THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

Adaptive Finite Element Methods for Optimal Control Problems

Karin Kraft

Department of Mathematical Sciences
Chalmers University of Technology and Göteborg University
Göteborg, Sweden 2008

Adaptive Finite Element Methods for Optimal Control Problems
Karin Kraft

©Karin Kraft, 2008

NO 2008:1
ISSN 1652-9715
Department of Mathematical Sciences
Chalmers University of Technology and Göteborg University
SE-412 96 Göteborg
Sweden
Telephone +46 (0)31 772 1000

Printed in Göteborg, Sweden 2008

Adaptive Finite Element Methods for Optimal Control Problems

Karin Kraft

Department of Mathematical Sciences
Chalmers University of Technology and Göteborg University

Sammanfattning

Vi har härlett en numerisk metod för att lösa optimala styrningsproblem. De nödvändiga villkoren för optimum härleds med variationskalkyl och diskretiseras sedan med en finita elementmetod. Till denna metod har vi bevisat en *a posteriori* feluppskattning och använt denna för att implementera en adaptiv finita elementmetod.

En alternativ adaptiv finita elementmetod utgår från optimalitetsvillkor härledda på funktionalform där man vill minimera felet i kostnadsfunktionalen. Feluppskattningen kan då uttryckas i dualviktade residualer. Beräkningen av feluppskattningen blir därmed betydligt effektivare. Vi har implementerat även denna metod.

Slutligen presenteras två numeriska exempel.

Abstract

We have derived a method for solving optimal control problems using variational calculus for the derivation of the optimality conditions and a finite element method for the discretisation of these conditions. Further, we have derived an *a posteriori* error estimate and based on this estimate, an adaptive finite element method has been implemented.

As an alternative, we have considered a similar method where the optimality conditions are derived in a functional form and the error estimate is derived using the dual weighted residuals approach. With this method, the computation of the error estimate is more effective. This method has been implemented as well.

Finally, these adaptive finite element methods have been tested on some examples.

Keywords: Adaptive finite element method, boundary value problem, optimal control, dual weighted residual, a posteriori, error estimate

Appended papers:

Paper 1: *Using an adaptive FEM to determine the optimal control of a vehicle during a collision avoidance manoeuvre,* Proceedings of the 48th Scandinavian Conference on Simulation and Modeling (SIMS 2007), P. Bunus, D. Fritzson and C. Führer (eds), Linköping University Electronic Press, `http://www.ep.liu.se/ecp/027/`, 2007 (with Stig Larsson and Mathias Lidberg)

Paper 2: *The dual weighted residuals approach to optimal control of ordinary differential equations,* (Preprint 2008:2) (with Stig Larsson)

Acknowledgements

I am grateful to my supervisor Stig Larsson for all discussions and for the guidance through mathematics and error estimates. I also wish to thank my assistant supervisor Mathias Lidberg.

Thomas Ericsson has been very helpful and inspiring. Thank you. I also thank Nils Svanstedt, Mohammad Asadzadeh, Kjell Holmåker and Jan Lennartsson for your help.

I want to thank Christoffer Cromvik, Anna Nyström, Milena Anguelova, David Heintz, Fardin Saedpanah and Caroline Olsson for all encouragement and for making it fun to go to work.

My work was supported by the Gothenburg Mathematical Modelling Centre (GMMC).

Finally, I am very thankful for all love and support from my family, Niklas, Jonatan and Daniel.

Karin Kraft
Göteborg, January 2008

Contents

1 Introduction

Consider a car trying to avoid an object that suddenly appears on the road. The driver has some ability to steer the car. Obviously, the driver faces the problem of finding the optimal way to manoeuver the car such that a collision is avoided. This is an example of a optimal control problem which consists of a dynamical system describing the way the car moves, and an objective function to be minimised or maximised. Optimal control problems appear also in other fields of engineering, such as chemical engineering, robotics and vehicle dynamics, and in economics ([4, 9, 20]).

There are two ways to obtain the solution of optimal control problems: the *direct* and the *indirect* approaches. The direct approach discretises the dynamical system and then looks for an optimal solution of the discrete problem. In the indirect approach one determines the necessary conditions for optimality, and solves them numerically. In this work we focus on the indirect approach and use variational calculus for the derivation of the necessary conditions for optimality, resulting in a system of differential algebraic equations to be solved. We choose to discretise this system using an adaptive finite element method. We base the error control on two different *a posteriori* error estimates, taking the common dual approach in the first method and the approach of dual weighted residuals in the second.

The following section contains a mathematical formulation of an optimal control problem and the history of the solution strategies of such problems. Section 3 includes a description of the most common numerical methods used to solve the optimal control problems and an introduction to the finite element method. Section 4 contains a description of the solution of an optimal control problem using variational calculus and an adaptive finite element method. The error estimate which is used for the adaptive method is also given. Section 5 describes the approach of dual weighted residuals to an optimal control problem and includes the description of the adaptive finite element method. Finally, Section 6 contains numerical examples of the two methods derived in previous sections and some comparisons to Matlab's boundary value solver `bvp4c` [27] are made. The last section contains the plans for future research.

2 The Optimal Control Problem

The optimal control problem can be formulated as follows. We have a dynamical system $\dot{x} = f(x, u)$, where $x(t) \in \mathbb{R}^d$ are the state variables which are continuous functions of time t, and $u(t) \in \mathbb{R}^m$ are the control variables and we want to minimise a cost functional $\mathcal{J}(x, u)$. This leads to the problem finding (x, u) such that

$$\begin{aligned} &\text{minimise} && \mathcal{J}(x,u) \\ &\text{subject to} && \dot{x}(t) = f(x(t), u(t)), \quad 0 < t < T, \\ &&& I_0 x(0) = x_0, \quad I_T x(T) = x_T. \end{aligned} \tag{1}$$

The last line specifies the boundary conditions of the dynamical system.

2.1 The direct and indirect approaches

The numerical solution of optimal control problems can be approached in two different ways, the *direct* and the *indirect* approaches [6]. In the direct approach the dynamical system is discretised and approximated by a finite number of parameters. After the discretisation, the problem is a finite-dimensional optimisation problem which can be solved using Non-Linear Programming solvers, for example SQP (see [6, 10, 17]).

In the indirect approach the necessary conditions for optimality are first determined using variational techniques, such as variational calculus [8] or Pontryagin's maximum principle [24], and then the resulting equations are discretised and solved. The necessary conditions for optimality consist of the differential equations from the original problem, an additional set of differential equations called the adjoint equations and a set of algebraic equations.

2.2 Variational calculus

Variational calculus was developed in the end of the 17th century. It is used for solving extremal problems and it was further developed for problems in mechanics by Lagrange and Hamilton [16]. More precisely, variational calculus computes an extreme point to a certain quantity of a system described as a dynamical system. For example, the classical *brachistochrone* problem is the problem of finding the curve which takes a particle, acted on by gravity, from point A to point B in shortest time. Using variational calculus, one finds that this time is minimised by a hyperbolic cosine curve, since any perturbation of this curve increases the time.

Variational calculus can also be used for optimal control problems as long as there are no constraints on the controls [18]. The difference to the extremal problems above is that the control problems depend not only on x and $\dot{x}$ but also on the controls, u. The controls are often signals to the system which are usually subject to limitations and therefore the problem includes constraints on the controls. In the 1950's Pontryagin and co-workers presented Pontryagin's maximum principle [24], which is a generalisation of the variational calculus to handle constraints on the controls.

In this thesis we take an indirect approach to the optimal control problem and derive the necessary conditions for optimality using variational calculus. We also take a modern approach and present the variational calculus in a functional analytic framework. We believe that the indirect approach in combination with the finite element method, which is described below, gives us the possibility to control the error in the solution and makes it possible to solve more difficult problems.

3 Numerical Solution Methods

The most common numerical methods for solving the discretised optimal control problems are the multiple shooting method and the collocation method [5].

3.1 Shooting and collocation

The shooting method is a numerical method for solving boundary value problems of the form

$$\dot{x} = f(t, x), \quad 0 < t < T \tag{2}$$

$$g(x(0), x(T)) = 0, \tag{3}$$

where $x, g \in \mathbb{R}^m$. The name of the method comes from the procedure of aiming a cannon so the cannonball hits the target [6, 25]. One considers the function $h(c) = g(c, x(T, c))$, where $x(T, c)$ is the value of $x(T)$ obtained by shooting with $x(0) = c$, that is propagating the differential equation from 0 to T. The equation $h(c) = 0$ can then be solved using any appropriate method.

The shooting method has been further developed into multiple shooting. In this method the computational interval is refined into smaller sub-intervals where the shooting method is applied in each sub-interval. This method has been used for optimal control problems, see for example [23].

The use of sub-intervals is present also in the collocation method [2]. One determines a continuous piecewise polynomial which fulfils the differential equation in the collocation points $t_n + c_i h$, where t_n is the initial time of the interval, h is the interval length and $0 \leq c_i \leq 1$ are suitable chosen points, for instance the roots of the Legendre polynomials [11].

We have used the Boundary Value Problem solver `bvp4c` [27] in Matlab to benchmark our results. This solver is based on the collocation idea. An improved collocation method that controls the error and the residual in the solution has recently been presented [26].

3.2 The finite element method

The finite element method was developed in the 1950's and 1960's, mainly by engineers, to solve equations in elasticity and structural mechanics. It was developed as a geometrically more flexible alternative to the finite difference method (see for example [28]). The finite element method is a special case of the Rayleigh-Ritz-Galerkin-methods which are used to approximate partial differential equations and it has a solid foundation in functional analysis [7]. This is one of its strengths, as is the possibility to use it on complicated domains. The mathematical foundation makes it easier to derive analytic error estimates which can be used to improve the approximate solutions.

Traditionally the finite element method has been used for partial differential equations. However, some work has been done on adaptive finite element methods for ODE:s, see for example [21, 22, 14, 15].

We illustrate how the finite element method works in the context of a simple boundary value problem:

$$\begin{aligned} -\ddot{x} &= f(t), \qquad 0 < t < T, \\ x(0) &= a, \qquad x(T) = b. \end{aligned} \tag{4}$$

We start by reformulating the problem in weak form by introducing a set of test functions V which satisfy the boundary conditions $v(0) = v(T) = 0$, multiply equation (4) by a test function $v \in V$, integrate over the interval $[0, T]$, and then integrate by parts. The weak form is: Find $x \in C^1([0, T])$ such that,

$$x(0) = a, \qquad x(T) = b, \tag{5}$$

$$\int_0^T \dot{x}\dot{v}\, dt = -\int_0^T f v\, dt, \qquad \text{for all } v \in V. \tag{6}$$

Let V_h be a subspace of V consisting of for instance piecewise linear functions on $[0, T]$ with sub-intervals of size h. We want to solve (6) for all $v \in V_h$ with the Ansatz $x_h(t) = a\varphi_0(t) + \sum_{n=1}^{N-1} x_n\varphi_n(t) + b\varphi_N(t)$, where $\varphi_n, n = 1, \ldots, N-1$ is a basis for V_h and φ_0 and φ_N are additional basis functions such that $\varphi_0(0) = \varphi_N(T) = 1$. In this example the so called trial space and test space, that is the spaces containing x and v, respectively, are discretised in the same way, but this need not be the case. The fact that the finite element methods are based on the weak form (6) rather than (4) makes it easier to use tools from functional analysis to derive error estimates.

There are two types of error estimates, *a priori* and *a posteriori* error estimates. The first type gives a bound of the error $e = x - x_h$, in terms of x, h, and the data a, b and f. Since the estimate depends on the unknown exact solution it cannot be explicitly computed and is used to investigate the convergence of the numerical method. In the second type of error estimate, the *a posteriori* error estimate, the error bound is expressed in terms of the data, x_h and h. The *a posteriori* error estimates can be explicitly computed, since they depend only on known or computable quantities. The *a posteriori* error estimates are used in constructing adaptive algorithms, see Algorithm 1, which solve the equation repeatedly on refined meshes.

Algorithm 1: An adaptive finite element method

Solve the equation on an initial mesh;
Compute the error estimate est;
while $|est| \geq TOL$ **do**
  Refine the mesh according to the error estimate, i.e., refine intervals that give large contributions to the error;
  Solve the equation on the refined mesh;
  Compute the error estimate on the refined mesh;
end

More about error estimates and adaptive finite element methods can be found in [7, 12, 13, 19].

In this work we use an adaptive finite element method similar to the one in [15]. We only consider *a posteriori* error estimates since we want to construct an adaptive algorithm. We derive an *a posteriori* error estimate minimising the error expressed in an arbitrary linear functional G. Further, we take a different approach and derive an *a posteriori* error estimate estimating the error, $\mathcal{J}(x,u) - \mathcal{J}(x_h, u_h)$, in the minimised objective functional $\mathcal{J}$. This approach is called dual weighted residuals [3].

4 The Finite Element Method for Optimal Control

In this section we solve an optimal control problem using an adaptive finite element method and the section is a summary of Paper 1.

4.1 Necessary conditions for optimality

Consider the optimal control problem

$$\begin{aligned} \text{minimise} \quad & \mathcal{J}(y(t),u(t)) = l(y(0),y(T)) + \int_0^T L(y(t),u(t))\,dt, \\ \text{subject to} \quad & \dot{y}(t) = f(y(t),u(t)), \qquad 0 < t < T, \\ & J_0 y(0) = y_0, \quad J_T y(T) = y_T, \end{aligned} \tag{7}$$

where

$$\begin{aligned} l &: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}, \\ L &: \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}, \\ f &: \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^d, \end{aligned}$$

are smooth functions and J_0 and J_T are diagonal matrices with zeroes or ones on the diagonals. We assume $y_0 \in R(J_0)$, $y_T \in R(J_T)$, where $R(A)$ denotes the range of a matrix A. We note that the boundary conditions are equality conditions and that there are no constraints on the controls u. We want to determine pairs (y,u) that fulfil (7). Taking an indirect approach, as described in Section 2, we first write down the necessary conditions for optimality derived by variational calculus. We introduce the Hamiltonian

$$H = L(y,u) + z^{\mathrm{T}} f(y,u),$$

and then the optimal (y^*, u^*, z^*) fulfil

$$\begin{aligned} \dot{y} &= \frac{\partial H}{\partial z} = f(y,u), \\ \dot{z} &= -\frac{\partial H}{\partial y} = -\frac{\partial L}{\partial y} - \left(\frac{\partial f}{\partial y}\right)^{\mathrm{T}} z, \\ 0 &= \frac{\partial H}{\partial u} = \frac{\partial L}{\partial u} + z^{\mathrm{T}} \frac{\partial f}{\partial u}, \\ J_0 y(0) &= y_0, \;\; J_T y(T) = y_T, \\ (I - J_0) z(0) &= z_0, \;\; (I - J_T) z(T) = z_T, \end{aligned} \tag{8}$$

where $z \in \mathbb{R}^d$ are called the co-states or adjoint variables, and z_0 and z_T are obtained from $\mathcal{J}$. We note here that since y_0 and y_T are in the ranges of J_0 and J_T, respectively, the boundary conditions are imposed on those components of z that are complementary to the components of x with boundary conditions.

4.2 Reducing the problem

To simplify the problem we assume in Paper 1 that the algebraic equation on the third line in (8) can be solved explicitly for u^* which is then substituted into the other equations. We then have a two point boundary value problem. In the case that (8) cannot be solved explicitly our equations constitute a system of differential algebraic equations (see Paper 2). We reformulate the problem by joining y and z into the new variable x and end up with the system

$$\begin{aligned} &\dot{x} = f(x), \qquad 0 < t < T, \\ &I_0 x(0) = x_0, \;\; I_T x(T) = x_T, \end{aligned} \tag{9}$$

where $x(t) \in \mathbb{R}^{2d}$ and I_0 and I_T are diagonal matrices with zeroes or ones on the diagonals and $\mathrm{rank}(I_0) + \mathrm{rank}(I_T) = 2d$.

4.3 Discretisation of the problem

We begin the discretisation of the problem (9) by writing it in a weak form and then continue with the definitions of the appropriate function spaces. Take the scalar product between (9) and a test function $v \in V = C^1([0,T])$, integrate over the interval $[0,T]$, leading to the weak formulation of the problem is: Seek $x \in V$ such that

$$\begin{aligned} &I_0 x(0) = x_0, \;\; I_T x(T) = x_T, \\ &F(x,v) = \int_0^T (\dot{x} - f(x), v)\, dt = 0, \;\; \forall v \in V. \end{aligned} \tag{10}$$

The problem in (10) is an infinite dimensional problem which we discretise to get a finite problem. We choose the following discretisation.

- Mesh: $0 = t_0 < t_1 < t_2 < \ldots < t_N = T$, $h_n = t_n - t_{n-1}$ and $I_n = (t_{n-1}, t_n)$.
- Trial space: $W_h = \mathbb{R}^d \times \{w : w|_{I_n} \in P^0(I_n)\} \times \mathbb{R}^d$, discontinuous piecewise constant functions.
- Test space: $V_h = \left\{v : v|_{I_n} \in P^1(I_n)\right\} \cap C^0([0,T])$, continuous piecewise linear functions.

The notation $P^k(I_n)$ refers to the $\mathbb{R}^d$-valued polynomials of degree k on the interval I_n. We also introduce the left and right limits $w_n^{\pm} = \lim_{t \to t_n^{\pm}} w(t)$, and jumps $[w]_n =$

$w_n^+ - w_n^-$. The two factors $\mathbb{R}^d$ in W_h contain the boundary values w_0^- and w_N^+. Now our finite element problem can be stated: Find a function $x_h \in W_h$ which fulfils

$$
\begin{aligned}
&I_0 X_0^- = x_0, \;\; I_T X_N^+ = x_T, \\
&F(X, v) = \sum_{n=1}^{N} \int_{I_n} (\dot{X} - f(X), v)\, dt + \sum_{n=0}^{N} ([X]_n, v_n) = 0, \;\; \forall v \in V_h.
\end{aligned}
\tag{11}
$$

Here the definition of the form F from (10) has been extended to include the contributions from the jump terms which appear since we use discontinuous trial functions. Since the trial space consists of piecewise constant functions, we have $\dot{X} = 0$. Hence, (11) results in a system of $(N+2)d$ equations, more precisely, d boundary conditions and $(N+1)d$ equations. With boundary conditions at both ends, the equations are coupled and thus we cannot use time stepping. Therefore, the equations in the system have to be solved simultaneously.

4.4 An a posteriori error estimate

In order to evaluate how good the computed solution is and to construct an adaptive finite element method we derive an *a posteriori* error estimate. We introduce the notation $\|v\|_{I_n} = \sup_{t \in I_n} \|v(t)\|$, where $\|\cdot\|$ denotes the norm in $\mathbb{R}^d$ or $\mathbb{R}^m$.

Theorem 4.1. *Let $e = x - x_h$ be the error in the finite element solution of the boundary value problem in (9). The error expressed in a linear functional G is bounded by*

$$
|G(e)| \leq \sum_{n=1}^{N} \mathcal{R}_n \mathcal{I}_n,
$$

where

$$
\begin{aligned}
\mathcal{R}_1 &= h_1 \big\|\dot{X} - f(X)\big\|_{I_1} + \big\| [X]_0 \big\| + \frac{h_1}{h_1 + h_2} \big\| [X]_1 \big\|, \\
\mathcal{R}_n &= h_n \big\|\dot{X} - f(X)\big\|_{I_n} + \frac{h_n}{h_n + h_{n-1}} \big\| [X]_{n-1} \big\| + \frac{h_n}{h_n + h_{n+1}} \big\| [X]_n \big\|, \\
&\quad n = 2, \dots, N-1, \\
\mathcal{R}_N &= h_N \big\|\dot{X} - f(X)\big\|_{I_N} + \frac{h_N}{h_N + h_{N-1}} \big\| [X]_{N-1} \big\| + \big\| [X]_N \big\|,
\end{aligned}
$$

$$
\mathcal{I}_n = C h_n \int_{I_n} \left| \ddot{\phi} \right| dt.
$$

C is a constant and ϕ is the solution to the linearised dual problem to (9) with data functional G.

In this error estimate, $\mathcal{R}_n$ mainly describes how well the approximate solution satisfies the differential equation and $\mathcal{I}_n$ describes how sensitive the error functional G is to local residuals.

We present an adaptive algorithm based on this error estimate (see Algorithm 1). It has been implemented (see also [1]) and bench-marked to the boundary value problem solver `bvp4c` in Matlab, see Paper 1. A numerical example is given in Section 6.

In order to compute the error estimate above we need to solve an additional dual problem of the same size as the original one, which means that for each step in the adaptive algorithm we double the size of the problem. For already large problems this is a major drawback and can force the user to choose another numerical method. However, in the following section we present a method where we have removed this feature.

5 The Dual Weighted Residuals Approach

This section is a summary of Paper 2. We express the optimal control problem in a general and abstract form by introducing smooth functionals $\mathcal{F}(x, u; \varphi)$ and $\mathcal{J}(x, u)$, where we use the notation that functionals depend non-linearly on the arguments before the semicolon and linearly on the arguments after the semicolon. For the proofs and the details of the definitions of the different spaces and functionals we refer the reader to Paper 2. The problem we study is this: Determine $x \in \tilde{W}$ and $u \in U$ such that

$$\begin{aligned} &\text{minimise} && \mathcal{J}(x, u), \\ &\text{subject to} && \mathcal{F}(x, u; \varphi) = 0, \quad \forall \varphi \in V. \end{aligned} \tag{12}$$

This is a constrained optimisation problem and the necessary condition for an optimum is expressed in terms of the Lagrange functional

$$\mathcal{L}(x, u; z) = \mathcal{J}(x, u) + \mathcal{F}(x, u; z), \quad (x, u, z) \in W \times U \times V,$$

where z are the adjoint variables.

5.1 Necessary conditions for optimality

The necessary conditions for an optimum are presented in the following theorem.

Theorem 5.1. *The necessary condition for an optimum $(x, u, z) \in \tilde{W} \times U \times V$ is given by*

$$\mathcal{L}'(x, u; z, \varphi) := \mathcal{L}'(x, u; z)\varphi = 0, \quad \forall \varphi \in \dot{W} \times U \times V, \tag{13}$$

that is,

$$\mathcal{J}_x'(x, u; \varphi_x) + \mathcal{F}_x'(x, u; z, \varphi_x) = 0, \qquad \forall \varphi_x \in \dot{W}, \tag{14}$$

$$\mathcal{J}_u'(x, u; \varphi_u) + \mathcal{F}_u'(x, u; z, \varphi_u) = 0, \qquad \forall \varphi_u \in U, \tag{15}$$

$$\mathcal{F}(x, u; \varphi_z) = 0, \qquad \forall \varphi_z \in V. \tag{16}$$

The proof of Theorem 5.1 can be found in Paper 2. We note that equation (16) is the original differential equation in (12) and (14) is the dual equation. We discretise these equations and derive an *a posteriori* error representation for the Galerkin approximation of the equations.

Theorem 5.2. *Let* $(x, u, z) \in \tilde{W} \times U \times V$ *and* $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ *be the exact and discrete solutions of* (14)-(16)*, respectively. Then*

$$\mathcal{J}(x, u) - \mathcal{J}(x_h, u_h) = \tfrac{1}{2}\rho_x + \tfrac{1}{2}\rho_z + \tfrac{1}{2}\rho_u + R, \tag{17}$$

with the residuals ρ_x*,* ρ_z*, and* ρ_u *defined as*

$$\begin{aligned}
\rho_x &= \mathcal{J}'_x(x_h, u_h; x - \tilde{x}_h) + \mathcal{F}'_x(x_h, u_h; z_h, x - \tilde{x}_h), \\
\rho_u &= \mathcal{J}'_u(x_h, u_h; u - \tilde{u}_h) + \mathcal{F}'_u(x_h, u_h; z_h, u - \tilde{u}_h), \\
\rho_z &= \mathcal{F}(x_h, u_h; z - \tilde{z}_h).
\end{aligned} \tag{18}$$

Here $(\tilde{x}_h, \tilde{u}_h, \tilde{z}_h) \in \tilde{W}_h \times U_h \times V_h$ *is arbitrary. The remainder term* R *is given by*

$$\begin{aligned}
R = \tfrac{1}{2} \int_0^1 \Big(& J'''(x_h + se_x, u_h + se_u; e, e, e) \\
& + \mathcal{F}'''(x_h + se_x, u_h + se_u, z_h + se_z; z, e, e, e)\Big) s(s-1)\, ds,
\end{aligned} \tag{19}$$

where $e = (e_x, e_u, e_z) \in \dot{W} \times U \times V$*,* $e_x = x - x_h$*,* $e_u = u - u_h$*, and* $e_z = z - z_h$*.*

The remainder term is cubic in the error and can therefore often be neglected. In particular, we note that $R = 0$ in the case when $\mathcal{F}(\cdot, \cdot; \cdot)$ is tri-linear and $\mathcal{J}(\cdot, \cdot)$ is bi-quadratic.

Using these theorems we derive the necessary conditions and error representations for an optimal control problem of the form (7) and for a linear/quadratic optimal control problem of the form,

$$\begin{aligned}
\text{minimise} \quad & \mathcal{J}(x, u) = \|x(0) - \bar{x}_0\|^2_{S_0} + \|x(T) - \bar{x}_T\|^2_{S_T} \\
& \qquad + \int_0^T \left(\|u - \bar{u}\|^2_R + \|x - \bar{x}\|^2_Q\right) dt, \\
\text{subject to} \quad & \dot{x} = A(t)x + B(t)u, \quad 0 < t < T, \\
& I_0 x(0) = x_0, \quad I_T x(T) = x_T,
\end{aligned} \tag{20}$$

The finite element discretisation of this problem is described in the following section.

5.2 A finite element discretisation

Applying Theorem 5.1 to the linear quadratic optimal control problem (20) we obtain

$$\begin{aligned}&\int_0^T (\varphi_x, 2Q(x-\bar{x}) - \dot{z} - A^{\mathrm{T}} z)\,dt \\ &\quad + (\varphi_{x,0}^-, 2S_0(x_0^- - \bar{x}_0) - z_0) + (\varphi_{x,N}^+, 2S_T(x_N^+ - \bar{x}_T) + z_N) = 0, \quad \forall \varphi_x \in \dot{W},\end{aligned} \tag{21}$$

$$\int_0^T (\varphi_u, 2R(u-\bar{u}) - B^{\mathrm{T}} z)\,dt = 0, \quad \forall \varphi_u \in U, \tag{22}$$

$$\int_0^T (\dot{x} - Ax - Bu, \varphi_z)\,dt = 0, \quad \forall \varphi_z \in V. \tag{23}$$

We discretise the state equation (23) with the same discontinuous Galerkin method as in the previous section and Paper 1, using W_h as trial space and V_h as test space: Seek $x_h \in W_h$ which fulfils

$$\begin{aligned}&I_0 x_{h,0}^- = x_0,\ I_T x_{h,N}^+ = x_T, \\ &\int_0^T (\dot{x}_h - Ax_h - Bu_h, \varphi)\,dt + \sum_{n=0}^{N} ([x_h]_n, \varphi_n) = 0, \quad \forall \varphi \in V_h.\end{aligned}$$

Since we have discontinuous trial functions we get an extra sum arising from the jump terms. The dual equation (21) is discretised using the continuous Galerkin method: Seek $z_h \in V_h$ which fulfils

$$\begin{aligned}&\int_0^T (\varphi, 2Q(x_h - \bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h)\,dt + (\varphi_0^-, 2S_0(x_{h,0}^- - \bar{x}_0) - z_{h,0}) \\ &\quad + (\varphi_N^+, 2S_T(x_{h,N}^+ - \bar{x}_T) + z_{h,N}) = 0 \quad \forall \varphi \in \dot{W}_h,\end{aligned} \tag{24}$$

and finally we discretise the equation for the controls, (22), using a continuous Galerkin method: Seek $u_h \in U_h$

$$\int_0^T (2Ru_h - B^{\mathrm{T}} z_h, \varphi_u)\,dt = 0, \quad \forall \varphi_u \in U_h. \tag{25}$$

We have three sets of equations which must be solved simultaneously in order to obtain the approximate solutions (x_h, u_h, z_h).

5.3 An a posteriori error estimate

In order to implement an adaptive finite element method (see Algorithm 1) we derive an *a posteriori* error estimate, using Theorem 5.2 with $R = 0$ since we have a linear/quadratic problem.

Theorem 5.3. *The* a posteriori *error estimate for the finite element discretisation of the linear quadratic optimal control problem described above is given by*

$$|\mathcal{J}(x,u) - \mathcal{J}(x_h,u_h)| \leq \frac{1}{2}\sum_{n=1}^{N}\Big(R_n^z\omega_n^x + R_n^u\omega_n^u + R_n^x\omega_n^z\Big), \tag{26}$$

where the residuals and weights are defined by

$$\begin{aligned}
R_n^z &= h_n\|\dot{z}_h + A^{\mathrm{T}}z_h + 2Q(x_h - \bar{x})\|_{I_n},\\
\omega_n^x &= \|x - \tilde{x}_h\|_{I_n},\\
R_n^x &= h_n\|\dot{x}_h - Ax_h - Bu_h\|_{I_n} + \frac{h_n}{h_n + h_{n+1}}\|\,[x_h]_n\,\| + \frac{h_n}{h_n + h_{n-1}}\|\,[x_h]_{n-1}\,\|,\\
\omega_n^z &= \|z - \tilde{z}_h\|_{I_n},\\
R_n^u &= h_n\|2Ru_h - B^{\mathrm{T}}z_h\|_{I_n},\\
\omega_n^u &= \|u - \tilde{u}_h\|_{I_n},
\end{aligned}$$

where $h_0 = h_{N+1} = 0$.

The error estimate depends only on the already computed numerical solution. This means that we do not need to compute any new solutions in order to use the error estimate. This can be compared to the adaptive finite element method in Section 4, where an additional dual problem of the same size as the whole system (21)–(22) has to be solved. However, using this method we can only control the error in $\mathcal{J}$.

In the explicit calculation of the error estimate (26) we use that the interpolation errors in ω_x, ω_z and ω_u are bounded by

$$\begin{aligned}
\|x - \tilde{x}_h\|_{I_n} &\leq h_n\|\dot{x}\|_{I_n},\\
\|z - \tilde{z}_h\|_{I_n} &\leq h_n^2\|\ddot{z}\|_{I_n},\\
\|u - \tilde{u}_h\|_{I_n} &\leq h_n^2\|\ddot{u}\|_{I_n},
\end{aligned}$$

where the derivatives are approximated by difference quotients of the discrete solution.

The refinement of the mesh in the adaptive algorithm is done according to the principle of equidistribution, that is, we want each interval to give equally large contribution to the error estimate and insert new nodes to fulfil this criterion.

A numerical example is given in the following section.

6 Numerical Examples

In this section we present numerical examples which have been solved by using the numerical methods described in the previous sections.

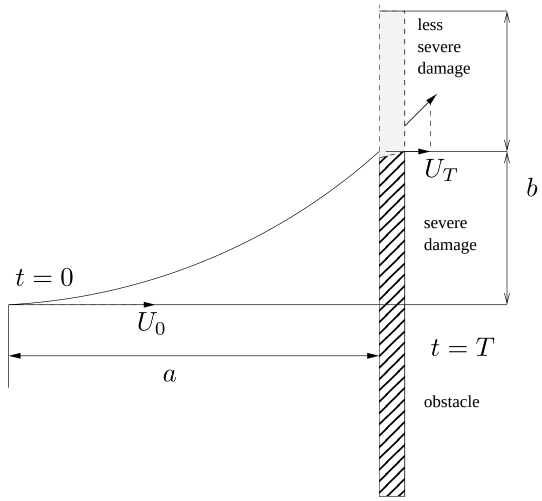

Figure 1: The collision avoidance manoeuvre.

6.1 A collision avoidance manoeuvre

The simplest way to model a vehicle is to model it as a point mass on which a force acts. We let β be the angle between this force and the direction of the initial track. We introduce the X-axis as the direction of the original track and the Y-axis as the axis perpendicular to the X-axis. The equations of planar motion for the vehicle then become

$$\begin{aligned} \ddot{X} &= -\mu g \cos(\beta), \\ \ddot{Y} &= \mu g \sin(\beta), \end{aligned} \tag{27}$$

where μ is the friction coefficient and g is the gravitational acceleration. This model is used in Paper 1.

We test our solver on an example from vehicle dynamics. It is a collision avoidance manoeuvre, where a vehicle should be steered in such a way that it avoids an object in the road and minimises the final velocity, see Figure 6.1.

We use the equations (27) to describe the dynamics and then the equations of motion for the vehicle are

$$\dot{z} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{U} \\ \dot{V} \end{bmatrix} = \begin{bmatrix} U \\ V \\ -\mu g \cos(\beta) \\ \mu g \sin(\beta) \end{bmatrix},$$

where U and V are the velocities in the X and Y directions, respectively, and β is the steering angle. In this model U, V, X, Y are the states and β is the control. We want

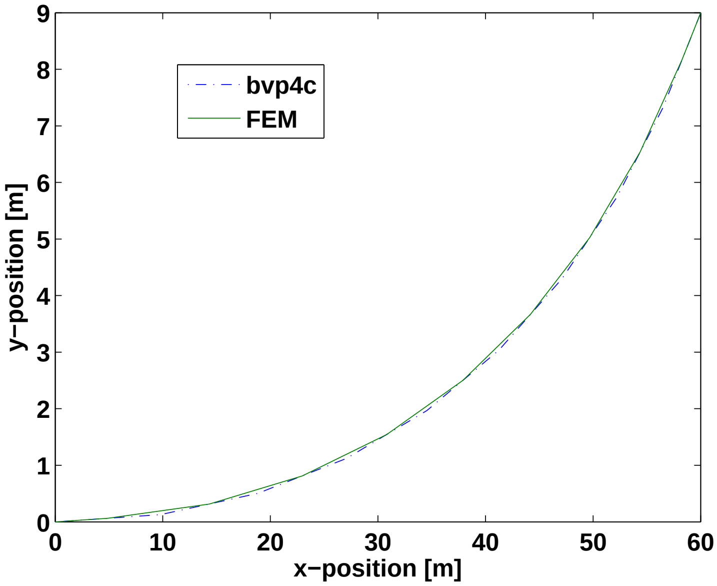

Figure 2: The position of the vehicle when it is manoeuvred in the optimal way for the manoeuvre distances $a = 60$ m and $b = 9$ m.

to minimise the speed at the time of the accident in order to reduce the damage. Therefore we formulate an optimal control problem: Find the state $z(t) \in \mathbb{R}^n$ and control $\beta(t) \in \mathbb{R}^m$ which fulfil the following minimisation problem

$$\begin{aligned} \text{minimise} \quad & \mathcal{J}(z,\beta) = c^{\mathrm{T}} z\,(T) \\ \text{subject to} \quad & \dot{z}(t) = f(z,\beta), \\ & J_0 z(0) = z_0, \quad J_T z(T) = z_T. \end{aligned}$$

Here J_0 and J_T are diagonal matrices with zeroes or ones on the diagonals, f is given by the right hand side of (28) and $c^{\mathrm{T}} = (0, 0, 1, 0)$. We solve this problem using the method described in Section 4.

Figure 2 shows some results from the case with initial velocity $u_0 = 90$ km/h (25 m/s) of the vehicle and the manoeuvre distances $a = 60$ m and $b = 9$ m, that is, the object appears 60 m in front of the vehicle. We see in Figure 2 that the solutions from the FEM solver and `bvp4c` almost coincide. The final velocity which is the quantity we minimise, is 31.0 km/h from both `bvp4c` and the FEM solver. More results can be found in Paper 1.

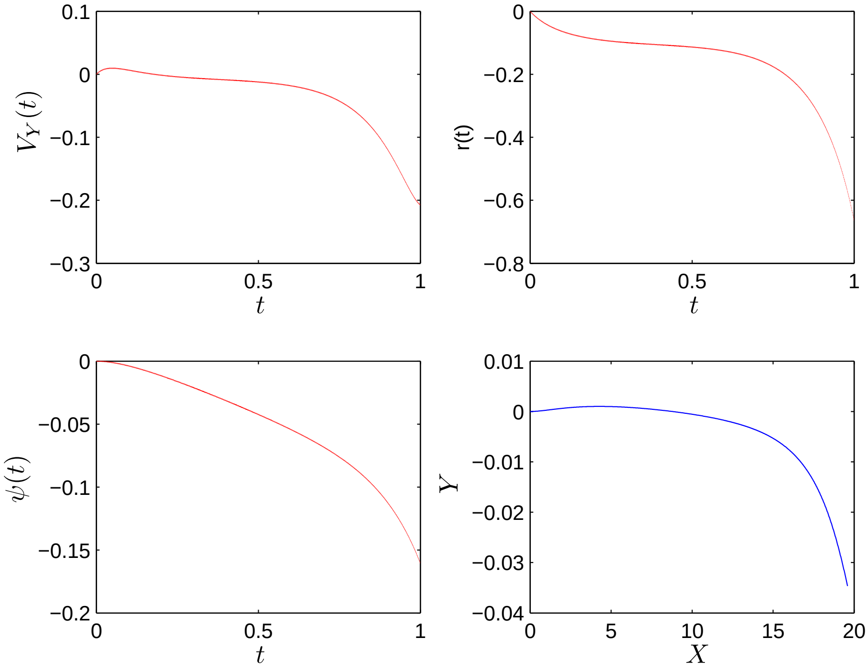

Figure 3: The optimal states. The last image shows the optimal track.

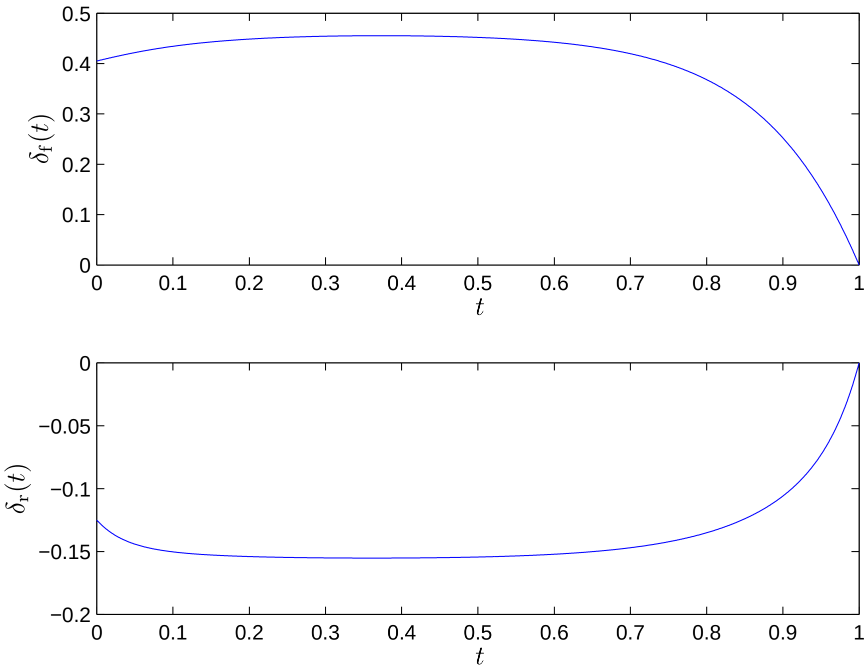

Figure 4: The optimal controls u.

6.2 A linear quadratic optimal control problem

We test the method described in Section 5. In order to get a linear/quadratic problem we use a simplified model from vehicle dynamics describing a vehicle that is braked on a surface with different friction on the wheel pairs.We let the control variable u be $u_1 = \delta_f$ and $u_2 = \delta_r$ and the state variable x is

$$x(t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} V_X \\ V_Y \\ r \\ \psi \\ X \\ Y \end{bmatrix}. \tag{28}$$

The differential equations are

$$\dot{x}(t) = \begin{bmatrix} \dot{V}_X \\ \dot{V}_Y \\ \dot{r} \\ \dot{\psi} \\ \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21}V_Y + a_{22}r + bf_1\delta_{\rm f} + br_1\delta_{\rm r} \\ a_{31}V_Y + a_{32}r + bf_2\delta_{\rm f} + br_2\delta_{\rm r} + r_{\rm brake} \\ r \\ V_X \\ V_Y \end{bmatrix} = Ax(t) + Bu(t) + b. \quad (29)$$

The goal functional is

$$J(x,u) = \int_0^T \tfrac{1}{2}\left(Y^2 + r^2 + V_Y^2 + \delta_{\rm f}^2 + \delta_{\rm r}^2\right) dt = \int_0^T \left(\|x\|_Q^2 + \|u\|_R^2\right) dt.$$

We have used the boundary condition $x_1(0) = 25$ and $x_2(0) = x_3(0) = x_4(0) = x_5(0) = x_6(0) = 0$. The numerical solution for the optimal states and controls can be found in Figures 3 and 4. The initial discretisation was made with 10 intervals and when the given tolerance of 10^{-6} was achieved the adaptive method had refined the mesh into 1072 intervals. The convergence rate of the solution in the numerical example is 2.03 and the theoretical order is 2 (see Paper 2).

7 Future Research

In this thesis we have presented a new approach to adaptive finite element solution of optimal control problems using the approach of dual weighted residuals. The theory for this method, including derivation of necessary conditions for optimality, error estimates, and finite element discretisations, is presented.

The numerical method has been tested on a linear/quadratic optimal control problem, but more extensive testing of the method on non-linear problems will be done. We also want to compare the performance of our method to direct methods. In addition, we plan to include constraints on the control and state variables over the entire interval in order to be able to solve more sophisticated models.

References

[1] C. Andersson, *Solving optimal control problems using FEM*, Master's thesis, Chalmers University of Technology, 2007.

[2] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, first ed., Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1988.

[3] R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation in finite element methods*, Acta Numer. **10** (2001), 1–102.

[4] E. Bertolazzi, F. Biral, and M. Da Lio, *Symbolic-numeric efficient solution of optimal control problems for multibody systems*, J. Comput. Appl. Math. **185** (2006), no. 2, 404–421.

[5] J. T. Betts, *Survey of numerical methods for trajectory optimization*, AIAA J. of Guidance, Control and Dynamics **21** (1998), 193–207.

[6] ______, *Practical Methods for Optimal Control Using Nonlinear Programming*, SIAM, Philadelphia, 2001.

[7] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, second ed., Springer-Verlag, New York, 2002.

[8] A. E. Bryson, Jr and Y. Ho, *Applied Optimal Control*, Hemisphere Publishing Corporation, Washington, D.C., 1975.

[9] C. Büskens and H. Maurer, *Nonlinear programming methods for real-time control of an industrial robot*, J. Optim. Theory Appl. **107** (2000), 505–527.

[10] C. Büskens and H. Maurer, *SQP-methods for solving optimal control problems with control and state constraints: adjoint variables, sensitivity analysis and real-time control*, J. Comput. Appl. Math. **120** (2000), 85–108, SQP-based direct discretization methods for practical optimal control problems.

[11] E. Eich-Soellner and C. Führer, *Numerical Methods in Multibody Dynamics*, B.G Teubner, Stuttgart, 1998.

[12] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Introduction to adaptive methods for differential equations*, Acta Numer. (1995), 105–158.

[13] ______, *Computational Differential Equations*, Cambridge University Press, Cambridge, 1996.

[14] D. Estep, *A posteriori error bounds and global error control for approximation of ordinary differential equations*, SIAM Journal on Numerical Analysis **32** (1995), no. 1, 1–48.

[15] D. Estep, D. H. Hodges, and M. Warner, *Computational error estimation and adaptive error control for a finite element solution of launch vehicle trajectory problems*, SIAM J. Sci. Comput. (1999), 1609–1631 (electronic).

[16] G. R. Fowles and G.L. Cassiday, *Analytical Mechanics*, Harcourt Brace College Publisher, Orlando, FL, 1993.

[17] P. E. Gill, L. O. Jay, M. W. Leonard, L. R. Petzold, and V. Sharma, *An SQP method for the optimal control of large-scale dynamical systems*, J. Comput. Appl. Math. **120** (2000), 197–213, SQP-based direct discretization methods for practical optimal control problems.

[18] D. E. Kirk, *Optimal Control Theory, An Introduction*, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1970.

[19] S. Larsson and V. Thomée, *Partial Differential Equations with Numerical Methods*, Springer-Verlag, Berlin, 2003.

[20] M. Lidberg, *Design of Optimal Control Processes for Closed-Loop Chain SCARA-Like Robots*, Ph.D. thesis, Chalmers University of Technology, 2004.

[21] A. Logg, *Multi-adaptive Galerkin methods for ODEs. I*, SIAM J. Sci. Comput. (2003), 1879–1902 (electronic).

[22] ______, *Multi-adaptive Galerkin methods for ODEs. II. Implementation and applications*, SIAM J. Sci. Comput. (2003/04), 1119–1141 (electronic).

[23] H. J. Oberle and W. Grimm, *BNDSCO a Program for the Numerical Solution of Optimal Control Problems*, Tech. report, Deutsche Forschungs- and Versuchsanstalt fur Luft- und Raumfahrt e.V., 1989.

[24] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*, revised printing ed., John Wiley and Sons, Inc., New York, 1962.

[25] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes, the Art of Scientific Computing*, first ed., Cambridge University Press, Cambridge, 1986.

[26] L. F. Shampine and J. Kierzenka, *A BVP solver that Controls Residual and Error*, Tech. report, MathWorks, 2007, `http://www.mathworks.com`.

[27] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, *Solving Boundary Value Problems for Ordinary Differential Equations in Matlab using* `bvp4c`, Tech. report, MathWorks, 2000, `http://www.mathworks.com`.

[28] G. Strang and G. J. Fix, *An Analysis of the Finite Element Method*, first ed., Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1973.

Paper 1

Using an adaptive FEM to determine the optimal control of a vehicle during a collision avoidance manoeuvre

Karin Kraft and Stig Larsson
Mathematical Sciences, Chalmers University of Technology and Göteborg University
karin.kraft@chalmers.se
stig@chalmers.se
Mathias Lidberg
Applied Mechanics, Chalmers University of Technology
mathias.lidberg@chalmers.se

Abstract

The optimal manoeuvering of a vehicle during a collision avoidance manoeuvre is investigated. A simple model where the vehicle is modelled as point mass and the mathematical formulation of the optimal manoeuvre are presented. The resulting two-point boundary problem is solved by an adaptive finite element method and the theory behind this method is described.

Keywords: Vehicle dynamics, collision avoidance manoeuvre, optimal control, boundary value problem, adaptive finite element method.

1 Introduction

Historically, active safety systems for vehicles are designed to ensure that the driver can steer and brake the vehicle. Automatic controls are being incorporated in conventional safety systems such as ESC with the ability to minimise driver errors. It is important to evaluate how such systems perform in various situations. For this purpose the American National Highway Traffic Safety Administration has proposed a test called "sine with dwell" to evaluate the performance of a car during a collision avoidance manoeuvre. In such a manoeuvre the driver of a vehicle tries to avoid an object that suddenly appears in front of the vehicle [1].

This article is a theoretical investigation of how to combine braking and steering to perform a collision avoidance manoeuvre in an optimal way. The optimisation function has two goals. The primary objective is to achieve a vehicle trajectory distance to avoid collision. If the primary objective cannot be met, then a secondary objective is to minimise the final velocity of the unavoidable collision. This is an optimal control problem, since we want to find controls and states which minimise a quantity subject to constraints consisting of a dynamical system.

Methods for solving optimal control problem can be classified as either a *direct* or an *indirect* approach [4]. The direct approach approximates the dynamical system and then looks for a solution, such that the objective function is minimised. The indirect approach determines the necessary conditions for optimality, and then seeks their solution. Taking the indirect approach means that we have to derive the adjoint equations and optimality conditions explicitly. However, we use this approach because the indirect approach in combination with the finite element method gives us the possibility to control the error in the numerical solution of the optimality conditions over the entire interval. We believe that this is important for an efficient solver. Our first attempts in this direction are described in the present work. The most common numerical methods for solving optimal control problems based on either a direct or an indirect approach are multiple shooting or collocation methods [4]. However, in this work we use an adaptive finite element method similar to the one in [8] to solve the necessary optimality conditions that arise in an indirect approach. In the presented adaptive finite element method we derive an a posteriori error estimate which is used as a basis for error control and adaptive mesh refinement. Since we estimate the error over the entire time interval we can use the computational power where it is best needed. This gives us the ability to choose the level of modelling for the FEM solver and we also believe that we will be able to solve optimal control problems for more advanced vehicle models.

2 The collision avoidance manoeuvre

A traffic situation that presents a safety risk is defined for the investigation. A vehicle is driven on a plane homogeneous surface. There is an obstacle in front of the vehicle. How shall the driver manoeuver the vehicle in the best way in order to avoid collision and, if that is not possible, minimise the collision severity? In Figure 1 we show a picture of the steering in a collision avoidance manoeuvre. The driver performs the avoidance manoeuvre by braking and steering simultaneously. The manoeuver starts at time $t = 0$, at a distance a from the obstacle and with velocity U_0. After the manoeuvre the car hits or passes the obstacle at time T, with velocity U_T and at distance b from the original track.

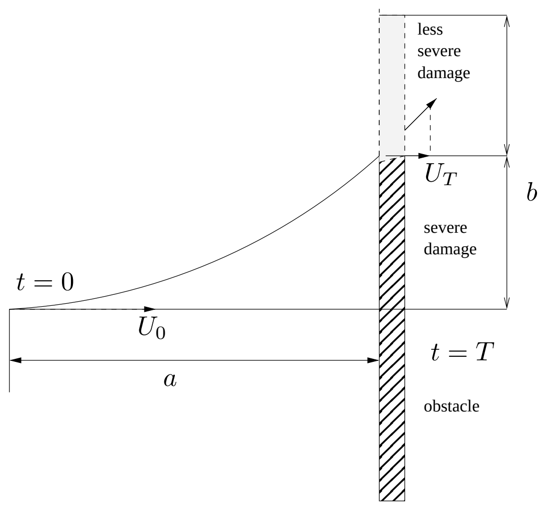

Figure 1: The collision avoidance manoeuvre

We know that the higher speed the vehicle has at the time of collision, the more severe the accident. Therefore we want to determine the best braking and steering strategy to avoid collision or minimise the speed perpendicular to the object at impact. This optimisation problem can be formulated as follows: given the manoeuvre distances a and b determine the braking and steering strategy that minimises the final velocity component U_T.

3 Point mass vehicle dynamics model

The driver controls the braking and steering but it is the friction forces acting on the car tyres that makes the vehicle move in a certain direction. For our purposes, the dynamics of the vehicle due to these forces can be modelled as a point mass [10]. We introduce the X-axis as the direction of the original track and the Y-axis as the axis perpendicular to the X-axis. The equations of planar motion for the vehicle then become

$$\begin{aligned} \ddot{X} &= -\mu g \cos(\beta), \\ \ddot{Y} &= \mu g \sin(\beta), \end{aligned} \tag{1}$$

where β is the angle between the X-axis and the sum of the forces between the tyres and the road, μ is the friction coefficient and g is the gravitational acceleration.

4 Optimal control theory for the collision avoidance manoeuvre

4.1 State-space formulation

To derive the necessary conditions of optimality, the final speed optimal control problem is formulated in state space by transforming differential equations (1) to first order differential equations. The equations of planar motion for the vehicle then become

$$\dot{z} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{U} \\ \dot{V} \end{bmatrix} = \begin{bmatrix} U \\ V \\ -\mu g \cos(\beta) \\ \mu g \sin(\beta) \end{bmatrix}, \tag{2}$$

where U and V are the velocities in the X and Y directions, respectively.

We want to minimise the speed at the time of the accident in order to reduce the damage. Therefore we formulate an optimal control problem: Find the state $z(t) \in \mathbb{R}^n$ and control $\beta(t) \in \mathbb{R}^m$ which fulfill the following minimisation problem

$$\begin{aligned} \min \quad & \mathcal{J}(z, \beta) = \bar{c}^{\mathrm{T}} z(T) \\ \text{s.t.} \quad & \dot{z}(t) = f(z, \beta), \\ & J_0 z(0) = z_0, \quad J_T z(T) = z_T. \end{aligned} \tag{3}$$

Here J_0 and J_T are diagonal matrices with zeroes or ones on the diagonals and f is given by the right hand side of (1) and $\bar{c}^{\mathrm{T}} = (0, 0, 1, 0)$.

Since this problem has a free terminal time we transform the time interval $t \in [0, T]$ into a normalised time interval $\tau \in [0, 1]$ by introducing the new independent variable

$$\tau = \frac{t}{T}, \tag{4}$$

rewrite the equations in (3) for the new variable τ and add the trivial equation $\dot{T} = 0$. This results in a problem of the form (3) but with a fixed time interval.

4.2 Necessary conditions for optimality

Introducing the Hamiltonian,

$$H = \lambda^{\mathrm{T}} f(z, \beta),$$

and then applying variational calculus [6] to (3) leads to the following necessary conditions for optimality.

The optimal solution $(z^*(t), \lambda^*(t), \beta^*(t))$ fulfills the optimality conditions

$$\dot{z} = \frac{\partial H}{\partial \lambda} = f(z), \tag{5}$$

$$\dot{\lambda} = -\frac{\partial H}{\partial z} = -\left(\frac{\partial f}{\partial z}\right)^{\mathrm{T}} \lambda, \tag{6}$$

$$0 = \frac{\partial H}{\partial \beta} = \left(\frac{\partial f}{\partial \beta}\right)^{\mathrm{T}} \lambda, \tag{7}$$

the boundary conditions

$$J_0 z(0) = z_0, \quad J_T z(T) = z_T, \tag{8}$$

and the transversality conditions

$$(J - J_0)\lambda(0) = \lambda_0, \quad (J - J_T)\lambda(T) = \lambda_T, \tag{9}$$

where λ_0 and λ_T are obtained from $\mathcal{J}$. We note here that $x_0 \in R(J_0)$ and $x_T \in R(J_T)$ which means that the components of the adjoint variable λ that have boundary values are the ones complementary to the components of x that have boundary values. To simplify the problem we assume that the optimality condition (7) can be solved explicitly for β^*.

4.3 Reformulating the boundary value problem into standard form

General purpose software for treating boundary value problems for ordinary differential equations usually requires the problem to be reformulated into standard form [3]. We make this conversion by joining the states z and the costates λ into a new variable $x \in \mathbb{R}^d$ for $d = 2n$, and then redefining f by merging the right hand sides of (5) and (6). The resulting system is a two point boundary value problem with fixed time interval and separated linear boundary conditions,

$$\begin{aligned} &\dot{x} = f(x), \\ &I_0 x(0) = x_0, \;\; I_T x(1) = x_T, \end{aligned} \tag{10}$$

where $\dot{x}$ denotes the derivative of x with respect to the new independent variable τ.

5 An adaptive finite element method

5.1 Weak formulation

In this section we derive an adaptive finite element method. It consists of the discretisation of the problem with definitions of the right function spaces and an a posteriori error estimate. We start with the so called weak formulation. To obtain the weak formulation we multiply (10) by a test function $v \in V = C^1([0,T])$, integrate over the interval $[0,T]$ and the weak formulation of the problem is: Seek $x \in V$ such that

$$\begin{aligned} &I_0 x(0) = x_0, \;\; I_T x(T) = x_T, \\ &F(x,v) = \int_0^T (\dot{x} - f(x), v)\, dt = 0, \;\; \forall v \in V, \end{aligned} \tag{11}$$

where $(\cdot,\cdot)$ is the Cartesian scalar product in $\mathbb{R}^d$.

5.2 Discretisation of the problem

The problem in (11) is an infinite dimensional problem which we discretise as follows to get a finite problem. We discretise the time axis and introduce the trial and test spaces as follows.

- Mesh: $0 = t_0 < t_1 < t_2 < \ldots < t_N = T$, $h_n = t_n - t_{n-1}$ and $I_n = (t_{n-1}, t_n)$.
- Trial space: $W_h = \mathbb{R}^d \times \{w : w|_{I_n} \in P^0(I_n)\} \times \mathbb{R}^d$, discontinuous piecewise constant functions.
- Test space: $V_h = \left\{v : v|_{I_n} \in P^1(I_n) \cap C^0([0,T])\right\}$, continuous piecewise linear functions.

The notation $P^k(I_n)$ refers to the $\mathbb{R}^d$-valued polynomials of degree k on the interval I_n. We also introduce the left and right limits $w_n^{\pm} = \lim_{t \to t_n^{\pm}} w(t)$, and jumps $[w]_n = w_n^+ - w_n^-$. The two factors $\mathbb{R}^d$ in W_h contain the boundary values w_0^- and w_N^+. Now our finite element problem can be stated: Find a function $X \in W_h$ which fulfills

$$\begin{aligned} &I_0 X_0^- = x_0, \;\; I_T X_N^+ = x_T, \\ &F(X,v) = \sum_{n=1}^{N} \int_{I_n} (\dot{X} - f(X), v)\, dt \\ &\qquad + \sum_{n=0}^{N} ([X]_n, v_n) = 0, \;\; \forall v \in V_h. \end{aligned} \tag{12}$$

Here the definition of the form F from (11) has been extended to include the contributions from the jump terms which appear since we use discontinuous trial functions. Since the trial space consists of piecewise constant functions, $\dot{X} = 0$. Hence, (12) results in a system of $(N+2)d$ equations that have to be solved, more precisely, d boundary conditions and $(N+1)d$ equations. With boundary conditions at both ends, the equations are coupled and thus we cannot use time stepping and therefore the equations in the system have to be solved simultaneously.

5.3 An a posteriori error estimate

An adaptive finite element method gives us the possibility to control the error in the numerical solution. In order to derive an a posteriori error estimate we introduce ϕ as the solution to the adjoint problem to (10) with data functional G. We want to construct an equation for the error, $e = X - x$ where $e \in W = \mathbb{R}^d \times \{w|_{I_n} : w \in C^1(I_n)\} \times \mathbb{R}^d$, the difference between the real and the computed solution. The details of the a posteriori error estimate are given below.

5.3.1 Proof of the error estimate

We subtract (11) from (12),

$$\begin{aligned} F(X,v) - &\underbrace{F(x,v)}_{=0,\forall v\in V} \\ = &\sum_{n=1}^{N}\int_{I_n} (\dot{X} - \dot{x} - (f(X) - f(x)), v)\,dt \\ &+ \sum_{n=0}^{N}([X-x]_n, v_n). \end{aligned} \tag{13}$$

Since f is nonlinear we linearise $f(X)-f(x)$ by rewriting it as follows

$$\begin{aligned} f(X) - f(x) &= \int_0^1 \frac{d}{d\theta} f(\theta(X(t)-x(t)) + x(t))\,d\theta \\ &= \underbrace{\int_0^1 Df(\theta(X(t)-x(t)) + x(t))\,d\theta}_{=A(t)}(X(t)-x(t)). \end{aligned}$$

Inserting this in (13) we get

$$\begin{aligned} F(X,v) &= \sum_{n=1}^{N}\int_{I_n} (\dot{X} - \dot{x} - (f(X) - f(x)), v)\,dt \\ &\quad + \sum_{n=0}^{N}([X-x]_n, v_n) \\ &= \sum_{n=1}^{N}\int_{I_n} (\dot{e} - A(t)e, v)\,dt \\ &\quad + \sum_{n=0}^{N}([e]_n, v_n), \quad \forall v \in V. \end{aligned} \tag{14}$$

Since (14) is linear in both e and v we introduce a bilinear form to simplify the notation. The bilinear form B is defined as

$$\begin{aligned} B(w,v) = &\sum_{n=1}^{N}\int_{I_n} (\dot{w} - A(t)w, v)\,dt + \sum_{n=0}^{N}([w]_n, v_n) \\ &+ (I_0 w_0^-, v_0) - (I_T w_N^+, v_N), \quad w \in W, v \in V, \end{aligned} \tag{15}$$

Now we can write the equation for the error (14) with the bilinear form as

$$\begin{aligned} &e \in W \\ &B(e,v) = F(X,v), \quad \forall v \in V. \end{aligned} \tag{16}$$

Partial integration of (15) gives us the backward form of the bilinear form

$$\begin{aligned} B(w,v) &= \sum_{n=1}^{N}\int_{I_n} (\dot{w} - A(t)w, v)\,dt \\ &\quad + \sum_{n=0}^{N}([w]_n, v_n) \\ &\quad + (I_0 w_0^-, v_0) - (I_T w_N^+, v_N) \\ &= \sum_{n=1}^{N}\int_{I_n} (w, -\dot{v} - A(t)^{\mathrm{T}} v)\,dt \\ &\quad - (w_0^-, (I - I_0)v_0) + (w_N^+, (I - I_T)v_N), \\ &\quad w \in W, v \in V. \end{aligned} \tag{17}$$

This suggests the dual problem with arbitrary data functional G

$$\begin{aligned} &\phi \in V \\ &B(w,\phi) = G(w), \quad \forall w \in W. \end{aligned} \tag{18}$$

We put $v = \phi$ in (16) and $w = e$ in (18) to obtain

$$G(e) = B(e,\phi) = F(X,\phi), \tag{19}$$

that is

$$\begin{aligned} G(e) &= B(e,\phi) \\ &= F(X,\phi) = \sum_{n=1}^{N}\int_{I_n} (\dot{X} - f(X), \phi)\,dt \\ &\quad + \sum_{n=0}^{N}([X]_n, \phi_n). \end{aligned} \tag{20}$$

Subtracting a Lagrange node interpolant $\tilde{\phi} \in V_h$ from ϕ in the right hand side of (20) using (12) gives us

$$G(e) = \sum_{n=1}^{N}\int_{I_n} (\dot{X} - f(X), \phi - \tilde{\phi})\,dt + \sum_{n=0}^{N}([X]_n, \phi_n - \tilde{\phi}_n).$$

Hence,

$$
\begin{aligned}
|G(e)| &\leq \Big| \sum_{n=1}^{N} \int_{I_n} (\dot{X} - f(X), \phi - \tilde{\phi})\, dt \Big| \\
&\quad + \Big| \sum_{n=0}^{N} ([X]_n, \phi_n - \tilde{\phi}_n) \Big| \\
&\leq \underbrace{\sum_{n=1}^{N} \int_{I_n} \|\dot{X} - f(X)\| \|\phi - \tilde{\phi}\|\, dt}_{I} \\
&\quad + \underbrace{\sum_{n=0}^{N} \|\, [X]_n \,\| \|\phi_n - \tilde{\phi}_n\|}_{II}.
\end{aligned}
\tag{21}
$$

Now we have the basis for an error estimate, but we want the method to be symmetric, meaning that we want each interior node to contribute to the error estimate on both sides of the node. To do this we rewrite the last term II in (21) as follows

$$
\begin{aligned}
\sum_{n=0}^{N} \|\, [X]_n \,\| \|\phi_n - \tilde{\phi}_n\| &= \|\, [X]_0 \,\| \|\phi_0 - \tilde{\phi}_0\| \\
&\quad + \frac{h_1}{h_1 + h_2} \|\, [X]_1 \,\| \|\phi_1 - \tilde{\phi}_1\| \\
&\quad + \sum_{n=2}^{N-1} \Big(\frac{h_n}{h_n + h_{n+1}} \|\, [X]_n \,\| \|\phi_n - \tilde{\phi}_n\| \\
&\qquad + \frac{h_n}{h_n + h_{n-1}} \|\, [X]_{n-1} \,\| \|\phi_{n-1} - \tilde{\phi}_{n-1}\| \Big) \\
&\quad + \frac{h_N}{h_{N-1} + h_N} \|\, [X]_{N-1} \,\| \|\phi_{N-1} - \tilde{\phi}_{N-1}\| \\
&\quad + \|\, [X]_N \,\| \|\phi_N - \tilde{\phi}_N\|.
\end{aligned}
\tag{22}
$$

At this stage we introduce the notation $\|v\|_{I_n} = \max_{I_n} \|v\|$ for the maximum norm of a function on an interval. Now we note that $\phi - \tilde{\phi} \in V$ is continuous and the following estimates $\|\phi_n - \tilde{\phi}_n\| \leq \|\phi - \tilde{\phi}\|_{I_n}$ and $\|\phi_n - \tilde{\phi}_n\| \leq \|\phi - \tilde{\phi}\|_{I_{n+1}}$ hold. Using this we can estimate the last term in (22) by

$$
\begin{aligned}
&\sum_{n=0}^{N} \|\, [X]_n \,\| \|\phi_n - \tilde{\phi}_n\| \\
&\leq \left(\|\, [X]_0 \,\| + \frac{h_1}{h_1 + h_2} \|\, [X]_1 \,\| \right) \|\phi - \tilde{\phi}\|_{I_1} \\
&\quad + \sum_{n=2}^{N-1} \Big(\frac{h_n}{h_n + h_{n+1}} \|\, [X]_n \,\| \\
&\qquad + \frac{h_n}{h_n + h_{n-1}} \|\, [X]_{n-1} \,\| \Big) \|\phi - \tilde{\phi}\|_{I_n} \\
&\quad + \left(\frac{h_N}{h_{N-1} + h_N} \|\, [X]_{N-1} \,\| + \|\, [X]_N \,\| \right) \|\phi - \tilde{\phi}\|_{I_N}.
\end{aligned}
\tag{23}
$$

The term I in (21) (where $\dot{X} = 0$) can be estimated as follows

$$
\begin{aligned}
&\sum_{n=1}^{N} \int_{I_n} \|\dot{X} - f(X)\| \|\phi - \tilde{\phi}\|\, dt \\
&\leq \sum_{n=1}^{N} h_n \|\dot{X} - f(X_n^-)\|_{I_n} \|\phi - \tilde{\phi}\|_{I_n}.
\end{aligned}
\tag{24}
$$

Collecting the estimates (24) and (23) we now have

$$
\begin{aligned}
|G(e)| &\leq \sum_{n=1}^{N} h_n \|\dot{X} - f(X)\|_{I_n} \|\phi - \tilde{\phi}\|_{I_n} \\
&\quad + \left(\|\, [X]_0 \,\| + \frac{h_1}{h_1 + h_2} \|\, [X]_1 \,\| \right) \|\phi - \tilde{\phi}\|_{I_1} \\
&\quad + \sum_{n=2}^{N-1} \Big(\frac{h_n}{h_n + h_{n+1}} \|\, [X]_n \,\| \\
&\qquad + \frac{h_n}{h_n + h_{n-1}} \|\, [X]_{n-1} \,\| \Big) \|\phi - \tilde{\phi}\|_{I_n} \\
&\quad + \Big(\frac{h_N}{h_{N-1} + h_N} \|\, [X]_{N-1} \,\| \\
&\qquad + \|\, [X]_N \,\| \Big) \|\phi - \tilde{\phi}\|_{I_N}.
\end{aligned}
$$

According to [5] we have the following error bound for the interpolan t, $\tilde{\phi}$,

$$
\left\| \phi - \tilde{\phi} \right\|_{I_n} \leq C h_n \int_{I_n} \left\| \ddot{\phi} \right\| dt.
$$

With

$$
\begin{aligned}
\mathcal{R}_1 &= h_1 \|\dot{X} - f(X)\|_{I_1} \\
&\quad + h_1 \|\, [X]_0 \,\| + \frac{h_1}{h_1 + h_2} \|\, [X]_1 \,\|, \\
\mathcal{R}_n &= h_n \|\dot{X} - f(X)\|_{I_n} \\
&\quad + \frac{h_n}{h_n + h_{n+1}} \|\, [X]_n \,\| \\
&\quad + \frac{h_n}{h_n + h_{n-1}} \|\, [X]_{n-1} \,\|, \quad n = 2, \ldots, N-1, \\
\mathcal{R}_N &= h_N \|\dot{X} - f(X)\|_{I_N} + \\
&\quad \frac{h_N}{h_{N-1} + h_N} \|\, [X]_{N-1} \,\| + \|\, [X]_N \,\|,
\end{aligned}
$$

and

$$
\mathcal{I}_n = C h_n \int_{I_n} \left\| \ddot{\phi} \right\| dt,
$$

we can write the error estimate as

$$
|G(e)| \leq \sum_{n=1}^{N} \mathcal{R}_n \mathcal{I}_n,
\tag{25}
$$

where $e = X - x$ is the error and $\mathcal{R}_n$ is essentially the residual, $\dot{X} - f(X)$, expressing how well the differential equation is satisfied by the numerical solution. The weights $\mathcal{I}_n$ depend on the solution to the adjoint problem, ϕ, and express the sensitivity of the error quantity $G(e)$ to the local residuals. The functional G is chosen to be the quantity in which we want to measure the error, for example, $G(e) = \frac{e}{\|e\|}$. The error resulting from approximate nonlinear equation solver is small compared to the error resulting from the discretisation and is therefore neglected in this estimate. Checking which intervals give large contributions to the error estimate (25), we can refine the intervals where the contributions are large and vice versa. Using (25) we obtain an adaptive procedure where we refine those intervals that give large contributions to the estimate and vice versa, see below.

5.4 Implementation

The finite element discretisation of (10) results in the system (12) to be solved. Since we have boundary conditions at both ends it is a coupled system of equations that we have to solve simultaneously. The system is also nonlinear and the nonlinearity is handled using a damped Newton method [7] to extend the convergence region. The initial guess is decided by a homotopy process [3]. Once a solution is calculated the error estimate above is computed. Then we apply the criterion

$$\sum_{n=1} \mathcal{R}_n \mathcal{I}_n \leq \delta,$$

where δ is a given tolerance. If the error is too large compared to the tolerance an iteration is made over the intervals and the mesh is refined where the error is large. New nodes are inserted according to the *principle of equidistribution*, that is, we want to insert nodes such that the contribution to the error is the same from each time interval. A new solution is calculated on the refined mesh and so on until the solution has reached the desired accuracy. In theory the mesh can also be coarsened but we have not implemented this.

The error estimate is dependent of the solution to the dual problem. We need to approximate the unknown data $G(e) = \frac{e}{\|e\|}$ to solve the dual problem numerically. We do this by a Richardson extrapolation using twice the number of nodes. Then the dual problem is solved with the finite element method.

The solver is a prototype solver and it has been implemented in Matlab. More information regarding the implementation can be found in [2].

6 Results

The indirect approach to our optimal control problem results in a boundary value problem. This makes it possible to compare our new approach to the boundary value solver `bvp4c` in Matlab [11]. In Table 1 we can see the results from various choices of the manoeuvre distances a and b. We have used the same initial velocity $u_0 = 90$ km/h and constant friction μ for all cases. We can see that the FEM code is almost always about three times slower than `bvp4c` but it always uses fewer nodes. There is also a remarkable case where `bvp4c` solves the problem in about 30 seconds and with 3415 nodes compared to 1.3 seconds and 21 nodes for the finite element solver. The problem becomes difficult to solve but our FEM solver performs well, maybe due to the adaptivity. There is also one problem that the finite element solver can solve but `bvp4c` cannot.

In some cases where `bvp4c` finds a solution the FEM solver seems to compute the wrong one, maybe by missing a singularity. On the other side of the singularity it continues on another solution. Some results about existence and uniqueness of solutions to boundary value problems can be found in [7] and [9]. This aspect of our solver is something that we have to investigate further.

Figure 2 and 3 show some results from the case with initial velocity $u_0 = 90$ km/h (25 m/s) and the manoeuvre distances $a = 50$ m and $b = 8$ m. We see in the figures that the solutions from the FEM solver and `bvp4c` coincide. The final velocity is 53.32 km/h from `bvp4c` and 53.37 km/h from the FEM solver.

a [m]	b [m]	FEM CPU [s]	FEM nodes	`bvp4c` CPU [s]	`bvp4c` nodes
40	6	2.54	15	0.27	25
50	9	1.59	90	0.36	41
50	8	0.61	10	0.24	28
50	5	0.52	10	0.27	37
50	3	1.45	10	-	-
60	8	1.45	21	18.39	1287
60	6	1.46	10	0.63	81
60	5	3.45	10	3.06	286

Table 1: Performance of the FEM solver and `bvp4c` measured in CPU time and number of nodes for different combinations of manoeuvre distances.

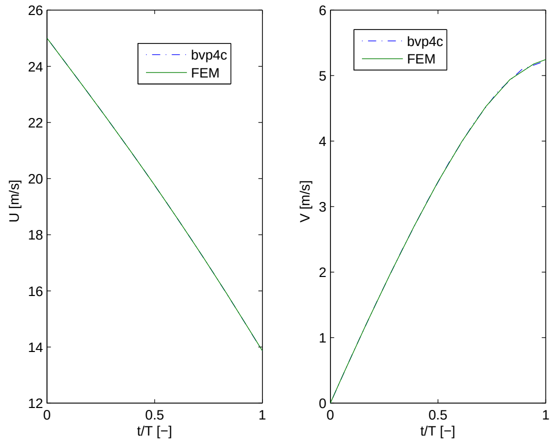

Figure 2: The optimal velocities in the X and Y-directions for the manoeuvre distances $a = 50$ m and $b = 8$ m.

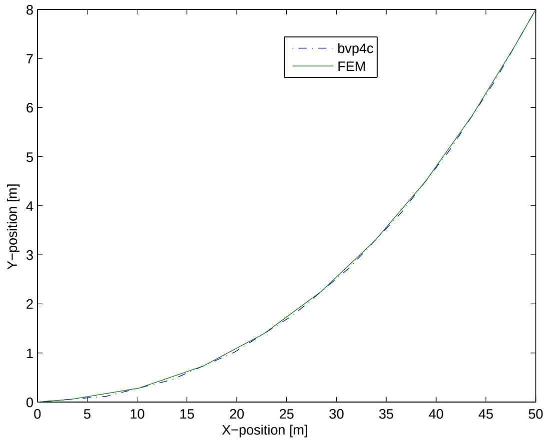

Figure 3: The position of the vehicle when it is manoeuvered in the optimal way for the manoeuvre distances $a = 50$ m and $b = 8$ m.

7 Conclusion

In this article we have presented an adaptive finite element method for solving optimal control in vehicle dynamics. Modelling the vehicle as a point mass, we obtain a system of ordinary differential equations which is solved, together with the constraints imposed by the manoeuvre, using the adaptive finite element method. With this approach, we can control the error and concentrate our resources to the most sensitive parts of the computations.

We have compared the finite element method to the Matlab solver `bvp4c` and found that there are at least some cases where our solver outperforms `bvp4c`. It is noteworthy that in all studied cases, our method uses fewer nodes to find the same solution. At the moment the finite element solver is slower than `bvp4c`, but up to this point no extra effort has been put in optimising the code. Thus, it is expected that the computation time can be reduced by a more efficient implementation. Further, these comparisions are very preliminary, since the accuracies of both methods depend on error tolerances that are not directly comparable. We are not sure that the settings are equal. Still, since the quality of the solutions have been similar throughout our computations, we feel confident that our comparision is reasonable.

We have also noted that our solver is sensitive to the initial guess. If we give the solver a poor initial guess for some component, it may fail to solve the problem or show a dramatical increase in computation time. To make the solver more useful we have to make it more robust to poor initial guesses.

The model used in this article may look simple. However, when it comes to evaluation of combined steering and braking versus braking and then steering, the behaviour of this model gives insights into the behaviour of the more realistic vehicle models and manoeuvres we will consider. In our future work we also intend to compare the performance of our indirect approach to the direct approach.

References

[1] National Highway Traffic Safety Administration, *Laboratory Test Instruction for FMVSS 126 Stability Control Systems*, `http://www.nhtsa.dot.gov`.

[2] C. Andersson, *Solving optimal control problems using FEM*, Master's thesis, Chalmers University of Technology, 2007.

[3] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, first ed., Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1988.

[4] J. T. Betts, *Survey of numerical methods for trajectory optimization*, AIAA J. Guidance Control Dynam. **21** (1998), 193–207.

[5] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, second ed., Springer-Verlag, New York, 2002.

[6] A. E. Bryson, Jr and Y. Ho, *Applied Optimal Control*, revised printing ed., Hemisphere Publishing Corporation, Washington, D.C, 1975.

[7] P. Deuflhard and F. Bornemann, *Scientific Computing with Ordinary Differential Equations*, Springer-Verlag New York Inc., New York, 2002.

[8] D. J. Estep, D. H. Hodges, and M. Warner, *Computational error estimation and adaptive error control for a finite element solution of launch vehicle trajectory problems*, SIAM J. Sci. Comput. **21** (1999), 1609–1631.

[9] H. B. Keller, *Numerical Methods for Two-Point Boundary-Value Problems*, Dover Publications, Inc., New York, 1992.

[10] B. Schmidtbauer, *Sväng och bromsa samtidigt (in Swedish)*, Teknisk Tidskrift (1971).

[11] L. F. Shampine, J. Kierzenka, and M. W. Reichelt, *Solving Boundary Value Problems for Ordinary Differential Equations in Matlab using* `bvp4c`, Tech. report, MathWorks, 2000, `http://www.mathworks.com`.

Paper 2

THE DUAL WEIGHTED RESIDUALS APPROACH TO OPTIMAL CONTROL OF ORDINARY DIFFERENTIAL EQUATIONS

KARIN KRAFT AND STIG LARSSON

ABSTRACT. The methodology of dual weighted residuals is applied to an optimal control problem for ordinary differential equations. The differential equations are discretised by finite element methods. An *a posteriori* error estimate is derived and an adaptive algorithm is formulated. The algorithm is implemented in Matlab and tested on a simple model problem from vehicle dynamics.

1. INTRODUCTION

The methodology of *dual weighted residuals* was developed in [1] in the context of finite element methods for partial differential equations. In this paper we adapt the methodology to optimal control problems of the form: Find states x and controls u which

$$\begin{aligned} &\text{minimise} && \mathcal{J}(x,u) = l(x(0),x(T)) + \int_0^T L(x,u)\,dt, \\ &\text{subject to} && \dot{x}(t) = f(x(t),u(t)), \quad 0 < t < T, \\ & && I_0 x(0) = x_0, \quad I_T x(T) = x_T. \end{aligned} \tag{1.1}$$

We present an adaptive finite element method with error control based on an *a posteriori* error estimate which is the sum of dual weighted residuals.

Optimal control problems can be solved numerically using two different approaches, the *direct* and the *indirect* [2]. In the direct approach the problem is first discretised and a finite dimensional minimisation problem is solved. In the indirect approach the necessary conditions for optimality are determined and these equations are then solved numerically. Traditionally, the necessary conditions for optimality are derived using variational calculus [3], and their solution can be obtained using different numerical methods such as finite element methods [5] or multiple shooting [2].

In the present work we use the indirect approach. We present the classical variational calculus in a weak form and derive the necessary conditions

Date: January 14, 2008.
1991 *Mathematics Subject Classification.* 65L60, 49K15.
Key words and phrases. finite element, a posteriori, error estimate, adaptive, dual weighted residual, boundary value problem, differential-algebraic, vehicle dynamics.
This work was supported by the Gothenburg Mathematical Modelling Centre (GMMC).

for optimality. These consist of a system of three equations: the linearised adjoint equation for the Lagrange multiplier z, the original state equation for x, and a non-linear algebraic equation for the control variable u. We approximate the equations by a finite element method and derive an *a posteriori* error representation formula and an estimate of the error in the goal functional $\mathcal{J}$. The error estimate is expressed as an element-wise sum of dual weighted residuals,

$$|\mathcal{J}(x,u) - \mathcal{J}(x_h,u_h)| \leq \sum_{n=1}^{N} \Big(R_n^z \omega_n^x + R_n^x \omega_n^z + R_n^u \omega_n^u \Big) + R,$$

where R_n^z, R_n^x, R_n^u are residuals from the adjoint equation, the state equation, and the algebraic equation for the control variable, respectively, and ω_n^x, ω_n^z, ω_n^u are weights computed from the solutions of the respective equations indicated by the superscripts, and R is a remainder which may be neglected.

Previous work, [5], [6], aims at controlling the error in an arbitrary linear functional (or a norm) of the variables and requires the solution of an additional adjoint problem of the same size as the optimality conditions. The main advantage of the dual weighted residual error estimate is that it only uses the equations introduced in the optimality conditions and no extra dual problem has to be solved. However, it can only be used for controlling the error in the goal functional $\mathcal{J}$.

We use the error estimate as the basis for an adaptive finite element method, which is implemented in Matlab and tested on an optimal control problem from vehicle dynamics with quadratic goal functional and linear state equation.

We begin in Section 2 by presenting an abstract framework for the optimal control problem where we can derive the necessary conditions for optimality as well as an *a posteriori* representation formula for the error in the goal functional $\mathcal{J}$. In Section 3 we apply these results to the optimal control problem. In Section 4 we specialise to a quadratic/linear optimal control problem. For this problem, we derive the *a posteriori* error estimate from the error representation formula and we describe the implementation of an adaptive finite element method based on the *a posteriori* error estimate. Finally, we solve an example from vehicle dynamics in Section 5.

2. An abstract framework

Following [1] we formulate the optimal control problem in an abstract way. Let W, U, V be normed vector spaces, let $\dot{W} \subset W$ be a subspace, let $\hat{x} \in W$ be fixed and define the affine space

$$\tilde{W} = \hat{x} + \dot{W} = \Big\{ w \in W : w - \hat{x} \in \dot{W} \Big\}.$$

The reason for using this affine space will be clear in Section 3, where we include boundary conditions in the problem formulation. Further we introduce smooth functionals

$$\mathcal{F} : W \times U \times V \to \mathbb{R},$$
$$\mathcal{J} : W \times U \to \mathbb{R}.$$

We assume that $\mathcal{F}(x, u; z)$ is linear in the third variable, z. We use the notation that the functionals depend non-linearly on the arguments before the semicolon and linearly on the arguments after the semicolon. For example, we denote the derivative $\mathcal{F}'_x(x, u; z)$ acting on a test function φ_x by $\mathcal{F}'_x(x, u; z, \varphi_x) = \mathcal{F}'_x(x, u; z)\varphi_x$.

We consider optimal control problems of the form: Determine $x \in \tilde{W}$ and $u \in U$ which

$$\begin{aligned} &\text{minimise} && \mathcal{J}(x, u), \\ &\text{subject to} && \mathcal{F}(x, u; \varphi) = 0, \quad \forall \varphi \in V. \end{aligned} \tag{2.1}$$

The main difference with [1] is the presence of the control variable u and that we need several spaces in order to allow for a Petrov-Galerkin method and non-homogeneous boundary conditions.

This is a constrained optimisation problem and the necessary condition for an optimum is expressed in terms of the Lagrange functional

$$\mathcal{L}(x, u; z) = \mathcal{J}(x, u) + \mathcal{F}(x, u; z), \quad (x, u, z) \in W \times U \times V.$$

Theorem 2.1. *The necessary condition for an optimum* $(x, u, z) \in \tilde{W} \times U \times V$ *is given by*

$$\mathcal{L}'(x, u; z, \varphi) = 0, \quad \forall \varphi \in \dot{W} \times U \times V, \tag{2.2}$$

that is,

$$\begin{aligned} \mathcal{J}'_x(x, u; \varphi_x) + \mathcal{F}'_x(x, u; z, \varphi_x) &= 0, \qquad \forall \varphi_x \in \dot{W}, \\ \mathcal{J}'_u(x, u; \varphi_u) + \mathcal{F}'_u(x, u; z, \varphi_u) &= 0, \qquad \forall \varphi_u \in U, \\ \mathcal{F}(x, u; \varphi_z) &= 0, \qquad \forall \varphi_z \in V. \end{aligned} \tag{2.3}$$

Proof. We expand $\mathcal{L}'$ in partial derivatives, noting that $\mathcal{L}'_z(x, u; z, \varphi_z) = \mathcal{F}'_z(x, u; z, \varphi_z) = \mathcal{F}(x, u; \varphi_z)$. □

Note that the third equation in (2.3) is the equation in the original problem (2.1) and the first equation in (2.3) is the linearised adjoint equation.

In order to formulate a Petrov-Galerkin approximation of the equations (2.3) we assume that we have subspaces $W_h \subset W$, $\dot{W}_h \subset \dot{W}$, $V_h \subset V$, $U_h \subset U$, and that $\hat{x} \in W_h$, so that

$$\tilde{W}_h = \hat{x} + \dot{W}_h \subset \tilde{W}.$$

The approximation of the necessary condition for optimality now becomes: find $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ such that

$$\mathcal{L}'(x_h, u_h; z_h, \varphi) = 0, \quad \forall \varphi \in \dot{W}_h \times U_h \times V_h, \tag{2.4}$$

that is,

$$
\begin{aligned}
\mathcal{J}'_x(x_h, u_h; \varphi_x) + \mathcal{F}'_x(x_h, u_h; z_h, \varphi_x) &= 0, \qquad \forall \varphi_x \in \dot{W}_h, \\
\mathcal{J}'_u(x_h, u_h; \varphi_u) + \mathcal{F}'_u(x_h, u_h; z_h, \varphi_u) &= 0, \qquad \forall \varphi_u \in U_h, \\
\mathcal{F}(x_h, u_h; \varphi_z) &= 0, \qquad \forall \varphi_z \in V_h.
\end{aligned} \tag{2.5}
$$

The following theorem provides an *a posteriori* representation formula for the error in the functional $\mathcal{J}$.

Theorem 2.2. *Let* $(x, u, z) \in \tilde{W} \times U \times V$ *and* $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ *be solutions of* (2.3) *and* (2.5), *respectively. Then*

$$
\mathcal{J}(x, u) - \mathcal{J}(x_h, u_h) = \tfrac{1}{2}\rho_x + \tfrac{1}{2}\rho_z + \tfrac{1}{2}\rho_u + R,
$$

with the residuals ρ_x, ρ_z, *and* ρ_u *defined as*

$$
\begin{aligned}
\rho_x &= \mathcal{J}'_x(x_h, u_h; x - \tilde{x}_h) + \mathcal{F}'_x(x_h, u_h; z_h, x - \tilde{x}_h), \\
\rho_u &= \mathcal{J}'_u(x_h, u_h; u - \tilde{u}_h) + \mathcal{F}'_u(x_h, u_h; z_h, u - \tilde{u}_h), \\
\rho_z &= \mathcal{F}(x_h, u_h; z - \tilde{z}_h).
\end{aligned}
$$

Here $(\tilde{x}_h, \tilde{u}_h, \tilde{z}_h) \in \tilde{W}_h \times U_h \times V_h$ *is arbitrary. The remainder term* R *is given by*

$$
\begin{aligned}
R = \tfrac{1}{2} \int_0^1 \Big(& J'''(x_h + s e_x, u_h + s e_u; e, e, e) \\
& + \mathcal{F}'''(x_h + s e_x, u_h + s e_u; z_h + s e_z, e, e, e) \Big) s(s-1)\, ds,
\end{aligned} \tag{2.6}
$$

where $e = (e_x, e_u, e_z) \in \dot{W} \times U \times V$, $e_x = x - x_h$, $e_u = u - u_h$, *and* $e_z = z - z_h$.

The remainder term is cubic in the error and can therefore often be neglected. In particular, we note that $R = 0$ in the important special case when $\mathcal{F}(\cdot, \cdot; \cdot)$ is tri-linear and $\mathcal{J}(\cdot, \cdot)$ is bi-quadratic.

Proof. We introduce the notation

$$
\begin{aligned}
\overline{\mathcal{L}}'(x, x_h, u, u_h; z, z_h, e) &= \mathcal{L}(x, u; z) - \mathcal{L}(x_h, u_h; z_h) \\
&= \int_0^1 \frac{d}{ds} \mathcal{L}(x_h + s e_x, u_h + s e_u; z_h + s e_z)\, ds \\
&= \int_0^1 \mathcal{L}'(x_h + s e_x, u_h + s e_u; z_h + s e_z, e)\, ds,
\end{aligned}
$$

where $e = (e_x, e_u, e_z) \in \dot{W} \times U \times V$. Using the third equation in (2.3) and the third equation in (2.5) we get

$$
\begin{aligned}
\mathcal{J}(x, u) - \mathcal{J}(x_h, u_h) &= \mathcal{L}(x, u; z) - \mathcal{F}(x, u; z) - \mathcal{L}(x_h, u_h; z_h) + \mathcal{F}(x_h, u_h; z_h) \\
&= \mathcal{L}(x, u; z) - \mathcal{L}(x_h, u_h; z_h) \\
&= \overline{\mathcal{L}}'(x, x_h, u, u_h; z, z_h, e) + \tfrac{1}{2}\mathcal{L}'(x_h, u_h; z_h, e) \\
&\quad - \tfrac{1}{2}\mathcal{L}'(x_h, u_h; z_h, e) - \tfrac{1}{2}\mathcal{L}'(x, u; z, e),
\end{aligned}
$$

where the last term is zero in view of (2.2). The last two terms are equal to an approximation of the first term by the trapezoidal rule. Hence, with R denoting the remainder in this approximation,

$$\begin{aligned}\mathcal{J}(x,u)-\mathcal{J}(x_h,u_h) &= \tfrac{1}{2}\mathcal{L}'(x_h,u_h;z_h,e)+R\\ &= \tfrac{1}{2}\mathcal{L}'(x_h,u_h;z_h,x-x_h,u-u_h,z-z_h)+R\\ &= \tfrac{1}{2}\mathcal{L}'(x_h,u_h;z_h,x-\tilde{x}_h,u-\tilde{u}_h,z-\tilde{z}_h)+R.\end{aligned}$$

Here we used the orthogonality property (2.4) to replace (x_h,u_h,z_h) by an arbitrary $(\tilde{x}_h,\tilde{u}_h,\tilde{x}_h)\in\tilde{W}_h\times U_h\times V_h$. By expanding $\mathcal{L}'$ in terms of partial derivatives we then obtain

$$\begin{aligned}\mathcal{J}(x,u)-\mathcal{J}(x_h,u_h) &= \tfrac{1}{2}\Big(\mathcal{J}'_x(x_h,u_h;x-\tilde{x}_h)+\mathcal{F}'_x(x_h,u_h;z_h,x-\tilde{x}_h)\Big)\\ &\quad+\tfrac{1}{2}\Big(\mathcal{J}'_u(x_h,u_h;u-\tilde{u}_h)+\mathcal{F}'_u(x_h,u_h;z_h,u-\tilde{u}_h)\Big)\\ &\quad+\tfrac{1}{2}\mathcal{F}(x_h,u_h;z-\tilde{z}_h)+R\\ &= \tfrac{1}{2}\rho_x+\tfrac{1}{2}\rho_u+\tfrac{1}{2}\rho_z+R.\end{aligned}$$

The remainder term R is

$$\begin{aligned}R &= \overline{\mathcal{L}}'(x,x_h,u,u_h;z,z_h,e)-\tfrac{1}{2}\mathcal{L}'(x_h,u_h;z_h,e)-\tfrac{1}{2}\mathcal{L}'(x,u;z,e)\\ &= \tfrac{1}{2}\int_0^1\mathcal{L}'''(x_h+se_x,u_h+se_u;z_h+se_z,e,e,e)s(s-1)\,ds\\ &= \tfrac{1}{2}\int_0^1\Big(J'''(x_h+se_x,u_h+se_u;e,e,e)\\ &\quad+\mathcal{F}'''(x_h+se_x,u_h+se_u;z_h+se_z,e,e,e)\Big)s(s-1)\,ds.\end{aligned}\tag{2.7}$$

□

3. An optimal control problem

We consider optimal control problems of the form

$$\begin{aligned}&\text{minimise} && l(x(0),x(T))+\int_0^T L(x(t),u(t))\,dt,\\ &\text{subject to} && \dot{x}(t)=f(x(t),u(t)),\quad 0<t<T,\\ &&& I_0x(0)=x_0,\quad I_Tx(T)=x_T.\end{aligned}\tag{3.1}$$

Here

$$\begin{aligned}l &: \mathbb{R}^d\times\mathbb{R}^d\to\mathbb{R},\\ L &: \mathbb{R}^d\times\mathbb{R}^m\to\mathbb{R},\\ f &: \mathbb{R}^d\times\mathbb{R}^m\to\mathbb{R}^d,\end{aligned}$$

are smooth functions and I_0 and I_T are diagonal matrices with zeroes or ones on the diagonals, and $x_0\in R(I_0)$, $x_T\in R(I_T)$, where $R(A)$ denotes the range of a matrix A.

In order to put this into the abstract framework of the previous section, we need to introduce function spaces $W, \dot{W}, \tilde{W}, V, U$ and functionals $\mathcal{J}$ and $\mathcal{F}$. The spaces must accommodate both the continuous functions x, z, u and the corresponding finite element functions. It is therefore convenient to begin by defining the finite element spaces.

We define a mesh $0 = t_0 < t_1 < t_2 < \ldots < t_N = T$, with steps $h_n = t_n - t_{n-1}$ and intervals $I_n = (t_{n-1}, t_n)$. Let $q \geq 0$ and let P^q denote the polynomials of degree $\leq q$. We introduce the spaces

$$\begin{aligned} W_h &= \mathbb{R}^d \times \Big\{ w : w|_{I_n} \in P^q(I_n, \mathbb{R}^d),\ n = 1, \ldots, N \Big\} \times \mathbb{R}^d, \\ \dot{W}_h &= R(I - I_0) \times \Big\{ w : w|_{I_n} \in P^q(I_n, \mathbb{R}^d),\ n = 1, \ldots, N \Big\} \times R(I - I_T) \\ &= \Big\{ w \in W_h : I_0 w_0^- = 0,\ I_T w_N^+ = 0 \Big\}, \end{aligned}$$

of (vector-valued) discontinuous piecewise polynomial functions of degree $\leq q$ and the space

$$V_h = \Big\{ v \in C([0,T], \mathbb{R}^d) : v|_{I_n} \in P^{q+1}(I_n, \mathbb{R}^d) \Big\},$$

of continuous piecewise polynomial functions of degree $\leq q+1$. For $w \in W_h$ we use the notation $[w]_n = w_n^+ - w_n^-$, $w_n^\pm = \lim_{t \to t_n^\pm} w(t)$ for the jump and the one-sided limits at t_n, and for $v \in V_h$ we write $v_n = v(t_n)$. The two factors $\mathbb{R}^d$ in W_h contain the boundary values w_0^- and w_N^+. We also select $\hat{x} \in W_h$ such that

$$I_0 \hat{x}_0^- = x_0, \quad I_T \hat{x}_N^+ = x_T,$$

where x_0, x_T are the boundary values in (3.1), and define the affine space

$$\begin{aligned} \tilde{W}_h = \hat{x} + \dot{W}_h &= \Big\{ w \in W_h : w - \hat{x} \in \dot{W}_h \Big\} \\ &= \Big\{ w \in W_h : I_0 w_0^- = x_0,\ I_T w_N^+ = x_T \Big\}. \end{aligned}$$

Finally, we define

$$U_h = \Big\{ v \in C([0,T], \mathbb{R}^m) : v|_{I_n} \in P^{q+1}(I_n, \mathbb{R}^m) \Big\}.$$

Note that

$$\begin{aligned} \dim(W_h) &= (N(q+1)+2)d, \\ \dim(\dot{W}_h) &= (N(q+1)+2)d - d_0 - d_T, \\ \dim(V_h) &= (N(q+1)+1)d, \\ \dim(U_h) &= (N(q+1)+1)m, \end{aligned} \tag{3.2}$$

where $d_0 = \operatorname{rank}(I_0)$, $d_T = \operatorname{rank}(I_T)$.

We now define the function spaces

$$\begin{aligned}
W &= \mathbb{R}^d \times \Big\{ w : w|_{I_n} \in H^1(I_n, \mathbb{R}^d),\ n = 1, \dots, N \Big\} \times \mathbb{R}^d, \\
\dot{W} &= R(I - I_0) \times \Big\{ w : w|_{I_n} \in H^1(I_n, \mathbb{R}^d),\ n = 1, \dots, N \Big\} \times R(I - I_T) \\
&= \Big\{ w \in W : I_0 w_0^- = 0,\ I_T w_N^+ = 0 \Big\}, \\
\tilde{W} &= \hat{x} + \dot{W} = \Big\{ w \in W : w - \hat{x} \in \dot{W} \Big\} \\
&= \Big\{ w \in W : I_0 w_0^- = x_0,\ I_T w_N^+ = x_T \Big\}, \\
V &= H^1((0,T), \mathbb{R}^d), \\
U &= H^1((0,T), \mathbb{R}^m).
\end{aligned}$$

The spaces are equipped with the maximum norm. Note that, by Sobolev's inequality, functions in $W, \dot{W}$ are continuous on each interval I_n with one-sided limits at the endpoints, and functions in V, U are continuous on $[0, T]$. Boundary values are accommodated in W in the same way as in W_h; of course, if $w \in W$ happens to be continuous, then $w_0^- = w_0^+ = w(0)$ and $w_N^- = w_N^+ = w(T)$ are the usual boundary values. The function spaces have been constructed so that $W_h \subset W$, $\dot{W}_h \subset \dot{W}$, $\tilde{W}_h \subset \tilde{W}$, $V_h \subset V$, and $U_h \subset U$.

The functional to be minimised is

$$\mathcal{J}(w,u) = l(w_0^-, w_N^+) + \int_0^T L(w,u)\,dt, \quad (w,u) \in W \times U,$$

and, for the weak formulation of the state equation, we define the functional

$$\mathcal{F}(w,u;v) = \sum_{n=1}^{N} \int_{I_n} (\dot{w} - f(w,u), v)\,dt + \sum_{n=0}^{N} ([w]_n, v_n), \qquad (w,u,v) \in W \times U \times V.$$

Here and below $(\cdot,\cdot)$ denotes the scalar product in $\mathbb{R}^d$ or $\mathbb{R}^m$. If x is a smooth function which satisfies the state equation in (3.1), then it also satisfies the weak problem: find $x \in \tilde{W}$ such that

$$\mathcal{F}(x,u;\varphi) = 0, \quad \forall \varphi \in V. \tag{3.3}$$

Here we used the fact that $x_0^- = x(0)$, $x_N^+ = x(T)$, $[x]_n = 0$, because x is continuous.

We now find it convenient to change the notation for partial derivatives. For a scalar-valued function

$$g : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R},$$

we denote by $g_i'(x,u)$ the partial derivative with respect to the i^{th} variable. It is a linear operator $\mathbb{R}^d \to \mathbb{R}$ for $i = 1$ and $\mathbb{R}^m \to \mathbb{R}$ for $i = 2$, which we may identify with a vector, so that

$$g_1'(x,u)y = (y, g_1'(x,u)),\ y \in \mathbb{R}^d, \quad g_2'(x,u)y = (y, g_2'(x,u)),\ y \in \mathbb{R}^m.$$

For a vector-valued function

$$f : \mathbb{R}^d \times \mathbb{R}^m \to \mathbb{R}^d,$$

the partial derivatives are linear operators $f_1'(x,u) : \mathbb{R}^d \to \mathbb{R}^d$ and $f_2'(x,u) : \mathbb{R}^m \to \mathbb{R}^d$ denoted by $y \mapsto f_1'(x,u)y,\ y \in \mathbb{R}^d$ and $y \mapsto f_2'(x,u)y,\ y \in \mathbb{R}^m$.

We note that, by integration by parts,

$$\begin{aligned} &\mathcal{F}_1'(w,u;v,\varphi) \\ &\quad = \sum_{n=1}^{N} \int_{I_n} (\dot\varphi - f_1'(w,u)\varphi, v)\,dt + \sum_{n=0}^{N} ([\varphi]_n, v_n) \\ &\quad = \sum_{n=1}^{N} \int_{I_n} (\varphi, -\dot v - f_1'(w,u)^* v)\,dt + (\varphi_N^+, v_N) - (\varphi_0^-, v_0), \\ &\quad \forall (w,u,v,\varphi) \in W \times U \times V \times \dot W. \end{aligned} \tag{3.4}$$

The Lagrange functional is

$$\mathcal{L}(x,u;z) = \mathcal{J}(x,u) + \mathcal{F}(x,u;z), \quad (w,u,z) \in W \times U \times V.$$

The necessary condition for optimality is that $(x,u,z) \in \tilde W \times U \times V$ and

$$\mathcal{L}'(x,u;z,\varphi) = 0, \quad \forall \varphi \in \dot W \times U \times V, \tag{3.5}$$

which yields

$$\begin{aligned} \mathcal{L}_1'(x,u;z,\varphi_x) &= \mathcal{J}_1'(x,u;\varphi_x) + \mathcal{F}_1'(x,u;z,\varphi_x) = 0, \quad \forall \varphi_x \in \dot W, \\ \mathcal{L}_2'(x,u;z,\varphi_u) &= \mathcal{J}_2'(x,u;\varphi_u) + \mathcal{F}_2'(x,u;z,\varphi_u) = 0, \quad \forall \varphi_u \in U, \\ \mathcal{L}_3'(x,u;z,\varphi_z) &= 0 + \mathcal{F}(x,u;\varphi_z) = 0, \quad \forall \varphi_z \in V. \end{aligned} \tag{3.6}$$

The first equation in (3.6) is, in view of the second form of $\mathcal{F}_1'$ in (3.4),

$$\begin{aligned} &\sum_{n=1}^{N} \int_{I_n} (\varphi, L_1'(x,u) - \dot z - f_1'(x,u)^* z)\,dt \\ &\quad + (\varphi_N^+, l_2'(x_0^-, x_N^+) + z_N) + (\varphi_0^-, l_1'(x_0^-, x_N^+) - z_0) = 0, \quad \forall \varphi \in \dot W. \end{aligned}$$

Assuming that $x, \dot z, \varphi$ are continuous, we may identify the strong form of this equation:

$$\begin{aligned} &\dot z + f_1'(x,u)^* z - L_1'(x,u) = 0, \quad 0 < t < T, \\ &(I - I_0)\big(z(0) - l_1'(x(0), x(T))\big) = 0, \\ &(I - I_T)\big(z(T) + l_2'(x(0), x(T))\big) = 0, \end{aligned}$$

which is the linearised adjoint equation to the state equation in (3.1). Note the complementary boundary conditions.

The second equation in (3.6) is

$$\int_0^T (\varphi, L_2'(x,u) - f_2'(x,u)^* z)\, dt = 0, \quad \forall \varphi \in U,$$

or, in strong form,

$$L_2'(x,u) - f_2'(x,u)^* z = 0, \quad 0 < t < T.$$

This a non-linear algebraic equation for u. The third equation is the same as (3.3).

We next formulate the finite element approximation of these equations. Find $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ such that

$$\mathcal{L}'(x_h, u_h; z_h, \varphi) = 0, \quad \forall \varphi \in \dot{W}_h \times U_h \times V_h, \tag{3.7}$$

which means that we want to determine $(x_h, u_h, z_h) \in W_h \times U_h \times V_h$ such that

$$\begin{aligned} &\sum_{n=1}^N \int_{I_n} (\varphi, L_1'(x_h, u_h) - \dot{z}_h - f_1'(x_h, u_h)^* z_h)\, dt \\ &\quad + \big(\varphi_N^+, l_2'(x_{h,0}^-, x_{h,N}^+) + z_{h,N}\big) + \big(\varphi_0^-, l_1'(x_{h,0}^-, x_{h,N}^+) - z_{h,0}\big) \\ &\quad \forall \varphi \in \dot{W}_h, \end{aligned} \tag{3.8}$$

$$\int_0^T (\varphi, L_2'(x_h, u_h) - f_2'(x_h, u_h)^* z_h)\, dt = 0, \quad \forall \varphi \in U_h, \tag{3.9}$$

$$\begin{cases} I_0 x_{h,0}^- = x_0, \quad I_T x_{h,N}^+ = x_T, \\ \displaystyle\sum_{n=1}^N \int_{I_n} (\dot{x}_h - f(x_h, u_h), \varphi)\, dt + \sum_{n=0}^N ([x_h]_n, \varphi_n) = 0, \quad \forall \varphi \in V_h. \end{cases} \tag{3.10}$$

Using (3.2) we easily verify that these are $N(q+1)(2d+m)+3d+m$ algebraic equations in equally many unknowns.

Since φ_0^- and φ_N^+ can be chosen arbitrarily in $R(I - I_0)$ and $R(I - I_T)$, respectively, we see that (3.8) implies

$$\begin{aligned} (I - I_0)\big(l_1'(x_{h,0}^-, x_{h,N}^+) - z_{h,0}\big) = 0, \\ (I - I_T)\big(l_2'(x_{h,0}^-, x_{h,N}^+) + z_{h,N}\big) = 0. \end{aligned} \tag{3.11}$$

The *a posteriori* error representation formula follows from Theorem 2.2.

Corollary 3.1. *Let* $(x, u, z) \in \tilde{W} \times U \times V$ *and* $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ *be solutions of* (3.5) *and* (3.7), *respectively. Then*

$$\mathcal{J}(x,u) - \mathcal{J}(x_h, u_h) = \tfrac{1}{2}\rho_x + \tfrac{1}{2}\rho_z + \tfrac{1}{2}\rho_u + R, \tag{3.12}$$

with the residuals ρ_x, ρ_z, and ρ_u defined as

$$\begin{aligned}\rho_x &= \sum_{n=1}^{N} \int_{I_n} \big(x - \tilde{x}_h, L_1'(x_h, u_h) - \dot{z}_h - f_1'(x_h, u_h)^* z_h\big)\, dt,\\ \rho_u &= \int_0^T (u - \tilde{u}_h, L_2'(x_h, u_h) - f_2'(x_h, u_h)^* z_h)\, dt, \\ \rho_z &= \sum_{n=1}^{N} \int_{I_n} (\dot{x}_h - f(x_h, u_h), z - \tilde{z}_h)\, dt + \sum_{n=0}^{N} ([x_h]_n, z_n - \tilde{z}_{h,n}),\end{aligned} \tag{3.13}$$

where $(\tilde{x}_h, \tilde{u}_h, \tilde{z}_h) \in \tilde{W}_h \times U_h \times V_h$ is arbitrary, and the remainder R is given by (2.6).

Proof. From Theorem 2.2 we have

$$\begin{aligned}\rho_x = \sum_{n=1}^{N} \int_{I_n} &\big(x - \tilde{x}_h, L_1'(x_h, u_h) - \dot{z}_h - f_1'(x_h, u_h)^* z_h\big)\, dt \\ &+ \big(x_N^+ - \tilde{x}_{h,N}^+, l_2'(x_{h,0}^-, x_{h,N}^+) + z_{h,N}\big) \\ &+ \big(x_0^- - \tilde{x}_{h,0}^-, l_1'(x_{h,0}^-, x_{h,N}^+) - z_{h,0}\big).\end{aligned}$$

Using (3.11) and $I_0(x_0^- - \tilde{x}_{h,0}^-) = 0$, $I_T(x_N^+ - \tilde{x}_{h,N}^+) = 0$, we find

$$\begin{aligned}\big(x_N^+ - \tilde{x}_{h,N}^+, l_2'(x_{h,0}^-, x_{h,N}^+) + z_{h,N}\big) &= 0, \\ \big(x_0^- - \tilde{x}_{h,0}^-, l_1'(x_{h,0}^-, x_{h,N}^+) - z_{h,0}\big) &= 0,\end{aligned}$$

and we obtain the desired form of ρ_x. The other residuals, ρ_u and ρ_z, follow directly from Theorem 2.2. □

4. A quadratic/linear optimal control problem

4.1. The continuous problem. In this section we specialise to the case when the functional to be minimised is quadratic and the state equation is linear. We use the notation $\|v\|_S^2 = (v, Sv)$, where $(\cdot,\cdot)$ is the scalar product and S is a symmetric, positive semidefinite matrix. The problem then reads

$$\begin{aligned}\text{minimise} \quad & \mathcal{J}(x, u) = \|x(0) - \bar{x}_0\|_{S_0}^2 + \|x(T) - \bar{x}_T\|_{S_T}^2 \\ & \qquad + \int_0^T \big(\|u - \bar{u}\|_R^2 + \|x - \bar{x}\|_Q^2\big)\, dt, \\ \text{subject to} \quad & \dot{x} = A(t)x + B(t)u, \quad 0 < t < T, \\ & I_0 x(0) = x_0, \quad I_T x(T) = x_T,\end{aligned} \tag{4.1}$$

where, for each t, $Q(t), S_0, S_T \in \mathbb{R}^{d\times d}$ are symmetric positive semidefinite matrices, $R(t) \in \mathbb{R}^{m\times m}$ is a symmetric positive definite matrix, and $A(t) \in \mathbb{R}^{d\times d}$ and $B(t) \in \mathbb{R}^{d\times m}$ are matrices. The matrices I_0 and I_T are diagonal matrices with zeroes or ones on the diagonals, and x_0, x_T, $\bar{x}_0$, $\bar{x}_T$, $\bar{x}(t)$, and $\bar{u}(t)$ are given.

Since we now have

$$
\begin{aligned}
&f(x,u) = Ax + Bu, \\
&f_1'(x,u) = A, && f_2'(x,u) = B, \\
&L_1'(x,u) = 2Q(x-\bar{x}), && L_2'(x,u) = 2R(u-\bar{u}), \\
&l_1'(x_0^-, x_N^+) = 2S_0(x_0^- - \bar{x}_0), && l_2'(x_0^-, x_N^+) = 2S_T(x_N^+ - \bar{x}_T),
\end{aligned}
$$

the equation (3.5) is now to find $(x,u,z) \in \tilde{W} \times U \times V$ such that

$$
\begin{aligned}
&\int_0^T (\varphi_x, 2Q(x-\bar{x}) - \dot{z} - A^{\mathrm{T}} z)\, dt \\
&\quad + (\varphi_{x,0}^-, 2S_0(x_0^- - \bar{x}_0) - z_0) \\
&\quad + (\varphi_{x,N}^+, 2S_T(x_N^+ - \bar{x}_T) + z_N) = 0, \quad \forall \varphi_x \in \dot{W},
\end{aligned} \tag{4.2}
$$

$$
\int_0^T (\varphi_u, 2R(u-\bar{u}) - B^{\mathrm{T}} z)\, dt = 0, \quad \forall \varphi_u \in U, \tag{4.3}
$$

$$
\int_0^T (\dot{x} - Ax - Bu, \varphi_z)\, dt = 0, \quad \forall \varphi_z \in V. \tag{4.4}
$$

4.2. **The finite element method.** Let the finite element spaces be as in Section 3. We discretise the state equation (4.4) by a discontinuous Galerkin method with W_h as trial space and V_h as test space: Seek $x_h \in W_h$ which fulfils

$$
\begin{aligned}
&I_0 x_{h,0}^- = x_0, \quad I_T x_{h,N}^+ = x_T, \\
&\int_0^T (\dot{x}_h - Ax_h - Bu_h, \varphi)\, dt + \sum_{n=0}^{N} ([x_h]_n, \varphi_n) = 0, \quad \forall \varphi \in V_h.
\end{aligned} \tag{4.5}
$$

The dual equation (4.2) is discretised by the continuous Galerkin method: Seek $z_h \in V_h$ which fulfils

$$
\begin{aligned}
&\int_0^T (\varphi, 2Q(x_h-\bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h)\, dt \\
&\quad + (\varphi_0^-, 2S_0(x_{h,0}^- - \bar{x}_0) - z_{h,0}) \\
&\quad + (\varphi_N^+, 2S_T(x_{h,N}^+ - \bar{x}_T) + z_{h,N}) = 0, \quad \forall \varphi \in \dot{W}_h,
\end{aligned} \tag{4.6}
$$

where we have used V_h as trial space and $\dot{W}_h$ as test space. Since we can vary the boundary values in $\dot{W}_h$ separately in $R(I - I_0)$ and $R(I - I_T)$, the boundary conditions become

$$
\begin{aligned}
&(I - I_0)(z_{h,0} - 2S_0(x_{h,0}^- - \bar{x}_0)) = 0, \\
&(I - I_T)(z_{h,N} + 2S_T(x_{h,N}^+ - \bar{x}_T)) = 0.
\end{aligned}
$$

The equation for the controls, (4.3), is discretised by a continuous Galerkin method: Seek $u_h \in U_h$

$$\int_0^T (\varphi, 2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h)\, dt = 0, \quad \forall \varphi_u \in U_h. \tag{4.7}$$

We now have three sets of linear algebraic equations which must be solved simultaneously in order to obtain the approximate solution (x_h, u_h, z_h).

4.3. **The error estimate.** We begin by repeating the error representation formula from Corollary 3.1 in the context of the linear/quadratic optimal control problem.

Corollary 4.1. *Let* $(x, u, z) \in \tilde{W} \times U \times V$ *and* $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ *be solutions of* (4.2)–(4.4) *and* (4.5)–(4.7)*, respectively. Then*

$$\mathcal{J}(x, u) - \mathcal{J}(x_h, u_h) = \tfrac{1}{2}\rho_x + \tfrac{1}{2}\rho_z + \tfrac{1}{2}\rho_u, \tag{4.8}$$

with ρ_x*,* ρ_z*, and* ρ_u *defined as*

$$\begin{aligned}
\rho_x &= \int_0^T (x - \tilde{x}_h, 2Q(x_h - \bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h)\, dt, \\
\rho_u &= \int_0^T (u - \tilde{u}_h, 2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h)\, dt, \\
\rho_z &= \int_0^T (\dot{x}_h - Ax_h - Bu_h, z - \tilde{z}_h)\, dt + \sum_{n=0}^{N} ([x_h]_n, z_n - \tilde{z}_{h,n}),
\end{aligned} \tag{4.9}$$

where $(\tilde{x}_h, \tilde{u}_h, \tilde{z}_h) \in \tilde{W}_h \times U_h \times V_h$ *is arbitrary.*

Proof. The proof is a straightforward calculation using Corollary 3.1. The remainder R is zero in this case, since we have a linear/quadratic problem and the remainder is the third derivative of the Lagrangian. □

In the following theorem we derive an *a posteriori* error estimate from the error representation formula. We use the notation $\|f\|_{I_n} = \sup_{t \in I_n} \|f(t)\|$, where $\|\cdot\|$ denotes the norm in $\mathbb{R}^d$ or $\mathbb{R}^m$.

Theorem 4.2. *Let* $(x, u, z) \in \tilde{W} \times U \times V$ *and* $(x_h, u_h, z_h) \in \tilde{W}_h \times U_h \times V_h$ *be solutions of* (4.2)–(4.4) *and* (4.5)–(4.7)*, respectively. Then*

$$|\mathcal{J}(x, u) - \mathcal{J}(x_h, u_h)| \le \tfrac{1}{2} \sum_{n=1}^{N} \Big(R_n^z \omega_n^x + R_n^u \omega_n^u + R_n^x \omega_n^z \Big), \tag{4.10}$$

where the residuals R_n *and weights* ω_n *are defined by (with* $h_0 = h_N = 0$*)*

$$\begin{aligned}
R_n^z &= h_n \|2Q(x_h - \bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h\|_{I_n}, \\
R_n^u &= h_n \|2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h\|_{I_n}, \\
R_n^x &= h_n \|\dot{x}_h - Ax_h - Bu_h\|_{I_n} + \frac{h_n}{h_n + h_{n+1}} \| \left[x_h\right]_n \| \\
&\quad + \frac{h_n}{h_n + h_{n-1}} \| \left[x_h\right]_{n-1} \|,
\end{aligned}$$

and, with arbitrary $(\tilde{x}_h, \tilde{u}_h, \tilde{z}_h) \in \tilde{W}_h \times U_h \times V_h$*,*

$$\omega_n^x = \|x - \tilde{x}_h\|_{I_n}, \quad \omega_n^u = \|u - \tilde{u}_h\|_{I_n}, \quad \omega_n^z = \|z - \tilde{z}_h\|_{I_n}.$$

Proof. We estimate the three contributions to the error representation (4.8) separately. The first term is

$$\begin{aligned}
|\rho_x| &\leq \sum_{n=1}^{N} \int_{I_n} \|x - \tilde{x}_h\| \|2Q(x_h - \bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h\| \, dt \\
&\leq \sum_{n=1}^{N} \|x - \tilde{x}_h\|_{I_n} \|2Q(x_h - \bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h\|_{I_n} h_n = \sum_{n=1}^{N} \omega_n^x R_n^z.
\end{aligned}$$

Similarly, for the second term we have

$$|\rho_u| \leq \sum_{n=1}^{N} \|u - \tilde{u}_h\|_{I_n} \|2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h\|_{I_n} h_n = \sum_{n=1}^{N} \omega_n^u R_n^u.$$

Finally,

$$\begin{aligned}
|\rho_z| &\leq \sum_{n=1}^{N} \int_{I_n} \|\dot{x}_h - Ax_h - Bu_h\| \|z - \tilde{z}_h\| \, dt + \sum_{n=0}^{N} \| \left[x_h\right]_n \| \|z_n - \tilde{z}_{h,n}\| \\
&\leq \sum_{n=1}^{N} \|\dot{x}_h - Ax_h - Bu_h\|_{I_n} \|z - \tilde{z}_h\|_{I_n} h_n + \sum_{n=0}^{N} \| \left[x_h\right]_n \| \|z_n - \tilde{z}_{h,n}\|.
\end{aligned}$$

Using the continuity of z we have

$$\|z_n - \tilde{z}_{h,n}\| \leq \|z - \tilde{z}_h\|_{I_n}, \quad \|z_n - \tilde{z}_{h,n}\| \leq \|z - \tilde{z}_h\|_{I_{n+1}},$$

so that

$$\begin{aligned}
&\sum_{n=0}^{N} \| [x_h]_n \| \| z_n - \tilde{z}_{h,n} \| \\
&= \sum_{n=1}^{N} \Big(\frac{h_n}{h_n + h_{n+1}} \| [x_h]_n \| \| z_n - \tilde{z}_{h,n} \| \\
&\quad + \frac{h_n}{h_n + h_{n-1}} \| [x_h]_{n-1} \| \| z_{n-1} - \tilde{z}_{h,n-1} \| \Big) \\
&\le \sum_{n=1}^{N} \Big(\frac{h_n}{h_n + h_{n+1}} \| [x_h]_n \| + \frac{h_n}{h_n + h_{n-1}} \| [x_h]_{n-1} \| \Big) \| z - \tilde{z} \|_{I_n},
\end{aligned}$$

where $h_0 = h_N = 0$. This yields

$$\begin{aligned}
|\rho_z| \le \sum_{n=1}^{N} \Big(h_n \| \dot{x}_h - A x_h - B u_h \|_{I_n} + \frac{h_n}{h_n + h_{n+1}} \| [x_h]_n \| \\
+ \frac{h_n}{h_n + h_{n-1}} \| [x_h]_{n-1} \| \Big) \| z - \tilde{z}_h \|_{I_n} = \sum_{n=1}^{N} R_n^x \omega_n^z.
\end{aligned}$$

□

We note that the error estimate does not introduce any additional adjoint equation. However, the weights depend on the exact solutions x, u, z and approximations $\tilde{x}_h, \tilde{u}_h, \tilde{z}_h$ of them. In practice, we approximate the weights by computable quantities. For example, when $q = 0$, by standard interpolation error estimates [4], we can find $\tilde{x}_h, \tilde{u}_h, \tilde{z}_h$ such that

$$\begin{aligned}
\| x - \tilde{x}_h \|_{I_n} &\le h_n \| \dot{x} \|_{I_n}, \\
\| u - \tilde{u}_h \|_{I_n} &\le h_n^2 \| \ddot{u} \|_{I_n}, \qquad (4.11) \\
\| z - \tilde{z}_h \|_{I_n} &\le h_n^2 \| \ddot{z} \|_{I_n},
\end{aligned}$$

where the derivatives are approximated by difference quotients of the discrete solutions. See also [1] for other approximations of the weights.

The above estimates of the weights indicate that the term ρ_z in the error estimate is $O(h)$, while ρ_x and ρ_u are $O(h^2)$. We therefore present the following error estimate, where all terms are formally $O(h^2)$. For simplicity we assume that $A(t) = A$ and $Q(t) = Q$ are constant.

Theorem 4.3. *Let $q = 0$ and assume that $A(t) = A$ and $Q(t) = Q$ are constant. Then*

$$\begin{aligned}
|\mathcal{J}(x,u) - \mathcal{J}_h(x_h,u_h)| \le \sum_{n=1}^{N} \Big(& h_n^3 \| \dot{x} \|_{I_n} \| 2Q\dot{\bar{x}} + A^{\mathrm{T}} \dot{z}_h \|_{I_n} \\
& + h_n^3 \| A x_h + B u_h \|_{I_n} \| \ddot{z} \|_{I_n} \\
& + h_n^3 \| 2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h \|_{I_n} \| \ddot{u} \|_{I_n} \Big).
\end{aligned}$$

Proof. We choose $\tilde{z}_h = I_h z$ and $\tilde{u}_h = I_h u$ to be the standard piecewise linear nodal interpolants, and we choose $\tilde{x}_h = P_h x$ to be the orthogonal projection onto the piecewise constant functions.

Then, using orthogonality, the fact that $z_n - \tilde{z}_{h,n} = 0$, and the error estimates (4.11), in the error representation formula (4.8), we obtain

$$
\begin{aligned}
&\mathcal{J}(x,u) - \mathcal{J}(x_h,u_h) \\
&\quad = \Big| \sum_{n=1}^{N} \Big(\int_{I_n} \big((I-P_h)x, (I-P_h)(2Q(x_h-\bar{x}) - \dot{z}_h - A^{\mathrm{T}} z_h)\big)\, dt \\
&\quad + \int_{I_n} \big(\dot{x}_h - Ax_h - Bu_h, (I-I_h)z\big)\, dt \\
&\quad + \int_{I_n} \big((I-I_h)u, 2R(u_h-\bar{u}) - B^{\mathrm{T}} z_h\big)\, dt \Big) \Big| \\
&\quad \le \sum_{n=1}^{N} \Big(h_n^3 \|\dot{x}\|_{I_n} \|2Q(\dot{x}_h - \dot{\bar{x}}) - \ddot{z}_h - A^{\mathrm{T}} \dot{z}_h\|_{I_n} \\
&\quad + h_n^3 \|\dot{x}_h - Ax_h - Bu_h\|_{I_n} \|\ddot{z}\|_{I_n} \\
&\quad + h_n^3 \|2R(u_h - \bar{u}) - B^{\mathrm{T}} z_h\|_{I_n} \|\ddot{u}\|_{I_n} \Big).
\end{aligned}
$$

Since $\dot{x}_h = 0$ and $\ddot{z}_h = 0$ we obtain the desired estimate. □

4.4. **An adaptive algorithm.** On the basis of the error estimate in the previous theorem we implement an adaptive finite element method, with $q = 0$, for the solution of the optimal control problem (4.1).

Algorithm 1: An adaptive finite element method

Solve the equation on a coarse initial mesh;
Compute the error estimate in Theorem 4.2, denote it by η;
while $\eta \geq TOL$ **do**
  Refine the mesh according to the error estimate, i.e., refine elements that give large contributions to the estimate;
  Solve the equation on the refined mesh;
  Compute the error estimate η on the refined mesh;
end

The refinement of the mesh is done according to the principle of equidistribution, that is, we want all intervals to give equally large contributions to the error estimate and insert new nodes to fulfil this criterion. A numerical example is given in the next section.

5. A numerical example

The adaptive finite element solver is tested on a linear/quadratic problem. In order to get a linear/quadratic problem we use a simplified model

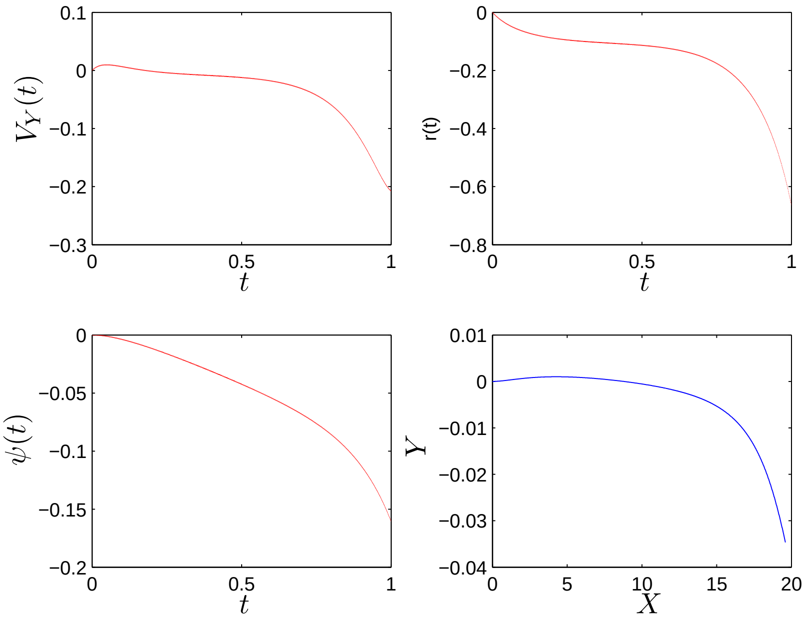

FIGURE 1. The optimal states. The last image shows the optimal track.

from vehicle dynamics describing a vehicle that is braked on a surface with different friction on the wheel pairs. We let the control variable u be $u_1 = \delta_\mathrm{f}$ and $u_2 = \delta_\mathrm{r}$ and the state variable x is

$$x(t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} V_X \\ V_Y \\ r \\ \psi \\ X \\ Y \end{bmatrix}. \tag{5.1}$$

The differential equations are

$$\dot{x} = \begin{bmatrix} \dot{V}_X \\ \dot{V}_Y \\ \dot{r} \\ \dot{\psi} \\ \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21}V_Y + a_{22}r + bf_1\delta_\mathrm{f} + br_1\delta_\mathrm{r} \\ a_{31}V_Y + a_{32}r + bf_2\delta_\mathrm{f} + br_2\delta_\mathrm{r} + r_{\mathrm{brake}} \\ r \\ V_X \\ V_Y \end{bmatrix} = Ax(t) + Bu(t) + b. \tag{5.2}$$

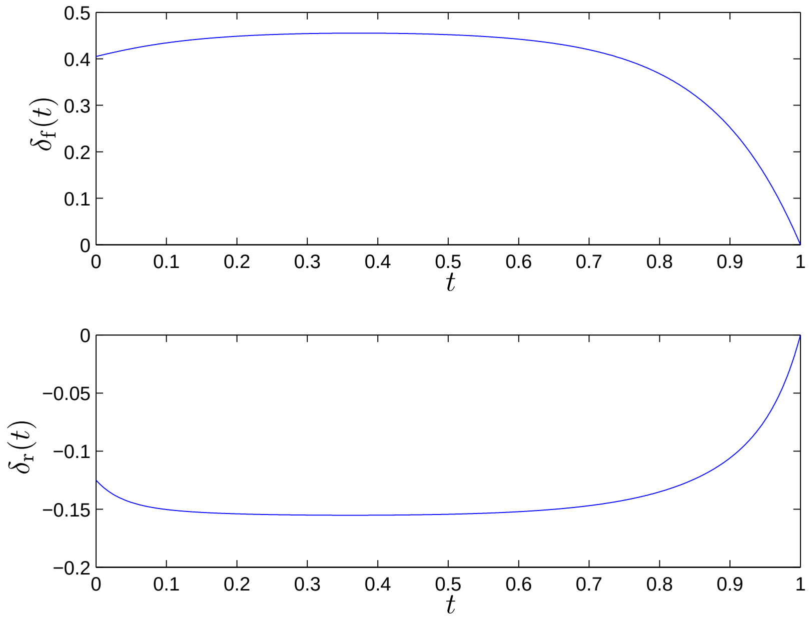

FIGURE 2. The optimal controls u.

The goal functional is

$$J(x,u) = \int_0^T \tfrac{1}{2}\left(Y^2 + r^2 + V_Y^2 + \delta_f^2 + \delta_r^2\right) dt = \int_0^T \left(\|x\|_Q^2 + \|u\|_R^2\right) dt.$$

We have used the boundary condition $x_1(0) = 25$ and $x_2(0) = x_3(0) = x_4(0) = x_5(0) = x_6(0) = 0$, which means that the state equation has no boundary conditions at the end of the time interval and therefore the dual equation has all its boundary conditions at the final time.

We have now formulated our problem in the form (4.1), but with an extra b in the right hand side:

$$\begin{aligned} &\text{minimise} && \mathcal{J}(x,u) = \int_0^T \left(\|u\|_R^2 + \|x\|_Q^2\right) dt, \\ &\text{subject to} && \dot{x} = A(t)x + B(t)u + b, \quad 0 < t < T, \\ &&& I_0 x(0) = x_0, \quad I_T x(T) = x_T, \end{aligned}$$

where I_0 is the identity matrix and I_T is zero. The coefficients A, B, Q, R and b can be found in the Appendix.

In Figure 1 we see the optimal states and Figure 2 shows the optimal controls and velocities. In Figure 3 we see the error estimate plotted as a function of the number of intervals. The convergence rate of the solution in the numerical example is 2.03.

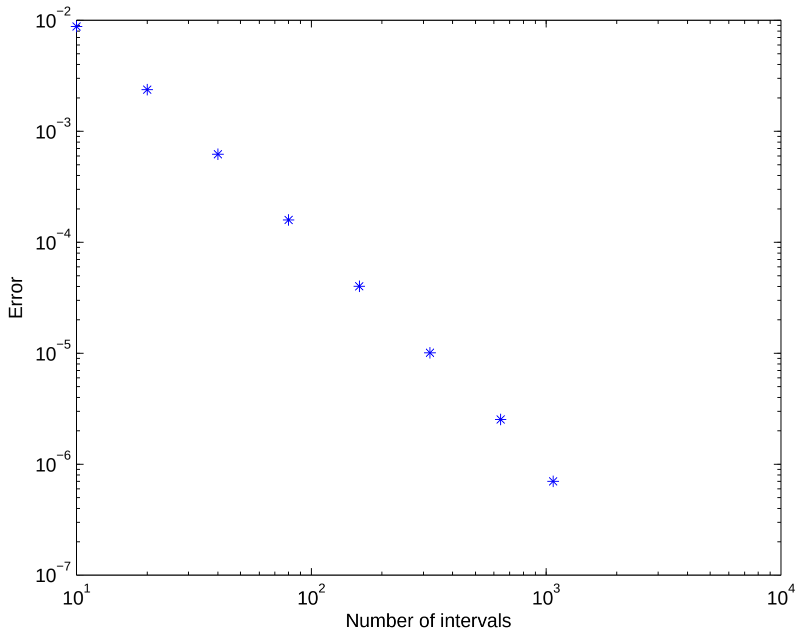

FIGURE 3. The error estimate as a function of the number of intervals. The convergence rate is 2.03.

REFERENCES

1. R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation in finite element methods*, Acta Numer. **10** (2001), 1–102.
2. J. T. Betts, *Practical Methods for Optimal Control Using Nonlinear Programming*, SIAM, Philadelphia, 2001.
3. A. E. Bryson, Jr and Y. Ho, *Applied Optimal Control*, Hemisphere Publishing Corporation, Washington, D.C., 1975.
4. K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Computational Differential Equations*, Cambridge University Press, Cambridge, 1996.
5. D. Estep, D. H. Hodges, and M. Warner, *Computational error estimation and adaptive error control for a finite element solution of launch vehicle trajectory problems*, SIAM J. Sci. Comput. (1999), 1609–1631 (electronic).
6. K. Kraft, S. Larsson, and M. Lidberg, *Using an adaptive FEM to determine the optimal control of a vehicle during a collision avoidance manoeuvre*, Proceedings of the 48th Scandinavian Conference on Simulation and Modeling (SIMS2007), Linköping University Electronic Press, 2007, `http://www.ep.liu.se/ecp/027/`.

APPENDIX

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{21} & a_{22} & 0 & 0 & 0 \\ 0 & a_{31} & a_{32} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ bf_1 & br_1 \\ bf_2 & br_2 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} a_{11} \\ 0 \\ r_{\text{brake}} \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

and

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}, \quad R = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix},$$

where

$$\begin{aligned} a_{21} &= -(C_f + C_r)/(mv_0), & a_{22} &= (C_r L_r - C_f L_f)/(mv_0), \\ a_{31} &= (C_r L_r - C_f L_f)/(I_z v_0), & a_{32} &= -(C_f(L_f^2) + C_r(L_r^2))/(I_z v_0)), \\ b_{f1} &= (C_f - F_{xf})/m, & b_{f2} &= (L_f F_{xf} + C_f L_f)/I_z, \\ b_{r1} &= (C_r - F_{xr})/m, & b_{r2} &= -(L_r F_{xf} + C_r L_r)/I_z, \\ a_{11} &= (F_{xf} + F_{xr})/m, \end{aligned}$$

and

$$\begin{aligned} F_{xrR} &= -mg\mu_{12}, & F_{xrL} &= -mg\mu_{34}, \\ F_{xfR} &= 0, & F_{xfL} &= 0, \\ F_{xr} &= F_{xrR} + F_{xrL}, & F_{xf} &= F_{xfR} + F_{xfL}, \\ r_{\text{brake}} &= -(F_{xrR} - F_{xrL})B_r/m, \end{aligned}$$

with numerical values

$$\begin{aligned} m &= (1500 + 150)\ \text{kg}, & I_z &= 3500\ \text{kg}\,\text{m}^2 \\ L &= 2.755\ \text{m}, & L_f &= 1.20\ \text{m} \\ L_r &= L - L_f, \\ C_f &= 20000\ \text{N/rad}, & C_r &= 40000\ \text{N/rad} \\ v_0 &= 20\ \text{m/s}, & g &= 9.82\ \text{m/s}^2 \\ B_f &= 1.563\ \text{m}, & B_r &= 1.560\ \text{m} \\ \mu_{12} &= 0.35, & \mu_{34} &= 0.75. \end{aligned}$$

Department of Mathematical Sciences, Chalmers University of Technology and Göteborg University, SE–412 96 Göteborg, Sweden
E-mail address: karin.kraft@chalmers.se

Department of Mathematical Sciences, Chalmers University of Technology and Göteborg University, SE–412 96 Göteborg, Sweden
E-mail address: stig@chalmers.se