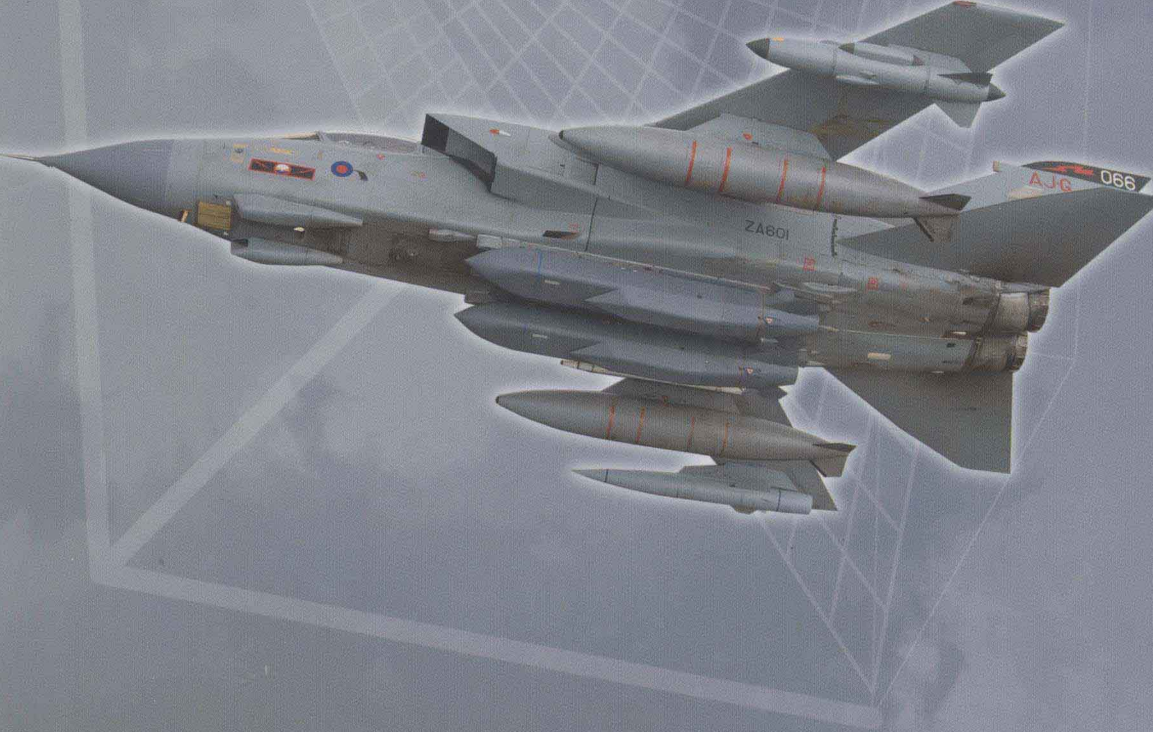
S. Subchan | R. Żbikowski

Computational Optimal Control

Tools and Practice



WILEY

Computational Optimal Control

Tools and Practice

Subchan Subchan

Cranfield University at Shrivenham, UK

Rafał Żbikowski

Cranfield University at Shrivenham, UK

WILEY

A John Wiley and Sons, Ltd, Publication

This edition first published 2009

Registered office
John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom.

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at www.wiley.com.

Library of Congress Cataloging-in-Publication Data

Subchan, S.
Computational optimal control : tools and practice / S. Subchan, R. Zbikowski.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-470-71440-9 (cloth)
1. Control theory. 2. Adaptive control systems. 3. Mathematical optimization. I. Zbikowski, R. (Rafal) II. Title.
TJ213.S789 2013
629.8'312–dc22 2009019927

A catalogue record for this book is available from the British Library.

ISBN 978-0-470-71440-9 (H/B)

Printed and bound in Great Britain by CPI Antony Rowe, Chippenham, Wiltshire

Computational Optimal Control

Preface

Computational Optimal Control Reliable and efficient ways of finding the best possible (optimal) solutions is a pervasive problem in science and engineering. In many such problems, especially in engineering, we can manipulate the optimised object/process through limited (constrained) influences (control) at our disposal which we can vary over time (dynamically). The dynamic control elicits dynamic responses from the optimised object/process, often in ways which are difficult to guess or intuit. A judicious choice of optimal control, obeying the constraints, must therefore be based on a systematic procedure. The well-documented theory of optimal control is exactly the mathematical tool necessary for that. Given a mathematical description of the optimised object/process, an optimisation criterion (performance index) and constraints, the theory of optimal control gives mathematical equations whose solution is the optimal control needed. In most realistic (and thus practical) engineering problems, application of the theory leads to complex equations.

The complexity of the optimal control equations is not a flaw—it properly reflects the mathematical details of the optimised problems; it is the details that make the problem realistic and hence practical. But the complexity of the optimal control equations means that, in industrial practice, the theory must be accompanied by calculations with digital computers, especially for advanced problems in aerospace and aeronautics. This blend of mathematical theory and numerical techniques is the essence of computational optimal control. Both the theory and the numerical algorithms involved are rather non-trivial in nature, and their interaction adds another layer of complexity.

Book Focus and Prerequisites This book focuses on informed use of computational optimal control rather than development of either theory or numerics. The aim of the book is to provide a hitherto unavailable computational optimal control self-study textbook for practising engineers, especially engineers working on challenging, real-world applications in aerospace and aeronautical industries. Graduate and post graduate students who want to specialise in advanced applications of optimal control should also find it of interest. The prerequisite knowledge is a general background in numerical analysis and ordinary differential equations plus familiarity with the FORTRAN computer language, usually acquired during graduate engineering studies. Some knowledge of optimal control would be helpful, but is not essential—the relevant theory can be picked up while studying this text.

Case Study The main thrust of the book is to explain how to use computational optimal control tools in engineering practice, employing an advanced aeronautical case study to provide a realistic setting for both theory and computation. The case study is focused on missile guidance in the form of trajectory shaping of a generic cruise missile attacking a fixed target which must be struck from above. The problem is reinterpreted using optimal control theory resulting in two formulations: (1) minimum time-integrated altitude and (2) minimum flight time. The resulting trajectory has a characteristic shape and hence the problem is also known as optimisation of the bunt manoeuvre. This eminently realistic and practical problem is quite hard, because realistic missile flight dynamics and practical control and flight path constraints are assumed. Due to its challenging nature, the bunt manoeuvre problem is an excellent illustration of advanced engineering practice, without comforting simplifications found in many textbooks. More importantly, the problem strongly exercises both the theoretical and numerical aspects of computational optimal control, so it robustly tests the true value of theoretical and software tools available to the practitioner. A detailed account of the actual performance of these tools is given in this book.

The bunt manoeuvre problem in its minimum time-integrated altitude and minimum flight time formulations is the only problem treated in this book. This allows a detailed, and often tutorial, presentation with insights into the structure and nature of the optimal solutions and also into the advantages and limitations of the available tools. Rather than moving from one simple example to another, and learning little of real-world computational optimal control, we prefer the reader to stay focused on one in-depth project. Introducing other challenging examples would, in our opinion, distract the reader from the main aim of this book: informed use of the tools of computational optimal control in advanced engineering practice.

Approach Each of the formulations of the bunt manoeuvre problem is solved using a three-stage approach. In stage 1, the problem is discretised, effectively transforming it into a nonlinear programming problem, and hence suitable for solution with the public-domain FORTRAN packages DIRCOL and NUDOCCCS or the commercial FORTRAN packages PROMIS or SOCS. The results of this direct approach are used to discern the structure of the optimal solution, i.e. type of active constraints, time of their activation, switching and jump points. The qualitative analysis of the solution structure, employing the results of stage 1 and optimal control theory, constitutes stage 2. Finally, in stage 3, the insights of stage 2 are made precise by rigorous mathematical formulation of the relevant two-point boundary value problems (TPBVPs), using appropriate theorems of optimal control theory. The TPBVPs obtained from this indirect approach are then solved using the public-domain FORTRAN package BNDSCO and the results compared with the appropriate solutions of stage 1. Additionally, a comparison is made with the results obtained by the commercial package GESOP (a software environment with PROMIS and SOCS solvers) whose solution approach can be considered as half-way between the approaches of stages 1 and 3.

For each formulation (minimum time-integrated altitude and minimum time) the influence of boundary conditions on the structure of the optimal solution and the performance index is investigated. Software implementation employing the public-domain packages DIRCOL, NUDOCCCS and BNDSCO, and also the commercial packages SOCS and PROMIS under the GESOP environment, which produced the results, is described and documented.

Book Features As explained earlier in this preface, this book focuses on informed use of computational optimal control for solving the terminal bunt manoeuvre, rather than development of either the underlying theory or the relevant numerics. In this context, the main features of this book are as follows:

- formulating trajectory shaping missile guidance as an optimal control problem for the case of terminal bunt manoeuvre;
- devising two formulations of the problem:
  - minimum time-integrated altitude
  - minimum flight time;
- proposing a three-stage hybrid approach to solve each of the problem formulations:
  - stage 1: solution structure exploration using a direct method
  - stage 2: qualitative analysis of the solution obtained in stage 1, using optimal control
  - stage 3: mathematical formulation of the TPBVP based on the qualitative analysis of stage 2;
- solving each of the problem formulations using the three-stage hybrid approach:
  - stage 1: by using DIRCOL/NUDOCCCS solvers
  - stage 2: by using the results of stage 1, understanding the underlying flight dynamics and employing optimal control theory
  - stage 3: by using optimal control theory and the BNDSCO solver;
- analysing the influence of boundary conditions on the structure of the optimal control solution of each problem formulation and the resulting values of the performance index;
- interpreting the results from the operational and computational perspectives, pointing out the trade-offs between the two;
- using effectively DIRCOL, NUDOCCCS, PROMIS and SOCS (under the GESOP environment) and BNDSCO and documenting their use.

Book Organisation We have striven to write the book so that each chapter is as much as possible independent of the others. Alas, we have not been able to attain the ideal of self-contained chapters, but we hope that a certain degree of independence has been achieved.

We begin with the introductory Chapter 1 which is deliberately brief: it gives a very concise historical context of the subject of computational optimal control, then defines the case study investigated in the rest of the book and concludes with a detailed summary of the book, chapter by chapter. Section 1.2 of Chapter 1 is an essential reference, as it describes the mathematical details of the case study problem. A reader who is pressed for time may want to glance at that section and move on.

Chapter 2 is a friendly (we hope) presentation of all the theory we use in the book. We have made an attempt to produce a reasonably readable narrative, as opposed to a dry recitation of formulae and theorems. It is not strictly necessary to read the whole of Chapter 2 in order to follow later analyses, especially if the reader is familiar with the basics of the theory. However, it might be useful at least to glance through the material—our hard-won experience shows that many "obvious" facts are far from clear, even for those who have already encountered problems of computational optimal control.

Chapters 3 and 4 are the core of the book in that they consider each variant of the case study analysed in this book. Each of these chapters is independent of the other, but both should be studied carefully if the reader is to learn anything real from this book. An impatient user may simply start reading the book from Chapters 3 or 4 and consult earlier chapters if necessary. However, these earlier chapters are not in this book as a result of a contractual obligation—we wrote them because we wished someone had written them when we embarked on the analysis of the terminal bunt problem. Perhaps reading those chapters will save the reader some of the frustration we experienced in our encounter with computational optimal control.

Chapter 5 is the part of our book which we strongly draw to the reader's attention, because it describes in practical detail—including code[1] listings—real (and tested) software implementations of the analyses presented in Chapter 4. We have striven not only to explain how to use various software packages, but also to share our (often hard-won) user experience. A separate tutorial on the most challenging of the software packages, BNDSCO, can be found in the Appendix.

Finally, we offer in Chapter 6 pragmatic conclusions based on our user experience with the tools and practice of computational optimal control. Moreover, we suggest a few ways of going beyond the approaches described in the book. If the reader is looking for "the bottom line", this is the place where we give it.

The book arose from a real-world, three-year project given to the authors by the UK Ministry of Defence, which was quite challenging and led, among others, to the first author's PhD thesis. Also, much of the material appearing in Chapter 3 appeared first in Subchan and Żbikowski (2007a) while the bulk of Chapter 4 comes from Subchan and Żbikowski (2007b).

Bibliographic Comments There are three categories of optimal control books currently available:

1. Introductory textbooks, focused mostly on theory:
   (a) Entry-level texts
      i. L. M. Hocking, *Optimal Control: An Introduction to the Theory with Applications*, Oxford University Press, 1991.
      ii. E. R. Pinch, *Optimal Control and the Calculus of Variations*, Oxford University Press, 1995.
      iii. A. E. Bryson, *Dynamic Optimization*, Addison-Wesley, 1999.
      iv. D. S. Naidu, *Optimal Control Systems*, CRC Press, 2002.
      v. D. G. Hull, *Optimal Control Theory for Applications*, Springer, 2003.
   (b) More advanced texts, but dealing with linear problems only
      i. F. L. Lewis and V. L. Syrmos, *Optimal Control*, John Wiley & Sons Ltd, 1995.
      ii. A. E. Bryson, *Applied Linear Optimal Control: Examples and Algorithms*, Cambridge University Press, 2002.
      iii. B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*, Dover, 2007.

[1] The code listings presented in this book are available electronically from http://www.wiley.com/go/zbikowski.

2. Advanced theoretical monographs

   (a) L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*, John Wiley & Sons, Inc., 1962.
   (b) A. E. Bryson and Y. C. Ho, *Applied Optimal Control. Optimization, Estimation, and Control* Revised Printing, Hemisphere, 1975.
   (c) G. Leitmann, *The Calculus of Variations and Optimal Control*, Plenum Press, 1981.
   (d) J. Macki and A. Strauss, *Introduction to Optimal Control Theory*, Springer, 1982.
   (e) V. M. Alekseev, V. M. Tikhomirov and S. V. Fomin, *Optimal Control*, Plenum Press, 1987.
   (f) T. L. Vincent and W. J. Grantham, *Nonlinear and Optimal Control Systems*, John Wiley & Sons Ltd, 1997.
   (g) R. Vinter, *Optimal Control*, Birkhäuser, 2000.
   (h) M. Athans and P. L. Falb, *Optimal Control: An Introduction to Theory and Its Application*, Dover, 2006.

3. More practically orientated monographs, including computational methods

   (a) R. Bulirsch (ed.), *Optimal Control: Calculus of Variations, Optimal Control Theory and Numerical Methods*, Birkhäuser, 1993.
   (b) R. Bulirsch and D. Kraft (eds), *Computational Optimal Control*, Birkhäuser, 1994.
   (c) J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming*, SIAM, 2001.

Among the above books, the Bryson and Ho monograph, see 2b, must be singled out as highly respected and widely used by practitioners—we benefited from it enormously. It comprehensively presents the theory in a well-organised, clear, readable and systematic way without distracting the reader by lengthy mathematical derivations. It is a ready theoretical reference for any serious user of optimal control. However, this book offers more up-to-date and expansive coverage of numerical methods than is included in Chapter 7 of Bryson and Ho.

Titles 3b and 3a as edited compilations serve as research references, and are not intended as textbooks. The book by Betts, see 3c above, is the only self-contained book on computational optimal control and was written by a practitioner from Boeing. However, it is dedicated to one approach to computational optimal control (direct method) and is focused on one commercial FORTRAN package, namely SOCS. Finally, it mainly gives an in-depth analysis of the numerical aspects of the direct method with one chapter briefly describing six examples of varying difficulty. By contrast, our book can be considered a self-study guide to real engineering practice of computational optimal control.

Shrivenham, England
2009

Acknowledgements

The investigation which was the subject of the case study considered in this book was initiated by the Director of Weapons Systems Sector, WS2, Defence Evaluation & Research Agency (DERA), Farnborough, England, and was carried out under the terms of Contract No. WSS/R4519. The study was sponsored by the UK Ministry of Defence and happened because of an exceptionally kind and dedicated professional, David East (formerly of DERA), who was instrumental in setting up the project; his gentlemanly support is gratefully acknowledged. John Cleminson (formerly of DERA and QinetiQ) kindly suggested the terminal bunt problem, provided the defining data and shared his solution. His eager support and personal modesty are much appreciated.

The GESOP (and DIDO) software used here was purchased with the funds of the Data and Information Fusion Defence Technology Centre (DIF DTC), a joint initiative between the UK Ministry of Defence and the UK defence industry. We especially thank Andy Tilbrook from General Dynamics UK for his support in obtaining funds for the GESOP (and DIDO) purchase.

Nomenclature

α angle of attack

$\boldsymbol{\lambda}$ co-state vector

$\boldsymbol{\psi}$ boundary condition

$\boldsymbol{C}$ mixed constraint

$\boldsymbol{S}$ pure state constraint

$\boldsymbol{u}, \boldsymbol{u}(t)$ control vector

$\boldsymbol{u}^*, \boldsymbol{u}^*(t)$ optimal control vector

$\boldsymbol{x}, \boldsymbol{x}(t)$ state vector

$\boldsymbol{x}^*, \boldsymbol{x}^*(t)$ optimal state vector

γ flight path angle

$\gamma(0), \gamma_0$ flight path angle initial condition

$\gamma(t_{\mathrm{f}}), \gamma_{t_{\mathrm{f}}}$ flight path angle terminal condition

ρ air density

C_d coefficient of axial aerodynamic force

C_l coefficient of normal aerodynamic force

D axial aerodynamic force

g gravitational constant

H Hamiltonian

h altitude

$h(0), h_0$ altitude initial condition

$h(t_{\mathrm{f}}), h_{t_{\mathrm{f}}}$ altitude terminal condition

h_{min} minimum altitude

J performance criterion

L normal aerodynamic force

L_{max} normalised maximum normal aerodynamic force

L_{min} normalised minimum normal aerodynamic force

m mass

S_{ref} reference area of the missile

T thrust

t_0 initial time

t_f final/terminal time

T_{max} maximum thrust

T_{min} minimum thrust

V speed

$V(0)$, V_0 speed initial condition

$V(t_f)$, V_{t_f} speed terminal condition

V_{max} maximum speed

V_{min} minimum speed

x horizontal position

$x(0)$, x_0 horizontal position initial condition

x_{t_f}, $x(t_f)$ horizontal position terminal condition

BNDSCO A software package for the numerical solution of optimal control problems using an indirect method

CAMTOS Collocation and Multiple Shooting Trajectory Optimization Software

DIRCOL Direct Collocation Method

GESOP Graphical Environment for Simulation and Optimization

GNC Guidance, Navigation and Control

GUI Graphical User Interface

KKT Karush–Kuhn–Tucker

MPBVP Multi-Point Boundary Value Problem

NLP Nonlinear Programming

NUDOCCCS Numerical Discretization Method for Optimal Control Problems with Constraints in Control and States

PROMIS Parameterized Trajectory Optimization by Direct Multiple Shooting

SOCS Sparse Optimal Control Software

SQP Sequential Quadratic Programming

TPBVP Two-Point Boundary Value Problem

TROPIC Trajectory Optimization by Direct Collocation

Contents

1

Introduction

The main focus of this short chapter is to define the case study investigated in this book (Section 1.2) and also to summarise the structure of the rest of the book (Section 1.3). These two sections are preceded with a brief account of the historical context of computational optimal control in Section 1.1.

Cruise missiles are guided weapons designed for atmospheric flight and whose primary mission is precision strike of fixed targets. This can be achieved only by a judicious approach to guidance, navigation and control (GNC). Navigation is the process of establishing the missile's location. Based on the location, guidance produces the trajectory that the missile should follow. Finally, control entails the use of actuators, so that the missile follows the desired trajectory.

The computational optimal control case study considered here arises from an approach to cruise missile guidance known as trajectory shaping. The essence of the approach is to compute an optimal trajectory together with the associated control demand. In other words, for given launch and strike conditions, find a missile trajectory which:

- hits the target in a pre-defined way;
- shapes the missile's flight in an optimal fashion;
- defines the control demand for optimal flight.

This setting leads naturally to expressing the guidance problem as an optimal control problem which cannot be solved analytically. Hence the solution approach for the trajectory shaping involves computational optimal control. This is a set of techniques which combines the theory of infinite-dimensional optimisation with numerical methods of finite-dimensional optimisation and boundary value problem solvers. Both the optimal control theory and the numerical algorithms involved are rather non-trivial in nature, and their interaction adds another layer of complexity. This book focuses on *informed use* of computational optimal control rather than development of either theory or numerics.

The theoretical and computational tools are used to elucidate the features of the special case of cruise missile trajectory shaping, the terminal bunt manoeuvre, defined in detail in Section 1.2 below. The tools are both powerful and complex. Their power gives insights into optimisation of the manoeuvre—operationally valuable knowledge. Their complexity not only challenges the analyst, but uncovers the limitations of the approach and, crucially,

Computational Optimal Control: Tools and Practice S. Subchan and R. Żbikowski