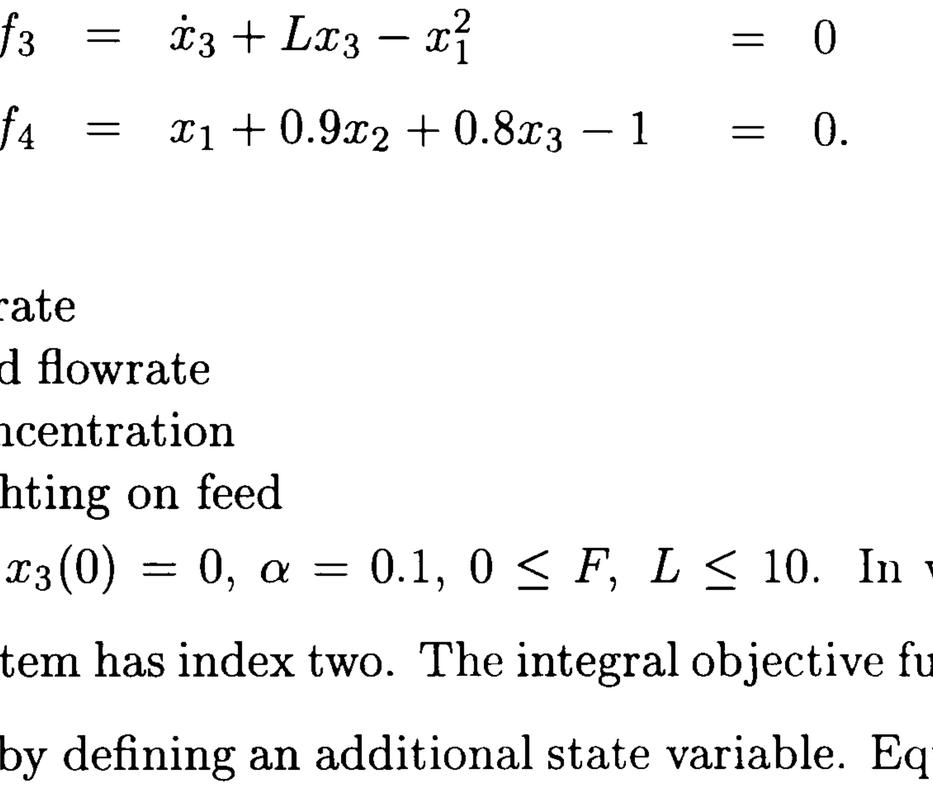


Figure 8.1: Parallel Reaction Scheme: Concentration Profiles

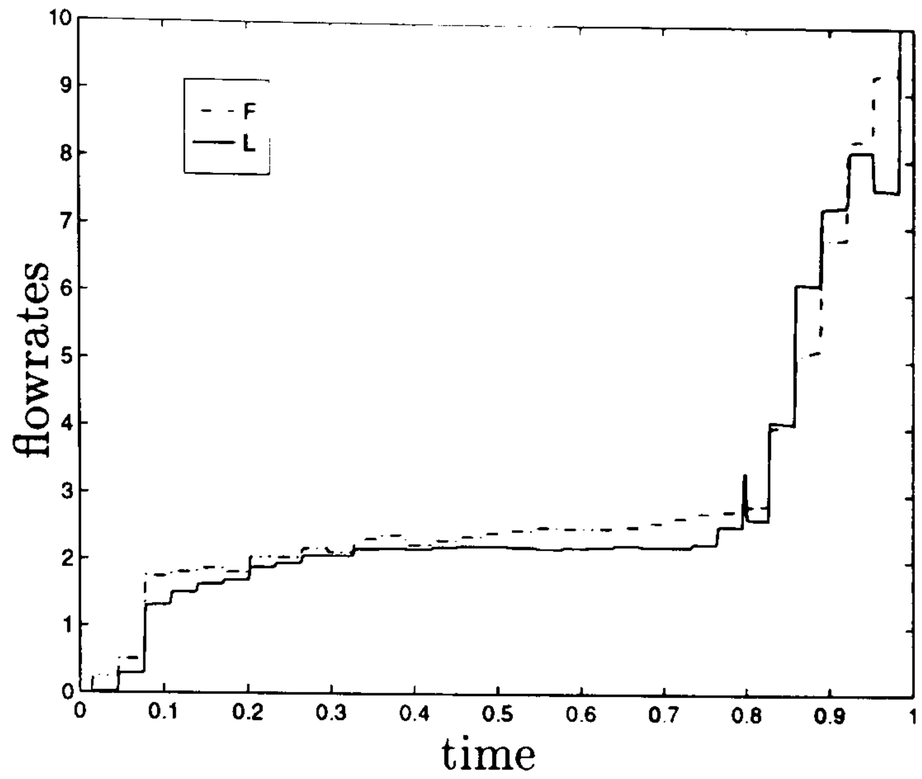


Figure 8.2: Parallel Reaction Scheme: Inlet and Outlet Profiles

The optimal objective value was found to be 0.278 for an integration tolerance of 10^{-4} and an optimisation tolerance of 10^{-2} and was achieved in 60 iterations. In addition to the endpoint constraint of $x_r(1) = 0$, we also imposed $x_1(1) + 0.9x_2(1) + 0.8x_3(1) = 1$ to improve convergence. Both a 2−norm and an absolute norm were considered. The absolute norm was found to yield a smoother result. The figures correspond to the use of an absolute norm. The final value for the endpoint constraint on the residual state was 0.010. This shows that the equality constraint was satisfied within 1% over all t.

8.6 Conclusions

We adopt here the approach of numerical partitioning of the system dynamics into an index one subsystem and a set of algebraic equality constraints involving only state variables $x(t)$. By integrating the sum of squares of the residuals of these algebraic equations, they are converted to a single endpoint constraint, and the resulting problem is then of the same form as that treated in Chapter 5. The chapter therefore demonstrates the equivalence of the solution of high index DAE systems and dynamic optimisation, and presents some numerical results to support the applicability of the method.

The numerical partitioning technique is easily incorporated into the factorisations which are in any case required, thus involving no extra work and this is more satisfactory than the use of a structural analysis, as proposed by earlier workers.

The numerical results obtained were for a system of index 2. As the index of the system

increases, one would expect poor approximations of the analytical solution. This is a direct result of the insensitivity of the endpoint constraint to changes in the control(s) due to the number of orders of integration separating the two. This expectation was supported by numerical results presented in (Jarvis, 1994) and suggests future work using monotonically increasing weighting functions on the penalty state, $x_r(t)$.

We note that the partitioning approach presented in this chapter transforms the high index problem to a singular formulation. The terms in the adjoint equations (5.16) convey no useful information as to the sensitivity of the endpoint constraint $x_r(t) = 0$ to changes in the control trajectory $u'(t)$, since the Jacobian of $\dot{x}_r(t) = \frac{1}{2}\hat{f}^T\hat{f}$ with respect to the states $\tilde{x}$ is zero at the solution $\hat{f} = 0$ and the Jacobian of $\dot{x}_r(t) = |\hat{f}|$ is undefined at the solution. The end result is that convergence as we approach the solution is slow, as was observed in the parallel reaction scheme example.

CHAPTER 9

Conclusions and Future Directions

The goal of this thesis has been the analysis and development of an algorithmic framework for the solution of general optimal control problems. Five major areas were addressed:

- Proof of existence of optimal controls for infinite horizon, or free endtime, formulations.
- Development of new necessary conditions for constrained DAE systems.
- Development of a general purpose optimal control algorithm.
- Extensions of the general algorithm to ensure robustness to model, measurement and parametric uncertainty.
- Extensions of the general algorithm to deal with DAE systems of any finite index.

A brief summary of each of these areas follows with discussion of future directions.

9.1 Existence of Optimal Controls

The conditions for existence of optimal controls for continuous time infinite horizon (free endtime) formulations were presented. Such conditions provide a method of assessing the suitability of the formulation of a control problem *a priori* and are important for completeness, as optimality conditions mean little in the absence of an optimal solution. This work provides a useful extension of finite horizon work (Vinter (1987), for example) and discrete time infinite horizon work (Keerthi and Gilbert (1985)).

The drawback of this analysis is the absence of state constraints along the trajectory. This continues to be an open question in existence theory. Furthermore, the proof of existence of optimal controls in Chapter 3 considers only ODE systems. While one can reduce higher order differential-algebraic equations to a system of first-order equations by the various methods discussed in Chapter 2, a useful extension of this proof would consider existence conditions for the original DAE system.

9.2 New Necessary Conditions for Constrained DAE Systems

New necessary conditions for nonlinear DAE systems constrained by equality constraints at the initial time and final time, as well as by general inequality constraints at the endpoints and along the path, were developed. A logarithmic barrier function was employed as a means of penalising deviations from the constrained, feasible region. These new necessary conditions are an innovative extension of existing necessary conditions for unconstrained ODE systems. The development in Chapter 4 resulted in a two point boundary value problem, characterised by very general specifications at the initial and endpoint.

The derivation of the necessary conditions does not include a formal proof, but the development is strong enough to demonstrate the existence of both the adjoints and multipliers when $\mu > 0$. A constraint qualification on both the endpoint equalities and inequalities is necessary to ensure $\pi > 0$ (i.e. a normal problem). A useful future extension of this work would be a formal proof to provide theoretical support for these new necessary conditions.

9.3 Optimal Control Algorithm

The proposed algorithm is based on solving an exact maximisation of the Hamiltonian at each time t. The algorithm provides an extension of this approach to constrained DAE systems. Of key importance is the algorithm's handling of inequality constraints. Slack variables which are strictly interior to the constraint space are introduced as a means of converting the inequalities to equality constraints. The interior point method is employed to enforce equality between the constraint function and the slack variable.

The routine operates in two stages. The first stage, integration, generates a state tra-

jectory for the current control trajectory and time-invariant parameter vector. The second stage, operating in the backward direction, generates the adjoint system corresponding to this state trajectory, as well as a new control trajectory. The algorithm operates iteratively in a forward and backward manner until an optimal trajectory is reached. The adjoint approach was chosen over control parameterisation based on past experience which showed the computation of sensitivities in the parameterisation approach to be costly. The case studies in Chapter 6 demonstrated a potential for this direct approach to be computationally more efficient. Furthermore, the approach developed in this thesis generates a more exact characterisation of the optimal trajectory.

Of theoretical interest is the proof of convergence of the interior point iterations. An informal convergence proof was presented in Chapter 5. The main stumbling block in formalising this proof is the varying Hessian in the iteration matrix. Future work will characterise the required modifications of the standard convergence proof for constant Hessian to the more general varying case.

Inherent in this approach is the assumption of stability of the initial state trajectory in the forward direction. Although this is a restriction, in most cases such a condition will be met. While collocation methods are able to deal with instability in the state dynamics, they suffer from an explosion in the size of the NLP to be solved as the system size increases.

The primary advantage of collocation schemes is their direct treatment of constraints. The main shortcoming of our approach is its handling of state-dependent constraints. As the state dynamics are assumed stable in the forward direction, the states are fixed when the controls are computed in the backward integration. The result may be a control trajectory which takes the system from a feasible to an infeasible region. In order to accommodate state constraints, we presented two approaches. Both relied on a modified system, incorporating the active state-dependent constraints as algebraic constraints on the system. The first approach made use of the modified system as a means of updating the control when an integration step from t to $t + \Delta t$ generated an infeasible state for the $u'(t)$ computed for fixed $x(t)$. The second approach exploits the stability of the state

and adjoint trajectories, integrating both the state and adjoint dynamics in parallel such that the computed control is directly related to the current state. On a state constraint boundary, we switch over to a new system where the state dynamics of the constrained state are now fixed by this constraint.

Ideally one would wish to deal with path constraints as we deal with control constraints. As a future direction, we shall consider modifying the interior point iteration to handle state inequality constraints. Currently, as we approach the boundary of a constraint, satisfaction of the interior point constraints (5.44) may be achieved by $\underline{z}$ or $\overline{z}$ tending to infinity, especially if the system is solved to a high accuracy. Then $\underline{z}\,\underline{s} \approx \mu$ or $\overline{z}\,\overline{s} \approx \mu$. A possible remedy to this problem would be to limit the growth of these multipliers. Thus, if $\underline{z}$ or $\overline{z}$ exceed some pre-determined bound, the routine would reset these values to the upper limit and recompute the corresponding slack variable s. A logical limit would be $\overline{z},\ \underline{z} \leq 1$ such that the error on the residual $g - s$ is of order μ as $\mu \to 0$. This approach will be tested in future work.

The presence of state equality constraints at the end point t_f results in an underdetermined system of equations at t_f. We presented two possible approaches to dealing with this problem within the basic algorithm's framework. The first, and preferred approach, is an extension of ideas presented in (Bryson and Ho, 1975) to constrained DAEs. The result is a partitioning of the endpoint equality constraints into those which depend on $x(t_f)$, $\bar{\phi}(x)$, and those which are independent of $x(t_f)$, $\tilde{\phi}(\dot{x}, u, v)$. The corresponding adjoint variable at t_f is partitioned into two parts, one corresponding to $\bar{\phi}(x)$ and one corresponding to the rest of the system equations at t_f. The result is an increase in the number of adjoint equations to be integrated in the backward pass from n_d to $(n_d + \bar{n}_e)(\bar{n}_e + 1)$, where $n_e = \bar{n}_e + \tilde{n}_e$ is the number of endpoint equalities. This imposes a greater computational burden, but still no greater than the computation of sensitivities in the control parameterisation approach. The second approach is an extension of Miele's (1973) ideas. For this case, we generate a basis of state trajectories from a basis of multipliers $\{\xi_i\}$, where the dimension of this basis is $\bar{n}_e + 1$. We generate an update for our state trajectory using a linear combination of the basis vectors such that both the cost and the violation of the equality constraint are reduced at every major iteration. To generate this update requires $2n_d(\bar{n}_e + 1)$ integrations. This approach was found to work well when a good estimate of the adjoint variables at the endpoints t_0 and t_f was

available (i.e. initialising a path constrained problem using the adjoint values from the unconstrained optimum). However, in general, one would expect the number of states to be greater than the number of endpoint equality constraints on the states and, therefore, the first approach requires fewer integrations to generate an update of ξ and, hence, the states.

9.4 Robustness

Recognising the inevitably of an imperfect model, as well as the presence of disturbances in the system, a method was outlined for ensuring that state constraints can be satisfied in the presence of uncertainty at least within the sampling interval. While $\mathcal{H}_\infty$ games fit nicely within a receding horizon framework, the saddlepoint solution was demonstrated to be invalid in the presence of inequality constraints. While the inequality constraints can be transformed to equality constraints using slack variables, thereby making the game well-defined, an approach based on the most-likely, or nominal, conditions will generate a less-conservative control than one based on a worst-case scenario. The approach in this thesis uses Gronwall's Lemma to redefine the inequality constraints over the sampling interval. This exploits the receding horizon framework by using a conservative inequality over the sampling interval. How conservative this inequality is depends on both the reliability of the model and the importance of the constraints. This requires the definition of an acceptable neighbourhood of deviation of the actual values of the state, disturbance and optimisation parameters from their nominal values. Therefore, the incorporation of robust inequality constraints requires only a slight modification of the nominal problem. The effectiveness and potential of this approach was demonstrated using a simple example.

9.5 High Index DAE systems

A differentiation free algorithm for dealing with optimal control of high index DAEs is proposed which is the numerical counterpart to the structural algorithm proposed by (Jarvis, 1994). This method provides a promising way of dealing with optimal control problems in which the system equations are high index since it avoids the need to repeatedly differentiate the system equations and augment the new dynamics to the original equations. Its effectiveness was demonstrated with a case study. This approach pro-

vides an improvement over collocation schemes, which are unable to deal directly with high-index systems.

The advantage of structural matrices is the efficiency of the matrix manipulations. It allows for exploitation of any sparsity in the Jacobian of the system. However, the detection of structural singularity is necessary but not sufficient for numerical singularity. Therefore, a numerical approach will detect singularities a structural approach would miss. The numerical approach has the advantage of providing a stronger characterisation of the partition. Furthermore, unlike the structural approach which assumes a constant index over the entire horizon, this approach potentially allows for an index which varies over the horizon.

The proposed algorithm for dealing with high index DAEs has not been extended to deal with systems whose index varies over the horizon and therefore a useful future direction would be to develop this approach. Furthermore, the effects of using monotonically increasing weights in the penalty state function could be investigated. This could be beneficial for convergence of higher index systems.

9.6 Computational Considerations

The main drawback of nonlinear control techniques is the sluggishness of the computations. However, work on parallelisation of codes such as DAEOPT has demonstrated significant reductions in computation time. The proposed algorithm of this thesis presents opportunities for parallelisation, most notably in the presence of state equality constraints at t_f. In this case, the integration of the partitioned adjoint variable may be accomplished using parallel processors.

The algorithm showed good convergence results from an iterational point of view. However, there are many opportunities for improving the convergence rates from a CPU point of view. The code as it stands now is written within the Matlab environment, with interfaces to Fortran code (DASOLV, for example). It could be easily translated into Fortran, where a wealth of efficient tools could be incorporated. In its current form, there is no exploitation of sparsity in the solution of the endpoint transversality conditions. The goal of this thesis was to explore the utility of this approach and hence Matlab provided a flexible environment within which these ideas could be tested. Undoubtedly, efficiency was sacrificed in favour of a user-friendly platform.

9.7 Conclusions

The thesis has discussed and developed new and interesting solutions to the control of constrained DAEs of a general description. Advances have been made in both theoretical and practical aspects of optimal control. At present it seems one of the biggest hurdles to the advancement to industrial application of control techniques such as those proposed in this thesis is computing power. However, it is the belief of this author that this is a problem of the moment, soon to be remedied by the rapid advancement of computing technology and power. This view is shared by many. We finish with a quote which provides motivation for continuing research in the area of advanced control techniques, such as those proposed in this thesis:

> *Any technology in which computing power is mentioned as a major impediment to today's implementation, will likely be implementable tomorrow*
> (Rawlings *et al.*, 1994)

References

R. B. Ash. *Measure, integration and functional analysis.* Academic Press, 1st edition, 1972.

R. Bachmann, L. Brüll, T. Mrzigold, and U. Pallaske. On methods for reducing the index of differential algebraic equations. *cce*, 14:1271–1273, 1990.

V. Balakrishnan, A. Zheng, and M. Morari. Constrained stabilization of discrete-time linear systems. In *Oxford Conference Proceedings: Advances in MBPC*, October 1993.

T. Basar and P. Bernhard. $\mathcal{H}_\infty-$ *Optimal Control and Related Minimax Design Problems. A Dynamic Game Approach.* Birkhäuser, 1991.

L. T. Biegler and J. B. Rawlings. Optimization approaches to nonlinear model predictive control. *Proceedings of Chemical Process Control IV*, pages 543–571, 1991.

R. R. Bitmead, M. Gevers, and V. Wertz. *Adaptive Optimal Control: The Thinking Man's GPC.* Prentice Hall International: London, 1st edition, 1990.

K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* SIAM, 2nd edition, 1996.

A. E. Bryson and Y. Ho. *Applied Optimal Control.* Halsted Press, 2nd edition, 1975.

S. L. Campbell. A computational method for general higher index nonlinear singular systems of differential equations. In *IMACS Transactions Scientific Computing*, pages 178–180, 1988.

D. A. Carlson, A. B. Haurie, and A. Leizarowitz. *Infinite Horizon Optimal Control: Deterministic and Stochastic Systems.* Springer-Verlag, 2nd edition, 1991.

Y. Chung and A. W. Westerberg. A proposed algorithm for solving nonlinear index problems. *Industrial and Engineering Chemistry Research*, 29:1234–1239, 1990.

D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized predictive control - part I. the basic algorithm. *Automatica*, 23(2):137–148, 1987a.

D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized predictive control - part II. extensions and interpretations. *Automatica*, 23(2):149 160, 1987b.

F. H. Clarke. *Optimization and Nonsmooth Analysis.* Wiley, 1st edition, 1983.

D. W. Clarke. Advances in model-based predictive control. In *Oxford Conference Proceedings: Advances in MBPC*, October 1993.

J. Cronin. *Differential equations : introduction and qualitative theory.* Dekker, Inc., 1st edition edition, 1980.

J. E. Cuthrell and L. T. Biegler. On the optimization of differential-algebraic process systems. *AIChE J.*, 33:1257–1270, 1987.

C. R. Cutler and B. L. Ramaker. Dynamic matrix control - a computer control algorithm. *JACC, San Francisco*, 1980.

I. Disli. *A Case Study in Optimal Process Operation: Control of a Carbon Dioxide Absorption Plant.* PhD thesis, Imperial College, University of London, 1996.

William F. Feehery and Paul I. Barton. Dynamic simulation and optimization wiht inequality path constraints. *Computers Chem. Engng.*, 20:S707–S712, 1996.

A. V. Fiacco and G. P. McCormick. *Nonlinear programming: sequential unconstrained minimization techniques.* John Wiley and Sons, 1st edition, 1968.

W. H. Fleming and R. W. Rishel. *Deterministic and Stochastic Optimal Control.* Springer-Verlag, 1st edition edition, 1982.

C. E. Garcia and M. Morari. Internal model control 1. a unifying review and some new results. *Ind. Eng. Chem. Process Des. Dev.*, 21:308–323, 1986.

C. E. Garcia and A. M. Morshedi. Quadratic programming solution of dynamic matrix control (qdmc). *Ind. Eng. Chem. Process Des. Dev.*, 21:308–323, 1986.

C. E. Garcia, D. M. Prett, and M. Morari. Model predictive control: Theory and practice - a survey. *Automatica*, 25(3):335–348, 1989.

C. W. Gear. Differential-algebraic equation index transformations. *SIAM Journal of Scientific and Statistical Computing*, 9:39–47, 1988.

H. Genceli and M. Nikolaou. Robust stability analysis of constrained l_1-norm model predictive control. *AIChE J.*, 39(12):1954–1965, 1993.

G. H. Golub and C. F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1^{st} edition, 1983.

M. Green and D. J. N. Limebeer. *Linear Robust Control.* Prentice Hall International: London, 1^{st} edition, 1995.

D. Gritsis. *The Dynamic Simulation and Optimal Control of Systems Described by Index Two Differential-Algebraic Equations.* PhD thesis, Imperial College, University of London, 1990.

D. J. Gunn and W. J. Thomas. Mass transport and chemical reaction in multifunctional catalyst systems. *Chem. Eng. Sci.*, 20:89–100, 1965.

H. Halkin. Necessary conditions for optimal control problems with infinite horizons. *Econometrica*, 42:267–272, 1974.

R. F. Hartl, S. P. Sethi, and R. G. Vickson. A survey of the maximum principles for optimal control problems with state constraints. *SIAM Review*, 37(2):181–218, 1995.

R. B. Jarvis and C. C. Pantelides. Numerical methods for differential-algebraic equations - new problems and new solutions. In *Proceedings of the AIChE Annual Meeting*, 1991.

R. B. Jarvis and C. C. Pantelides. Dasolv: A differential-algebraic equation solver, version 1.2. Technical report, Imperial College, University of London, 1992.

R. B. Jarvis. *Robust Dynamic Simulation of Chemical Engineering Processes.* PhD thesis, Imperial College, University of London, 1994.

B. R. Keeping and C. C. Pantelides. On the implementation of optimisation algorithms for large-scale transient systems on a MIMD computer. In *Proceedings of the AIChE Annual Meeting*, 1995. Paper 176e.

S. S. Keerthi and E. G. Gilbert. An existence theorem for discrete-time infinite-horizon optimal control problems. *IEEE Transactions on Automatic Control*, $AC-30(9)$:907–909, 1985.

L. Kershenbaum, D. Q. Mayne, R. Pytlak, and R. B. Vinter. Receding horizon control. In *Oxford Conference Proceedings: Advances in MBPC*, October 1993.

J. F. C. Kingman and S. J. Taylor. *Introduction to Measure and Probability.* Cambridge University Press, 1966.

H. Kwakernaak. Robust control and $\mathcal{H}_\infty$-optimization - tutorial paper. *Automatica*, 29(2):255–273, 1993.

W. H. Kwon and A. E. Pearson. A modified quadratic cost problem and feedback stabilization of a linear system. *IEEE Transactions on Automatic Control*, 22:838–842, 1977.

S. Lall and K. Glover. A game theoretic approach to moving horizon control. In *Oxford Conference Proceedings: Advances in MBPC*, October 1993.

P. L. Lee, G. R. Sullivan, and W. Zhou. Process/model mismatch compensation for model-based controllers. *Chem. Eng. Comm.*, 80:33–51, 1989.

J. S. Logsdon and L. T. Biegler. Decomposition strategies for large-scale dynamic optimization problems. *Chem. Eng. Sci.*, 47(4):851–864, 1992.

J. S. Logsdon. *Efficient Determination of Optimal Control Profiles for Differential Algebraic Systems.* PhD thesis, Carnegie Mellon University, 1990.

S. E. Mattson and G. Söderlind. Index reduction in differential-algebraic equations using dummy derivatives. *SIAM J. Sci. Stat. Comput.*. 14:677–692, 1993.

R. McGill. Optimal control, inequality constraints and the generalized newton-raphson algorithm. *SIAM J. Control*, 3:291–298, 1965.

E. S. Meadows and J. B. Rawlings. Receding horizon control with an infinite horizon. In *Proceedings of the American Control Conference*, pages 2926-2930, June 1993.

H. Michalska and D. Q. Mayne. Receding horizon control of nonlinear systems. *IEEE Transactions on Automatic Control*, AC-35:814–824. 1990.

H. Michalska and D. Q. Mayne. Implementable receding horizon control of constrained nonlinear systems. Technical report, Centre for Process Systems Engineering, Imperial College, 1993.

A. Miele. *Optimization and Design*, chapter Gradient Methods in Optimal Control Theory. Prentice-Hall, Inc., 1^{st} edition, 1973.

M. Morari and J. C. Doyle. A unifying framework for control system design under uncertainty and its implications for chemical process control. *Proceedings of Chemical Process Control III*, pages 5–51, 1986.

J. J. Moré and M. Y. Cosnard. Numerical solution of nonlinear equations. *ACM Transactions on Mathematical Software*, 5:64–85, 1979.

B. A. Ogunnaike and W. H. Ray. *Process Dynamics, Modeling and Control.* Oxford University Press, 1^{st} edition, 1994.

J. M. Ortega and W. C. Rheinboldt. *Iterative Solutions of Nonlinear Equations in Several Variables.* Academic Press, 1^{st} edition edition, 1970.

J. R. Paloschi. Steady state process simulation on mimd machines: Solving nonlinear equations. *Computers and Chemical Engineering*, 19S:S721–S728, 1995.

C. C. Pantelides, R. W. H. Sargent, and V. S. Vassiliadis. Optimal control of multistage systems described by high-index differential-algebraic equations. In *Computational Optimal Control*, pages 177–191. Birkhäuser, Bulirsch, R., Kraft, D., eds., 1994.

R. Polyak. Modified barrier functions (theory and methods). *Mathematical Programming*, 54:177–222, 1992.

L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes.* John Wiley and Sons, 1^{st} edition, 1962.

J. B. Rawlings, E. S. Meadows, and K. R. Muske. Nonlinear model predicitive control: A tutorial and survey. In *ADCHEM '94 Proceedings, Kyoto, Japan*. 1994.

W. H. Ray. *Advanced Process Control*. McGraw Hill. 1^{st} edition. 1981.

J. C. Renfro, A. M. Morshedi, and O. A. Asbjornsen. Simultaneous optimization and solution of systems described by differential-algebraic equations. *Comput. Chem. Engrg.*, 11:503–517, 1987.

J. Richalet, A. Rault, J. L. Testud, and J. Papon. Model predictive heuristic control: Applications to industrial processes. *Automatica*, 14:413–428. 1978.

E. Roxin. The existence of optimal controls. *Michigan Math. J.*, 9:109–119. 1962.

H. L. Royden. *Real Analysis*. Collier-Macmillan, 2^{nd} edition, 1968.

A. P. Sage and C. C. White. *Optimum Systems Control*. Prentice-Hall. Inc., 2^{nd} edition. 1977.

R. W. H. Sargent and G. R. Sullivan. The development of an efficient optimal control package. In *Lecture Notes in Control and Information Sciences 7 (Stoer, J., editor)*, pages 158–168. IFIP Conference on Optimization Techniques, Springer-Verlag, 1978.

R. W. H. Sargent. An infeasible interior point method for generalized monotone linear complementarity problems. Technical report, Imperial College, London, PSE Centre Report No. $C94-12$ 1994 (revised 1995).

E. D. Sontag. *Mathematical Control Theory, Texts in Applied Mathematics*. Springer Verlag, volume 6 edition, 1990.

J. L. Speyer and A. E. Bryson. Optimal programming problems with a bounded state space. *AIAA*, 6(8):1488–1491, 1968.

J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag. 2^{nd} edition. 1993.

G. R. Sullivan. *Development of Feed Changeover Policies for Refinery Distillation Units*. PhD thesis, Imperial College. University of London, 1977.

G. Tadmor. Receding horizon revisited: An easy way to robustly stabilise an LTV system. *Systems and Control Letters*, 18:285–294, 1992.

P. Tanartkit and L. T. Biegler. Stable decomposition for dynamic optimization. *Ind. Eng. Chem. Res.*, 34:1253–1266, 1995.

F. A. Valentine. *The Problem of Lagrange with Differential Inequalities as Added Side Conditions. Contributions to the Calculus of Variations 1933-1937*. University of Chicago Press, 1937.

S. Vasantharajan and L. T. Biegler. Simultaneous strategies for optimization of differential-algebraic systems with enforcement of error criteria. *Comput. Chem. Engng.*, 14(10):1083–1100, 1990.

V. S. Vassiliadis, C. C. Pantelides, and R. W. H. Sargent. Solution of a class of multistage dynamic optimization problems. part i - problems without path constraints. *Ind. and Eng. Chem. Res.*, 33(9):2111–2122, 1994a.

V. S. Vassiliadis. *DAEOPT* - a differential-algebraic dynamic optimization code, *v*.2.0. Technical report, Centre for Process Systems Engineering, Imperial College, London, 1993a.

V. Vassiliadis. *Computational Solution of Dynamic Optimization Problems with General Differential-Algebraic Constraints*. PhD thesis, Imperial College, University of London, 1993b.

V. S. Vassiliadis. Application of the modified barrier method in large-scale quadratic programming problems. *Computers Chem. Engng.*, 20:S243–S248, 1996.

R. B. Vinter and H. Michalska. Receding horizon control for general nonlinear systems: Stability and robustness. Technical report, Centre for Process Systems Engineering, Imperial College, 1993.

R. B. Vinter. Optimal control theory. Technical report, Summer School "Optimization, Control and Related Subjects". Nice, August-September 1987.

S. P. Walsh. *Integrated design of chemical waste water treatment systems*. PhD thesis, Imperial College, University of London, 1993.

J. Warga. *Optimal Control of Differential and Functional Equations.* Academic Press, Inc., 1^{st} edition, 1972.

S. J. Wright. Interior point methods for optimal control of discrete time systems. *Journal of Optimization Theory and Applications*, 77(1):161–187, 1993.

G. Zames. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Transactions on Automatic Control*, 26:301–320, 1981.

Z. Q. Zheng and M. Morari. Robust stability of constrained model predictive control. In *Proceedings of the 1993 American Control Conference*, pages 379–383, 1993.

Consider the postage stamp: its usefulness consists in the ability to stick to one thing till it gets there.

–Josh Billings

BADL.
LONDIN.
UNIV.