Optimal Control of Vehicle Systems

Giacomo Perantoni

New College

Department of Engineering Science

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2013

Abstract

Optimal Control of Vehicle Systems

Giacomo Perantoni — New College

DPhil in Engineering Science — Michaelmas 2013

This thesis studies the optimal control of vehicular systems, focusing on the solution of minimum-lap-time problems for a Formula 1 car. The basic optimal control theory is summarised as an infinite-dimensional extension of optimisation theory. The relevant numerical techniques for optimisation and integral approximation are compared in view of the application to vehicle systems.

The classical brachistochrone problem is revisited from an optimal control perspective, with two vehicle-relevant generalisations. Closed-form solutions are derived for both the optimal trajectory and transit time. Problems involving a steerable disc rolling on the interior surface of a hemisphere are studied. For three-dimensional problems of this type, which involve rolling bodies and nonholonomic constraints, numerical solutions are used.

The identification of 3D race track models from measured GPS data is treated as a problem in the differential geometry of curves and surfaces. Curvilinear coordinates are adopted to facilitate optimal control solutions. The track is specified in terms of three displacement-dependent curvatures and two edge variables. The differential model is smoothed using numerical optimal control techniques. The Barcelona track is considered as an illustrative example.

The minimum-lap-time problem for a Formula 1 car on a flat track is solved using direct transcription. The driven line and multiple car setup parameters are optimised simultaneously. It is shown that significant lap-time reductions can be obtained from track-specific setup parameter optimisation. Reduced computing times are achieved using a combination of analytical derivatives, model non-dimensionalisation and problem scaling.

The optimal control of the car on a 3D track is studied; the results are compared with flat-track solutions. Contemporary kinetic energy-recovery systems are studied and compared with future hybrid kinetic-thermal energy-recovery systems. It is demonstrated that these systems can produce contemporary lap time using approximately two-thirds of the fuel required by present-day vehicles.

dedicata a mia madre, Valeria

Contents

Acknowledgements v

Publications vi

Mathematical Notation vii

Symbols viii

Acronyms xi

List of Figures xii

List of Tables xviii

1 Introduction 1

2 Numerical Optimal Control 8

2.1 Parameter Optimisation . . . 9

2.1.1 Equality Constraints . . . 11

2.1.2 Inequality Constraints . . . 14

2.1.3 Optimal Control of Multi-Stage Systems . . . 16

2.2 The Calculus of Variations . . . 19

2.2.1 Lagrangian Mechanics . . . 22

2.2.2 Continuous-Time Optimal Control . . . 23

2.2.2.1 Variable Time Horizon . . . 25

2.3 Numerical Methods . . . 26

2.3.1 Optimisation . . . 29

2.3.1.1 Root-Finding . . . 29

2.3.1.2 Equality-Constrained Optimisation . . . 31

2.3.1.3 Inequality-Constrained Optimisation . . . 32

2.3.2 Initial-Value Problem . . . 34

2.3.2.1 Basic Rule . . . 35

2.3.2.2 Runge-Kutta Schemes . . . 37

2.3.2.3 Collocation . . . 38
2.3.2.4 Error Control . . . 40
2.3.2.5 Differential-Algebraic Equations . . . 41
2.3.3 Transcription . . . 42
2.3.3.1 Single-Shooting . . . 44
2.3.3.2 Multiple-Shooting . . . 46
2.3.3.3 Collocation . . . 48
2.3.4 Direct and Indirect Methods . . . 49
2.3.4.1 Euler-Collocation Equivalence . . . 54
2.4 Summary . . . 55

3 Optimal Control of Nonholonomic Systems 57
3.1 The Brachistochrone . . . 60
3.1.1 The Classical Problem . . . 60
3.1.2 The Unrestrained Brachistochrone . . . 62
3.1.2.1 State Continuity . . . 65
3.1.2.2 Transit Time . . . 66
3.1.2.3 Solution Structure . . . 66
3.1.2.4 Impulsive Control . . . 68
3.1.2.5 Solution Possibilities . . . 69
3.1.3 The Rolling Unrestrained Brachistochrone . . . 70
3.1.3.1 Zero-Normal-Load Trajectory . . . 73
3.1.3.2 Transit Time . . . 75
3.1.3.3 Numerical Optimal Control . . . 77
3.2 Disc Rolling in a Hemisphere . . . 78
3.2.1 Results . . . 81
3.3 Summary . . . 85

4 Track Modelling and Identification 89
4.1 Geometry of Ribbons . . . 92
4.1.1 Rotations and Cross Products . . . 92
4.1.2 Curves . . . 94
4.1.3 Ribbons . . . 95
4.1.4 Euler Angles . . . 97
4.2 Numerical Optimal Control . . . 100
4.2.1 Finite-Differencing . . . 101
4.2.2 Noise Influences . . . 101
4.2.3 Problem Formulation . . . 103
4.2.3.1 Dynamics . . . 103
4.2.3.2 Cost . . . 104
4.2.3.3 Constraints . . . 105

4.3 Results . . . 106
4.3.1 Lemniscate Ribbon . . . 106
4.3.2 Circuit de Catalunya . . . 109
4.4 Summary . . . 114

5 Minimum-Lap-Time Problem 115
5.1 Track and Vehicle Models . . . 121
5.1.1 Track Model . . . 121
5.1.1.1 Change of Independent Variable . . . 124
5.1.2 Dynamic Model . . . 125
5.1.3 Tyre Forces . . . 126
5.1.3.1 Load Transfer . . . 126
5.1.3.2 Non-Positive Tyre Loads . . . 127
5.1.4 Aerodynamics . . . 129
5.1.5 Wheel Torque Distribution . . . 130
5.1.5.1 Brakes . . . 130
5.1.5.2 Differential . . . 130
5.2 Numerical Optimal Control . . . 131
5.2.1 Problem Formulation . . . 131
5.2.2 Transcription . . . 132
5.2.3 State and Control Scaling . . . 135
5.2.3.1 Non-Dimensionalisation . . . 135
5.2.3.2 Affine Transformations . . . 136
5.2.4 Smoothing Functions . . . 136
5.3 Results . . . 137
5.3.1 Optimal Lap . . . 139
5.3.2 Vehicle Stability . . . 143
5.3.3 Parameter Optimisation . . . 145
5.3.4 Neglected and Slow Dynamics . . . 150
5.4 Summary . . . 151

6 Vehicle Model Extensions 153
6.1 Speed-Dependent Aerodynamic Coefficients . . . 154
6.1.1 Nominal Car Performance . . . 155
6.2 3D Track . . . 157
6.2.1 Car Model . . . 158
6.2.1.1 Kinematics . . . 159
6.2.1.2 Dynamics . . . 161
6.2.1.3 Load Transfer . . . 165
6.2.2 Optimal-Control Problem Formulation . . . 165
6.2.3 Results . . . 166

6.3 Energy-Recovery Systems . . . 172
6.3.1 Kinetic Energy Recovery . . . 172
6.3.1.1 Results . . . 174
6.3.2 Kinetic and Thermal Energy Recovery . . . 175
6.3.2.1 Results . . . 181
6.4 Summary . . . 186

7 Conclusions **188**

8 Proposed Developments **191**
8.1 Parallel Computing . . . 191
8.2 Model Extensions . . . 192
8.3 Optimisation Studies . . . 194
8.4 Robustness and Feedback Control . . . 194
8.5 Validation . . . 196

A Tyre Friction **198**

B Vehicle and Tyre Data **202**

C Ribbon Tangent Plane **204**
C.1 Tangent Frame . . . 205
C.1.1 Embedded Curve . . . 206
C.2 Kinematics . . . 207
C.3 Dynamics . . . 208

Bibliography **211**

Acknowledgements

A number of people deserve acknowledgement for their contribution to the fulfilment of the work presented in this thesis.

My sincerest thanks go to my supervisor Prof. David Limebeer, who was an inestimable guide, mentor and research collaborator, invariably present and supportive. I must also acknowledge the fundamental role of Dr Simos Evangelou, who provided the right encouragement to embark on this project.

Many academics and industry experts deserve thanks for their guidance and the insightful discussions I had with them, which contributed to my understanding of the subjects covered in this thesis. In particular I must thank: Dr Amrit Sharma, Dr Matteo Massaro and Prof. Roberto Lot for discussions on vehicle modelling; Prof. Richard Vinter, Prof. Anil Rao, Dr Paola Falugi and Dr Eric Kerrigan for discussions on the theory and application of optimal control; Prof. Igo Besselink for the experience he shared with me on tyre modelling; Prof. Arend Shwab and Dr Jaap Meijaard for dialogues on bicycle dynamics; Dr Enrico Sampò, Dr Giacomo Tortora and Luca De Bernardi for their suggestions on the application of optimal control to racing cars.

Among my colleagues I owe special thanks to Dr Matthew Arthington, who assisted me in all aspects of my work. Konstantinos-Nikolaos Bakis, Ingrid Salisbury, Thomas Prescott and Peter Fussey also deserve to be mentioned for the help they gave on various occasions.

I cannot forget the support received from many friends in Oxford throughout this doctorate; in particular I want to mention Dr Emanuele Ruspini, Laura Murray-White, Sei Fukao, Solana Mizrahi, Gökhan Bilgin, Dr Samuel Slattery, Dr Araz Taeihagh, Ignacio Lamata Martinez, Andrew Spurr, Neo Vincen Malvaez and all the members of the Oxford University Canoe and Kayak Club.

I will never stop thanking my family, Carlo, Valeria and Sergio, and my girlfriend Francesca for their unconditional affection and love.

This educational journey would not have been as enjoyable without the hospitality offered by New College, the Department of Engineering Science, the University of Oxford and Scuderia Ferrari.

This work was supported by the UK Engineering and Physical Sciences Research Council.

Publications

The following papers are the result of research conducted during my doctoral studies:

1. G. Perantoni and D. J. N. Limebeer. A dynamic motorcycle tyre model based on lugre friction. *The 11th International Symposium on Advanced Vehicle Control*, September 2012. [63]

2. G. Perantoni and D. J. N. Limebeer. A direct method for optimal control and optimal design of two-wheeled vehicles. *The 11th International Symposium on Advanced Vehicle Control*, September 2012. [62]

3. G. Perantoni and D. J. N. Limebeer. Optimal control of a two-mass skate bicycle without steering. *IEEE International Conference on Industrial Technology*, February 2013. [66]

4. G. Perantoni and D. J. N. Limebeer. Time-optimal control of rolling bodies. *International Journal of Control*, 86, 2013. [67]

5. G. Perantoni and D. J. N. Limebeer. Minimum-lap-time optimal control for a formula one car with variable parameters. *Vehicle System Dynamics.* under review, 2013. [64]

6. G. Perantoni and D. J. N. Limebeer. Optimal control of a formula one car on a three-dimensional track part 1: Track modelling and identification. *Journal of Dynamic Systems, Measurement and Control.* under review, 2013. [65]

7. D. J. N. Limebeer and G. Perantoni. Optimal control of a formula one car on a three-dimensional track part 2: Optimal control. *Journal of Dynamic Systems, Measurement and Control.* under review, 2013. [46]

8. D. J. N. Limebeer and G. Perantoni. Optimal control of formula one car energy recovery systems. *International Journal of Control.* under review, 2013. [47]

Mathematical Notation

Lower-case unbold italic symbols are used for scalars; lower-case bold symbols denote vectors; upper-case italic symbols are used for matrices; upper-case bold symbols represent points; calligraphic characters are used for sets; physical units are given using the SI convention in plain fonts. Exceptions to the rules above are made for forces F, moments M, energy E, power P, the objective function J, the Lagrangian L, the kinetic energy T, the potential energy U and the number of samples N, following standard notation. The upper-case symbol $\boldsymbol{\Omega}$ is used for a vector.

Unit vectors are denoted by a hat, e.g. $\hat{\boldsymbol{x}}$. A vector (or a matrix) with an upper-case subscript is expressed in the reference frame indicated by the subscript; e.g. $\boldsymbol{x}_I$ is the vector $\boldsymbol{x}$ expressed in the inertial frame I. The i^{th} element of a vector is identified by the unbolded vector symbol with subscript i; e.g. x_i is the i^{th} element of $\boldsymbol{x}$. The length of a vector is represented by the underlined unbolded vector symbol; e.g. $\underline{x}$ is the length of $\boldsymbol{x}$.

The gradient of a scalar function $f(\boldsymbol{x})$ with respect to a vector $\boldsymbol{x}$ is denoted by $\nabla_{\boldsymbol{x}} f(\boldsymbol{x})$ and is a $1 \times \underline{x}$ row vector. The Jacobian of a vector-valued function $\boldsymbol{f}(\boldsymbol{x})$ with respect to a vector $\boldsymbol{x}$ is denoted by $\nabla_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x})$. The Jacobian is a $\underline{f} \times \underline{x}$ matrix. The Hessian of a scalar function $f(\boldsymbol{x})$ with respect to a vector $\boldsymbol{x}$ is denoted by $\nabla_{\boldsymbol{x}\boldsymbol{x}} f(\boldsymbol{x})$. The Hessian is a square $\underline{x} \times \underline{x}$ matrix.

The cardinality (number of elements) of a set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. A subset of the elements of a vector (or a matrix) is identified using a set as a subscript. For instance consider the set $\mathcal{A} = \{i, j\}$. The notation $\boldsymbol{x}_{\mathcal{A}}$ indicates a vector formed by the i^{th} and j^{th} elements of $\boldsymbol{x}$. An equivalence between a scalar c and a vector $\boldsymbol{x}$ shall be interpreted element-wise. The same holds for inequalities and sums.

Axis labels in figures specify units and multiplicative coefficients (when required) for the prescribed quantities; e.g. the label P / [100 kW] indicates that values on the axis correspond to values of power scaled by 100 kW ($1.2 \rightarrow 120$ kW). Circled numbers indicate corners of a racing track, numbered in increasing order from the start-finish line; e.g. (2) denotes the 2^{nd} turn on a track. The operator $:=$ is used for definitions; e.g. $a := b$ means that a is defined to be equal to b. The $(\cdot)$ notation refers to function operators, as opposed to function values; e.g. $f(\cdot)$ denotes the operator $f : \mathcal{D}_1 \rightarrow \mathcal{D}_2$, while $f(x)$ is the function value at x. Dot subscripts signify all the possible subscript variants of a symbol; e.g. $\omega_{\cdot\cdot}$ means the angular speeds of the four vehicle wheels $\{\omega_{fl}, \omega_{fr}, \omega_{rl}, \omega_{rr}\}$.

Symbols

Upper-Case Roman

Symbol	Description
$\mathcal{B}_l$, $\mathcal{B}_r$	left- and right-hand ribbon's boundaries
$\mathcal{C}$	curve (e.g. ribbon's spine)
$\mathbf{C}_m$	vehicle's centre of mass
$\mathbf{C}_p$	vehicle's centre of pressure
E_s	stored energy
$F_{..x}$, $F_{..y}$	longitudinal and lateral components of the tyre friction forces
$F_{..z}$	tyre loads
F_{ax}, F_{az}	longitudinal and vertical components of the aerodynamic force
$\mathbf{G}$	vehicle's geometric centre
H	Hamiltonian
I	vehicle's inertia tensor
$I_{n\times n}$	identity matrix of size $n \times n$
I_z	vehicle's moment of inertia around the body-fixed z-axis
J	performance function
L	Lagrangian
N	number of quadrature intervals used for transcription
P	power
$\mathcal{R}$	ribbon surface
R	rotation matrix
$R_z(\alpha)$	elementary rotation matrix for a rotation of angle α around the z-axis
$S(\cdot)$	skew-symmetric matrix operator
T	kinetic energy
U	potential energy

Lower-Case Roman

Symbol	Description
$\mathbf{c}$	position vector of a point on the ribbon's spine
$\hat{\mathbf{e}}_x$, $\hat{\mathbf{e}}_y$, $\hat{\mathbf{e}}_z$	base vectors for the canonical base
$\mathbf{f}$	vector of constraints (implicit notation)
$\boldsymbol{f}$	vector of constraints (explicit notation)
$\mathbf{g}$	vector of algebraic equality constraints
g	gravitational acceleration
$\mathbf{h}$	vector of inequality constraints
$\hat{\mathbf{m}}$	unit vector normal to the ribbon's surface (at the spine)
$\hat{\mathbf{n}}$	second unit vector in the ribbon frame: $\hat{\mathbf{n}} = \hat{\mathbf{m}} \times \hat{\mathbf{t}}$
n	curvilinear ordinate (distance between $\mathbf{G}$ and the ribbon's spine)
n_l, n_r	left- and right-hand-side width of the ribbon
$\boldsymbol{p}$	vector of parameters in a control problem
$\mathbf{r}$	position vector of a point on the ribbon's surface
$\boldsymbol{s}$	vector of slack variables
s	curvilinear abscissa (distance travelled)
s_t	independent coordinate transformation coefficient $s_t = dt/ds$
t	time
$\hat{\mathbf{t}}$	unit vector tangent to the ribbon's spine
t_f	final time
$\boldsymbol{u}$	control vector
u	longitudinal component of $\mathbf{v}$
$u_{-1}(\cdot)$	Heaviside step function
$u_{-2}(\cdot)$	ramp function
$\mathbf{v}$	absolute speed of $\mathbf{G}$ in the vehicle's body-fixed frame
v	lateral component of $\mathbf{v}$
$\mathbf{v}_m$	absolute speed of $\mathbf{C}_m$ in the vehicle's body-fixed frame
w	vertical component of $\mathbf{v}$
$\boldsymbol{x}$	state vector
x, y, z	inertial Cartesian coordinates
x_B, y_B	axes in the vehicle's body-fixed reference frame
$\mathbf{x}_m$	position of $\mathbf{C}_m$ in the vehicle's body-fixed frame
$\boldsymbol{z}$	vector of optimisation parameters

Greek Symbols

Symbol	Description
δ	steering angle
θ	absolute yaw angle of the ribbon frame
$\boldsymbol{\kappa}$	curvature vector
κ	curvature
$\boldsymbol{\lambda}$	Lagrange multipliers (or costate)
$\boldsymbol{\mu}$	vector of Lagrange multipliers for inequality constraints
μ	pitch angle
$\boldsymbol{\nu}$	vector of Lagrange multipliers for algebraic constraints
ξ	z-angle between the ribbon frame and the vehicle frame
ρ	radius of curvature
τ	torsion
ϕ	roll angle
χ	yaw angle of the vehicle relative to the track's spine
ψ	yaw angle
$\boldsymbol{\omega}$	angular velocity vector
$\bar{\boldsymbol{\omega}}$	angular velocity vector of the vehicle
$\omega_{..}$	wheel angular speed
$\boldsymbol{\Omega}$	generalised curvature vector ($\boldsymbol{\Omega}=\boldsymbol{\omega}\, dt/ds$)
Ω_x	ribbon's relative torsion
Ω_y	ribbon's normal curvature
Ω_z	ribbon's geodesic curvature

Acronyms

Acronym	Expansion
3D	Three-Dimensional
BPF	Final Ballistic Parabola
BPI	Initial Ballistic Parabola
BVP	Boundary-Value Problem
CV	Calculus of Variations
DAE	Differential-Algebraic Equation
EOM	Equation Of Motion
ERS	Energy-Recovery System
ES	Energy Storage
F1	Formula 1
GPS	Global Positioning System
HJBE	Hamilton-Jacobi-Bellman Equation
ICE	Internal-Combustion Engine
ICLOCS	Imperial College London Optimal Control Software [22]
IPOPT	Interior Point OPTimiser [87]
IVP	Initial-Value Problem
LQ	Linear Quadratic
KERS	Kinetic Energy-Recovery System
KKT	Karush-Kuhn-Tucker
MGU-H	Motor-Generator-Unit-Heat
MGU-K	Motor-Generator-Unit-Kinetic
MPC	Model-Predictive Control
MR	Multiplier Rule
NLP	NonLinear Programming
OC	Optimal Control
ODE	Ordinary Differential Equation
PMP	Pontryagin Maximum Principle
RK	Runge-Kutta

List of Figures

2.1 Numerical optimal control through transcription. 28

3.1 Original definition of the brachistochrone problem, published by Johan Bernoulli in June 1696 in the 'Acta Eruditorum' [3]. 58
3.2 The classical brachistochrone problem. 60
3.3 The unrestrained brachistochrone problem. The tangential force F_t is zero in the point-mass case. Any symbol associated with a vector represents the vector magnitude. 63
3.4 General form of the solution to the unrestrained brachistochrone problem. The initial ballistic parabola, labelled BPI, is determined by the initial velocity v_i and the terminal ballistic parabola, labelled BPF, is determined by the terminal velocity v_f. The cycloid's directrix is at an elevation for which the bead's kinetic energy is zero. The intersection point for the free-fall parabolae is $\mathbf{P}_x$ and the cycloid tangent to both parabolae has points of tangency $\mathbf{P}_a$ and $\mathbf{P}_b$. 65
3.5 The unrestrained brachistochrone problem. The point mass has an initial speed of 5 m/s and a launch angle of 45^o (relative to the x-axis). There are two alternative terminal points at (10 m, -1 m) and (10 m, 0 m), with terminal angles of 30^o. The green diamonds mark the transition points between the ballistic parabolae and the central cycloid. The transit times in the two cases are 2.2265 s and 1.9791 s respectively. 67
3.6 Sensitivity of the optimal solution to small changes in the boundary conditions (see [75]). The start of the trajectory is $\mathbf{P}_i$ and the ballistic parabola BPI is determined by the initial velocity. The points $\mathbf{P}_{f1}$ and $\mathbf{P}_{f2}$ are alternative terminal points in close proximity falling on either side of BPI. In the case of $\mathbf{P}_{f1}$ impulsive control in the vicinity of $\mathbf{P}_i$ selects the optimal trajectory Γ_1, while in the case of $\mathbf{P}_{f2}$ BPI is used in combination with a cycloidal subarc. In both cases terminal impulsive control is used to meet the final angle condition. 69
3.7 The unrestrained rolling-body brachistochrone problem. A rolling body, with α=2, has an initial speed of 5 m/s and an initial launch angle of 45^o (relative to the x-axis). There are two alternative terminal points at (10 m, -1 m) and (10 m, 0 m), with a terminal entry angle of 30^o. Green diamonds mark the transition points between Gaussian hypergeometric functions and the central cycloid. The transit times in the two cases are 2.2300 s and 2.0908 s respectively. 77

3.8 Disc rolling on the interior surface of a hemisphere. 78
3.9 Contact geometry for a disc rolling on the inner surface of a sphere. 80
3.10 Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m, 0 m) to (x, y)=(0 m, 4 m). The initial roll angle is zero with the initial yaw angle a free decision variable. There is no externally applied steering torque. The optimal transfer time is 2.5614 s. The coloured cartoons show the disc at equispaced instants in time (from light red to dark blue). The solid black line is the segment of the great circle that joins the starting and terminal points. The roll and yaw angles are constant and equal to, respectively, 0 rad and $-\pi/4$ rad. 82
3.11 Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial yaw and roll angles are free decision variables. The initial spin rate is zero and there is no external steering. The resulting transfer time is 2.4163 s. On the right, the roll and yaw angles are plotted against time, in the black and grey curves respectively. 84
3.12 Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial yaw and roll angles are free with an initial spin rate of 100 rad/s. There is no external control and the transfer time is 0.6152 s. 85
3.13 Projections on the ground plane of the optimal trajectories of the disc's mass centre computed for different initial spin rates. The line segment (dot-dashed black line) and the approximated geodesic (dashed black curve) are shown as limit references in the figure. 86
3.14 Disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial angular rates are zero, with the initial yaw and roll angles free decision variables. A stage cost $L(t) = 10^{-2} \cdot M(t)^2$ results in a transition time of 2.1018 s. A three-dimensional plot of the disc's trajectory is shown on the left. On the right, the normalised steering rate ω_s (projection of the disc's angular speed onto the contact radius, with scaling factor 120.5 rad/s) and the normalised steering moment M (scaling factor 3.307 Nm) are plotted against time in the black and grey curves respectively. 87
3.15 The disc's energy (left) and the ground-plane projection of the mass centre (right). The dark blue plots correspond to no external steering, the green plots correspond to a stage cost of $M^2(t)$, and the light red plots correspond to a stage cost of $10^{-2} \cdot M^2(t)$. In the energy diagram, the solid curves represent the total stored energy of the disc, while the dashed and dashed-dotted curves are, respectively, the kinetic and the potential energy. The vertical dashed lines identify the transition times. 88

4.1 A ribbon $\mathcal{R}$ supported by a spine curve $\mathcal{C}$. 91
4.2 Three-dimensional ribbon for the extended lemniscate of Bernoulli. The solid and dashed lines are the optimised boundaries and centre line, respectively. The dots represent the noisy simulated boundary-point measurements on which the solution is based. The ribbon grey scale represents the z-coordinate of the optimised centre line and is calibrated according to the colour bar given. 108

4.3 Parameters for the extended lemniscate ribbon. The theoretically correct values are the red dashed curves, while the optimal estimates are the solid black curves. The top plot shows the relative torsion Ω_x, the second the normal curvature Ω_y, and the third the geodesic curvature Ω_z. The distances between the ribbon boundaries and its spine are shown in the bottom diagram. The theoretical boundaries come from (4.40). 110
4.4 Ribbon-based model of the Circuit de Catalunya. The solid lines are the optimised track boundaries. The track spine's z-coordinate is shown using a grey scale. 111
4.5 Magnified views of corner (10) (left-hand side), and corners (14) and (15) (right-hand side). The measured boundary points are shown as black dots, with the dashed red curve the spine estimate. 111
4.6 Estimated curvature of the Circuit de Catalunya. The plots show, respectively, the relative torsion Ω_x, the normal curvature Ω_y and the geodesic curvature Ω_z. The optimal model-based estimates are shown as the solid black curves, while the finite-difference approximations are the red dashed curves. 112
4.7 Distances to the right-hand (solid black) and left-hand (dashed red) boundaries of the track as measured from the track's spine. 113
4.8 Error estimates for the track boundaries (as calculated in the ribbon tangent plane). The solid black curve corresponds to the right-hand-side boundary, while the dashed red curve corresponds to the left-hand boundary.113

5.1 Side view of a Formula 1 car showing its mass centre $\mathbf{C}_m$, its centre of pressure $\mathbf{C}_p$ and its geometric centre $\mathbf{G}$. The other symbols refer to vehicle parameters described in Table B.2 120
5.2 Curvilinear-coordinate-based description of a track segment $\mathcal{R}$. The independent variable s represents the elapsed centre-line distance, with ρ the radius of curvature and n_l and n_r defining the track width; x and y represent the inertial-frame axes. The arrows used for angles start from the corresponding reference direction. Angles are positive when measured clockwise. 121
5.3 Plan view of a Formula 1 car with its basic geometric parameters. The body-fixed axes x_B and y_B are in the road plane. 125
5.4 Tyre force system. 126
5.5 Plan view of the Circuit de Catalunya (current Spanish Formula 1 Grand Prix). The start-finish line is labelled 'SF'; the corners are numbered from start to finish (this is a clockwise circuit). 137
5.6 Geodesic curvature for the plane projection of the Circuit de Catalunya. The labels are consistent with Figure 5.5. 138
5.7 Car speed as a function of the distance travelled from the start-finish line on the Barcelona Formula 1 track. The track model is the one given in Figures 5.5 and 5.6. The corner numbers correspond to those given in the same figures. 140

5.8 Optimal trajectory of the car's geometric centre through turns (1), (5), (10) - (11) and (14) - (15). The markers associated with different points in the corners are as follows: The start of the manoeuvre is marked with a ◯; the start of the braking phase is marked with a △; the start of the cornering phase is marked with a □; the ◊ marks the beginning of the corner-exit acceleration phase; the ▽ symbol marks a reversal in the direction of the tyre side forces, while the ☆ is the end of the manoeuvre. The marker points defined for turn (1) correspond to specific transition points in the tyre forces shown in Figure 5.9. 141
5.9 Tyre forces in turn (1). The first column gives the longitudinal and lateral tyre forces for the left-hand wheels, with the front-left wheel at the top. The longitudinal forces are on the vertical axes, while the side forces are on the horizontal axes. The second column gives the longitudinal and lateral tyre forces for the right-hand wheels, with the front-right wheel at the top. The third column gives the normal loads as functions of the distance travelled from the start-finish line, with the front wheels at the top; the left-hand wheels are the solid lines, while the right-hand wheels are the dashed lines. 142
5.10 Sensitivity of the minimum lap time to arc length mesh size. The solid black line shows the optimal lap time as a function of the arc length sampling distance (on a semi-log scale). The dashed blue line shows the NLP solution time (on a log-log scale) as a function of the arc length sampling distance. The red line is the approximate computation time given by $t_{NLP} \approx 2 \cdot 10^{-3} N^{1.65}$. 143
5.11 Yaw-mode stability as a function of the location of the aerodynamic centre of pressure. The plot shows the real parts of the yaw-mode eigenvalues as functions of the distance between the front axle and the centre of pressure (see Figure 5.1); negative real parts are displayed in blue, while positive ones are red. The model is linearised at a constant speed of 65 m/s in a corner with radius of curvature 150 m, with the aerodynamic centre of pressure moved towards the front axle at 0.0033 m/s. 144
5.12 Minimum lap time as a function of the centre of mass position a, the position of the centre of pressure a_a, the roll balance d_r, the differential viscosity k_d and the vehicle mass m. The bottom-right plot is a magnified view of the one just above, representing variations in k_d. These curves were computed by varying each parameter in isolation. The black dashed lines correspond to the nominal configuration and its corresponding lap time, while the dashed red lines correspond to optimised parameter values. . . . 146
5.13 Speed profiles for the nominal (red line) and optimised (blue line) parameter sets. The black line shows the time gain for the optimised vehicle against distance travelled from the start-finish line. The left-hand y-axis gives the speed, while the right-hand y-axis is the time gained by the optimised car. 148
5.14 Change in the racing line in turn (4) due to parameter variations. The trajectory for the nominal car is shown in red, while the trajectory for the optimised car is shown in blue. 149

6.1 Car aerodynamic maps: the drag coefficient c_d is the green curve; the downforce coefficient c_l is the blue curve; the distance of the aerodynamic centre of pressure from the front axle a_a is given by the red curve (in metres). The '$\circ$' symbols represent reference data points. 156
6.2 Speed profiles for the vehicle with constant (red line) and speed-dependent (blue line) aerodynamic coefficients. The black line shows the time gain for the vehicle with speed-dependent parameters against distance travelled from the start-finish line. The left-hand y-axis gives the speed, while the right-hand y-axis is the time gained. 156
6.3 Change in the racing line in turn (4) due to speed-dependent aerodynamic properties. The constant-parameter trajectory is shown in red, while the trajectory for the car with speed-dependent parameters is shown in blue. . 157
6.4 Elevation z and geodesic curvature Ω_z of the track against distance travelled. The elevation is given by the shaded area; the shading is consistent with Figure 4.4. The black line represents the geodesic curvature. 167
6.5 Product of the track yaw rate and camber angle against distance travelled. Positive values are indicative of positive cambering, while negative values are associated with adverse cambering. 167
6.6 Speeds and time difference for the 2D and 3D track. The red curve gives the speed of the car on the flat track, while the blue curve is the speed on the 3D track. The black curve is the time difference; positive values indicate that the car is faster on the 3D track. The left-hand y-axis gives the speed, while the right-hand y-axis refers to the time gained. 169
6.7 Changes in the racing line between turns (2) and (3) (on the left) and turn (4) (on the right) due to three-dimensional track features. The trajectory for the flat track is shown in red, while the trajectory for the 3D track is shown in blue. 170
6.8 Speed difference Δv (in black) and the gravitational force component in the car's x_B-direction F (in red). The speed difference is positive when the car is faster on the 3D track. The car is travelling down hill when the gravitational force component is positive. 170
6.9 Speed difference Δv (in black) and the centripetal force in the car's z-direction F (in red). The speed difference is positive when the car is faster on the 3D track. The centripetal force provides additional downforce when it is positive. 171
6.10 Operating regime of a KERS. The ordinate represents the KERS power flow P_k, while the abscissa represents the power P_m delivered to the rear wheels. 173
6.11 Speeds and time difference for the vehicle with and without KERS. The red curve gives the speed of the car with no KERS, while the blue curve is the speed using KERS. The (solid) black curve is the time difference; positive values indicate that the car is faster with KERS. The left-hand y-axis gives the speed, while the right-hand y-axis refers to the time gained. 174

6.12 KERS optimal control for a race lap on the three-dimensional Circuit de Catalunya. The black curve shows the vehicle's speed; the red curve is the power delivered by the rear wheels; the blue curve is the power to/from the KERS (the KERS is propelling for positive values of power); the green curve is the stored energy; the yellow curve is the energy released from the KERS. The units on the left-hand side y-axis refer to speed and power; energy values appear on the right-hand side. 176
6.13 KERS optimal control for a race lap on the plane projection of the Circuit de Catalunya. The notation is consistent with Figure 6.12 177
6.14 Operating regime of the 2014 engine and thermal ERS at full power; power absorbed by the MGU-H is deemed positive. 179
6.15 Kinetic and Thermal ERS optimal control for a race lap on the three-dimensional Circuit de Catalunya. The solid black curve shows the vehicle's speed; the solid red curve is the power delivered by the rear wheels; the blue curve is the power to/from the MGU-K (the MGU-K is propelling for positive values of power); the cyan curve is the power to/from the MGU-H (the MGU-H generates energy for negative values of power); the purple line represents the state of the waste gate (1 is open); the green curve is the stored energy; the yellow curve is the energy transferred from the energy store to the MGU-K, while the orange curve is the energy transferred from the MGU-K to the energy store; the dashed black curve is the mass of fuel consumed; the dashed red curve is the normalised fuel rate. The units on the left-hand side y-axis refer to speed, power, waste-gate opening and normalised fuel rate; energy values and fuel mass appear on the right-hand side. 184
6.16 Kinetic and Thermal ERS optimal control for a qualifying lap on the three-dimensional Circuit de Catalunya. The notation is equivalent to the one in Figure 6.15 185

A.1 Longitudinal tyre force versus longitudinal slip. The left-hand diagram considers five normal loads (at $\alpha = 0°$); the lightest yellow curve corresponds to 1 kN with the normal load then increased in steps of 1 kN. The right-hand diagram considers combined slip for five values of side-slip angle at a normal load of 2 kN; the darkest red curve corresponds to a slip angle of 0° with the slip angle increased in steps of 5°. 201
A.2 Modulus of the tyre force as a function of the longitudinal slip κ and the slip angle α for a normal load of 2 kN. The right-hand diagram shows contours of equal tyre-force (in kN). 201

List of Tables

3.1 Solution and trajectory types for (3.34). The period t_{cyc} is the time spent on the cycloidal subarc and is given by (3.13). The $\bar{\omega}_i$ and $\bar{\omega}_f$ columns refer to the possible need for impulsive control at the initial and/or final points respectively. When impulsive control may be required, the corresponding table entry is marked with a '+' symbol. 70

5.1 Approximate distances to mid-corner on the Circuit de Catalunya (from the start-finish line). 138
5.2 Independently optimised parameter values with corresponding lap times. . 147
5.3 Parameter values from a multi-parameter optimisation study. The resulting lap time is 80.178 s. 148

B.1 Tyre friction parameters. 202
B.2 Vehicle parameters; values marked with an asterisk relate to the 2014 car. . 203
B.3 Polynomial coefficients for the speed-dependent aerodynamic model. 203

CHAPTER 1

Introduction

THERE is an elegant parallel between the principle of stationary action in physics and the necessary conditions for optimality in optimal control. Both involve the search for a stationary point in the integral of a Lagrangian function, which in mechanics is known as the *action* integral. Combining the two fields is the key to efficiency and high performance in mechanical systems. This thesis is focused on the application of optimal control to vehicular mechanics. The intent is two-fold: firstly to identify general techniques for the numerical solution of such problems; secondly to find and analyse the solutions to the aforementioned applications.

The overarching theme is the solution of minimum-lap-time problems for Formula 1 cars racing on three-dimensional circuits. A renewed interest in this sport is born by the recently updated 'environment-friendly' technical regulations, that downsize the engines in favour of more powerful motor-generator units, and that allow energy recovery from exhaust gases in addition to braking [51]. Formula 1 cars are becoming prototypes of high-performance hybrid vehicles, and technical developments of hybrid technology will be likely to reach production vehicles once mature and proven effective.

Within the automotive context, the remarkable feature of Formula 1 is the concurrent and extreme pursuit of maximum performance and efficiency, where small gains attract large returns, and solutions must be available in a short time. Optimal control is the natural framework to study minimum-time problems for dynamic systems. The solution

apparatus proposed in this thesis provides reference lap times for any given vehicle-track pair, and can be used to search optimal vehicle configurations that maximise performance for specific circuits. The optimal control strategy is calculated, which includes the driven line and the associated driver control — drive, brake and steer. Following the introduction of hybrid energy-recovery systems and restrictions on fuel usage, energy management becomes an issue of key importance; the optimal allocation of driving and braking power across the available powertrain devices (i.e. internal combustion engine, motor-generator units, brakes, etc.) is crucial to guarantee energy efficiency while retaining high performance.

The work in this thesis is described in a series of papers, which have either been published or are currently being reviewed. A list of these papers can be found in the Publication section at the beginning of this document.

The aim of this introduction is to give an outline of the topics covered in the thesis, and to identify the gaps in the current scientific knowledge addressed herein. Due to the multi-disciplinary nature of this work it is convenient to set out the specialised literature review in the preamble to each of the subsequent chapters. The general conclusions are drawn in Chapter 7, with some of the possible developments of this work proposed in Chapter 8.

The first part of Chapter 2 covers the basic theory of optimisation and optimal control. Optimal control is presented as an infinite-dimensional extension of optimisation, and the first-order necessary conditions for optimality are derived for both. The Euler-Lagrange equations, obtained as necessary conditions for optimality, reveal the parallel with the Lagrangian approach for the derivation of the equations of motion in mechanics (see [43]).

It is often hard to obtain closed-form solutions to optimal-control problems when the controlled system is nonlinear. In many cases the only feasible route to a solution is via numerical approximations: the optimal-control problem is first reduced to a finite-dimensional optimisation problem using integral-approximation techniques. Numerical

optimisation methods are then invoked to search for a solution to the approximated problem. This topic is briefly covered in the second part of Chapter 2. A variety of alternative approaches are reviewed, which include direct and indirect optimisation methods in combination with shooting, or collocation approaches for discretisation. Each method is qualified with advantages and disadvantages, and the choice of direct collocation for subsequent applications is motivated. The second chapter, apart from being a concise overview of existing background theory, aspires to present and highlight the dependencies amongst the necessary tools used in the subsequent numerical investigations. A novel perspective on the equivalence between direct and indirect collocation methods for numerical optimal control is given, which considers the Euler rule for time discretisation.

Ideally, a solution method should work regardless of the system to be controlled. In practice, distinct methods can behave in different ways depending on the model and on the modelling approach; some of these issues are explained in Chapter 2. It is widely recognised that simulation and control problems associated with mechanical systems can be significantly affected by the modelling process: a problem as simple as the analysis of the motion of a pendulum can become much harder if the mechanism is described using alternative coordinate systems (i.e. using Cartesian in place of polar coordinates). An efficient model is crucial within optimal control applications; it must include all, and only, the features strictly relevant to the problem analysis. When a numerical method is to be applied to a model, it is necessary to ensure that the two are compatible; besides, the knowledge of numerical methods for optimal control is fundamental in order to apply them efficiently.

The higher-complexity models presented throughout this thesis are assembled from simpler sub-systems. For instance full vehicular models include friction laws and dynamics of rolling bodies; these sub-systems are first analysed independently to allow for a careful selection of the relevant effects to be included in a full vehicle model, that is subsequently used in optimal control calculations. Some studies on the motion of rolling bodies are presented in Chapter 3, which are also used as validation benchmarks for numerical

optimal control methods. The material in this chapter is published in [67]. Starting from the classical brachistochrone problem and some of its variants, a series of problems are defined for increasingly complex systems of nonholonomic rolling bodies; the recurring theme is the optimal exchange between potential and kinetic energy. These studies evince that, in the context of minimum-time vehicle control, the influences of changes in potential and kinetic energy are crucial, which suggests the need for three-dimensional road models. At the same time the gyroscopic and rotational inertial effects acting on rolling bodies are not as influential, especially considering the low ratio between the wheel inertia and the total vehicle mass.

One of the sub-systems considered is expanded further in Appendix A, which describes a road-tyre friction model based on empirical formulae. This model was used in [35]; it is based on the well-known Pacejka Magic Formulae (see [55]) and was successfully employed in vehicle optimal-control applications. The longitudinal and lateral components of the road-tyre friction force are functions of the slip angle, the slip coefficient and the tyre normal load, accounting for the effects of combined longitudinal and lateral slip, friction force saturation and dependency of the friction peak on tyre load. Although alternative friction models were considered (see [63]), the model in Appendix A represents a trade-off between accuracy and simplicity that suits the overall modelling accuracy and the requirements herein dictated by the application of numerical optimal control methods.

Recognising the tyre friction dependency on load, any effect that influences the normal force at the road-tyre interface becomes critical to performance maximisation. Phenomena such as the weight transfer across the wheels due to longitudinal and lateral accelerations, or the aerodynamic downforce and its distribution across the axles, must be included in the vehicle model.

The road geometry is another source of variations in tyre load: humps and dips affect the road-tyre normal load due to apparent inertial forces; the same happens when a vehicle drives through a cambered corner. In order to capture these phenomena, the road model must account for the three-dimensional nature of the road surface. Three-

dimensional road modelling in relation to minimum-lap-time studies has not yet been observed in the literature. Prior vehicle-optimal-control applications were limited to flat track models; Chapter 4 studies a new modelling technique for three-dimensional roads. Roads are abstracted as ribbons in solid space using a differential geometric description of the surface, that associates local curvature and width to the distance travelled along the road. This approach is an extension of the track description based on curvilinear coordinates employed in [4] and [19] for the optimal control of racing vehicles. A novel method is proposed for track identification based on GPS data relating to the position of the road boundaries. Optimal-control techniques are exploited to address problems associated with integration drift and track closure previously recognised in the literature (see [35]). These 3D track modelling and identification techniques are covered in [65].

Chapter 5 describes a basic Formula 1 car model for the solution of minimum-lap-time problems on flat tracks: this model includes a rigid chassis with 3 degrees of freedom and the tyre friction model described in Appendix A; other model features are included, such as aerodynamic forces, load transfer distribution, a viscous differential and equal pressure at the brake callipers. This chapter also explains the particular modelling corrections introduced to improve the effectiveness of the numerical optimal-control methods: the problem scaling based on non-dimensionalisation of physical units is of pivotal importance; non-smooth features are replaced by smooth approximations in order to guarantee the existence of function derivatives used by gradient-descent numerical methods. As a result, a significantly reduced computation time is achieved for the minimum-lap-time calculations compared to that reported in prior work (see [18, 35]).

A convenient feature of direct optimal-control methods is that model parameters can be optimised alongside state and control trajectories. Some of the vehicle parameters are first studied independently, considering changes from their nominal values for one parameter at the time. The goal is to find the values that achieve the minimum lap time. The sensitivity of the lap time with respect to independent parametric changes is calculated too. The parameters are subsequently optimised in concert in order to improve

the vehicle performance even further. These unprecedented studies are intended to show that part of the prototyping phase of a vehicle can be conducted offline through optimal-control simulations, with the possibility of concurrent optimisation of multiple parameters. The exploitation of this last feature could speed-up the time-consuming iterative process that is often employed for vehicle tuning. Minimum-lap-time and parameter-optimisation calculations for the basic model given in Chapter 5 reproduce the results obtained in [64], which were based on a slightly different model parametrisation.

The basic vehicle model is extended in Chapter 6. The first extension applies to the aerodynamic model, accounting for speed-varying aerodynamic features. The vehicle dynamics are then augmented for the use on a three-dimensional track. New dynamic influences appear, such as centripetal forces in the vertical direction of the vehicle, and gravitational forces in the longitudinal and lateral directions. Three-dimensional studies of this type have not yet appeared in the known literature, and are currently being considered for publication in [46].

The last original extensions allow for the inclusion of energy-recovery systems allowed by present — kinetic energy-recovery systems in 2013 — and forthcoming — kinetic and heat energy-recovery systems from 2014 — Formula 1 regulations (see [50, 51]). These devices add another layer of optimisation, that is the administration of the stored and recoverable energy (both kinetic and from exhaust heat). Optimal-control calculations are used to estimate the management of the energy storage (i.e. the battery) under different operating conditions. Alternative scenarios are considered, that correspond to race and qualifying laps. In the first case the energy level at the end of the lap must coincide with that at the beginning, in order to guarantee the repeatability of the lap. During qualification, the available energy can be deployed prodigally without constraining relations between the initial and final energy charge. The section relating to ERS control problems is based on the studies appearing in [47], and extends them to account for three-dimensional tracks; the numerical results reported here are in broad agreement with the ones given in [47], which were obtained using an alternative numerical optimal-control

method, that is pseudospectral direct collocation.

The nominal tyre and vehicle parameters used for the optimal-control calculations can be found in Appendix B. An alternative three-dimensional modelling procedure is developed in Appendix C to investigate possible improvements in the simulation accuracy.

It will be demonstrated that minimum-lap-time problems, inclusive of parameter optimisation, can now be solved for realistic road and vehicle models, without resorting to approximation techniques for nonlinear dynamics. The time taken to perform these calculations on a desktop PC is within an order of magnitude of the time taken to drive the lap in real-world F1 circuits. This research represents significant progress in the field of vehicular control.

CHAPTER 2

Numerical Optimal Control

"Nothing at all takes place in the universe in which some rule of maximum or minimum does not appear.", Leonhard Euler.
"... optimal control and mathematical programming problems are essentially one and the same thing ..." , [16].

SOME readers may be familiar with the theory of optimal control and its connection with the calculus of variations. These subjects have motivated countless chapters in textbooks, theses or scholarly papers, and there is little to be said that cannot be found in the literature. The *Calculus of Variations* (CV) and *Optimal Control* (OC) have a long history that started in the 17^{th} century. The history of CV and OC is a captivating subject thoroughly covered by surveys such as the ones by Bryson, Goldstine, and Sussmann and Willems (see [12], [27] and [78]).

One purpose of this work is to review and analyse existing numerical methods for OC, which is a vast topic on its own; by no means is this chapter intended to cover numerical OC entirely, and references to the literature will be given when appropriate. The discussion will be restricted to the concepts in the CV and OC theory that are strictly necessary or relevant to this specific field. Inevitably, some fundamental results from the theory of *optimisation* and *numerical integration* will be introduced to fulfil a need for a self-contained dissertation.

This chapter is intended to portray the development of *numerical methods* for OC that followed the appearance and evolution of digital computers. The duality between

the *direct* and *indirect* approaches will be described as a fundamental result in the advance of such methods, ascribing the merit of their success to the efficiency of modern numerical integration schemes and optimisation algorithms.

Attention is given to the relationship between problems of functional optimisation, i.e. problems in the CV, and problems of parametric optimisation, which lend themselves as finite-dimensional approximations of the former. From a reversed perspective (see [13]), problems in the CV can be interpreted as limit cases of parameter-optimisation problems when the number of optimisation variables tends to infinity. This thesis considers the OC of systems governed by continuous-time Differential-Algebraic Equations (DAEs) as the fundamental problem; any finite-dimensional approximation is introduced as a means to find approximate solutions.

The following sections provide the building blocks of standard numerical OC techniques, and justify the particular choice of the family of methods applied within this thesis. Section 2.1 introduces the parameter optimisation theory, in a way that is suited to the formulation of discrete-time OC problems. Section 2.2 presents the fundamental results in the CV to provide optimality conditions for continuous-time OC problems. A concise derivation of Lagrangian mechanics from the application of the CV is given in Section 2.2.1. Section 2.3 sketches a series of numerical methods, starting from the solution of optimisation problems (in Section 2.3.1) and numerical integration problems (in Section 2.3.2), to eventually synthesise the aforementioned topics and compare the direct and indirect OC approaches in Section 2.3.4.

2.1 Parameter Optimisation

Parameter optimisation consists of searching for an *extremum* value of a real-valued function of a finite number of numerical parameters. In the context of optimal control such a function is often called the *objective function*. The parameters can be concisely grouped

in a vector representation

$$\boldsymbol{z} = \begin{bmatrix} z_1 \\ \vdots \\ z_{\underline{z}} \end{bmatrix} \in \mathcal{D}, \tag{2.1}$$

where the underlined symbol $\underline{z}$ is the *dimension* of the decision vector $\boldsymbol{z}$, and $\mathcal{D}$ is the parameter *domain*. The objective function is

$$J(\boldsymbol{z}) \ : \ \mathcal{D} \to \mathbb{R}, \tag{2.2}$$

which maps the parameter domain into the real numbers. Heretofore the set $\mathcal{D}$ is assumed to be $\mathbb{R}^{\underline{z}}$, and $J(\boldsymbol{z})$ to be twice-continuously differentiable on $\mathcal{D}$.

A common parameter-optimisation problem is the problem of finding the vector of real parameters that attains the minimum[1]

$$\min_{\boldsymbol{z} \in \mathbb{R}^{\underline{z}}} J(\boldsymbol{z}); \tag{2.3}$$

for an arbitrary nonlinear function $J(\boldsymbol{z})$ the problem (2.3) is called *unconstrained optimisation*. In minimisation problems $J(\boldsymbol{z})$ is also called the *cost* function.

The decision vector identifies a *point* in the parameter domain. A *stationary point* $\check{\boldsymbol{z}}$ of $J(\boldsymbol{z})$ is one at which

$$\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}}) := \frac{\partial}{\partial \boldsymbol{z}} J(\check{\boldsymbol{z}}) = 0, \qquad \text{that is} \qquad \frac{\partial}{\partial z_i} J(\check{\boldsymbol{z}}) = 0, \quad \forall i = 1, \dots, \underline{z}, \tag{2.4}$$

where the ∂ symbol denotes a partial derivative. A stationary point can be a *minimum*, a *maximum*, a *saddle point* or a *point of inflection*; a full characterisation requires higher order derivatives.

A *local minimum* of $J(\boldsymbol{z})$ is a point $\check{\boldsymbol{z}}$ for which the value of $J(\check{\boldsymbol{z}})$ is not larger than $J(\boldsymbol{z})$ for any $\boldsymbol{z}$ in an arbitrarily small neighbourhood of $\check{\boldsymbol{z}}$; formally, an $\epsilon > 0$ exists such that

$$J(\check{\boldsymbol{z}}) \leq J(\boldsymbol{z}), \ \ \forall \boldsymbol{z} \in \mathbb{R}^{\underline{z}} : ||\boldsymbol{z} - \check{\boldsymbol{z}}|| < \epsilon. \tag{2.5}$$

The symbol $||\boldsymbol{z}|| = \sqrt{\boldsymbol{z}^T \boldsymbol{z}} = \sqrt{\sum_i^{\underline{z}} z_i^2}$ represents the Euclidean norm of $\boldsymbol{z}$.

[1] A maximum problem can, in general, be defined as a minimum problem inverting the sign of the objective function.

A minimum is *global* if the performance function is not larger than at any other point in the parameter domain:

$$J(\check{\boldsymbol{z}}) \leq J(\boldsymbol{z}), \quad \forall \boldsymbol{z} \in \mathbb{R}^{\underline{z}}. \tag{2.6}$$

If in (2.5) or in (2.6) the strict inequality $J(\check{\boldsymbol{z}}) < J(\boldsymbol{z})$ holds for all $\boldsymbol{z} \neq \check{\boldsymbol{z}}$, then the minimum is *strict*.

When $J(\boldsymbol{z})$ is convex — i.e. $J\big(t\,\boldsymbol{z}_1 + (1-t)\boldsymbol{z}_2\big) \leq tJ(\boldsymbol{z}_1) + (1-t)J(\boldsymbol{z}_2), \quad \forall \boldsymbol{z}_1, \boldsymbol{z}_2 \in \mathbb{R}^{\underline{z}}, \quad \forall t \in [0,1]$ — a stationary point is a local minimum, and a local minimum is also a global minimum; if $J(\boldsymbol{z})$ is strictly convex any minimum is strict and unique (see [6]).

From differential calculus it is known that a necessary condition for a point $\check{\boldsymbol{z}}$ to be a minimum is that it be a stationary point, and that the second-order inequality condition

$$\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}}) := \frac{\partial^2}{\partial \boldsymbol{z}^2} J\left(\check{\boldsymbol{z}}\right) \geq 0 \tag{2.7}$$

be satisfied, where the ≥ 0 matrix-inequality condition means that the matrix is *positive-semi-definite*, i.e. al the eigenvalues are ≥ 0.

The stationary-point condition (2.4) and

$$\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}}) > 0, \tag{2.8}$$

together are *sufficient* for $\check{\boldsymbol{z}}$ to be a strict local minimum. The condition (2.8) means that $\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}})$ is *positive-definite*, i.e. all the eigenvalues of $\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}})$ are > 0.

A stationary point at which $\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}}) = 0$ holds element-wise is known as a *singular* point. It follows from the necessary condition (2.7) that if $J(\boldsymbol{z})$ is linear or affine in $\boldsymbol{z}$ there are no stationary points.

2.1.1 Equality Constraints

The cost minimisation might be subject to constraints on the decision parameters. Such constraints define the *feasible region* in the parameter domain. A *feasible point* is one that satisfies all the constraints.

Considering *equality constraints*, the optimisation problem becomes:

$$\begin{aligned} &\min_{\boldsymbol{z} \in \mathbb{R}^{\underline{z}}} J(\boldsymbol{z}) \\ \text{subject to} \quad &\mathbf{f}(\boldsymbol{z}) = 0, \end{aligned} \tag{2.9}$$

where $\mathbf{f}(\boldsymbol{z})$ is a vector-valued function mapping $\mathbb{R}^{\underline{z}}$ into $\mathbb{R}^{\underline{f}}$, with $\underline{f}$ the number of equality constraints; in short $\mathbf{f}(\boldsymbol{z}) : \mathbb{R}^{\underline{z}} \to \mathbb{R}^{\underline{f}}$. In this thesis the constraint function $\mathbf{f}$ is assumed to be twice-continuously-differentiable.

A point $\check{\boldsymbol{z}}$ where the rows of $\nabla_{\boldsymbol{z}}\mathbf{f}(\check{\boldsymbol{z}})$ are linearly independent is called a *regular* point. If any feasible point is regular, it follows that $\underline{f} \leq \underline{z}$, i.e. the number of equality constraints is not greater than the number of parameters. An optimisation problem is called regular when all the feasible points are regular.

In the constrained case, the definition and characterisation of a minimum follow from (2.5) and (2.6) where $\mathbb{R}^{\underline{z}}$ is replaced by the feasible region.

At a regular point $\check{\boldsymbol{z}}$ the rows of the Jacobian of the equality constraints represent a basis $\mathcal{B}_{\perp}(\check{\boldsymbol{z}})$ of the linear vector space orthogonal to the constraints

$$V_{\perp}(\check{\boldsymbol{z}}) = \left\{ \nabla_{\boldsymbol{z}}^T \mathbf{f}(\check{\boldsymbol{z}})\boldsymbol{\lambda} : \ \boldsymbol{\lambda} \in \mathbb{R}^{\underline{f}} \right\}. \tag{2.10}$$

For $\check{\boldsymbol{z}}$ to be a minimum, the *descent direction* $-\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}})$ must be in $V_{\perp}(\check{\boldsymbol{z}})$; otherwise a reduction of the cost would be feasible.

The *Lagrange Multiplier Theorem* is derived from these arguments, and defines necessary conditions for optimality[2].

Theorem 2.1 ([6, page 278]). *If a regular feasible point $\check{\boldsymbol{z}}$ is a local minimiser of the performance $J(\boldsymbol{z})$, then a unique vector $\check{\boldsymbol{\lambda}}$ exists, such that*

$$\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}}) + \check{\boldsymbol{\lambda}}^T \nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}}) = 0. \tag{2.11}$$

In addition, if $J(\boldsymbol{z})$ and $\mathbf{f}(\boldsymbol{z})$ are twice-continuously-differentiable, then

$$\mathbf{q}^T \left(\nabla_{\boldsymbol{z}\boldsymbol{z}} J(\check{\boldsymbol{z}}) + \sum_{i=1}^{\underline{f}} \check{\lambda}_i \nabla_{\boldsymbol{z}\boldsymbol{z}} \mathrm{f}_i(\check{\boldsymbol{z}}) \right) \mathbf{q} \geq 0, \qquad \forall \mathbf{q} \in V(\check{\boldsymbol{z}}), \tag{2.12}$$

where $V(\check{\boldsymbol{z}})$ is the subspace of first-order feasible variations

$$V(\check{\boldsymbol{z}}) = \left\{ \mathbf{q} \in \mathbb{R}^{\underline{z}} : \nabla_{\boldsymbol{z}}^T \mathbf{f}(\check{\boldsymbol{z}})\mathbf{q} = 0 \right\}. \tag{2.13}$$

[2]Proofs of the optimality conditions are omitted from this thesis; the textbook [6] thoroughly covers the parameter optimisation theory.

The vector $\boldsymbol{\lambda}$ is known as the vector of *Lagrange multipliers*.

The first- and second-order necessary conditions (2.11) and (2.12) can be compactly rewritten defining the *Lagrangian*:

$$L(\boldsymbol{z}, \boldsymbol{\lambda}) := J(\boldsymbol{z}) + \boldsymbol{\lambda}^T \mathbf{f}(\boldsymbol{z}), \qquad L : \mathbb{R}^{(\underline{z}+\underline{\mathrm{f}})} \to \mathbb{R}. \tag{2.14}$$

The conditions (2.11) and (2.12) become, respectively

$$\left.\begin{aligned} \nabla_{\boldsymbol{z}} L(\check{\boldsymbol{z}}, \check{\boldsymbol{\lambda}}) = 0 \\ \nabla_{\boldsymbol{\lambda}} L(\check{\boldsymbol{z}}, \check{\boldsymbol{\lambda}}) = 0 \end{aligned}\right\} \tag{2.15}$$

$$\mathbf{q}^T \nabla_{\boldsymbol{z}\boldsymbol{z}} L(\check{\boldsymbol{z}}, \check{\boldsymbol{\lambda}}) \mathbf{q} \geq 0, \qquad \forall \mathbf{q} \in V(\check{\boldsymbol{z}}). \tag{2.16}$$

The conditions (2.15) on their own define a stationary point. In combination, (2.15) and (2.16) satisfied for a strict inequality provide sufficient conditions for $\check{\boldsymbol{z}}$ to be a strict local minimum.

The Lagrange multipliers $\check{\boldsymbol{\lambda}}$ at a regular minimum $\check{\boldsymbol{z}}$ can be derived from (2.11)

$$\check{\boldsymbol{\lambda}}^T = -\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}}) \left[\nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}})\right]^{-1} = -\nabla_{\mathbf{f}} J(\check{\boldsymbol{z}}). \tag{2.17}$$

The multipliers thus represent the sensitivity of the performance index with respect to variations of the constraints in a neighbourhood of a local minimum. A constraint variation is intended as a change in the equality constraints from $\mathbf{f}(\check{\boldsymbol{z}}) = 0$ to $\mathbf{f}(\check{\boldsymbol{z}}) = \delta\mathbf{f}$, with $||\delta\mathbf{f}||$ small. Geometrically, the Lagrange multipliers are the coordinates of the negative performance gradient $-\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}})$ expressed in the basis $\mathcal{B}_{\perp}(\check{\boldsymbol{z}})$.

If the problem is regular, the *implicit-function theorem* implies that, at each feasible point $\check{\boldsymbol{z}}$, a function $\boldsymbol{f}(\boldsymbol{u})$ exists such that the parameter vector can be partitioned into two vectors $\check{\boldsymbol{z}} = \begin{bmatrix} \check{\boldsymbol{x}} \\ \check{\boldsymbol{u}} \end{bmatrix}$ with

$$\check{\boldsymbol{x}} = \boldsymbol{f}(\check{\boldsymbol{u}}), \qquad \boldsymbol{f} : \mathbb{R}^{\underline{u}} \to \mathbb{R}^{\underline{x}}, \qquad \text{with } \underline{x} = \underline{f}. \tag{2.18}$$

The elements of the $\boldsymbol{x}$ and $\boldsymbol{u}$ partitions can be identified, respectively, as the *dependent* and the *independent* parameters of the optimisation problem. In general the choice of the partitions is not unique. When (2.18) is defined uniquely over the whole domain, the

dependent variables can be *eliminated* and the optimisation problem reformulated as an unconstrained minimisation in the independent variables:

$$\min_{\boldsymbol{u}\in\mathbb{R}^{\underline{u}}} J\left(\begin{bmatrix} \boldsymbol{f}(\boldsymbol{u}) \\ \boldsymbol{u} \end{bmatrix}\right). \tag{2.19}$$

2.1.2 Inequality Constraints

The feasible region can be further restricted by inequality constraints on the optimisation parameters. An optimisation problem subject to inequality constraints is also known as a *NonLinear Programming* (NLP) problem. A standard definition is

$$\begin{aligned} &\min_{\boldsymbol{z}\in\mathbb{R}^{\underline{z}}} J(\boldsymbol{z}) \\ \text{s.t.} \quad &\begin{cases} \mathbf{f}(\boldsymbol{z}) &= 0 \\ \mathbf{h}(\boldsymbol{z}) &\leq 0 \end{cases}, \end{aligned} \tag{2.20}$$

where $\mathbf{h}$ is a vector-valued function mapping $\mathbb{R}^{\underline{z}}$ into $\mathbb{R}^{\underline{\mathrm{h}}}$, with $\underline{\mathrm{h}}$ the number of inequality constraints; in short $\mathbf{h} : \mathbb{R}^{\underline{z}} \to \mathbb{R}^{\underline{\mathrm{h}}}$. The function $\mathbf{h}$ is assumed to be twice-continuously differentiable.

For any feasible point $\check{\boldsymbol{z}}$ each inequality constraint is satisfied either as an equality or as a strict inequality. The set of inequality constraints can thus be partitioned into two sub-sets: the *active set*, for which equalities hold, identified by the indices in $\mathcal{A}(\check{\boldsymbol{z}}) := \{i : \mathrm{h}_i(\check{\boldsymbol{z}}) = 0\}$; and the *inactive set*, for which the strict inequalities hold, identified by the indices in $\bar{\mathcal{A}}(\check{\boldsymbol{z}}) := \{i : \mathrm{h}_i(\check{\boldsymbol{z}}) < 0\}$.

At a minimum, the inactive constraints do not affect the solution. In fact, a minimum for the complete problem (2.20) is still a local minimum if the inactive constraints are removed. The necessary and the sufficient conditions defined for equality-constrained problems apply to NLP problems if the inactive constraints are neglected. A point $\check{\boldsymbol{z}}$ is regular when the rows of $\nabla_{\boldsymbol{z}}\mathbf{f}(\check{\boldsymbol{z}})$ and of $\nabla_{\boldsymbol{z}}\mathbf{h}_{\mathcal{A}}(\check{\boldsymbol{z}})$ are linearly independent[3].

At a minimum $\check{\boldsymbol{z}}$ the descent direction $-\nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}})$ must be orthogonal to all the equality constraints and the active inequality constraints. However, in this case, each active

[3]The number of inequality constraints is, in general, not limited by the number of parameters or of equality constraints. However, the regularity of a point limits the number of active inequality constraints to $|\mathcal{A}| = \underline{z} - \underline{\mathrm{f}}$, where $|\mathcal{A}|$ is the cardinality of $\mathcal{A}$.

inequality constraint would be violated only for a variation of $\boldsymbol{z}$ in the direction of the constraint's gradient. This means that the space of *infeasible orthogonal directions* is the cone

$$V_{\wedge}(\check{\boldsymbol{z}}) = \Big\{ \mathbf{q} \in \mathbb{R}^{\underline{z}} : \mathbf{q} = \nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}}) \boldsymbol{\lambda} + \nabla_{\boldsymbol{z}} \mathbf{h}_{\mathcal{A}}(\check{\boldsymbol{z}}) \boldsymbol{\mu},\ \boldsymbol{\lambda} \in \mathbb{R}^{\underline{f}},\ \boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{A}|},\ \boldsymbol{\mu} > 0 \Big\}. \tag{2.21}$$

A point in $V_{\wedge}(\check{\boldsymbol{z}})$ is a linear combination of the rows of $\nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}})$ and $\nabla_{\boldsymbol{z}} \mathbf{h}_{\mathcal{A}}(\check{\boldsymbol{z}})$, with the restriction that the coefficients for $\nabla_{\boldsymbol{z}} \mathbf{h}_{\mathcal{A}}(\check{\boldsymbol{z}})$ be positive.

At a minimum $\check{\boldsymbol{z}}$ the cost descent direction must be in $V_{\wedge}(\check{\boldsymbol{z}})$, otherwise a cost reduction would be feasible. Following these arguments, first-order necessary conditions for optimality are defined for NLP problems. The Lagrangian is defined as

$$\begin{aligned} L(\boldsymbol{z}, \boldsymbol{\lambda}, \boldsymbol{\mu}) &:= J(\boldsymbol{z}) + \boldsymbol{\lambda}^T \mathbf{f}(\boldsymbol{z}) + \boldsymbol{\mu}^T \mathbf{h}(\boldsymbol{z}) \\ &= J(\boldsymbol{z}) + \boldsymbol{\lambda}^T \mathbf{f}(\boldsymbol{z}) + \boldsymbol{\mu}_{\mathcal{A}}^T \mathbf{h}_{\mathcal{A}}(\boldsymbol{z}) + \boldsymbol{\mu}_{\bar{\mathcal{A}}}^T \mathbf{h}_{\bar{\mathcal{A}}}(\boldsymbol{z}) \end{aligned}. \tag{2.22}$$

If a feasible regular point $\check{\boldsymbol{z}}$ is a local minimiser for the problem (2.20), then unique Lagrange multiplier vectors $\check{\boldsymbol{\lambda}}$ and $\check{\boldsymbol{\mu}}$ exist, such that

$$\begin{cases} \nabla_{\boldsymbol{z}} L(\check{\boldsymbol{z}}) = 0 \\ \check{\boldsymbol{\mu}} \geq 0 \\ \check{\boldsymbol{\mu}}_{\bar{\mathcal{A}}} = 0 \end{cases}. \tag{2.23}$$

These are known as the *Karush-Kuhn-Tucker Necessary Conditions* (KKT, [40] [6, p.309]). The condition $\check{\boldsymbol{\mu}}_{\bar{\mathcal{A}}} = 0$ is equivalent to $\check{\boldsymbol{\mu}}^T \mathbf{h}(\check{\boldsymbol{z}}) = 0$, which is known as the *complementary slackness*. Under the assumption that $J(\boldsymbol{z})$, $\mathbf{f}(\boldsymbol{z})$ and $\mathbf{h}(\boldsymbol{z})$ be twice-continuously differentiable, there holds

$$\mathbf{q}^T \nabla_{\boldsymbol{z}\boldsymbol{z}} L(\check{\boldsymbol{z}}, \check{\boldsymbol{\lambda}}, \check{\boldsymbol{\mu}}) \mathbf{q} \geq 0, \tag{2.24}$$

for all $\mathbf{q} \in \mathbb{R}^{\underline{z}}$ such that

$$\nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}}) \mathbf{q} = 0, \qquad \nabla_{\boldsymbol{z}} \mathbf{h}_{\mathcal{A}}(\check{\boldsymbol{z}}) \mathbf{q} = 0. \tag{2.25}$$

Together, the conditions (2.23, 2.24, 2.25) satisfied for a strict inequality in (2.24), for $\mathbf{q} \neq 0$ and assuming $\check{\boldsymbol{\mu}}_{\mathcal{A}} > 0$, are sufficient for the qualification of a local minimiser $\check{\boldsymbol{z}}$ (see [6, p.313]). Furthermore, the minimum is strict if $\check{\boldsymbol{\mu}}_{\mathcal{A}} > 0$; this condition is called the *strict complementarity slackness*, Again, the multipliers $\check{\boldsymbol{\lambda}}$ and $\check{\boldsymbol{\mu}}_{\mathcal{A}}$ represent the sensitivity of

the optimal performance in a neighbourhood of $\check{\boldsymbol{z}}$, with respect to small variations in the equality and active inequality constraints [6, p.315].

Considering the Lagrangian (2.22), a solution of the problem (2.20) is [6, p.315]

$$\begin{aligned} \check{\boldsymbol{z}} &= \arg \min_{\{\boldsymbol{z},\boldsymbol{\lambda},\boldsymbol{\mu}\}} L(\boldsymbol{z},\boldsymbol{\lambda},\boldsymbol{\mu}) \\ \text{s.t.} \quad & \begin{cases} \mathbf{f}(\boldsymbol{z}) &= 0 \\ \mathbf{h}(\boldsymbol{z}) &\leq 0 \\ \boldsymbol{\mu}^T \mathbf{h}(\boldsymbol{z}) &= 0 \\ \boldsymbol{\mu} &\geq 0 \end{cases} . \end{aligned} \tag{2.26}$$

Although this redefinition per se does not provide an explicit solution, it offers an alternative equivalent formulation of the problem, which can be convenient when employing numerical solution techniques.

The existence of other conditions for the point qualification in optimisation problems must be mentioned: these specialise to particular classes of problems (e.g. convex cost or constraints), or relax certain assumptions to extend their applicability (e.g. the Mangasarian-Fromovitz or the Slater constraint qualification conditions). These topics are not strictly necessary to the development of this thesis, and have been omitted. However, the interested reader can refer to textbooks such as [6] for further details.

2.1.3 Optimal Control of Multi-Stage Systems

Consider an N-stage system governed by the explicit *Ordinary-Difference Equations*

$$\Delta^+ \boldsymbol{x}(i) = \boldsymbol{f}\Big(i, \boldsymbol{x}(i), \boldsymbol{u}(i), \boldsymbol{p}\Big), \qquad i = 1, \dots, N-1, \tag{2.27}$$

where $\Delta^+ \boldsymbol{x}(i) := \boldsymbol{x}(i+1) - \boldsymbol{x}(i)$. In the context of control theory the vector of dependent variables $\boldsymbol{x}(i)$ is the *state* of the system at the i^{th} stage, and the vector of independent variables $\boldsymbol{u}(i)$ is called the *control*. The symbol $\boldsymbol{p}$ represents a vector of independent system *parameters*, constant for all the N stages.

Also consider a constraint involving the initial and final stages

$$\Delta^+ \boldsymbol{x}(N) = \boldsymbol{f}\Big(N, \boldsymbol{x}(N), \boldsymbol{u}(N), \boldsymbol{p}\Big), \tag{2.28}$$

where $\boldsymbol{x}(N+1) := \boldsymbol{x}(1)$; a constraint of this type is referred to as a *cyclic* or *periodic* constraint[4]. Such a constraint may be used to impose the periodicity of the system *trajectory*, which is the sequence of values assumed by the state and control vectors.

Through the dynamic formulation (2.27, 2.28) a cyclic N-stage OC problem is set up as a parameter optimisation problem of the form (2.9):

$$\begin{aligned} &\min_{\{\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),\boldsymbol{p}\}} J\big(\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),\boldsymbol{p}\big) \\ \text{s.t.}\quad &\Delta^{+}\boldsymbol{x}(i) = \boldsymbol{f}\big(i,\boldsymbol{x}(i),\boldsymbol{u}(i),\boldsymbol{p}\big) \;=\; 0, \qquad i = 1,\dots,N, \end{aligned} \tag{2.29}$$

with

$$\begin{aligned} J\big(\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),\boldsymbol{p}\big) &= \sum_{i=1}^{N} l\big(i,\boldsymbol{x}(i),\boldsymbol{u}(i),\boldsymbol{p}\big), \\ \boldsymbol{x}(\cdot) \in \mathbb{R}^{(\underline{x}\times N)}, \qquad \boldsymbol{u}(\cdot) &\in \mathbb{R}^{(\underline{u}\times N)}, \qquad \boldsymbol{p} \in \mathbb{R}^{\underline{p}}, \end{aligned} \tag{2.30}$$

where the cost addend $l\big(i,\boldsymbol{x}(i),\boldsymbol{u}(i),\boldsymbol{p}\big)$ depends on the stage number, the corresponding state and control vectors, and the system parameters. The symbols $\boldsymbol{x}(\cdot)$ and $\boldsymbol{u}(\cdot)$ represent the state and control vectors at all times, i.e. their trajectories. The constants $\underline{x}$ and $\underline{u}$ are, respectively, the size of the state and control vectors at any one stage.

The problem (2.29) is a special case of an equality-constrained parameter-optimisation problem. All the point-qualification conditions from Section 2.1.1 thus hold for system trajectories. A trajectory for the OC problem is in fact a 'point' for the parameter-optimisation problem. This is evident considering the definition

$$\boldsymbol{z} := \begin{bmatrix} \boldsymbol{x}(1) \\ \boldsymbol{u}(1) \\ \vdots \\ \boldsymbol{x}(N) \\ \boldsymbol{u}(N) \\ \boldsymbol{p} \end{bmatrix}. \tag{2.31}$$

The Lagrange multipliers at a stationary point $\{\check{\boldsymbol{x}}(\cdot),\check{\boldsymbol{u}}(\cdot),\check{\boldsymbol{p}}\}$ must satisfy the first-order necessary conditions[5] (2.11), which imply

$$\Delta^{-}\check{\boldsymbol{\lambda}}^{T}(i) = \nabla_{\boldsymbol{x}(i)} H\big(i,\check{\boldsymbol{x}}(i),\check{\boldsymbol{u}}(i),\check{\boldsymbol{p}},\check{\boldsymbol{\lambda}}(i)\big), \tag{2.32}$$

[4] Although not common in the standard formulation of multi-stage systems, cyclic constraints are necessary when the trajectory must be repeatable, which will be useful in the following vehicle-control studies.

[5] In (2.11) consider $\mathbf{f}(i,\boldsymbol{x}(i),\boldsymbol{u}(i),\boldsymbol{x}(i+1)) = \boldsymbol{f}(i,\boldsymbol{x}(i),\boldsymbol{u}(i)) - \Delta^{+}\boldsymbol{x}(i+1)$.

with

$$H(i, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, \boldsymbol{\lambda}) := l(i, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) + \boldsymbol{\lambda}^T \boldsymbol{f}(i, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}), \tag{2.33}$$

$$\Delta^- \boldsymbol{\lambda}(i) := \boldsymbol{\lambda}(i-1) - \boldsymbol{\lambda}(i) \quad \text{and} \quad \check{\boldsymbol{\lambda}}(0) := \check{\boldsymbol{\lambda}}(N). \tag{2.34}$$

The function $H(i, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, \boldsymbol{\lambda})$ is known within the OC theory as the *control Hamiltonian* [6, p.207]. The Lagrange multipliers are thus governed by difference equations; they are also referred to as the *costate* of the system.

For N-stage OC problems, necessary conditions for a stationary point can be rewritten as

$$\left.\begin{array}{rcl} \nabla_{\boldsymbol{x}(i)} H(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}, \check{\boldsymbol{\lambda}}(i)) &=& \Delta^- \check{\boldsymbol{\lambda}}^T(i) \\ \nabla_{\boldsymbol{u}(i)} H(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}, \check{\boldsymbol{\lambda}}(i)) &=& 0 \\ \nabla_{\boldsymbol{\lambda}(i)} H(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}, \check{\boldsymbol{\lambda}}(i)) &=& \Delta^+ \check{\boldsymbol{x}}^T(i) \\ \nabla_{\boldsymbol{p}} H(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}, \check{\boldsymbol{\lambda}}(i)) &=& 0 \end{array}\right\} i = 1, \ldots, N, \tag{2.35}$$

where the first three vector equations are called, respectively, the *costate dynamics* (or *adjoint equations*), the *control equations* and the *state dynamics* (or *dynamic constraints*); the last row implies optimality of the cost with respect to the system parameters.

It is common within OC problems to include *algebraic* equality constraints and inequality constraints on the decision variables; these extend the formulation (2.29) to

$$\begin{aligned} &\min_{\{\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot), \boldsymbol{p}\}} J\big(\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot), \boldsymbol{p}\big) \\ \text{s.t.} \quad &\left\{\begin{array}{rcl} \Delta^+ \boldsymbol{x}(i) - \boldsymbol{f}\big(i, \boldsymbol{x}(i), \boldsymbol{u}(i), \boldsymbol{p}\big) &=& 0 \\ \mathbf{g}\big(i, \boldsymbol{x}(i), \boldsymbol{u}(i), \boldsymbol{p}\big) &=& 0 \\ \mathbf{h}\big(i, \boldsymbol{x}(i), \boldsymbol{u}(i), \boldsymbol{p}\big) &\leq& 0 \end{array}\right. , \quad i = 1, \ldots, N, \end{aligned} \tag{2.36}$$

In this case the Hamiltonian can be modified to account for the additional constraints using three sets of Lagrange multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\nu}$ and $\boldsymbol{\mu}$ (some of the function arguments are omitted to simplify the notation):

$$H\big(i, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\mu}\big) = l(i) + \boldsymbol{\lambda}^T \boldsymbol{f}(i) + \boldsymbol{\nu}^T \mathbf{g}(i) + \boldsymbol{\mu}^T \mathbf{h}(i) \ . \tag{2.37}$$

Then the the KKT conditions (2.23) can be applied to define necessary conditions for optimality of a solution $\big\{\check{\boldsymbol{x}}(\cdot), \check{\boldsymbol{u}}(\cdot), \check{\boldsymbol{p}}\big\}$ and the associated multipliers $\big\{\check{\boldsymbol{\lambda}}(\cdot), \check{\boldsymbol{\nu}}(\cdot), \check{\boldsymbol{\mu}}(\cdot)\big\}$:

$$\left.\begin{array}{lll} \nabla_{\boldsymbol{x}(i)} \check{H}(i) = \Delta^- \check{\boldsymbol{\lambda}}^T(i), & \nabla_{\boldsymbol{u}(i)} \check{H}(i) = 0, & \nabla_{\boldsymbol{p}} \check{H}(i) = 0, \\ \nabla_{\boldsymbol{\lambda}(i)} \check{H}(i) = \Delta^+ \check{\boldsymbol{x}}^T(i), & \nabla_{\boldsymbol{\nu}(i)} \check{H}(i) = 0, & \nabla_{\boldsymbol{\mu}(i)} \check{H}(i) \leq 0, \\ \check{\boldsymbol{\mu}}(i)^T \mathbf{h}\big(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}\big) = 0, & \check{\boldsymbol{\mu}}(i) \geq 0 & \end{array}\right\} i = 1, \ldots, N, \tag{2.38}$$

where $\check{H}(i) := H(i, \check{\boldsymbol{x}}(i), \check{\boldsymbol{u}}(i), \check{\boldsymbol{p}}, \check{\boldsymbol{\lambda}}(i), \check{\boldsymbol{\nu}}(i), \check{\boldsymbol{\mu}}(i))$.

The Lagrange multipliers provide the sensitivity of the optimal cost with respect to variations of the associated constraints; in particular, the costate represents the sensitivity with respect to variations in the system dynamics.

Alternative necessary optimality conditions for the problem (2.36) can be formulated in terms of the minimisation of the control Hamiltonian (2.37) (see [5, p.110]). In order for a trajectory to be optimal, the Hamiltonian must be minimised at each stage i for all possible variations of the control $\boldsymbol{u}(i)$ and the parameters $\boldsymbol{p}$. This is a finite-dimensional version of the *Pontryagin Maximum Principle* (PMP), which will be introduced for continuous-time OC in Section 2.2.2.

Sufficient conditions for global optimality are given by the *Bellman's principle of optimality* (see [2, p.15]), which is at the basis of *Dynamic Programming*. This approach will not be considered here because the solution of these sufficient conditions becomes prohibitively expensive for most practical problems; a difficulty that Bellman himself recognised in [2, p.204] as the 'curse of dimensionality'.

2.2 The Calculus of Variations

The results hitherto obtained apply to finite-dimensional optimisation problems, i.e. with a finite number of decision parameters. Physical problems are often defined on infinite-dimensional domains, such as the optimisation of the profile of a curve, or a surface. These are also referred to as *functional-optimisation* problems, and are studied by the calculus of variations.

The discussion commences with the standard definition of a problem in the CV:

> *"The fundamental problem of the calculus of variations is as follows. The ... vectors $\boldsymbol{z}_0$, $\boldsymbol{z}_f$ are prescribed .... The function $L(\tau, \boldsymbol{x}, \boldsymbol{y})$ is a given function of the three independent variables τ, $\boldsymbol{x}$, $\boldsymbol{y}$. We consider a set of functions $\boldsymbol{z}(\tau)$, called the* 'admissible functions'*, each of which has the value $\boldsymbol{z}_0$ at* 0 *and the value $\boldsymbol{z}_f$ at* 1*,*
>
> $$\boldsymbol{z}(0) = \boldsymbol{z}_0, \quad \boldsymbol{z}(1) = \boldsymbol{z}_f. \tag{2.39}$$
>
> *The problem is to find among the admissible functions one that minimises the*

integral

$$J\big(\boldsymbol{z}(\tau)\big) = \int_0^1 L\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau)\big)d\tau. \tag{2.40}$$

With each $\boldsymbol{z}(\tau)$ is associated the number $J\big(\boldsymbol{z}(\tau)\big)$. If we think of $\boldsymbol{z}(\tau)$ as the independent variable, $J\big(\boldsymbol{z}(\tau)\big)$ is a 'functional', *i.e. a numerical-valued function on $\boldsymbol{z}(\tau)$."*

[58, p.1] (rephrased to account for vector-valued functions)

The following discussion is restricted to analytical admissible functions $\boldsymbol{z}(\tau)$ and functions $L\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau)\big)$, in order to obtain results directly relevant to the problems discussed in the remainder of the thesis. The integrand L is also known as the *Lagrangian* function (see [45, p.32]). It is implicitly assumed in the standard definition that the functions $\boldsymbol{z}(\tau)$ and $\dot{\boldsymbol{z}}(\tau)$ be mutually dependent according to the differential relation

$$\dot{\boldsymbol{z}}(\tau) := \frac{d}{d\tau}\boldsymbol{z}(\tau) = \lim_{h\to 0}\frac{\boldsymbol{z}(\tau+h) - \boldsymbol{z}(\tau)}{h}. \tag{2.41}$$

Considering (2.41) the fundamental problem in the CV can be reformulated as the problem of finding the 'independent' vector-valued functions $\boldsymbol{z}(\tau)$ and $\dot{\boldsymbol{z}}(\tau)$, subject to differential constraints:

$$\begin{aligned} &\min_{\{\boldsymbol{z}(\tau),\,\dot{\boldsymbol{z}}(\tau)\}} \int_0^1 L\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau)\big)d\tau \\ &\text{s.t.} \quad \dot{\boldsymbol{z}}(\tau) - \frac{d}{d\tau}\boldsymbol{z}(\tau) = 0 \end{aligned}. \tag{2.42}$$

An augmented cost function $\bar{J}$ is defined, that is equivalent to the original cost function J when the constraints are satisfied

$$\begin{aligned} \bar{J} &:= \int_0^1 \left[L\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau)\big) + \boldsymbol{\lambda}^T(\tau)\left(\dot{\boldsymbol{z}}(\tau) - \frac{d}{d\tau}\boldsymbol{z}(\tau)\right)\right]d\tau \\ &= \int_0^1 \left[H\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau), \boldsymbol{\lambda}(\tau)\big) - \boldsymbol{\lambda}^T(\tau)\frac{d}{d\tau}\boldsymbol{z}(\tau)\right]d\tau \end{aligned} \tag{2.43}$$

with the *Hamiltonian* [13, p.48] defined as

$$H\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau), \boldsymbol{\lambda}(\tau)\big) := L\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau)\big) + \boldsymbol{\lambda}^T(\tau)\dot{\boldsymbol{z}}(\tau). \tag{2.44}$$

After integration by parts the augmented cost can be rewritten as

$$\bar{J} = \int_0^1 \left[H\big(\tau, \boldsymbol{z}(\tau), \dot{\boldsymbol{z}}(\tau), \boldsymbol{\lambda}(\tau)\big) + \frac{d}{d\tau}\boldsymbol{\lambda}^T(\tau)\boldsymbol{z}(\tau)\right]d\tau + \boldsymbol{\lambda}^T(1)\boldsymbol{z}(1) - \boldsymbol{\lambda}^T(0)\boldsymbol{z}(0). \tag{2.45}$$

Now consider the variation of $\bar{J}$ for small variations[6] of $\boldsymbol{z}(\tau)$ and $\dot{\boldsymbol{z}}(\tau)$; the variations of the decision functions $\delta\boldsymbol{z}(\tau)$ and $\delta\dot{\boldsymbol{z}}(\tau)$ are independent. The variations of the augmented cost are

$$\delta_{\dot{\boldsymbol{z}}(\tau)}\bar{J} = \int_0^1 \left[\frac{\partial}{\partial\dot{\boldsymbol{z}}(\tau)}L\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big)\delta\dot{\boldsymbol{z}}(\tau) + \boldsymbol{\lambda}^T(\tau)\delta\dot{\boldsymbol{z}}(\tau)\right]d\tau \tag{2.46}$$

$$\begin{aligned}\delta_{\boldsymbol{z}(\tau)}\bar{J} &= \int_0^1 \left[\frac{\partial}{\partial\boldsymbol{z}(\tau)}L\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big)\delta\boldsymbol{z}(\tau) + \frac{d}{d\tau}\boldsymbol{\lambda}^T(\tau)\delta\boldsymbol{z}(\tau)\right]d\tau \\ &+ \boldsymbol{\lambda}^T(1)\delta\boldsymbol{z}(1) - \boldsymbol{\lambda}^T(0)\delta\boldsymbol{z}(0),\end{aligned} \tag{2.47}$$

where $\delta_{\boldsymbol{z}(\tau)}\bar{J}$ and $\delta_{\dot{\boldsymbol{z}}(\tau)}\bar{J}$ represent first-order variations of $\bar{J}$ associated with variations of $\boldsymbol{z}(\tau)$ and $\dot{\boldsymbol{z}}(\tau)$, respectively.

In order to attain a minimum, the first-order variations of the augmented cost must vanish for all possible variations of the decision functions. The boundary values of $\boldsymbol{z}(\tau)$ are now left unconstrained, thus allowing variations at the boundaries $\delta\boldsymbol{z}(0)$ and $\delta\boldsymbol{z}(1)$:

$$\delta_{\dot{\boldsymbol{z}}(\tau)}\bar{J} = 0, \quad \forall\delta\dot{\boldsymbol{z}}(\tau) \quad \rightarrow \quad \boldsymbol{\lambda}^T = -\nabla_{\dot{\boldsymbol{z}}}L\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big) \tag{2.48}$$

$$\delta_{\boldsymbol{z}(\tau)}\bar{J} = 0, \quad \forall\delta\boldsymbol{z}(\tau) \quad \rightarrow \quad \begin{cases} \dfrac{d}{d\tau}\boldsymbol{\lambda}^T(\tau) + \nabla_{\boldsymbol{z}}H\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big) = 0 \\ \boldsymbol{\lambda}(0) = 0, \quad \boldsymbol{\lambda}(1) = 0 \end{cases} \tag{2.49}$$

The conditions (2.48, 2.49) are first-order necessary conditions for a local minimiser; substituting (2.48) into (2.49) one obtains the *Euler-Lagrange equation*

$$\nabla_{\boldsymbol{z}(\tau)}L\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big) - \frac{d}{d\tau}\Big[\nabla_{\dot{\boldsymbol{z}}(\tau)}L\big(\tau,\boldsymbol{z}(\tau),\dot{\boldsymbol{z}}(\tau)\big)\Big] = 0. \tag{2.50}$$

If periodic constraints of the kind

$$\boldsymbol{z}(0) = \boldsymbol{z}(1) \rightarrow \delta\boldsymbol{z}(0) = \delta\boldsymbol{z}(1) \tag{2.51}$$

are enforced on the decision functions, the necessary boundary condition on the multipliers follows,

$$\boldsymbol{\lambda}(0) = \boldsymbol{\lambda}(1). \tag{2.52}$$

[6]Small variations are considered for both the functions $\boldsymbol{z}(\tau)$ and $\dot{\boldsymbol{z}}(\tau)$; these conditions can be used to characterise a *weak minimiser* of the cost function; for a detailed discussion on *weak* and *strong optimality* refer to [9] and [58].

This is in fact an application of the *Multiplier Rule* (MR) for a problem of Lagrange in the CV (see [58, p.236]).

Together, the boundary conditions on the multipliers, at $\tau = 0$ and $\tau = 1$, and the Euler-Lagrange equation (2.50) define a two-point *Boundary-Value Problem* (BVP) in the variables $\boldsymbol{z}(\tau)$ and $\boldsymbol{\lambda}(\tau)$. A solution of said BVP represents a trajectory at which the cost functional is stationary. The optimality of the trajectory should then be validated by means of sufficiency conditions. The reader interested in a more detailed discussion on the optimality conditions in the CV may refer to [9] and [58].

The calculus of variations is a general theory on the study of functional extrema. Amongst its applications to many fields of science, the ones that are most relevant to this thesis are the derivations of the equations of motion in mechanics and the definition of optimality conditions in optimal control. The CV represents in fact a strong connection between Lagrangian mechanics and optimal control; it is thus a fundamental tool in the development of this work.

2.2.1 Lagrangian Mechanics

This section presents a derivation of the equations of motion for a free particle in a conservative force field. The derivation is based on the application of CV theory to the principle of stationary action. The *principle of stationary action*, also known as *principle of least action*, is a fundamental principle in physics, according to which the path taken by a particle provides a stationary value of the *action* integral

$$S\big(\mathbf{r}(t)\big) = \int_{t_1}^{t_2} T\big(\dot{\mathbf{r}}(t)\big) - U\big(\mathbf{r}(t)\big)dt; \tag{2.53}$$

the functions T and U are, respectively, the kinetic and potential energy, with

$$T\big(\dot{\mathbf{r}}(t)\big) = \frac{1}{2}m\dot{\mathbf{r}}^2(t), \tag{2.54}$$

where the constant m is the mass of the particle and the vector $\mathbf{r}(t)$ describes its position at time t; t_1 and t_2 are the initial and final instants of the motion. In this case the

optimisation functions are the Cartesian coordinates of the particle's trajectory, i.e. the components of $\mathbf{r}$.

The action integral S corresponds to the cost functional J for a problem in the calculus of variations. The integrand function is the difference between kinetic and potential energy, which, following a nomenclature consistent with the CV, is known in classical mechanics as the Lagrangian $L = T - U$. A stationary value is thus found solving the Euler-Lagrange equation (2.50) for the action integral:

$$\frac{\partial U}{\partial \mathbf{r}} - \frac{d}{dt}\frac{\partial T}{\partial \dot{\mathbf{r}}} = 0. \tag{2.55}$$

The vector equation (2.55) gives the equation of motion for the particle. It can be proven that the principle of stationary action is equivalent to Newton's second law of motion: first, in a conservative field the force acting on a particle is $F = \partial U/\partial \mathbf{r}$; second, from (2.54) one obtains $\partial T/\partial \dot{\mathbf{r}} = m\dot{\mathbf{r}}$, which is the momentum of the particle. Substituting these two equalities into (2.55), and differentiating with respect to time gives Newton's second law

$$F = m\ddot{\mathbf{r}}. \tag{2.56}$$

This reasoning can be extended to systems of particles and rigid bodies.

The Newtonian and Lagrangian approaches will be used interchangeably in subsequent chapters for the derivation of vehicle-system dynamics. An appealing property of the Lagrangian approach is that it is directly applicable using any choice of generalised coordinates for the mechanical system, given that the coordinate transformation between the chosen and the Cartesian coordinates — in which the particle position $\mathbf{r}$ is expressed — is invertible.

2.2.2 Continuous-Time Optimal Control

In problems of continuous-time OC the minimisation of the objective function is subject to differential and algebraic constraints in form of equalities or inequalities (or both). The physical concept of 'time' is introduced in association with the independent variable τ. A

sufficiently general formulation for the purpose of this thesis is

$$\min_{\{\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\}} J\big(\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big)$$
$$\text{s.t.} \begin{cases} \dfrac{d\boldsymbol{x}(\tau)}{d\tau} &= \boldsymbol{f}\big(\tau,\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(\tau,\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(\tau,\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big) \end{cases}, \tag{2.57}$$

with the cost functional J given in the *Lagrange form*

$$J\big(\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big) := \int_0^1 L\big(\tau,\boldsymbol{x}(\tau),\boldsymbol{u}(\tau),\boldsymbol{p}\big)d\tau. \tag{2.58}$$

The differential constraints in (2.57) involving $d\boldsymbol{x}(\tau)/d\tau$ are referred to as the *dynamics*. The vector-valued function $\boldsymbol{x}(\tau)$ subject to differential constraints represents the *state* of the system, $\boldsymbol{u}(\tau)$ is the *control*. The constant vector $\boldsymbol{p}$ contains the system *parameters*. Constraints that affect the trajectories of state and control throughout the integration interval are referred to as *path* constraints.

The multiplier approach of Section 2.2, combined with the arguments on inequality constraints used in Section 2.1.2, provide necessary conditions for optimality based on the control Hamiltonian. The control Hamiltonian for continuous time systems is

$$H\big(\tau,\boldsymbol{x},\boldsymbol{u},\boldsymbol{p},\boldsymbol{\lambda},\boldsymbol{\nu},\boldsymbol{\mu}\big) := L(\tau) + \boldsymbol{\lambda}^T\boldsymbol{f}(\tau) + \boldsymbol{\nu}^T\mathbf{g}(\tau) + \boldsymbol{\mu}^T\mathbf{h}(\tau); \tag{2.59}$$

first-order necessary conditions for feasibility and optimality are

$$\left.\begin{array}{lll} \nabla_{\boldsymbol{x}(\tau)}\check{H}(\tau) = -d\check{\boldsymbol{\lambda}}^T(\tau)/d\tau, & \nabla_{\boldsymbol{u}(\tau)}\check{H}(\tau) = 0, & \nabla_{\boldsymbol{p}}\check{H}(\tau) = 0, \\ \nabla_{\boldsymbol{\lambda}(\tau)}\check{H}(\tau) = d\check{\boldsymbol{x}}^T(\tau)/d\tau, & \nabla_{\boldsymbol{\nu}(\tau)}\check{H}(\tau) = 0, & \nabla_{\boldsymbol{\mu}(\tau)}\check{H}(\tau) \leq 0, \\ \check{\boldsymbol{\mu}}(\tau)^T\mathbf{h}\big(\tau,\check{\boldsymbol{x}}(\tau),\check{\boldsymbol{u}}(\tau),\check{\boldsymbol{p}}\big) = 0, & \check{\boldsymbol{\mu}}(\tau) \geq 0 & \end{array}\right\} \forall\tau\in[0,1], \tag{2.60}$$

where $\check{H}(\tau) := H(\tau,\check{\boldsymbol{x}}(\tau),\check{\boldsymbol{u}}(\tau),\check{\boldsymbol{p}},\check{\boldsymbol{\lambda}}(\tau),\check{\boldsymbol{\nu}}(\tau),\check{\boldsymbol{\mu}}(\tau))$.

If the state is unconstrained at the boundaries, then, as in (2.49), there holds $\boldsymbol{\lambda}(0) = 0$, $\boldsymbol{\lambda}(1) = 0$. If a cyclic constraint is imposed, condition (2.52) must be instead satisfied by the multipliers.

The boundary conditions on the multipliers combined with the conditions (2.60) define an inequality-constrained BVP in the variables $\boldsymbol{x}(\tau)$, $\boldsymbol{u}(\tau)$, $\boldsymbol{\lambda}(\tau)$, $\boldsymbol{\nu}(\tau)$, $\boldsymbol{\mu}(\tau)$ and $\boldsymbol{p}$.

When the set of active inequality constraints does not change throughout the optimisation horizon one obtains a two-point BVP.

If the active set changes, the horizon is divided into subarcs characterised by distinct constraint sets; the points of transition between these subarcs are called *junction points*. In this case the necessary conditions define a sequence of two-point BVP on the subarcs, which is also known as a *multi-point* BVP. So far, no general method has been defined to establish the evolution of the active set within an optimal solution. This is one of the reasons why the solution of continuous-time OC problems through the satisfaction of the necessary conditions (2.60) can be hard in general.

An alternative definition of necessary conditions for optimality can be derived from the principle that "the Hamiltonian must be minimised over the set of all possible $\boldsymbol{u}(\tau)$" [12, p.108] and $\boldsymbol{p}$. This is known as the *Pontryagin Minimum Principle*[7].

Sufficient conditions for global optimality can be defined applying Bellman's principle of optimality in the form of the *Hamilton-Jacobi-Bellman Equation* (HJBE). However, the search of an optimal trajectory requires the solution of a set of *Partial-Differential Equations*, which is comparatively harder and more expensive than the solution of the necessary-condition BVP only. For a discussion on the PMP and the HJBE the reader may refer to [5, pp.91-102] and [45].

At this point, the parallel between the necessary conditions for optimality in multi-stage and continuous-time systems should be evident; a clarifying example will be given in Section 2.3.4.1. This parallelism will play a key role in the definition of numerical methods for continuous-time OC problems.

2.2.2.1 Variable Time Horizon

Thus far only problems over the optimisation horizon [0, 1] have been considered; however, the presented results can be easily extended to problems with arbitrary finite horizon

[7] The original formulation in [68] is given for a maximisation problem; the principle is also referred to as the *Maximum Principle*.

$[a, b]$, with $b > a$. This is achieved applying a bijective[8] mapping to the independent variable, that transforms the first horizon into the second. Within this thesis only analytic mappings will be considered, of the form

$$\tau = f(t), \qquad \tau \in [0, 1], \qquad t \in [a, b]. \tag{2.61}$$

A simple example is given by the affine mapping

$$t = a + \tau(b - a). \tag{2.62}$$

The substitution of the independent variable allows one to change the optimisation horizon by adjusting parameters in the mapping. Continuous-time OC problems with variable initial and final times can thus be defined considering these parameters as decision variables, and rearranging the formulation given in (2.57):

$$\min_{\{\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p},a,b\}} \int_a^b L\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big)dt \qquad \min_{\{\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p},a,b\}} \int_0^1 L\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big)\frac{dt}{d\tau}d\tau$$
$$\text{s.t.} \begin{cases} \dfrac{d\boldsymbol{x}(t)}{dt} &= \boldsymbol{f}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \end{cases} \iff \text{s.t.} \begin{cases} \dfrac{d\tau}{dt}\dfrac{d\boldsymbol{x}(t)}{d\tau} &= \boldsymbol{f}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \end{cases} \tag{2.63}$$

with t substituted by its value in (2.62) within the right-hand-side formulation. The left-hand-side formulation in (2.63) encompasses minimum-time problems for the special case $L\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) = 1$.

2.3 Numerical Methods

For centuries the calculus of variations has been developed and used to seek exact solutions of optimal control problems, but most of the 'real' problems one encounters are too hard to be solved analytically. The alternative route is the solution of variational problems through finite-difference approximations. However, the computational burden associated

[8] A mapping between two domains is bijective if for every element in each domain there exists one, and one only, element in the other domain.

with the resulting optimisation problems often rendered this route equally tortuous for the pioneers of the discipline.

A game-changing transition in numerical optimisation was marked by the advent of digital computers in the 1940s. The size and execution speed of solvable problems blossomed at the same rate as the computational power and memory available on computers, to the extent that infinite-dimensional problems could be accurately approximated by large finite-dimensional ones[9]. In 1955 it was recognised that

> *"The high speed computing machines make it possible to enjoy the advantages of intricate methods"* [31].

Thenceforth it did not take more than two decades before the mature theory of the CV and the developing field of numerical integration converged into a unified approach for the numerical solution of optimal-control problems, which eventually reduces to parameter optimisation (see [12]).

The fundamental idea is to convert, or *transcribe*, the infinite-dimensional optimal-control problem into a finite-dimensional problem, using numerical approximations based on *quadrature formulae*. This technique is in fact a generalised version of *time discretisation* of continuous-time dynamics. A *transcribed* problem is in the form of (2.20) and can be studied using the standard NLP techiniques introduced in Section 2.1.2.

The essential paradigm of combined discretisation and optimisation branched into two main streams that are currently identified by the attributes *'direct'* and *'indirect'*. The Figure 2.1 is a representation of the basic techniques and methods associated with numerical optimal control through transcription. The direct and indirect solution avenues are represented using different arrow styles according to the legend at the bottom. The order in which the optimisation and the discretisation phases are performed defines the paradigm: discretisation comes first in a direct method, optimisation comes first in an indirect one. It is evident that the numerical integration plays a key role in either approach, in that it allows for the conversion from infinite- to finite-dimensional problems.

[9] Contemporary nonlinear-optimisation software admits problems with 10^9 optimisation variables and twice as many constraints [14].

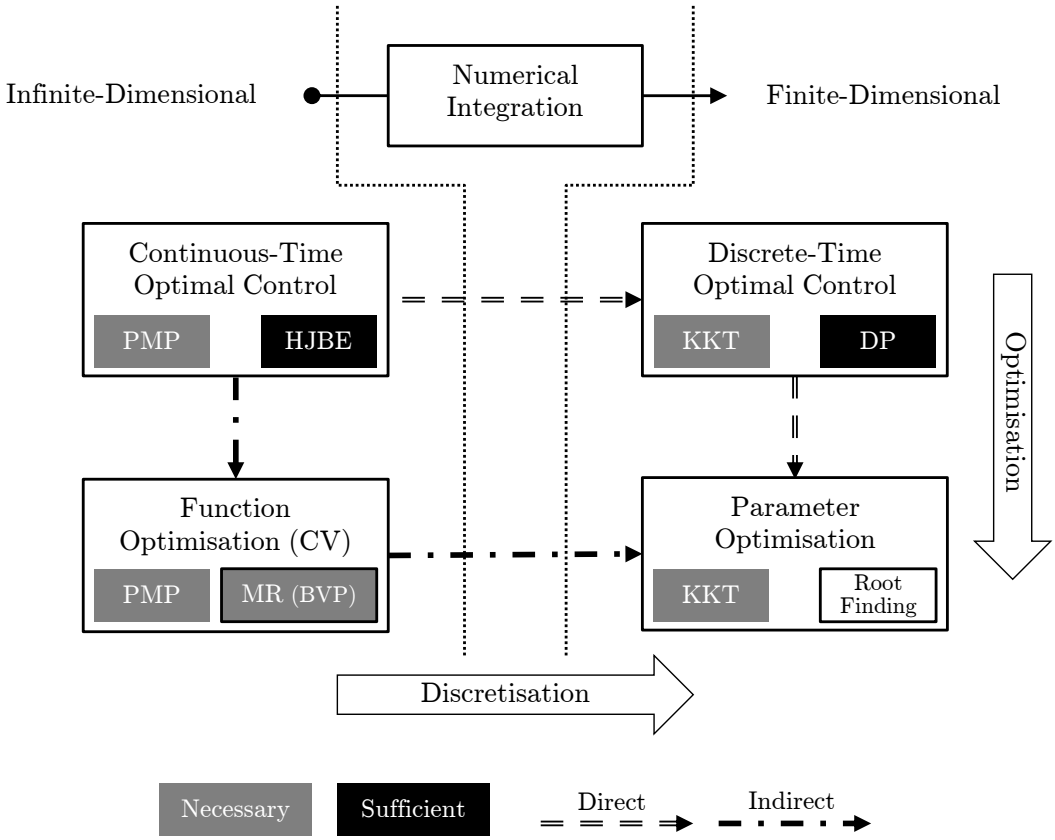

Figure 2.1: Numerical optimal control through transcription.

The grey and the black boxes give, respectively, necessary or sufficient conditions for an optimal solution in the corresponding problem classes. Within functional optimisation, the necessary conditions include both the Pontryagin Maximum Principle and the Multiplier Rule. After discretisation, these alternative approaches will respectively translate either into an optimisation or a root-finding problem. Inevitably, some of the elements in the diagram require further discussion, and will be explained in the sections to come.

It has already been mentioned that DP and the HJBE are not viable approaches for the range of applications considered in this thesis due to an unsustainable computational burden. The presence of nonlinear dynamics and constraints also banishes linear optimal control from the spectrum of applicable methods. Notwithstanding the relevance of these techniques in the broad field of OC (see [12]), it is not convenient to consider them further in this context.

2.3.1 Optimisation

So far only conditions for point qualification in optimisation problems have been considered: a candidate solution 'can be' an optimum, if it satisfies the necessary conditions, or actually 'is' an optimum, if it satisfies sufficient conditions.

As a notable example one can consider the necessary optimality conditions for an inequality-constrained optimisation problem: in accordance with Section 2.1.2, a solution $\check{\boldsymbol{z}}$ must satisfy

$$\nabla_{\boldsymbol{z}} L(\check{\boldsymbol{z}}, \check{\boldsymbol{\lambda}}, \check{\boldsymbol{\mu}}) = 0, \quad \mathbf{f}(\check{\boldsymbol{z}}) = 0, \quad \mathbf{h}(\check{\boldsymbol{z}}) \leq 0, \quad \check{\boldsymbol{\mu}} \geq 0, \quad \check{\boldsymbol{\mu}}^T \mathbf{h}(\check{\boldsymbol{z}}) = 0, \tag{2.64}$$

where the Lagrangian $L(\boldsymbol{z}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ is defined in (2.22). The nonlinear inequality constraints can be transformed into more treatable bound constraints introducing a set of *slack* variables $\boldsymbol{s}$, and replacing the constraints $\mathbf{h}(\boldsymbol{z}) \leq 0$ with

$$\boldsymbol{s} = \mathbf{h}(\boldsymbol{z}), \qquad \boldsymbol{s} \leq 0. \tag{2.65}$$

It is now evident that the satisfaction of necessary optimality conditions corresponds to the problem of finding the roots of a set of functions within given bounds on their arguments. If the aforementioned conditions cannot be solved in closed form, then a numerical procedure must be invoked to search for a solution.

This section begins by introducing some basic techniques for the numerical solution of root-finding problems. These will then be extended to cope with optimisation problems through the introduction of the known optimality conditions. Root-finding is interpreted here as a differential counterpart of optimisation.

2.3.1.1 Root-Finding

A root-finding problem is the problem of finding the values of the independent vector variable $\boldsymbol{z}$ that satisfy the vector-valued equality $\mathbf{f}(\boldsymbol{z}) = 0$. The variable values can be bounded from above and below, and it is assumed that a solution to this problem exists within these bounds.

The numerical solution of a root-finding problem is now a mature field that encompasses algorithms as simple as the bisection method, the well-known Newton-Rhapson method, or methods based on polynomial interpolation (see [69, ch.9]).

Any numerical method starts from an estimate $\bar{\boldsymbol{z}}_0$ of the solution $\check{\boldsymbol{z}}$. At each iteration one attempts to improve the estimate, with the aim of converging to a solution, or to an estimate that is sufficiently close to it. There are thus two fundamental rules to be defined:

1. the iteration procedure, which updates the estimate;
2. the termination criterion, which establishes when one estimate is sufficiently accurate and the iteration process can be stopped.

The most famous numerical method for root-finding is the Newton-Raphson method. The idea behind this method is readily explained for problems in one dimension: at each i^{th} iteration, one considers the line tangent to the function $f(z)$ at the previous estimate $\bar{z}_{i-1}$. The current estimate $\bar{z}_i$ is then defined to be the root of the tangent function

$$\bar{z}_i := \bar{z}_{i-1} - h_{i-1}, \qquad \text{with} \quad h_{i-1} := \frac{f(\bar{z}_{i-1})}{\frac{d}{dt} f(\bar{z}_{i-1})}. \tag{2.66}$$

The extension of the Newton-Raphson method to many dimensions is

$$\bar{\boldsymbol{z}}_i := \bar{\boldsymbol{z}}_{i-1} - \mathbf{h}_{i-1}, \qquad \text{with} \quad \nabla_{\boldsymbol{z}} \mathbf{f}(\bar{\boldsymbol{z}}_{i-1}) \mathbf{h}_{i-1} = \mathbf{f}(\bar{\boldsymbol{z}}_{i-1}). \tag{2.67}$$

The direction of the vector $\mathbf{h}_{i-1}$ is known as the *search direction*, and its magnitude as the *step length*. Determining the vector $\mathbf{h}_i$ involves the solution of a system of linear equations of the general form $A_i \mathbf{h}_i = \mathbf{b}_i$. The matrix of linear coefficents A_i is the Jacobian of $\mathbf{f}(\boldsymbol{z})$ evaluated at $\bar{\boldsymbol{z}}_i$ in this case, the constant term $\mathbf{b}_i$ is given by $\mathbf{f}(\bar{\boldsymbol{z}}_i)$. From the regularity assumption one knows that a unique solution exists (see Section 2.1.1). The numerical solution of this system requires in general a number of elementary operations proportional to $\underline{z}^3$. However, this complexity can be reduced if the A matrix is sparse[10] and its sparsity

[10] A matrix is said *sparse* if most of its elements are 0; the sparsity pattern of a matrix is a map of the positions of the non-zero elements.

pattern is known. Being able to exploit the sparsity of the A matrix is crucial to the efficiency of any of the numerical methods presented herein.

A termination criterion is often based on the satisfaction of an error tolerance in the function evaluation. For instance the iteration process can be arrested when $||\mathbf{f}(\bar{\boldsymbol{z}}_i)|| \leq \epsilon$, with an arbitrarily small positive threshold ϵ.

The choice and combination of the iteration procedure and the termination criterion determine the effectiveness of a numerical method. In particular, the iteration procedure determines the *region of attraction* and the *rate of convergence* of the method, the termination criterion affects the number of required iterations. The reader may also refer to [8, ch.1] for a more detailed introduction to this topic.

2.3.1.2 Equality-Constrained Optimisation

The application of the Newton-Raphson method to the first-order necessary optimality conditions (2.15) provides a numerical method for the solution of constrained optimisation problems. The resulting method is also known as Newton's method for optimisation. In this case the constant term $\mathbf{b}_i$ comes from the evaluation of the Jacobian of the Lagrangian (2.14) at the current solution estimate $\bar{\boldsymbol{z}}_i$ (the roots of the Jacobian satisfy the necessary conditions); the matrix of linear coefficients A_i is the Hessian of the Lagrangian. A root-finding problem is a special case of a constrained minimisation problem, thus any numerical method for constrained minimisation is also capable of finding the roots of a system of equations.

In optimisation problems, not only does one want to find the roots of the first order necessary conditions for optimality, but also verify that the second order sufficient (2.8) (or at least necessary (2.7)) conditions are satisfied. Furthermore, since the cost function is known, and it has to be optimised, one can define additional criteria within the iteration procedure to guarantee that the solution estimate is evolving towards diminishing values of the cost function; these might include: corrections in both the search direction and the step length, the definition of a *trust region* for the allowed progress of a single iteration, or the use of a *merit function* to establish whether the search is proceeding in the right

direction (the reader may refer to [8, pp.21-27] for further details).

The definition of a merit function is naturally simplified within optimisation problems by the presence of a cost. The same task is not as straightforward when treating root-finding problems. For this reason, it is generally recognised (see [7]) that optimisation problems have a larger *region of attraction* than the root-finding counterparts, which means that a numerical solution algorithm might converge for a less accurate initial estimate. In the following sections it is argued that this repesents a practical advantage for direct over indirect optimal control methods.

2.3.1.3 Inequality-Constrained Optimisation

The presence of inequality constraints introduces the combinatorial problem of finding the active set (see Section 2.1.2). A family of numerical approaches, known as the *active-set* methods, search for a solution by solving a series of equality-constrained optimisation problems, each of which takes into account only a sub-set of active inequality constraints. The fundamental principle is to add inequality constraints to the active set if they are violated by the current solution, and to remove them if the associated Lagrange multiplier is < 0 (feasible direction); the process is then iterated until the active set remains unchanged within two consecutive solutions.

The problem of finding the correct active set is known to be a *NP-hard* combinatorial problem (see [86]). In order to circumvent this combinatorial complexity, one can incorporate the inequality constraints by means of smooth approximations, and solve a series of relaxed problems for increasing accuracy of the constraints.

Barrier Methods are numerical solution methods for inequality-constrained parameter-optimisation problems that model inequality constraints introducing additional terms in the problem cost. An inequality constraint is replaced by a term in the cost that increases considerably when the optimisation parameters violate (or are close to violating) the constraints. The underlying idea is that, since a minimum has to be achieved, 'costly' terms associated with constraint violation will be avoided.

Consider the inequality-constrained optimisation problem (2.20); using a barrier method the cost function may be modified to

$$J_b(\boldsymbol{z}, \beta) := J(\boldsymbol{z}) - \beta \sum_{i=1}^{\mathrm{h}} \ln\big(-\mathrm{h}_i(\boldsymbol{z})\big), \tag{2.68}$$

where the *barrier parameter* $\beta > 0$ determines the accuracy of the problem approximation. Since according to definition (2.68) the parameter vector $\boldsymbol{z}$ is always kept in the interior of the region defined by the inequality constraints, a method of this type is also referred to as an *interior-point* method.

A solution of the problem

$$\begin{aligned} &\min_{\boldsymbol{z}} J_b(\boldsymbol{z}, \beta) \\ \text{s.t.} \quad &\mathbf{f}(\boldsymbol{z}) = 0 \end{aligned} \tag{2.69}$$

would be an approximation of the solution to the original problem (2.20), with increasingly accurate approximations for $\beta \to 0$. This is known as the *primal* approach.

It can be recognised that the gradient of the modified cost function $\nabla_{\boldsymbol{z}} J_b(\boldsymbol{z}, \beta)$ is not defined when an inequality constraint is active ($\mathrm{h}_i(\boldsymbol{z}) = 0$ for any i), which means that an exact optimum cannot be reached. This issue is overcome introducing a vector of *dual* variables $\boldsymbol{\mu}$, with elements defined as $\mu_i = -\beta/\mathrm{h}_i(\boldsymbol{z})$. These variables are then bounded to be non-negative $\boldsymbol{\mu} \geq 0$ in order to enforce the original inequality constraint $\mathbf{h}(\boldsymbol{z}) \leq 0$ (since $\beta > 0$).

Combining the first-order necessary conditions for optimality to the definition of the dual variables one obtains the *perturbed KKT conditions* (see [86]):

$$\begin{cases} \nabla_{\boldsymbol{z}} J(\check{\boldsymbol{z}}) + \check{\boldsymbol{\lambda}}^T \nabla_{\boldsymbol{z}} \mathbf{f}(\check{\boldsymbol{z}}) + \check{\boldsymbol{\mu}}^T \nabla_{\boldsymbol{z}} \mathbf{h}(\check{\boldsymbol{z}}) = 0, \\ \check{\boldsymbol{\mu}} \geq 0 \\ \check{\boldsymbol{\mu}}^T \mathbf{h}(\check{\boldsymbol{z}}) - \beta = 0 \end{cases} . \tag{2.70}$$

Notice that for $\beta \to 0$ the third condition becomes the complementary slackness, and the dual variables $\check{\boldsymbol{\mu}}$ become the Lagrange multipliers associated with the inequality constraints (see (2.23) for a parallel with the KKT conditions). Methods that involve the definition of dual variables are called *primal-dual* methods. The iteration procedure usually includes a strategy for the choice of the barrier parameter, which is also considered

within the termination criterion to guarantee that the problem approximation is sufficiently accurate.

2.3.2 Initial-Value Problem

The following sections consider the problem of representing a function by means of a finite number of parameters, and of formulating any differential constraint on the function in terms of algebraic constraints on the parameters. This problem is tightly connected with the numerical solution of *Initial-Value Problems* (IVP). A brief introduction on IVP methods will lay the foundations for all the transcription methods described herein. For an exhaustive description of the relevant theory the reader may to refer to [11, 23, 28].

An initial-value problem for a set of *explicit Ordinary-Differential Equations* (ODE) is the problem of finding a function $\boldsymbol{x}(t)$ that satisfies

$$\left.\begin{aligned} &\frac{d}{dt}\boldsymbol{x}(t) = \boldsymbol{f}\big(t, \boldsymbol{x}(t)\big), \quad t \in [0,1] \\ &\boldsymbol{x}(0) = \boldsymbol{x}_0 \end{aligned}\right\}. \tag{2.71}$$

Additional arguments of $\boldsymbol{f}$, such as external inputs $\boldsymbol{u}(t)$ or parameters $\boldsymbol{p}$ can be included without affecting the solution method; a minimal definition of $\mathbf{f}$ is adopted here for notational simplicity. Initial-value problems are ubiquitous in physical problems where one wants to predict the trajectory of the state $\boldsymbol{x}(t)$ of a dynamic system starting from a known initial condition $\boldsymbol{x}_0$. It goes without saying that if the problem horizon is different from $[0, 1]$ it can be transformed using a variable substitution technique as explained in Section 2.2.2.1. When the value of $\boldsymbol{x}(t)$ is given at the end of the horizon one obtains a *final-value problem*; one such problem can be converted to the standard IVP form (2.71) through variable substitution, mapping the horizon $[1, 0]$ into $[0, 1]$.

For some special families of $\boldsymbol{f}$ functions (e.g. linear in $\boldsymbol{x}$ and independent of t) the solution can be found in closed form, but this is not true in the general case of nonlinear functions. When an analytical solution cannot be found, one resorts to numerical approximation methods.

A multitude of methods have been devised to attain accuracy and convergence of the numerical solutions. In most cases these methods can be classified into *one-step* or

multi-step methods, *explicit* or *implicit*, with constant or variable *step length*, and with determined *stage* and *order*. The discussion must be restricted to a brief introduction of these concepts; however, a detailed account and classification of these methods can be found in [11, 23, 28].

Most numerical methods are based on *marching techniques*. These techniques produce approximations $\bar{\boldsymbol{x}}(t)$ for the value of $\boldsymbol{x}(t)$ sequentially in time, starting from the known condition $\boldsymbol{x}(0)$. When the value of $\boldsymbol{x}(t)$, or its approximation, is known at some instant in time $t_i \in [0, 1)$, a *basic rule* is applied to the interval $[t_i, t_{i+1}]$ in order to generate an estimate of $\boldsymbol{x}(t_{i+1})$, with $t_i < t_{i+1} \leq 1$; starting from $t_0 = 0$ this procedure is iterated until an estimate of $\boldsymbol{x}(1)$ becomes available.

2.3.2.1 Basic Rule

Consider a single interval $[t_i, t_{i+1}]$ as introduced in the previous paragraph; a *basic rule* has the purpose of approximating the value of a definite integral:

$$\boldsymbol{\gamma}(i) \approx \int_{t_i}^{t_{i+1}} \boldsymbol{f}\big(t, \boldsymbol{x}(t)\big) dt. \tag{2.72}$$

The approximation can then be used to compute the estimate

$$\bar{\boldsymbol{x}}(t_{i+1}) := \bar{\boldsymbol{x}}(t_i) + \boldsymbol{\gamma}(i). \tag{2.73}$$

In this context only one-step methods are considered, as they are more suitable for a general description of numerical OC techniques. The general formulation of a one-step basic rule is

$$\bar{\boldsymbol{x}}(t_{i+1}) := \bar{\boldsymbol{x}}(t_i) + h_i \boldsymbol{\psi}\big(t_i, \bar{\boldsymbol{x}}(t_i), h_i\big) \tag{2.74}$$

with $h_i := t_{i+1} - t_i$; some of the possible techniques to define the function $\boldsymbol{\psi}\big(t_i, \bar{\boldsymbol{x}}(t_i), h_i\big)$ will be described later on.

A one-step method is said to have *order* p if it satisfies the condition

$$|\bar{\boldsymbol{x}}(t_{i+1}) - \boldsymbol{x}(t_{i+1})| \leq M h_i^{p+1}, \qquad \forall h_i : 0 \leq h_i \leq \bar{h} \tag{2.75}$$

where $\bar{\boldsymbol{x}}(t_{i+1}) - \boldsymbol{x}(t_{i+1})$ is the *local error*, M and $\bar{h}$ are constants, and all possible t_i's and $\boldsymbol{x}(t_i)$'s are considered (see [23]).

For notational ease a basic rule is described for a single arbitrary interval: the interval $[t_i, t_{i+1}]$ is herein transformed into $[0, 1]$ through variable substitution; the differential variable becomes τ and the integral approximation is denoted by $\boldsymbol{\gamma}$.

The standard approach is to substitute $\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big)$ with a function for which the integral is known in closed form. Due to the nature of the problem, these methods are closely related to *quadrature* methods for numerical approximation of a definite integral (see [41]), and share with the latter a spectrum of possible approximating functions. Most of these methods are based on polynomial substitution:

$$\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big) \approx \boldsymbol{\pi}(\tau) := \sum_{i=0}^{q} \boldsymbol{\beta}_i \tau^i, \tag{2.76}$$

where $\boldsymbol{\pi}(\tau)$ is a vector of q^{th}-order polynomials, and $\boldsymbol{\beta}_0, \ldots, \boldsymbol{\beta}_q$ are vectors of constant coefficients in $\mathbb{R}^{\underline{f}}$. The polynomial coefficients are chosen to interpolate the values of $\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big)$ at a set of r *quadrature* points identified by the vector $\boldsymbol{\rho} = [\rho_1 \ \ldots \ \rho_r]^T$ within the interval $[0, 1]$, with $\rho_i < \rho_{i+1}$ for each i (unless $r = 1$); the elements of $\boldsymbol{\rho}$ represent the values of τ at the quadrature points. For an introduction to polynomial interpolation techniques the reader can refer to [1, 41]. The polynomial is thus integrated to produce an approximation of the integral

$$\boldsymbol{\gamma} := \int_0^1 \boldsymbol{\pi}(\tau) d\tau = \sum_{i=0}^{q} \frac{\boldsymbol{\beta}_i}{i+1}. \tag{2.77}$$

The fundamental difference between methods for numerical integration and for differential equations is the dependence of $\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big)$ on the integral function $\boldsymbol{x}(\tau)$. Within numerical integration the 'integrand' is not an explicit function of the 'integral'. When solving differential equations, the values of $\boldsymbol{x}(\tau)$ have to be estimated at the quadrature points before the function $\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big)$, and its approximant, can be evaluated. Therefore, the integral quadrature rule needs to be accompanied by a compatible interpolation rule, which provides the required values of $\boldsymbol{x}(\tau)$ at the quadrature points. Although not necessary, the quadrature and the interpolation rules may be based on the same polynomial approximation. These concepts will be clarified in the following section.

2.3.2.2 Runge-Kutta Schemes

The *Runge-Kutta* (RK) schemes are a family of one-step methods. The definite integral in (2.72) is approximated by a linear combination of the estimates of $\boldsymbol{f}\big(\tau, \boldsymbol{x}(\tau)\big)$ evaluated at the quadrature points; these estimates are denoted by $\boldsymbol{f}_i$ with i the index of the corresponding quadrature point. The general formulation of a r-stage RK scheme is

$$\bar{\boldsymbol{x}}(1) := \bar{\boldsymbol{x}}(0) + h \sum_{j=1}^{r} b_j \boldsymbol{f}_j, \tag{2.78}$$

with

$$\boldsymbol{f}_j := \boldsymbol{f}\left(h\rho_j, \boldsymbol{x}(0) + h \sum_{i=1}^{r} a_{ji} \boldsymbol{f}_i\right), \qquad j = 1, \ldots, r\,, \tag{2.79}$$

where h is the interval length ($h = 1$ for the interval $[0, 1]$), the a_{ji}'s and the b_j's are constant coefficients, and $\rho_j = \sum_{i=1}^{r} a_{ji}$ for $j = 1, \ldots, r$ (see [28, 30]). The formulae (2.78) and (2.79) define, respectively, the quadrature and the interpolation rules. A complete set of coefficients defines a specific RK scheme, which can be represented in the compact form

$$\begin{array}{c|ccc} \rho_1 & a_{11} & \cdots & a_{1r} \\ \vdots & \vdots & & \vdots \\ \rho_r & a_{r1} & \cdots & a_{rr} \\ \hline & b_1 & \cdots & b_r \end{array} \qquad \text{or} \qquad \begin{array}{c|c} \boldsymbol{\rho} & A \\ \hline & \boldsymbol{b}^T \end{array}, \tag{2.80}$$

also known as the *Butcher array* (see [15]). When the elements in the upper triangular part of A are all 0, that is $a_{ji} = 0$ for $i \geq j$, the associated RK scheme is said to be explicit. In this case the $\boldsymbol{f}_j$'s can be explicitly computed for increasing $j = 1, \ldots, r$ using (2.79). Otherwise, the evaluation of one $\boldsymbol{f}_j$ requires an estimate $\boldsymbol{f}_i$, with $i \geq j$, before the latter has been computed. In general, implicit schemes require the simultaneous solution of a system of r equations defined by (2.79). If the dynamics are nonlinear, the resulting root-finding problem might require an iterative numerical technique for a solution to be found (see Section 2.3.1.1). The initial root estimates can be generated through an explicit method with equal quadrature points, in what is called the *predictor* step. The iterative root-finding phase of an implicit method is instead called the *corrector* step.

The most popular RK scheme is the *Euler method*. Its application to the interval $[0, 1]$

produces

$$\bar{\boldsymbol{x}}(1) := \bar{\boldsymbol{x}}(0) + \boldsymbol{f}\big(0, \bar{\boldsymbol{x}}(0)\big), \tag{2.81}$$

with associated Butcher array

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}. \tag{2.82}$$

The Euler method interpolates the integrand with the line tangent to the function $\boldsymbol{f}\big(t, \boldsymbol{x}(t)\big)$ at the initial point. This is a 1-stage explicit method of order 1.

Some well-known implicit RK methods are based on the *backward Euler* ($p = 1$, $r = 1$), the *midpoint* ($p = 2$, $r = 1$) and the *trapezoidal* ($p = 2$, $r = 2$) rules, with respective Butcher arrays

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}, \qquad \begin{array}{c|c} 1/2 & 1/2 \\ \hline & 1 \end{array}, \qquad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}. \tag{2.83}$$

The order p of a RK method, as well as properties such as the *convergence* and *stability*, can be determined verifying certain conditions on the coefficients in the Butcher array (refer to [30] for further details).

2.3.2.3 Collocation

Collocation methods are a sub-class of the Runge-Kutta methods. In a collocation method the derivatives of the integral approximant $\bar{\boldsymbol{x}}(\tau)$ match the values of the integrand function $\boldsymbol{f}\big(\tau, \bar{\boldsymbol{x}}(\tau)\big)$ at the quadrature points, which in this case are called *collocation points* (see [30, p.27]):

$$\frac{d}{dt}\bar{\boldsymbol{x}}(h\rho_i) = \boldsymbol{f}\big(h\rho_i, \bar{\boldsymbol{x}}(h\rho_i)\big), \qquad i = 1, \dots, r. \tag{2.84}$$

The Butcher arrays in (2.82) and (2.83) describe collocation methods. The collocation nodes can be chosen arbitrarily in order to obtain desirable properties for the RK scheme. If the nodes are equi-spaced the rule belongs to the family of *Newton-Cotes formulae*. The order of the quadrature rule is maximised choosing the collocation points as the roots of an orthonormal polynomial; this choice identifies the family of *Gauss formulae*. The polynomials in a set $\{\pi_i(\tau) : i = 1, \dots, n\}$ are said to be *orthogonal* over the range

$[a, b]$ if they satisfy the condition

$$\int_a^b w(\tau)\pi_i(\tau)\pi_j(\tau)d\tau = \begin{cases} c_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}, \tag{2.85}$$

where $w(\tau)$ is a weighting function; variable substitution can be used to change the integration interval. Moreover, if $c_i = 1$ for any i, then the polynomials are *orthonormal*. A representative example of orthogonal collocation method uses the *Gauss-Legendre* quadrature, with collocation points located at the roots of the r^{th} order Legendre polynomial ($a = -1, b = 1, w(\tau) = 1, c_i = 1\ \forall i$) (see [1, p.65]); an r-stage Gauss-Legendre method has order $2r$ (see [1, p.215]). Orthogonal collocation methods are also known as *pseudospectral* methods.

There are also sub-optimal versions of the Gauss formulae where a number r_1 of nodes are chosen arbitrarily, and the remaining $r_2 = r - r_1$ nodes are selected optimally, which can guarantee order $r_1 + 2r_2$ for the quadrature method (see [1, p.70]).

Quadrature methods can be further distinguished depending on the location of the nodes with respect to the interval limits. If the initial and final points of the interval are collocation points, the method is a *Lobatto* method. If there is a collocation point only at one end of the interval, that is a *Radau* method. One particular example of a Lobatto 2-stage collocation method is the trapezoidal rule, which will be further adopted to solve optimal control problems.

As demonstrated in [15], Gauss collocation methods are special cases of RK methods, and they can be represented in the form of Butcher arrays. As an example, the 2-stage Gauss quadrature rule based on Legendre polynomials on the interval $[0, 1]$ has associated Butcher array

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}, \tag{2.86}$$

while the 1-stage method is the midpoint rule.

2.3.2.4 Error Control

The quadrature error is intimately related to the function $\boldsymbol{f}\big(t, \boldsymbol{x}(t)\big)$, its derivatives (see [1, 23]), and to the basic rule employed. There are a number of elements to consider in order to obtain both accuracy and efficiency in the numerical solution of an IVP. Recalling the definition (2.75), the accuracy of the solution can be improved either using higher order methods or shortening the quadrature intervals. However, the behaviour of $\boldsymbol{f}\big(t, \boldsymbol{x}(t)\big)$ might change significantly across the time and state domains, and the integration error may vary accordingly, increasing where the derivatives of $\boldsymbol{f}\big(t, \boldsymbol{x}(t)\big)$ change more rapidly. A shrewd approach would determine the length of each basic interval $[t_i, t_{i+1}]$ on the basis of the local properties of $\boldsymbol{f}\big(t, \boldsymbol{x}(t)\big)$; shorter intervals improve accuracy, longer favour the overall IVP solution speed. A finer level of control may involve the choice of the basic rule at each time interval. Some algorithms may also control the stage of the method. Efficiency is pursued equalising the error throughout the integration. The capability to control the quadrature error, still delivering computational efficiency, is the major technical challenge undertaken by general-purpose numerical IVP solvers.

Popular examples of explicit RK schemes with automatic error correction are based on the *Fehlberg method* (stage 4 and 5; see [1]) and the *Dormand-Prince method* (stage 5 and 6 used within the popular MATLAB function *ode45*; see [53]). According to the Fehlberg method, the quadrature error is estimated comparing the results given by a 4^{th} and a 5^{th} order methods associated with the Butcher array

$$\begin{array}{c|c} \boldsymbol{\rho} & A \\ \hline & \boldsymbol{b}_4^T \\ & \boldsymbol{b}_5^T \end{array}, \tag{2.87}$$

where $\boldsymbol{b}_4$ and $\boldsymbol{b}_5$ are the sets of coefficients for the 4^{th} and a 5^{th} order methods, respectively. The underlying idea is to maintain the error estimate below a certain threshold controlling the step length.

2.3.2.5 Differential-Algebraic Equations

An initial-value problem for a set of *Differential-Algebraic Equations* (DAE) is the problem of finding the vector-valued function $\boldsymbol{y}(t)$ that satisfies

$$\left.\begin{aligned} &\mathbf{f}\left(t, \boldsymbol{y}(t), \frac{d}{dt}\boldsymbol{y}(t)\right) = 0, \quad t \in [0,1] \\ &\boldsymbol{y}(0) = \boldsymbol{y}_0 \end{aligned}\right\}. \tag{2.88}$$

Such problems are, in general, harder to solve than the ones limited to the special case of explicit ODEs (2.71). In the first place, even if the function $\mathbf{f}$ were analytical, the first equation might not admit solutions, which gives rise to the problem of *solvability* of a DAE IVP. A comprehensive account of the concepts and conditions associated with DAEs and the numerical solution of IVPs can be found in [11].

An approximate solution to (2.88) may be sought by extending the RK interpolation rule (2.79) to

$$\mathbf{f}\left(h\rho_j, \boldsymbol{y}(0) + h\sum_{i=1}^{r} a_{ji}\mathbf{f}_i, \mathbf{f}_j\right) = 0, \qquad j = 1, \ldots, r\,; \tag{2.89}$$

here $\mathbf{f}_i$ is an estimate of $\frac{d}{d\tau}\boldsymbol{y}(\rho_i)$. The DAEs are enforced on the estimates of $\boldsymbol{y}(\tau)$ and $\frac{d}{d\tau}\boldsymbol{y}(\tau)$ at each quadrature node. These conditions show that any RK method for a DAE IVP is in general implicit, because it involves the solution of a set of nonlinear equations. Methods that were explicit for ODEs may become implicit for general DAEs, regardless of the Butcher matrix A. Like all implicit methods, these require predictor-corrector schemes.

Semi-explicit Differential-Algebraic Equations are a special family of DAEs described by the system

$$\left.\begin{aligned} \frac{d}{dt}\boldsymbol{x}(t) &= \boldsymbol{f}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t)\big) \\ 0 &= \mathbf{g}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t)\big) \end{aligned}\right\}, \tag{2.90}$$

where both *differential* equations and *algebraic* equations appear. Accordingly, the variables $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$ are referred to as 'differential' and 'algebraic'. It can be recognised that this formulation is compatible with the definition of dynamics and equality constraints in a continuous-time optimal-control problem (2.57). Note that, in order to solve

an associated IVP through a RK scheme, the control $\boldsymbol{u}(t)$ must be either prescribed or determined at the quadrature points. Similarly to what was done for the definition of necessary conditions for optimality, in Section 2.1.1, it is assumed that the functions $\boldsymbol{f}$ and $\mathbf{g}$ satisfy a regularity condition over their entire domains. Under this assumption a DAE IVP can be solved using a time-marching technique. In this case the rule (2.79) is extended to define the constraints

$$\left.\begin{aligned} \boldsymbol{f}_j &= \boldsymbol{f}\left(h\rho_j, \boldsymbol{x}(0) + h\sum_{i=1}^{r} a_{ji}\boldsymbol{f}_i, \boldsymbol{u}(\rho_j)\right) \\ 0 &= \mathbf{g}\left(h\rho_j, \boldsymbol{x}(0) + h\sum_{i=1}^{r} a_{ji}\boldsymbol{f}_i, \boldsymbol{u}(\rho_j)\right) \end{aligned}\right\}, \qquad j = 1, \ldots, r\,. \tag{2.91}$$

2.3.3 Transcription

Consider the constrained functional-optimisation problem, including differential, algebraic and boundary constraints:

$$\min_{\{\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p}\}} J\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big)$$
$$\text{s.t.} \quad \begin{cases} \dfrac{d\boldsymbol{x}(t)}{dt} &= \boldsymbol{f}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(t, \boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) \\ 0 &= \mathbf{g}_b\big(\boldsymbol{x}(0), \boldsymbol{x}(1), \boldsymbol{p}\big) \end{cases}, \tag{2.92}$$

with the cost functional $J\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big)$ given in the *Lagrange form*

$$J\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big) := \int_0^1 L\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}\big)dt. \tag{2.93}$$

It is evident that a BVP is a special case of (2.92).

> *"In any computation carried out on a digital computer, we can do no better than to obtain a finite set of real numbers. Thus in solving a continuous optimal control problem ... we are forced to resort to some form of discretisation." [16]*

The fundamental *transcription* problem is the problem of converting (2.92) into an approximating parameter-optimisation problem.

In the definition (2.92), the boundary conditions are *mixed* between the initial and the final time, as they may affect the values of the problem functions at both the boundaries

simultaneously; if the entries of the vector $\mathbf{g}_b$ constrain independently inital and final conditions, then they are called *separated* boundary conditions. Similarly to the DAE IVP case, the solvability of the problem has to be studied. If a solution exists, but cannot be found in closed form, an approximation can be sought through numerical methods. A major difference from an IVP is that the boundary conditions may be defined at both ends of the horizon. The function $\boldsymbol{x}(t)$ may thus not be entirely available at one boundary. This means that a standard marching technique cannot be initialised at any point, and the numerical methods for IVP are not directly applicable.

This problem also involves the minimisation of a cost functional. The vector-valued function $\boldsymbol{u}(t)$ and the vector $\boldsymbol{p}$ are decision variables in such minimisation. Moreover, if not subject to boundary constraints, either $\boldsymbol{x}(0)$ or $\boldsymbol{x}(1)$ may become optimisation variables too. It is assumed that a subset $\boldsymbol{u}_c(t)$ of the control vector is uniquely determined by the algebraic constraints $\mathbf{g}$, and the remaining unconstrained $\boldsymbol{u}_u(t)$ can be chosen arbitrarily. The same separation is assumed for the parameter vector $\boldsymbol{p}$ with respect to the point constraints $\mathbf{g}_b$.

In order to apply numerical IVP solution methods one needs to *guess* a complete estimate of $\boldsymbol{x}(t)$ at any one boundary; values for the unconstrained parameters $\boldsymbol{p}_u$, and a trajectory for $\boldsymbol{u}_u(t)$, must be guessed too. Once these quantities are defined, the trajectories of $\boldsymbol{x}(t)$ and $\boldsymbol{u}_c(t)$ can be determined solving the resulting DAE IVP.

The trajectory of the unconstrained control shall be defined in terms of a finite set of parameters. A standard choice is to use a polynomial function

$$\boldsymbol{u}_u(t) := \boldsymbol{\pi}(t) = \sum_{i=0}^{q} \boldsymbol{\beta}_i t^i, \tag{2.94}$$

where $\boldsymbol{\beta}_i \in \mathbb{R}^{u_u}$. In order to simplify the notation, the matrix of polynomial coefficients $B = [\boldsymbol{\beta}_0 \cdots \boldsymbol{\beta}_q]$ is introduced. Alternative parametrisations can be used if deemed appropriate.

Transcription methods are in fact based on an idea not dissimilar to the predictor-corrector scheme, which starts from a solution guess and then refines it iteratively. The quality of an initial guess is a critical issue, and is generally recognised as a limiting

feature of any numerical scheme. There is no general theory to assist the guessing process, although in many cases some 'common sense' is sufficient to obtain good results. This topic will be further discussed for special applications in the subsequent chapters.

Transcription methods are essentially made of two ingredients:

1. a procedure for the definition of a finite set of parameters that approximate continuous functions and

2. a procedure that converts the differential relations into algebraic constraints on the parameters.

Depending on the choice of these procedures, the methods can be classified into three families [1, 8]:

1. *single-shooting*,

2. *multiple-shooting* and

3. *collocation* (or *finite-differencing*).

2.3.3.1 Single-Shooting

> *[8, p.94]: " An early practical application of the method required that a cannon be aimed such that the cannonball hit its target, hence explaining the colorful name."*

A single-shooting method consists of the iteration of two steps:

1. *'aim'* – guess a complete initial condition for the function $\bar{\boldsymbol{x}}(0)$, a parametrisation $\boldsymbol{\pi}(\cdot)$ for the decision function $\boldsymbol{u}_u(\cdot)$, and a set of parameters $\boldsymbol{p}$;

2. *'shoot'* – solve the IVP for the selected guess.

These are repeated for improved guesses until the trajectory satisfies all the constraints and is qualified as a minimum. This method can be formalised using concepts introduced in former sections. Consider the IVP defined only by the equality constraints in (2.92), excluding constraints acting at the final boundary — standard IVP numerical techniques

do not admit inequality constraints: an IVP solution method maps a set of initial conditions, control functions and parameters into final estimates $\bar{\boldsymbol{x}}(1)$; the function $\boldsymbol{\psi}$ defines such a map:

$$\bar{\boldsymbol{x}}(1) = \bar{\boldsymbol{x}}(0) + \boldsymbol{\psi}\big(\bar{\boldsymbol{x}}(0), \boldsymbol{u}_u(\cdot), \boldsymbol{p}\big). \tag{2.95}$$

The same IVP technique can be applied to estimate the value of the cost function

$$\bar{J} = \theta\big(\bar{\boldsymbol{x}}(0), \boldsymbol{u}_u(\cdot), \boldsymbol{p}\big). \tag{2.96}$$

A single-shooting method thus transforms the problem (2.92) into

$$\begin{aligned} &\min_{\{\bar{\boldsymbol{x}}(0), B, \boldsymbol{p}\}} \bar{J} \\ &\text{s.t.} \begin{cases} 0 = \bar{\boldsymbol{x}}(1) - \bar{\boldsymbol{x}}(0) - \boldsymbol{\psi}\big(\bar{\boldsymbol{x}}(0), B, \boldsymbol{p}\big) \\ 0 \geq \mathbf{g}\big(0, \bar{\boldsymbol{x}}(0), B, \boldsymbol{p}\big) \\ 0 \geq \mathbf{g}\big(1, \bar{\boldsymbol{x}}(1), B, \boldsymbol{p}\big) \\ 0 \geq \mathbf{h}\big(0, \bar{\boldsymbol{x}}(0), B, \boldsymbol{p}\big) \\ 0 \geq \mathbf{h}\big(1, \bar{\boldsymbol{x}}(1), B, \boldsymbol{p}\big) \\ 0 = \mathbf{g}_b\big(\bar{\boldsymbol{x}}(0), \bar{\boldsymbol{x}}(1), \boldsymbol{p}\big) \end{cases}, \end{aligned} \tag{2.97}$$

where $\boldsymbol{u}_u(\cdot)$ was replaced by its parameterisation, defined in (2.94) as a polynomial function with coefficients in B. The redefined problem (2.97) is a parameter-optimisation problem.

Using standard IVP methods the inequality constraints are enforced at the boundaries only. This may not represent a problem when the original problem is a two-point BVP, but it is a limitation when the solution admits changes in the active constraint set (see Section 2.2.2).

The extent to which the transcribed problem approximates the original is intimately related to the accuracy of the quadrature techniques. It is also true that, if the IVP solver introduces noise in the approximated cost and constraint functions, the convergence of the ultimate NLP solution might be significantly degraded. This may happen when the IVP method includes step-length and order adaptation techniques for error control (see [8]). A transcription method is said to be *consistent* if it performs the same sequence of operations for any given set of arguments. The inappropriate choice of an IVP method may

result in an *inconsistent* transcription. In general, consistency is preferable to catalyse the convergence of the numerical optimisation (or root-finding).

If the dynamic equations are unstable, the solution of the IVP may diverge rapidly for most choices of initial conditions. Amongst the issues associated with dynamic instabilities the most critical are:

1. high sensitivity of the cost and constraints with respect to the initial guess,
2. large truncation error in the quadrature scheme, and
3. amplified round-off error due to finite machine precision.

These effects possibly result in a failure to converge to a solution of the transcribed NLP.

One practicable remedy is to guess the state value at the end of the horizon $\bar{\boldsymbol{x}}(1)$ and solve a final-value problem instead, integrating backwards in time. This approach could circumvent the stability issues if the time-reversed dynamics are not unstable. Unfortunately, two-point BVP defined as the necessary conditions for optimality of a continuous-time OC problem (in Section 2.2.2) are often characterised by unstable dynamics both forwards and backwards in time (see [13]). When this is the case, an alternative to single-shooting should be sought in order to mitigate the difficulties associated with unstable dynamics.

2.3.3.2 Multiple-Shooting

An intuitive solution attempted to overcome the instability problems associated with single-shooting was to subdivide the optimisation horizon into a series of N shorter intervals. A time mesh is defined $\boldsymbol{t} = [t_0\ t_1\ \dots\ t_N]^T$, with $t_0 = 0$ and $t_N = 1$. The initial state $\bar{\boldsymbol{x}}_i(t_{i-1})$ and the control parametrisation B_i are guessed for each interval. The parameter vector $\boldsymbol{p}$ is also guessed. A single-shooting method is thus applied within each interval to estimate the state at the final boundary $\bar{\boldsymbol{x}}_i(t_i)$.

Similarly to the single-shooting transcription, the functions $\boldsymbol{\psi}$ and θ can be defined

for given $i = 1, \dots, N$:

$$\bar{\boldsymbol{x}}_i(t_i) = \bar{\boldsymbol{x}}_i(t_{i-1}) + \boldsymbol{\psi}_i\big(\bar{\boldsymbol{x}}_i(t_{i-1}), B_i, \boldsymbol{p}\big), \qquad \bar{J}_i = \theta_i\big(\bar{\boldsymbol{x}}_i(t_{i-1}), B_i, \boldsymbol{p}\big) \tag{2.98}$$

The continuity of $\bar{\boldsymbol{x}}(t)$ across the intervals is enforced imposing the conditions $\bar{\boldsymbol{x}}_{i+1}(t_i) = \bar{\boldsymbol{x}}_i(t_i)$ for $i = 1, \dots, N-1$. Such conditions allow one to define a matrix of guesses for the state values at the mesh points $\bar{X} = [\bar{\boldsymbol{x}}_0 \ \dots \ \bar{\boldsymbol{x}}_N]$, having $\bar{\boldsymbol{x}}_1(t_0) = \bar{\boldsymbol{x}}_0$ and $\bar{\boldsymbol{x}}_i(t_i) = \bar{\boldsymbol{x}}_i$ for $i = 1, \dots, N$. This is effectively a concatenation of single-shooting solutions, also known as *multiple-shooting*. This method can be employed for the solution of *multi-point* Boundary-Value Problems.

A multiple-shooting method transcribes the original problem (2.92) into[11]

$$\begin{aligned} &\min_{\{\bar{X}, B_1, \dots, B_N, \boldsymbol{p}\}} \sum_{i=1}^{N} \bar{J}_i \\ &\text{s.t.} \left\{ \begin{array}{l} 0 = \bar{\boldsymbol{x}}_i - \Big(\bar{\boldsymbol{x}}_{i-1} + \boldsymbol{\psi}_i\big(\bar{\boldsymbol{x}}_{i-1}, B_i, \boldsymbol{p}\big)\Big) \\ 0 = \mathbf{g}\big(t_{i-1}, \bar{\boldsymbol{x}}_{i-1}, B_i, \boldsymbol{p}\big) \\ 0 \geq \mathbf{h}\big(t_{i-1}, \bar{\boldsymbol{x}}_{i-1}, B_i, \boldsymbol{p}\big) \end{array} \right\} \ i = 1, \dots, N \\ &\quad \left\{ \begin{array}{l} 0 = \mathbf{g}\big(1, \bar{\boldsymbol{x}}_N, B_N, \boldsymbol{p}\big) \\ 0 \geq \mathbf{h}\big(1, \bar{\boldsymbol{x}}_N, B_N, \boldsymbol{p}\big) \\ 0 = \mathbf{g}_b\big(\bar{\boldsymbol{x}}_0, \bar{\boldsymbol{x}}_N, \boldsymbol{p}\big) \end{array} \right. \end{aligned} . \tag{2.99}$$

which can be solved through standard NLP techniques.

This method generates a transcribed problem with an increased number of decision variables and constraints compared to single-shooting; however, the sensitivity of these functions with respect to the initial guesses is reduced, which is crucial in the case of unstable dynamics. Since the shooting problems within the sub-intervals are mutually independent they can be solved in parallel, which allows for a reduction in the computation time of the associated local costs and constraints; this feature motivates the alternative name of *parallel shooting*.

[11] In the proposed multiple-shooting transcription, no assumption is made on the continuity of the control at the mesh points. Path constraints are selectively enforced for the control values evaluated on the right-hand-side of the intermediate mesh points. Alternative definitions are possible where appropriate.

2.3.3.3 Collocation

In the limit case, the IVP within each sub-interval of a multiple-shooting transcription can be solved using a single one-step rule. A method of this type can be referred to as a *collocation*[12] method.

The method can be formulated recalling the definition (2.80) of a Butcher array. The same notation used for multiple-shooting methods is adopted here for the segmentation of the time interval. The vector of time increments $\Delta\boldsymbol{t}$ is defined as $\Delta\boldsymbol{t} = [t_1 \ \dots \ t_{N-1} \ \ 1] - [0 \ \ t_1 \ \ \dots \ \ t_{N-1}]$. In addition, a quadrature rule of stage r_i, with Butcher coefficients $\{\boldsymbol{\rho}_i, \boldsymbol{b}_i, A_i\}$, is assigned to each interval i. A collocation method consists of finding the solution to the following parameter-optimisation problem:

$$
\begin{aligned}
&\min_{\{\bar{X},\bar{F},B_1,\dots,B_N,\boldsymbol{p}\}} \sum_{i=1}^{N} \bar{J}_i \\
&\text{s.t.} \quad \left\{ \begin{array}{l}
0 = \bar{\boldsymbol{x}}_i - \left(\bar{\boldsymbol{x}}_{i-1} + \Delta t_i \sum_{j=1}^{r_i} ({}_ib_j)({}_i\boldsymbol{f}_j) \right) \\
0 = ({}_i\boldsymbol{f}_j) - \boldsymbol{f}\left(t_{i-1} + \Delta t_i({}_i\rho_j), \right. \\
\qquad\qquad \left. \bar{\boldsymbol{x}}_{i-1} + \Delta t_i \sum_{k=1}^{r_i} ({}_ia_{jk})({}_i\boldsymbol{f}_k), B_i, \boldsymbol{p} \right), \quad j = 1, \dots, r_i \\
0 = \mathbf{g}\Big(t_{i-1}, \bar{\boldsymbol{x}}_{i-1}, B_i, \boldsymbol{p} \Big) \\
0 \geq \mathbf{h}\Big(t_{i-1}, \bar{\boldsymbol{x}}_{i-1}, B_i, \boldsymbol{p} \Big) \\
0 = \mathbf{g}\Big(1, \bar{\boldsymbol{x}}_N, B_N, \boldsymbol{p} \Big) \\
0 \geq \mathbf{h}\Big(1, \bar{\boldsymbol{x}}_N, B_N, \boldsymbol{p} \Big) \\
0 = \mathbf{g}_b\Big(\bar{\boldsymbol{x}}_0, \bar{\boldsymbol{x}}_N, \boldsymbol{p} \Big)
\end{array} \right\} \quad i = 1, \dots, N
\end{aligned}
\tag{2.100}
$$

where $\bar{F} = [({}_1\boldsymbol{f}_1) \ \dots \ ({}_1\boldsymbol{f}_n) \ \ ({}_2\boldsymbol{f}_1) \ \dots \ ({}_N\boldsymbol{f}_1) \ \dots \ ({}_N\boldsymbol{f}_N)]$ is the matrix of guesses for the derivative vectors at the collocation points; $({}_ia_{jk})$ is the element in the j^{th} row and k^{th} column of A_i, $({}_ib_j)$ is the j^{th} element of $\boldsymbol{b}_i$ and $({}_i\rho_j)$ is the j^{th} element of $\boldsymbol{\rho}_i$. The vec-

[12] Strictly speaking, for a transcription method to be of '*collocation*' type, the quadrature points should also be collocation points. However, this nomenclature has been extended to methods that include the values of the state (and the control) at the quadrature points as decision parameters of the final NLP problem. Sometimes, the generic name 'transcription' is used to refer to methods of this type.

tors $({}_i\boldsymbol{f}_j)$ represent the approximations of $\boldsymbol{f}$ at the time instants associated with each $({}_i\rho_j)$. Alternatively to the estimates of $\boldsymbol{f}$ one could use the estimates of $\boldsymbol{x}$ at the collocation points as decision variables, which allows one to impose path constraints within the collocation intervals. Each scalar $\bar{J}_i$ is the local cost measured for the i^{th} interval, approximated using the same collocation method applied to the dynamics; an alternative collocation method can be used for the cost if deemed appropriate. This formulation applies to explicit ODEs. However, the extension to implicit DAEs is possible using the concepts introduced in Section 2.3.2.5.

In a collocation method the sensitivity with respect to dynamic instabilities can be reduced shortening the quadrature segments. The fact that each constraint in (2.100) affects only adjacent segments avoids the propagation of the sensitivity with respect to any state or control guess (or current estimate) across the intervals. In addition, the Jacobian and the Hessian of the resulting optimisation Lagrangian with respect to the decision variables are sparse. Notoriously, numerical methods that can exploit this sparsity are considerably more efficient (see [8]). Analysing (2.100) it can be argued that the sparsity depends on the number of sub-intervals N and on choice of the A_i's. Gaussian quadratures are characterised by full-rank matrices in the Butcher array, which entails the presence of dense matrix blocks associated with the corresponding segments. Although Gaussian quadrature methods ensure high accuracy, a trade-off between accuracy and efficiency must be sought in practice limiting the stage of the quadrature rules and increasing the number of sub-intervals.

2.3.4 Direct and Indirect Methods

Numerical methods for OC are mathematical methods that combine techniques of numerical approximation and optimisation. These methods were briefly introduced at the beginning of Section 2.3, and broadly portrayed in Figure 2.1. Numerical approximation techniques for infinite-dimensional problems were described in Section 2.3.3. What is left to discuss is how known optimisation techniques, discussed in Sections 2.1 and 2.2 are

exploited for the solution of the OC problem.

One possible approach is to apply a transcription method 'directly' to the OC problem. One such approach is referred to as a *direct* method. An alternative approach is to seek optimality for the OC problem imposing the optimality conditions from the calculus of variations (see Section 2.2.2); a transcription method is then applied to solve the resulting BVP. Since this solution avenue considers an intermediate BVP problem before applying transcription, methods of this kind are called *indirect*.

According to [12], indirect methods appeared in the OC literature prior to direct ones. This was probably an effect of the strong attention devoted to optimality conditions through variational techniques at the dawn of OC theory. The lack of powerful computing facilities may have affected this trend too. However, direct methods have later become more appealing due to their relative simplicity and practicality. The duality of the two approaches is also being increasingly recognised within the OC community. None of the methods is generally accepted as superior to the other, and the choice is based on the particular OC application or even on the personal taste of the user.

Both direct and indirect methods require at some some stage the transcription of the continuous-time OC problem; any suitable transcription technique can be employed. For instance an indirect multiple-shooting method fist generates a continuous-time BVP, from the necessary conditions of the CV, and then transforms it into a root-finding problem, using the multiple-shooting transcription technique. A direct collocation method would instead apply collocation to directly convert continuous time OC into constrained parameter optimisation. Six possible combinations for the transcription techniques introduced in Section 2.3.3 can be recognised.

Here some of the distinctive features of the alternative paradigms are presented, with the intent to motivate the choice of direct collocation for the solution of OC problems in this thesis. The methods can be compared considering four features:

1. the nature of the resulting problem and the available solution techniques;

2. the problem size, in terms of number of constraints and decision parameters;

3. the problem complexity, in terms of function evaluations;

4. the optimality qualification of the solution.

Problem type – Indirect methods require the numerical solution of a BVP, which is generally recognised as harder than the solution of a DAE IVP. This is due to the fact that in a BVP some of the constraints are defined only at the boundaries, with no immediate information on their implications on the set of feasible system trajectories. In a DAE IVP the algebraic constraints affect the state and control throughout the whole horizon, which allows one to guarantee the satisfaction of the path constraints within the incremental numerical integration process. One advantage of direct methods is that they avoid the definition of boundary constraints on the costate that are not accompanied by algebraic path constraints.

Within the indirect approach it is hard to guess trajectories both for the states and the costates that are 'close enough' to a solution (see [13]). Indirect methods regard the feasibility of the BVP solution as a measure for optimality, and the associated root-finding problem might have a relatively small region of convergence. On the contrary, a dynamically feasible initial guess for a direct method can be defined by solving an IVP, starting from a feasible initial state guess and a feasible control parametrisation. In practice, a feasible sub-optimal solution is better than an infeasible one. It should be pointed out that this is not generally true when final boundary constraints are imposed at the end of the horizon or inequality constraints are considered, in which cases the solution of an IVP might not be sufficient for the feasibility of the initial guess. In any case, the number of boundary constraints to be dealt with using a direct approach is not greater than for an indirect method.

Problem size – The introduction of the costate dynamics doubles the size of the system state and of the dynamic constraints. This is a disadvantage associated with indirect methods, because the number of variables and constraints in the final NLP problem will

in turn increase. Direct methods do not suffer from this condition, because the costate dynamics do not appear explicitly in the solution procedure.

Information on the costate might be important if a sensitivity analysis is of interest. Under these circumstances an indirect method may seem more appealing. However, estimates for the trajectory of the costate can be obtained within a direct solution too. These are extrapolated from the values of the Lagrange multipliers associated with the discretised counterparts of the dynamic constraints. This methodology is explained in [8], and an example is given in Section 2.3.4.1 for a particular choice of basic rule.

The number of optimisation parameters in the transcribed NLP is proportional to the number of sampling nodes defined within the chosen transcription scheme. This is evident from the definitions in Section 2.3.3: a single-shooting method has the fewest optimisation variables and constraints, which increase in number for multiple-shooting and collocation. The problem size obviously translates into memory requirements when the solution method is implemented on a computer. Another critical issue connected with the length of the decision vector is the computational burden associated with the solution of the linear system within the NLP iterations (see Section 2.3.1.1 and 2.3.1.2). However, it is worth stressing that the described transcription methods guarantee a degree of sparsity of the matrices that define the linear system; this is a consequence of the horizon segmentation, which avoids mutual dependencies between parameters from non-adjacent segments. The choice of low order basic rules enhances sparsity. A more detailed discussion on how to exploit sparsity can be found in [8].

Problem complexity – The addition of the costate dynamics within an indirect method increases the complexity of the solution on two levels: first, the adjoint equations must be derived from the optimality conditions; second, they have to be computed (possibly with their derivatives) at each iteration of the numerical optimisation. This additional complexity does not appear in direct methods.

The complexity of the constraints in the final parameter optimisation problem is in-

evitably dependent on the transcription technique. For a given system of differential equations, the accurate numerical solution of an IVP is arguably more complicated for increasing length of the horizon. In general, collocation methods, as opposed to shooting, generate problem transcriptions with simpler constraints. The choice of the basic rule affects the complexity of the constraints too, with simple constraints for low-order rules.

Optimality – Indirect methods apply necessary optimality conditions, thus satisfying rigorous criteria for the qualification of a solution. It might be argued that a solution obtained via direct methods is not corroborated by any qualification criterion for the original continuous-time OC problem, and that optimality must be verified a posteriori using variational techniques. However, even within indirect methods the eventual solution is sought numerically, and the necessary conditions for optimality at continuous time will only be satisfied for a finite-dimensional approximation of the BVP. Furthermore, it can be proved that the KKT conditions within a direct method converge to the necessary variational conditions for functional optimality for increasing resolution of the transcription process (see [7]). These facts suggest that the optimality of a solution can not be in general guaranteed for either approach, and that it primarily depends on the accuracy of the transcription.

Motivated by the above discussion, direct collocation methods will be used for the numerical optimal-control calculations within this thesis. A direct approach is meant to expand the region of convergence and to avoid the introduction of costate dynamics. The choice of collocation with a large number of mesh points negates sensitivity problems and reinforces the convergence of the NLP solution. Short integration segments also guarantee the sparsity of the matrices fed to the NLP linear solver, and a balanced trade-off between the constraint complexity and the problem size. In the subsequent optimal control applications it will become evident that the complexity of dynamics and constraints represents a primary challenge, and that the computation speed is more critical than the size for

most of the analysed problems.

2.3.4.1 Euler-Collocation Equivalence

As a final remark, it is demonstrated that the direct and indirect approaches are equivalent if consistent transcription methods are used. In particular, collocation based on the Euler approximation is considered as a case-study. This will draw a clear parallel between continuous-time OC and the control of multi-stage systems as presented in Section 2.1.3.

Recall the definition of a continuous-time optimal-control problem given in (2.57). A direct method for optimal control starts with the conversion of the problem into a finite-dimensional parameter space. First, a set of intervals are defined, on which the quadrature rule shall be applied. These intervals are delimited by a finite set of time instants within the optimisation horizon; these are collected in the vector $\boldsymbol{t} = [t_0 \ t_1 \ \dots \ t_N]^T$ with $t_i < t_j$ for $i < j$, $t_0 = 0$ and $t_N = 1$. The vector of time increments is defined as $\Delta\boldsymbol{t} = [t_1 \ \dots \ t_N] - [t_0 \ \dots \ t_{N-1}]$. Then, one defines a vector of state samples at the time instants in $\boldsymbol{t}$, and of control samples at $[0 \ \dots \ t_{N-1}]$:

$$\bar{X} = \left[\ \boldsymbol{x}_0 \ \ \cdots \ \ \boldsymbol{x}_N \ \right], \qquad \bar{U} = \left[\ \boldsymbol{u}_0 \ \ \cdots \ \ \boldsymbol{u}_{N-1} \ \right]. \tag{2.101}$$

Using the explicit Euler rule the problem is transcribed into

$$\begin{aligned} &\min_{\{\bar{X},\bar{U},\boldsymbol{p}\}} \ J\big(\bar{X},\bar{U},\boldsymbol{p}\big) \\ &\text{s.t.} \quad \begin{cases} \boldsymbol{x}_{i+1} &= \boldsymbol{x}_i + \Delta t_i \, \boldsymbol{f}\big(t_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(t_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(t_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{p}\big) \end{cases} , \quad i = 1, \dots, N-1, \end{aligned} \tag{2.102}$$

with

$$J\big(\bar{X},\bar{U},\boldsymbol{p}\big) = \sum_{i=1}^{N-1} \Delta t_i \, L\big(t_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{p}\big), \tag{2.103}$$

which is an N-stage system of the form (2.36), and for which an optimal control problem can be solved applying the techniques presented in Sections 2.1.3 and 2.3.1. A numerical solution will thus satisfy the conditions (2.38).

An indirect method applies the optimality conditions for the continuous optimal-control problem. The conditions (2.60) are used to define a BVP that may then be solved

either in closed form or through numerical methods. Now consider a finite-dimensional approximation of the equations in (2.60); the state dynamics are approximated using the explicit Euler rule, while the adjoint equations are approximated using the implicit (backwards) Euler rule. It is easily recognised by inspection that, with this choice of approximation schemes, the discretised optimality conditions (2.60) are equivalent to the optimality conditions (2.38) for the aforementioned multi-stage optimal-control problem (2.102).

2.4 Summary

This chapter begins with a review of the fundamental theory for parameter optimisation, the calculus of variation and optimal control. The first- and second-order necessary and sufficient conditions for optimality are derived for finite-dimensional problems with equality and inequality constraints. Multi-stage optimal-control problems are solved as a sub-class of the general parameter-optimisation problem.

The fundamental problem in the calculus of variation is considered as an infinite-dimensional extension of parameter optimisation. The optimality conditions for functional-optimisation problems are stated in the form of the Euler-Lagrange equation. The calculus of variation is then applied to derive the equations of motion for a particle in a conservative force field from the principle of stationary action in classical mechanics. Optimal control is also treated using arguments from the calculus of variations, accounting for equality and inequality constraints, variable system parameters and variable time horizon.

The numerical methods for the solution of continuous-time optimal-control problems are then reviewed. First, gradient-descent methods for root-finding and optimisation are introduced. Second, quadrature methods for numerical integration are employed in the solution of initial-value problems. These involve the description of techniques for the sampling of the integration horizon and for the discretisation of the differential equations. A series of basic quadrature rules are given in the form of Runge-Kutta schemes.

The single-shooting, multiple-shooting and collocation paradigms are described for

the transcription of a functional-optimisation problem with a Lagrange cost, and multiple classes of constraints. These approaches are compared, and the choice of direct collocation for subsequent applications is motivated. The final results prove the equivalence of the direct and indirect approaches to numerical optimal control for a given quadrature rule.

CHAPTER 3

Optimal Control of Nonholonomic Systems

"The straight line is certainly the line of shortest distance between A and B, but it is not the one which is travelled in the shortest time.", Johan Bernoulli.

DURING track cycling competitions it is not uncommon to see athletes riding low and then high on steeply banked velodrome tracks, since riding high is an effective way of storing gravitational potential energy. At an opportune moment the high rider can then attack the field by 'diving', thereby translating potential energy into speed that can be used to overtake other riders. Aircraft pilots are also faced with the problem of stored energy management. In air combat a pilot may choose to break off an engagement and escape by diving to provide a speed advantage; this is another example of 'buying' speed with potential energy. Alternatively, by climbing, a pilot can use gravity to reduce the aircraft's speed and prevent an overshoot of the target.

The present work is motivated by an interest in fuel conserving strategies for road vehicles that make use of the interchange between potential and kinetic energy due to height variations in a gravitational field. With such vehicular problem in mind, a series of optimal control problems for mechanical systems are considered. In some cases analytical solutions are possible, while in others, particularly those involving nonholonomic[1]

[1] A constraint on a dynamic system is *holonomic* when it does not affect state derivatives; it is *nonholonomic* otherwise.

constraints, one must resort to numerical optimisation techniques.

The earliest problem of this type was posed by Johann Bernoulli in 1696, when he asked about the trajectory taken by a bead moving in a uniform gravitational field between prescribed points in least time. This problem, reported in its original formulation in Figure 3.1, is known as the *brachistochrone problem* (least time).

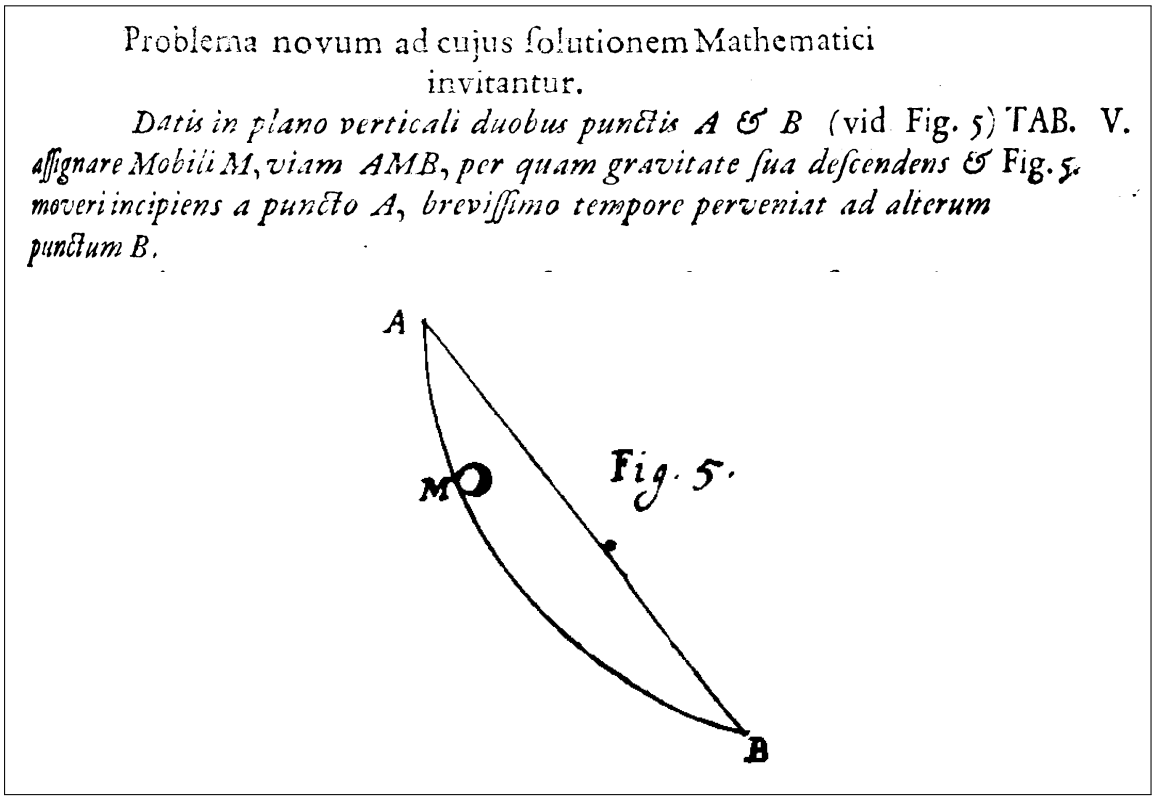

Figure 3.1: Original definition of the brachistochrone problem, published by Johan Bernoulli in June 1696 in the 'Acta Eruditorum' [3].

The brachistochrone problem was subsequently solved by Isaac Newton, Jacob Bernoulli, Gottfried Leibniz and Guillaume de L'Hôpital, who all showed the solution to be a cycloid — the curve traced out by a point on the rim of a rolling disc. The interested reader can refer to [79] for an excellent review of the history and solution techniques associated with this problem. Techniques based on optimal control are of particular interest, since these methods can be adapted to solve far more complicated problems of this type.

The original brachistochrone problem has been modified and extended by many authors over its three-hundred-year history, with solutions coming from the calculus of vari-

ations, optimal-control theory and differential geometry. Interesting variants include the constrained brachistochrone problem (see [21]), the unrestrained brachistochrone problem (see [76, 75]), the unrestrained rolling brachistochrone problem (see [90]), the brachistochrone with Coulomb friction (see [32]), the quantum brachistochrone problem (see [17]), the relativistic brachistochrone problem (see [26]), the giant brachistochrone problem (see [57]) and the brachistochrone problem with drag (see [85]). The fundamental and evergreen nature of this problem and its variants is widely recognised.

Section 3.1.1 offers a brief review of an optimal-control solution to the classical brachistochrone problem. Section 3.1.2 presents an optimal-control study of the unrestrained brachistochrone problem. In the unrestrained case the bead slides along the track surface, and contact with the track is maintained using an inequality constraint on the contact-maintaining reaction force. This is in contrast to the classical problem where the bead is constrained to the track kinematically. In Section 3.1.3 a step towards road-vehicle-relevant problems is taken by replacing the unrestrained brachistochrone's point mass with a rolling body. This problem is still two-dimensional, but it contains rotational inertia effects and a rolling contact. This is referred to as the rolling unrestrained brachistochrone, that was first studied in [90]. The problem formulation is extended to allow for a wider set of terminal conditions as well as correct some technical errors in the original treatment. Closed formulae for the optimal transition times are also derived. The theoretical optimal control solutions are compared to numerical results obtained using direct collocation in Section 3.1.3.3. The remainder of the chapter relates to the numerical solution of minimum-time optimal control-problems for rolling bodies in three dimensions. Section 3.2 studies the problem of a disc rolling between pre-specified points on the interior surface of a hemispherical shell. The results are given in Section 3.2.1, with a summary of the chapter and concluding remarks in Section 3.3. The contents of this chapter are published in [67].

3.1 The Brachistochrone

3.1.1 The Classical Problem

In Figure 3.2 a bead is allowed to slide along a wire, in a uniform gravitational field, between the starting point **A** and the prescribed terminal point **B**. Without loss of generality the origin of the inertial coordinate system is located at point **A**. The bead has a pre-specified initial velocity v_0 (usually assumed to be zero) and the control problem is to find the shape of the wire so that the bead travels from **A** to **B** in minimum time.

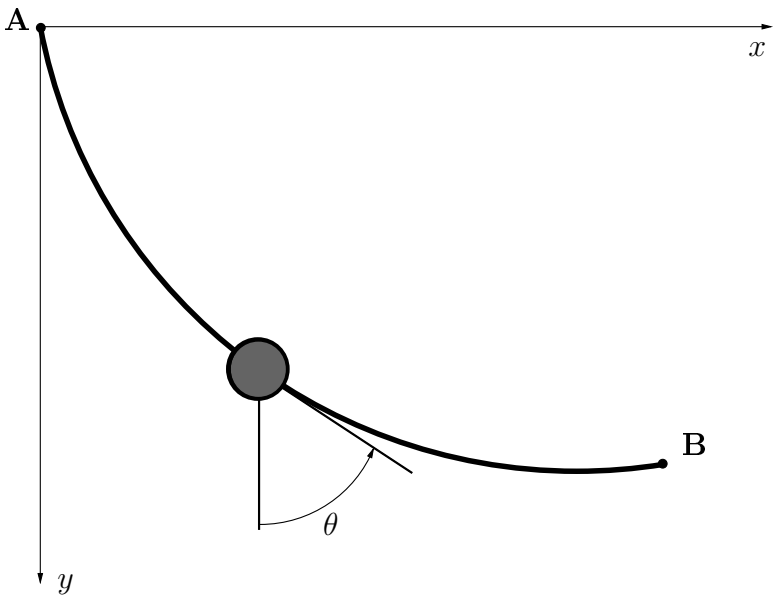

Figure 3.2: The classical brachistochrone problem.

It follows from simple Newtonian mechanics that the optimal-control formulation of the Brachistochrone problem is:

$$\min_{\theta\in[0,\pi]} \int_0^{t_f} dt$$
$$\text{subject to } \begin{cases} \dot{x} = v\sin\theta, & x(t_0) = 0, \quad x(t_f) = x_f \\ \dot{y} = v\cos\theta, & y(t_0) = 0, \quad y(t_f) = y_f \\ \dot{v} = g\cos\theta, & v(t_0) = v_0, \end{cases} \tag{3.1}$$

in which t_f is the final time, v_0 is the initial speed, g is the acceleration due to gravity, x and y are the Cartesian coordinates of the bead, being (x_f, y_f) the coordinates of **B**, v is the bead's speed and θ is the angle between the velocity vector and the y-axis (measured

anticlockwise). The variables x, y, v represent the state of the system and θ is the control. The states and control are all functions of time.

It will be shown that a cycloid satisfies the first-order necessary conditions for optimality. From the constraints in (3.1) one obtains

$$v\dot{v} = g\dot{y}, \tag{3.2}$$

which can be solved by separating the variables to obtain

$$v = \sqrt{2g\,(y + y_d)}, \tag{3.3}$$

where $y_d = v_0^2/2g$ is the value of y at which the mass has zero kinetic energy.

If one defines $\bar{y} = y + y_d$, the constraints in (3.1) can be replaced by

$$\left\{ \begin{array}{lll} \dot{x} = \sqrt{2g\bar{y}}\sin\theta, & x(t_0) = 0, & x(t_f) = x_f \\ \dot{y} = \sqrt{2g\bar{y}}\cos\theta, & \bar{y}(t_0) = y_d, & \bar{y}(t_f) = y_f + y_d. \end{array} \right. \tag{3.4}$$

This reformulation is equivalent to exploiting the conservation of the total stored energy. The Hamiltonian for the optimal control problem is

$$H = 1 + (\lambda_x \sin\theta + \lambda_y \cos\theta)\sqrt{2g\bar{y}}, \tag{3.5}$$

where λ_x and λ_y are the costates associated with the dynamic constraints in (3.4). The necessary conditions for optimality (see Section 2.2.2 or [13]) are

$$\left. \begin{array}{rcl} 0 & = & H_\theta \\ \dot{\lambda}_x & = & -H_x \\ \dot{\lambda}_y & = & -H_y, \end{array} \right\} \forall t \in [0, t_f] \tag{3.6}$$

where $H_{(\cdot)}$ represents the partial derivative $\partial H/\partial(\cdot)$. The necessary conditions (3.6) imply

$$\left. \begin{array}{lll} \lambda_y & = \lambda_x \cot\theta & \\ \dot{\lambda}_x & = 0 \quad \Rightarrow \lambda_x = \text{ Const} & \\ \dot{\lambda}_y & = -\left(\lambda_x \sin\theta + \lambda_y \cos\theta\right)\sqrt{g/2\bar{y}} & = -g\lambda_x/\dot{x}. \end{array} \right\} \forall t \in [0, t_f] \tag{3.7}$$

From (3.4) and the first of the optimality conditions in (3.7) one obtains

$$y' = \frac{dy}{dx} = \frac{\dot{y}}{\dot{x}} = \frac{\lambda_y}{\lambda_x} = \cot\theta, \tag{3.8}$$

and so

$$1 + y'^2 = \csc^2\theta. \tag{3.9}$$

From (3.8) and the third equation in (3.7) one obtains

$$y'' = \frac{dy'/dt}{dx/dt} = \frac{\dot{\lambda}_y}{\lambda_x \dot{x}} = -\frac{g}{\dot{x}^2} = \frac{\csc^2\theta}{2\bar{y}}. \tag{3.10}$$

Substituting (3.9) into (3.10) gives the differential equation that must be satisfied by the optimal trajectory:

$$2\bar{y}y'' + y'^2 + 1 = 0. \tag{3.11}$$

It can be verified by direct substitution that the cycloid parametrised by

$$\begin{array}{l} x = \bar{r}\ (2\theta - \sin 2\theta) - x_d \\ y = \bar{r}\ (1 - \cos 2\theta) - y_d \end{array}, \tag{3.12}$$

for $\theta \in [0, \pi]$, is a solution of the differential equation (3.11), where $\bar{r}$ is the radius of the cycloid's generating circle. The constants $\bar{r}$, x_d, $\theta(t_0)$ and $\theta(t_f)$ can be computed in terms of v_0, x_f and y_f.

The final result pertaining to the classical brachistochrone relates to the journey time, which can be computed as

$$\begin{aligned} t_f &= \int_{path} \frac{ds}{v} = \int_{path} \sqrt{\frac{1+y'^2}{2g\bar{y}}}dx \\ &= \int_{\theta(t_0)}^{\theta(t_f)} \sqrt{\frac{1+y'^2}{2g\bar{y}}}\bar{r}(2-2\cos 2\theta)d\theta \\ &= 2\sqrt{\frac{\bar{r}}{g}}\big(\theta(t_f) - \theta(t_0)\big). \end{aligned} \tag{3.13}$$

3.1.2 The Unrestrained Brachistochrone

In the standard brachistochrone the bead slides along the track without the possibility of ever losing contact with it. As a result, there are no constraints on the acceleration that might be experienced by the mass. In the unrestrained case, in which the mass slides on the surface of the track, the mass would lose contact with the track if the normal reaction F_n were to change sign; see Figure 3.3. This means that there is a constraint on the bead's centripetal acceleration that has to be introduced into the problem formulation as an inequality constraint. In order to describe this constraint mathematically the normal force is expressed in terms of the track inclination angle and its time derivative. For this

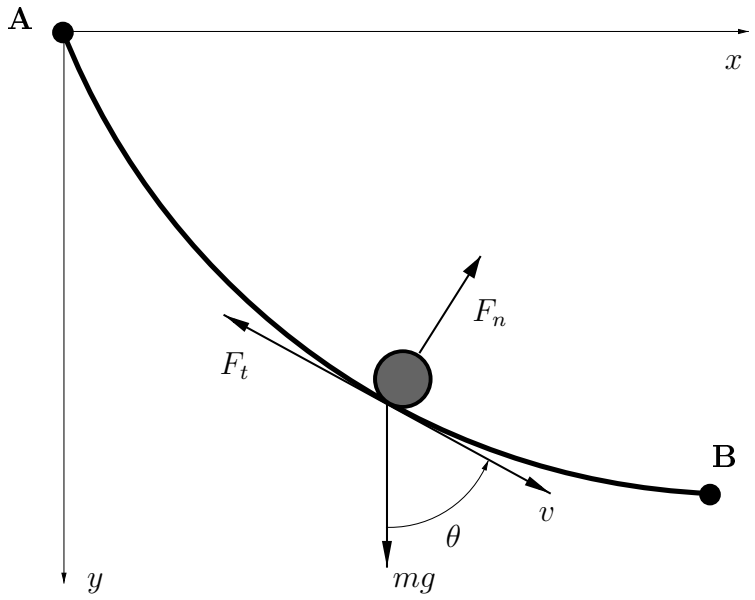

Figure 3.3: The unrestrained brachistochrone problem. The tangential force F_t is zero in the point-mass case. Any symbol associated with a vector represents the vector magnitude.

reason it is convenient to use the time rate of change of the track inclination angle ω as the controlled input and the inclination angle θ as a newly introduced state. This means that initial and final track inclination angles can be given as prescribed boundary conditions.

The optimal control formulation of the unrestrained brachistochrone problem is

$$\min_{\omega} \int_0^{t_f} dt$$
$$\text{s.t.} \begin{cases} \dot{x} = v\sin\theta, & x(t_0) = 0, \quad x(t_f) = x_f, \\ \dot{y} = v\cos\theta, & y(t_0) = 0, \quad y(t_f) = y_f, \\ \dot{v} = g\cos\theta, & v(t_0) = v_0, \\ \dot{\theta} = \omega, & \theta(t_0) = \theta_0, \quad \theta(t_f) = \theta_f, \\ g\sin\theta + v\omega \geq 0, \end{cases} \tag{3.14}$$

in which the inequality constraint ensures that the mass does not leave the track. Invoking the arguments used to derive (3.4), the problem constraints in (3.14) can be reduced to

$$\begin{cases} \dot{x} = \sqrt{2g\bar{y}}\sin\theta, & x(t_0) = 0, \quad x(t_f) = x_f, \\ \dot{y} = \sqrt{2g\bar{y}}\cos\theta, & \bar{y}(t_0) = y_d, \quad \bar{y}(t_f) = y_f + y_d, \\ \dot{\theta} = \omega, & \theta(t_0) = \theta_0, \quad \theta(t_f) = \theta_f, \\ g\sin\theta + \omega\sqrt{2g\bar{y}} \geq 0, \end{cases} \tag{3.15}$$

that satisfies the conservation of energy with $y_d = v_0^2/2g$.

The control Hamiltonian for this problem is

$$H = 1 + (\lambda_x \sin\theta + \lambda_y \cos\theta)\sqrt{2g\bar{y}} + \lambda_\theta \omega + \mu\left(g\sin\theta + \omega\sqrt{2g\bar{y}}\right), \tag{3.16}$$

in which λ_θ and μ are Lagrange multipliers associated with the θ-dynamics and inequality constraint respectively. The necessary conditions for optimality are

$$\left.\begin{array}{rcll} 0 &=& H_\omega &= \lambda_\theta + \mu\sqrt{2g\bar{y}} \\ \dot{\lambda}_x &=& -H_x &= 0 \Rightarrow \lambda_x = \text{ Const} \\ \dot{\lambda}_y &=& -H_y &= -\sqrt{g/2\bar{y}}\,(\mu\omega + (\lambda_x \sin\theta + \lambda_y \cos\theta)) \\ \dot{\lambda}_\theta &=& -H_\theta &= -(\lambda_x \cos\theta - \lambda_y \sin\theta)\sqrt{2g\bar{y}} - \mu g \cos\theta. \end{array}\right\} \forall t \in [0, t_f] \tag{3.17}$$

According to the PMP two alternative conditions can arise:

$$\left\{\begin{array}{lll} \mu = 0 & \text{and} & g\sin\theta + \omega\sqrt{2g\bar{y}} > 0 \\ \mu < 0 & \text{and} & g\sin\theta + \omega\sqrt{2g\bar{y}} = 0. \end{array}\right. . \tag{3.18}$$

In the first case ($\mu = 0$) the mass slides on the track and the necessary conditions (3.17) revert to those given in (3.7). To see this observe that $\mu = 0 \Rightarrow \lambda_\theta = 0 \Rightarrow \dot{\lambda}_\theta = 0 \Rightarrow \lambda_x = \lambda_y \tan\theta$. As shown in Section 3.1.1 a cycloid again satisfies the necessary conditions for optimality in the unconstrained phase.

When $\mu < 0$ in (3.18) (the normal load constraint being active), the mass is not subject to a normal reaction force and so it follows a ballistic parabola.

The switching structure of the optimal trajectories for the general unrestrained brachistochrone is analysed in [75], where it is shown that when an optimal trajectory exists, it is composed of up to three subarcs comprising a cycloidal portion embedded between two ballistic parabolae. This three-subarc structure is illustrated in Figure 3.4. Degenerate versions of this general solution occur when the length of one or two subarcs is zero. This means that trajectories of the form P, C, PC, CP and PCP can occur, where the 'C' represents a cycloidal subarc and 'P' a parabolic one. As explained in [75], impulsive control is sometimes required to enforce the boundary angle conditions.

The initial parabolic trajectory can be described by

$$\dot{x} = v_0 \sin\theta_0 \tag{3.19}$$

$$\dot{y} = gt + v_0 \cos\theta_0, \tag{3.20}$$

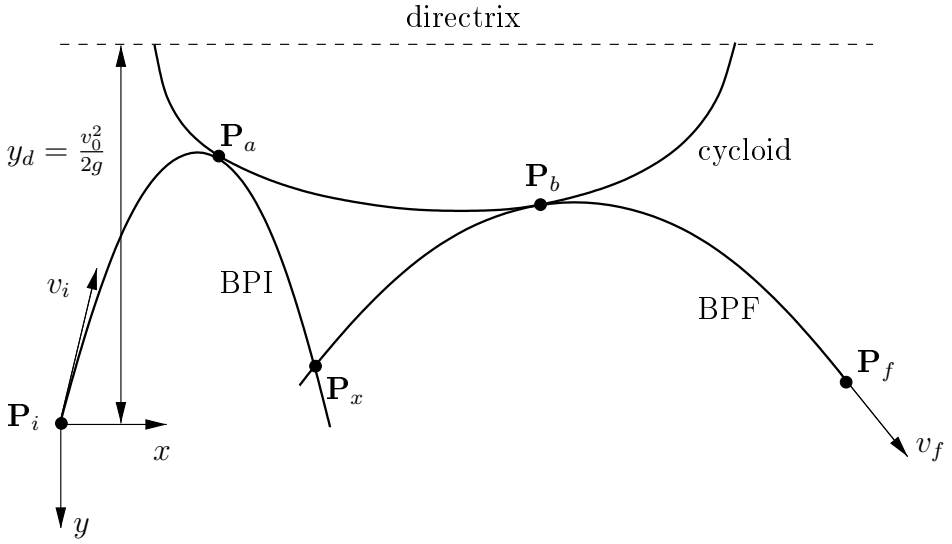

Figure 3.4: General form of the solution to the unrestrained brachistochrone problem. The initial ballistic parabola, labelled BPI, is determined by the initial velocity v_i and the terminal ballistic parabola, labelled BPF, is determined by the terminal velocity v_f. The cycloid's directrix is at an elevation for which the bead's kinetic energy is zero. The intersection point for the free-fall parabolae is $\mathbf{P}_x$ and the cycloid tangent to both parabolae has points of tangency $\mathbf{P}_a$ and $\mathbf{P}_b$.

while the final ballistic subarc is given by

$$\dot{x} = -v_f \sin\theta_f \tag{3.21}$$

$$\dot{y} = gt - v_f \cos\theta_f, \tag{3.22}$$

in which v_f is given by $v_f^2 = v_0^2 + 2gy_f$, since the system is conservative.

3.1.2.1 State Continuity

It has been established that the trajectory is cycloidal when the normal load constraint in (3.15) is inactive, and parabolic otherwise. The purpose here is to study the relationship between the parabolic and cycloidal subarcs at the junction points.

In optimal-control problems involving inequality constraints on the control, discontinuous changes in the control may be encountered. These points of discontinuity are called *corners*. It shall be proven that for this problem both the optimal state trajectory and the control are continuous at the junction points between cycloidal and parabolic phases. The first two state constraints in (3.15) show that $\dot{x}$ and $\dot{y}$ are finite and so the x and y

trajectories are continuous at the transition points.

Let us denote the time just before a junction point by t_- and the time just after by t_+. From standard theory [13] it is known that

$$\lambda(t_-) = \lambda(t_+) \tag{3.23}$$

$$H(t_-) = H(t_+). \tag{3.24}$$

It follows from (3.23) and the first equation in (3.17) that $\lambda_\theta(t_-) = \lambda_\theta(t_+) = 0$ and $\mu(t_-) = \mu(t_+) = 0$ when $\bar{y} \neq 0$; the case $\bar{y} = 0$ implies a degenerate solution. Using this in combination with (3.16), (3.24) and the x and y components of (3.23) establishes that $\tan\big(\theta(t_-)\big) = \tan\big(\theta(t_+)\big)$ and so for $\theta \in [0, \pi]$ the cycloid and parabolae have the same slope at the junction points.

3.1.2.2 Transit Time

The total journey time can be calculated by summing the transit times associated with the individual subarcs. Dividing (3.20) by (3.19) using (3.8) shows that the transit time for the first parabolic subarc is

$$t_{fA} = \frac{v_0}{g} \sin\theta_0 (\cot\theta_A - \cot\theta_0). \tag{3.25}$$

A similar calculation using (3.21) and (3.22) shows that the transit time for the second parabolic subarc is

$$t_{fB} = \frac{v_f}{g} \sin\theta_f (\cot\theta_f - \cot\theta_B). \tag{3.26}$$

The total transit time is thus

$$t_f = t_{fA} + t_{fB} + 2\sqrt{\frac{\bar{r}}{g}}(\theta_B - \theta_A). \tag{3.27}$$

Figure 3.5 shows two examples of solutions to the unrestrained brachistochrone problem with three-subarc optimal trajectories.

3.1.2.3 Solution Structure

In the general PCP case, there are two junction points where the x- and y-components of the cycloid and the parabolae must agree. At the junction points the gradients of the

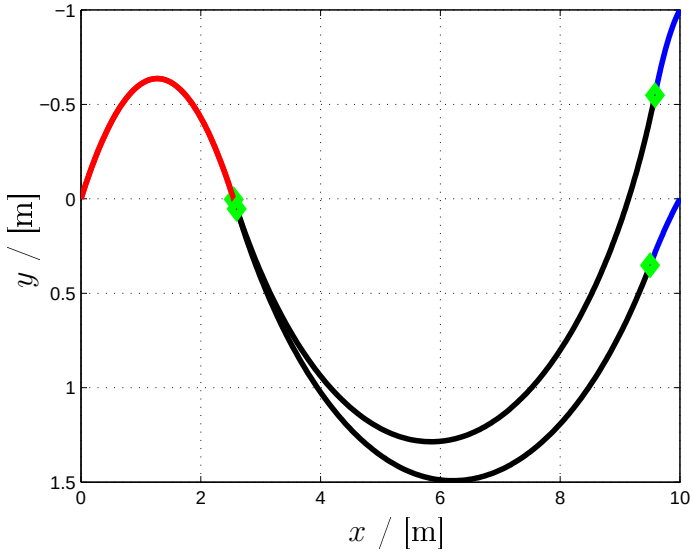

Figure 3.5: The unrestrained brachistochrone problem. The point mass has an initial speed of 5 m/s and a launch angle of 45^o (relative to the x-axis). There are two alternative terminal points at (10 m, -1 m) and (10 m, 0 m), with terminal angles of 30^o. The green diamonds mark the transition points between the ballistic parabolae and the central cycloid. The transit times in the two cases are 2.2265 s and 1.9791 s respectively.

cycloidal and parabolic subarcs must also match and so there holds

$$y'_A = \cot\theta_A \tag{3.28}$$

$$y'_B = \cot\theta_B. \tag{3.29}$$

In order to match the x- and y-components of the cycloid and the parabolae at the junctions, equations (3.19) to (3.22) are integrated to obtain

$$x_A = t_{fA} v_0 \sin\theta_0 \tag{3.30}$$

$$y_A = \frac{g t_{fA}^2}{g} + t_{fA} v_0 \cos\theta_0 \tag{3.31}$$

$$x_B = -t_{fB} v_f \sin\theta_f \tag{3.32}$$

$$y_B = \frac{g t_{fB}^2}{g} - t_{fB} v_f \cos\theta_f. \tag{3.33}$$

Equating (3.30) to (3.33), to the cycloid generating equations (3.12) at the junctions, using (3.25), (3.26), (3.28) and (3.29), one obtains the following four equations that must

be solved for x_d, $\bar{r}$, θ_A and θ_B:

$$\begin{cases} \bar{r}(2\theta_A - \sin 2\theta_A) - x_d &= 2y_d \sin^2\theta_0(\cot\theta_A - \cot\theta_0) \\ \bar{r}(1 - \cos 2\theta_A) - y_d &= y_d \sin^2\theta_0(\cot^2\theta_A - \cot^2\theta_0) \\ \bar{r}(2\theta_B - \sin 2\theta_B) - x_d &= x_f - 2(y_d + y_f)\sin^2\theta_f(\cot\theta_f - \cot\theta_B) \\ \bar{r}(1 - \cos 2\theta_B) - y_d &= y_f + (y_d + y_f)\sin^2\theta_f(\cot^2\theta_B - \cot^2\theta_f). \end{cases} \tag{3.34}$$

In the above equations the unknown parameters $\bar{r}$ and x_d characterise the cycloidal trajectory, while θ_A and θ_B are the cycloidal angles at the junction points — the corresponding junction times can be found using (3.25) and (3.26).

3.1.2.4 Impulsive Control

Impulsive control can be used to increase the initial slope of the trajectory to a value greater than θ_0 in an arbitrarily short time. This type of control is required in some cases to force the trajectory to meet the directional component of the initial boundary conditions. Since the inequality constraint in (3.15) is only violated for negative values of ω, the trajectory's direction can be increased by any positive amount in an arbitrarily short time with a sufficiently large (impulsive) control $\omega > 0$. A symmetrical argument can be applied to the terminal angle of approach of the optimal trajectory, which can take on values less than the prescribed θ_f for arbitrarily short periods prior to journey's end. In fact, if the given initial and final angles are respectively 0 and π radians, the problem reduces to a classical brachistochrone problem. Impulsive control is used to adjust the boundary angles to values compatible with the optimal cycloidal trajectory; boundary angles are not specified in the classical problem formulation.

Small changes to the boundary conditions can result in major changes to the optimal trajectory and the transit time. To see how this might occur one can consider Figure 3.6 (see [75]), in which the initial conditions are fixed, while there is an arbitrarily small change in the location of the terminal point. The initial conditions determine the ballistic parabola BPI and two closely-spaced terminal points $\mathbf{P}_{f1}$ and $\mathbf{P}_{f2}$ on either side of BPI are considered. The angle of arrival is required to be 0^o. In the case of $\mathbf{P}_{f1}$, BPI cannot be used without violating the track-contact inequality constraint and so an alternative parabolic trajectory Γ_1 must be used with impulsive control near the end points to meet

the boundary angle constraints. This is therefore a P problem with impulsive control. In the case of $\mathbf{P}_{f2}$, BPI can be used (without initial impulsive control) to get the bead moving towards the terminal point. Near the end of the trajectory a short cycloidal subarc is used in combination with impulse control to meet the terminal boundary conditions — this is thus a PC trajectory.

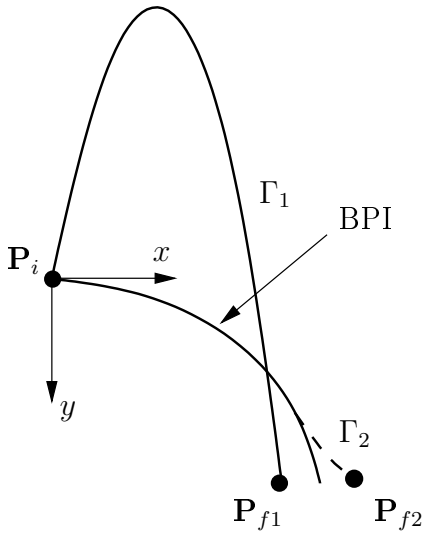

Figure 3.6: Sensitivity of the optimal solution to small changes in the boundary conditions (see [75]). The start of the trajectory is $\mathbf{P}_i$ and the ballistic parabola BPI is determined by the initial velocity. The points $\mathbf{P}_{f1}$ and $\mathbf{P}_{f2}$ are alternative terminal points in close proximity falling on either side of BPI. In the case of $\mathbf{P}_{f1}$ impulsive control in the vicinity of $\mathbf{P}_i$ selects the optimal trajectory Γ_1, while in the case of $\mathbf{P}_{f2}$ BPI is used in combination with a cycloidal subarc. In both cases terminal impulsive control is used to meet the final angle condition.

3.1.2.5 Solution Possibilities

The various solution possibilities relating to (3.34) are now given. The numerical solution of (3.34) provides the approximate junction conditions for a non-degenerate three-phase PCP trajectory, and provides an insight into the structure of the optimal trajectory in the other cases; the various solution possibilities are illustrated in Table 3.1. If the terminal potential energy of the bead is required to be higher than the total initial energy, then the problem is insoluble and (3.34) does not have a real solution; this case corresponds to the first row of Table 3.1. The second row of the table corresponds to the case that

there is no solution that does not violate the contact constraint given in (3.14). The third row corresponds to the generic PCP trajectory, where no impulsive control is required. The last four rows correspond to degenerate cases in which impulsive control is usually required in order to meet the boundary conditions.

Solution Type	t_A	t_B	t_{cyc}	$\bar{\omega}_i$	$\bar{\omega}_f$
No Sol^n.	—	—	—	—	—
No Sol^n.	≥ 0	≥ 0	< 0	—	—
PCP	> 0	> 0	> 0	—	—
PC	> 0	≤ 0	> 0	—	$+$
CP	≤ 0	> 0	> 0	$+$	—
C	≤ 0	≤ 0	> 0	$+$	$+$
P	≥ 0	≥ 0	$= 0$	$+$	$+$

Table 3.1: Solution and trajectory types for (3.34). The period t_{cyc} is the time spent on the cycloidal subarc and is given by (3.13). The $\bar{\omega}_i$ and $\bar{\omega}_f$ columns refer to the possible need for impulsive control at the initial and/or final points respectively. When impulsive control may be required, the corresponding table entry is marked with a '+' symbol.

3.1.3 The Rolling Unrestrained Brachistochrone

The problems analysed so far considered the motion of a point mass sliding on the surface of a track. In order to bring us a step closer to minimum-time vehicular problems, the point mass is replaced with a rolling body that has both mass and moment of inertia (see [90]); the problem is still two dimensional. The radius of curvature of the track shall be assumed larger than the radius of the rolling body so that there is a unique contact point. In addition to the position and speed of the mass centre, the state of the rolling body is characterised by its orientation angle ψ and its angular speed $\dot{\psi}$. The equations of motion for the rolling body are defined using Newton-Euler arguments based on force and moment balances. Resolving forces tangential and normal to the trajectory, in combination with a moment balance for the spinning body gives

$$m\dot{v} = mg\cos\theta - F_t, \tag{3.35}$$

$$mv\omega = F_n - mg\sin\theta, \tag{3.36}$$

$$rF_t = I_z\ddot{\psi}, \tag{3.37}$$

in which r is the radius of the rolling body, I_z is its moment of inertia, and F_n and F_t are, respectively, the normal and tangential components of the force applied to the rolling body at the contact point; see Figure 3.3. Non-slip rolling is enforced by

$$v = \dot{\psi} r. \tag{3.38}$$

The optimal-control problem for the unrestrained rolling brachistochrone can now be defined by replacing (3.14) with

$$\begin{aligned} &\min_{\omega} \int_0^{t_f} dt \\ &\text{s.t.} \quad \begin{cases} \dot{x} = v \sin\theta, & x(t_0) = 0, \quad x(t_f) = x_f, \\ \dot{y} = v \cos\theta, & y(t_0) = 0, \quad y(t_f) = y_f, \\ \dot{v} = g \cos\theta - F_t/m, & v(t_0) = v_0, \\ \dot{\theta} = \omega, & \theta(t_0) = \theta_0, \quad \theta(t_f) = \theta_f, \\ \ddot{\psi} = r F_t / I_z, & \psi(t_0) = \psi_0 \\ g \sin\theta + v\omega \geq 0, \\ v = \dot{\psi} r, \end{cases} \end{aligned} \tag{3.39}$$

using (3.35), (3.36), (3.37) and (3.38).

Through arguments analogous to those given in Section 3.1.2, the normal force F_n is again constrained to remain non-negative so that the rolling body cannot leave the track. In this problem the rolling body is geared to the track. Perfect rolling is thus assumed even when the normal load is vanishingly small. The last constraint equation in (3.39) allows one to eliminate the rolling body's spin velocity $\dot{\psi}$ from the problem. One can now find the value of the tangential force F_t by combining (3.35), (3.37) and (3.38) to get

$$F_t = \frac{I_z g^\star \cos\theta}{r^2}, \tag{3.40}$$

where $m^\star = (m + I_z/r^2)$ and $\alpha = (m^\star/m) = (g/g^\star)$. Using (3.40), (3.35) becomes

$$\dot{v} = g^\star \cos\theta. \tag{3.41}$$

Using (3.40) and the second and third constraint equations in (3.39) one obtains

$$v\dot{v} = g^\star \dot{y}, \tag{3.42}$$

and so

$$v = \sqrt{2g^\star(y + y_d^\star)}. \tag{3.43}$$

Setting $\bar{y} = y + y_d^\star$, the constraints in (3.39) can be replaced by

$$\begin{cases} \dot{x} = \sqrt{2g^\star\bar{y}}\sin\theta, & x(t_0) = 0, \quad x(t_f) = x_f, \\ \dot{y} = \sqrt{2g^\star\bar{y}}\cos\theta, & \bar{y}(t_0) = y_d^\star, \quad \bar{y}(t_f) = y_f + y_d^\star, \\ \dot{\theta} = \omega, & \theta(t_0) = \theta_0, \quad \theta(t_f) = \theta_f, \\ g\sin\theta + \sqrt{2g^\star\bar{y}}\omega \geq 0. \end{cases} \tag{3.44}$$

It is interesting to note that (3.43) is equivalent to

$$m(v^2 - v_0^2) + I_z(\dot{\psi}^2 - \dot{\psi}_0^2) = 2mgy, \tag{3.45}$$

which is a conservation of energy condition.

The control Hamiltonian for the system is

$$H = 1 + (\lambda_x\sin\theta + \lambda_y\cos\theta)\sqrt{2g^\star\bar{y}} + \lambda_\theta\omega + \mu\left(g\sin\theta + \omega\sqrt{2g^\star\bar{y}}\right). \tag{3.46}$$

The necessary conditions for optimality are

$$\left.\begin{array}{lll} 0 = H_\omega & = \lambda_\theta + \mu\sqrt{2g^\star\bar{y}} & \\ \dot{\lambda}_x = -H_x & = 0 \quad \Rightarrow \lambda_x = \text{Const} & \\ \dot{\lambda}_y = -H_y & = -\sqrt{g^\star/2\bar{y}}\,(\mu\omega + (\lambda_x\sin\theta + \lambda_y\cos\theta)) & \\ \dot{\lambda}_\theta = -H_\theta & = -(\lambda_x\cos\theta - \lambda_y\sin\theta)\sqrt{2g^\star\bar{y}} - \mu g\cos\theta & \end{array}\right\} \forall t \in [0, t_f]. \tag{3.47}$$

Considering the inequality constraint two conditions can arise:

$$\begin{cases} \mu = 0 \quad \text{and} \quad g\sin\theta + \omega\sqrt{2g^\star\bar{y}} > 0 \\ \mu < 0 \quad \text{and} \quad g\sin\theta + \omega\sqrt{2g^\star\bar{y}} = 0. \end{cases} \tag{3.48}$$

When the first of the conditions in (3.48) ($\mu = 0$) holds, the body rolls on the track and the necessary conditions (3.47), revert to (3.7), with only a modified gravitational acceleration constant:

$$\left.\begin{array}{lll} \lambda_\theta = 0 & & \\ \dot{\lambda}_\theta = -(\lambda_x\cos\theta - \lambda_y\sin\theta)\sqrt{2g^\star\bar{y}} = 0 & \Rightarrow \lambda_x = \lambda_y\tan\theta & \\ \dot{\lambda}_x = 0 & \Rightarrow \lambda_x = \text{Const} & \\ \dot{\lambda}_y = -g^\star\lambda_x/\dot{x} & & \end{array}\right\} \forall t \in [0, t_f]. \tag{3.49}$$

As shown in Section 3.1.1, a cycloidal path satisfies the necessary conditions for optimality in an unconstrained ($\mu = 0$) phase. When $\mu < 0$ the normal load vanishes, but a tangential

force remains when the body is accelerating, which ensures that the non-slip condition (3.38) remains in force. The presence of this tangential force means that the constrained arcs no longer follow ballistic trajectories of the type given in Section 3.1.2.

3.1.3.1 Zero-Normal-Load Trajectory

Here, the differential equations describing the motion of the rolling body under zero-normal-load conditions are derived. The general force balance conditions in the inertial x- and y-directions are given by

$$m\frac{dv_x}{dt} = F_n\cos\theta - F_t\sin\theta, \tag{3.50}$$

$$m\frac{dv_y}{dt} = mg - F_t\cos\theta - F_n\sin\theta, \tag{3.51}$$

where $v_x = v\sin\theta$ and $v_y = v\cos\theta$. In the case that $F_n = 0$, these become

$$m\frac{dv_x}{dt} = -F_t\sin\theta, \tag{3.52}$$

$$m\frac{dv_y}{dt} = mg - F_t\cos\theta, \tag{3.53}$$

with (3.36) reducing to

$$v\dot{\theta} = -g\sin\theta. \tag{3.54}$$

It can be verified that

$$\begin{aligned}\frac{dv^2}{dt} &= 2\left(v_x\frac{dv_x}{dt} + v_y\frac{dv_y}{dt}\right)\\ &= 2\left(gv\cos\theta - \frac{F_t}{m}v\cos^2\theta - \frac{F_t}{m}v\sin^2\theta\right)\\ &= 2\left(gv_y - \frac{F_t}{m}v\right)\\ &= 2v_y g^\star\end{aligned} \tag{3.55}$$

follows by direct calculation using (3.40), (3.52) and (3.53). Next, v_y is replaced with an expression involving y', using (3.54) and the four identities

$$\arctan\left(\frac{1}{y'}\right) = \theta, \tag{3.56}$$

$$\sin\left(\arctan\left(\frac{1}{x}\right)\right) = \frac{1}{\sqrt{1+x^2}}, \tag{3.57}$$

$$\cos\left(\arctan\left(\frac{1}{x}\right)\right) = \frac{x}{\sqrt{1+x^2}}, \tag{3.58}$$

$$\frac{d}{dt}\left(\arctan\left(\frac{1}{x}\right)\right) = \frac{-\dot{x}}{1+x^2}. \tag{3.59}$$

Combining these relationships one obtains

$$v = g\frac{\sqrt{1+y'^2}}{\dot{y}'}, \tag{3.60}$$

and

$$v_y = g\frac{y'}{\dot{y}'}. \tag{3.61}$$

Combining (3.55) and (3.61) gives

$$\begin{aligned}\frac{d\left(v^2\right)}{dt} &= 2\frac{g^2}{\alpha}\frac{y'}{\dot{y}'} \\ &= \frac{2}{\alpha}\frac{v^2 y'}{1+y'^2}\frac{dy'}{dt},\end{aligned} \tag{3.62}$$

using (3.60). Equation (3.62) is equivalent to Equation (10) in [90].

Equation (3.62) can be solved by separating the variables in combination with (3.43) to give

$$y\left(y'\right) = \frac{v_0^2}{2g^\star}\left(\left(\frac{1+y'^2}{\gamma}\right)^{\frac{1}{\alpha}} - 1\right), \tag{3.63}$$

with $\gamma = 1 + y_0'^2$, where y_0' and v_0 are respectively the initial slope and speed.

In order to get an expression for x as a function of y' it is required

$$\frac{d\left(v^2\right)}{dy'} = 2g^\star dx, \tag{3.64}$$

which comes from (3.43). Substituting (3.64) into (3.62) gives

$$dx = \frac{v_0^2}{g\gamma^{\frac{1}{\alpha}}}\left(1+y'^2\right)^{\frac{1-\alpha}{\alpha}} dy', \tag{3.65}$$

which can be integrated to give

$$x\left(y'\right) = \frac{v_0^2}{g\gamma^{\frac{1}{\alpha}}}\left[y'\ {}_2F_1\left(\frac{1}{2}, \frac{\alpha-1}{\alpha}, \frac{3}{2}, -y'^2\right)\right]_{y_0'}^{y'}. \tag{3.66}$$

The expression ${}_2F_1\left(a,b,c,z\right)$ is the Gaussian hypergeometric function defined in terms of the hypergeometric series:

$${}_2F_1\left(a,b,c,z\right) = \sum_{n=0}^{\infty} \frac{\left(a\right)_n \left(b\right)_n}{\left(c\right)_n} \frac{z^n}{n!}, \tag{3.67}$$

in which $\left(\cdot\right)_n$ is the Pochhammer symbol

$$\left(x\right)_n = \begin{cases} 1 & \text{if } n = 0 \\ \prod_{k=0}^{n-1}\left(x+k\right) & \text{if } n > 0. \end{cases} \tag{3.68}$$

The optimal trajectory of the rolling body is thus a sequence of alternating Gaussian hypergeometric functions and cycloids. As in Section 3.1.2.1, the corner conditions (3.23) and (3.24) are used to establish the continuity of the states along the optimal trajectory.

In the case that $\alpha = 1$, (3.66) becomes

$$x\left(y'\right) = \frac{v_0^2}{g\gamma}(y' - y_0'), \tag{3.69}$$

which is equivalent to the parabolic curve generated by (3.19) and (3.20).

These results extend the ones in [90] to a problem for which the start and the end of the trajectory are at different heights (as opposed to equal height), and the initial and final angles are assigned arbitrarily (as opposed to being equal to 0). Furthermore, the x-coordinate of the mass centre of the rolling body is expressed in terms of a Gaussian hypergeometric function, which corrects the solution given in [90], according to which the x-coordinate was defined by the incomplete Beta function $B(p,q;z)$ with $p = 1/2$, $q = (1+\alpha)/2\alpha$ and $z = y'^2/\left(1+y'^2\right)$ (refer to [90] for a definition of the incomplete Beta function).

3.1.3.2 Transit Time

Arguments similar to the ones used in Section 3.1.2 prove that, when an optimal trajectory exists, it is composed of up to three subarcs comprising a cycloidal portion embedded

between two hypergeometric subarcs. The minimum transit time is again given by the sum of the time spent on each of the subarcs of the optimal trajectory. The time spent on the cycloidal portion of the track is given in (3.13). Using (3.8) this equation can be expressed in the alternative form

$$t_{cyc} = 2\sqrt{\frac{\bar{r}}{g^\star}}\left(\arctan\left(\frac{1}{y'_B}\right) - \arctan\left(\frac{1}{y'_A}\right)\right), \tag{3.70}$$

in which y'_A and y'_B are the trajectory slopes at the first and second junction points.

The time spent by a rolling body following the hypergeometric trajectory is computed using the integral

$$t_f = \int_{path} \frac{\sqrt{1+y'^2}}{v} dx, \tag{3.71}$$

given already in (3.13). Next, equations (3.43) and (3.66) are used to obtain the time spent in the initial hypergeometric subarc

$$\begin{aligned} t_A &= \frac{v_0}{g\left(1+y_0'^2\right)^{\frac{1}{2\alpha}}} \int_{y'_0}^{y'_A} \left(1+y'^2\right)^{\frac{1-\alpha}{2\alpha}} dy' \\ &= \frac{v_0}{g}(\sin\theta_0)^{\frac{1}{\alpha}} \left[y' \; {}_2F_1\left(\frac{1}{2}, \frac{\alpha-1}{2\alpha}, \frac{3}{2}, -y'^2\right)\right]_{y'_0}^{y'_A} \end{aligned} \tag{3.72}$$

from (3.71), in which y'_0 is the initial (prespecified) slope. In much the same way one can compute the time spent in the terminal hypergeometric subarc as

$$t_B = \frac{v_f}{g}(\sin\theta_f)^{\frac{1}{\alpha}} \left[y' \; {}_2F_1\left(\frac{1}{2}, \frac{\alpha-1}{2\alpha}, \frac{3}{2}, -y'^2\right)\right]_{y'_f}^{y'_B}, \tag{3.73}$$

where y'_f is the pre-specified terminal slope. Thus

$$t_f = t_A + t_B + t_{cyc}. \tag{3.74}$$

In the point-mass case that $\alpha = 1$, (3.72) and (3.73) reduce to (3.25) and (3.26) respectively.

Figure 3.7 shows two examples of minimum-time trajectories for a rolling body with two and three subarcs.

A set of nonlinear algebraic equations , similar to (3.34), has to be solved to locate the transition points between the subarcs. Again, the solution of these equations determines the existence of the optimal trajectory and the number of subarcs.

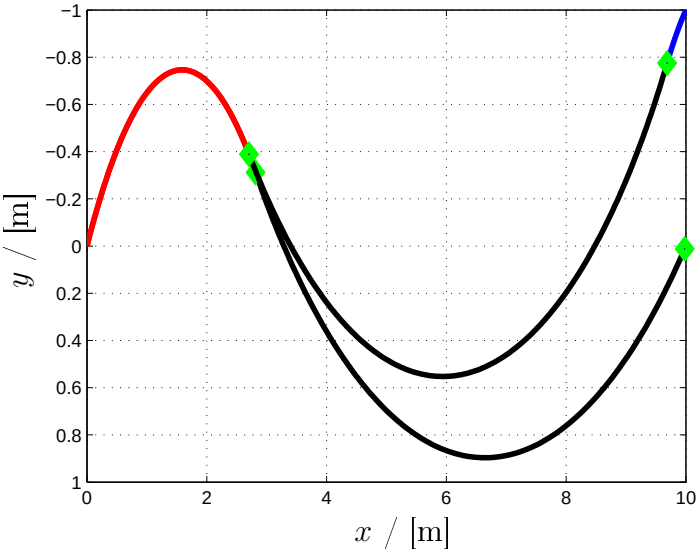

Figure 3.7: The unrestrained rolling-body brachistochrone problem. A rolling body, with α=2, has an initial speed of 5 m/s and an initial launch angle of 45^o (relative to the x-axis). There are two alternative terminal points at (10 m, -1 m) and (10 m, 0 m), with a terminal entry angle of 30^o. Green diamonds mark the transition points between Gaussian hypergeometric functions and the central cycloid. The transit times in the two cases are 2.2300 s and 2.0908 s respectively.

3.1.3.3 Numerical Optimal Control

An important goal of this work is to evaluate the effectiveness and accuracy of numerical methods for solving OC problems involving mechanical systems. The solutions are obtained using a direct collocation method that transforms the original continuous-time OC problem into a finite-dimensional parameter-optimisation problem, through a full discretisation of time, the state and the control; many of these ideas are explained in detail in Chapter 2 and in [8]. The transformed problem is then solved as a NLP problem.

A direct collocation method based on the trapezoidal rule (see Section 2.3.3) is applied to the problem given in (3.39). The problem is discretised at 200 sampling points equally distributed in time.

The numerical optimal-control calculations are in close agreement with the analytic solution given in Figure 3.7, with the numerical and theoretical transit times matching to six significant figures. Since the NLP algorithm is provided with the Hessian of the performance index, the solution of the transcribed problem is fully characterised up to the

second-order derivatives; increasing the sampling resolution did not affect the structure, or shape, of the trajectories.

The optimisation algorithm generates the Lagrange multipliers associated with the dynamic constraints and the inequality constraint at the discretisation points. These multipliers are respectively related to the costate trajectories and the inequality constraint multiplier of the continuous-time optimal control problem. Using the appropriate relation (see [8]) between the discrete- and continuous-time multipliers for the trapezoidal approximation, it was verified that a numerical optimal solution satisfies the corresponding theoretical conditions given (3.47).

3.2 Disc Rolling in a Hemisphere

The first example of optimal control problem involving a three-dimensional mechanical system consists of a thin disc rolling on the interior surface of a hemispherical shell in a uniform gravitational field. Non-slip rolling is again assumed, which is a standard assumption in simple vehicle modelling exercises. Figure 3.8 shows the position and orientation of the disc defined by the Cartesian coordinates of the disc's mass centre **G**, in combination with the disc's yaw ψ, roll ϕ and the spin angle (around the disc's axis $\hat{\boldsymbol{\gamma}}$). Since the spin angle is a cyclic[2] coordinate there is no need to assign a symbolic reference to it.

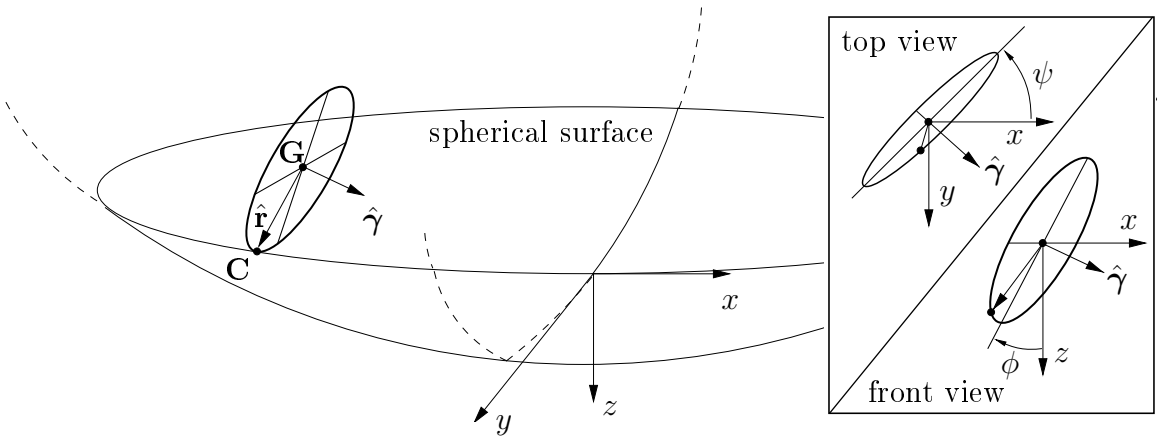

Figure 3.8: Disc rolling on the interior surface of a hemisphere.

[2]A coordinate q is *cyclic* if $\partial L/\partial q = 0$, where L is the system Lagrangian.

Gravity is positive in the z-direction. It is assumed that the disc remains in contact with the hemispherical surface at a unique moving contact point **C**. The rolling contact between the two surfaces is represented by one holonomic and two nonholonomic constraints. The holonomic constraint enforces the contact condition. The two nonholonomic constraints ensure perfect (non-slip) rolling. In effect, the constraints ensure that the material point **C** on the disc that is in contact with the sphere is absolutely stationary. For dynamic modelling purposes the disc is described by its mass m and its radius r. The inertia matrix can then be defined in terms of a body-fixed reference frame with its origin at the disc's centre; the y-axis of this reference frame coincides with the spin axis given by $\hat{\boldsymbol{\gamma}}$ in Figure 3.8. The inertia matrix relative to this body-fixed axis system is given by

$$I_B = \text{diag}\left\{\frac{mr^2}{4},\ \frac{mr^2}{2},\ \frac{mr^2}{4}\right\}.$$

A number of problems are now considered, which relate to the minimisation of the transfer time for the disc rolling between given initial and final points under the influence of gravity. At a later point active steering is introduced as a control input that can be optimised. In the unactuated case the trajectory of the disc is fully determined by its initial generalised coordinates and generalised speeds. Since the holonomic contact condition must be satisfied, the z-axis coordinate of the disc's centre is selected to be the dependent variable associated with the loss of one degree of freedom. As a result, the pre-specified initial and final points can be described in terms of the x- and y-axis coordinates of the disc's mass centre. The roll angle can be treated as either pre-specified, or as a free decision variable. The yaw angle is always treated as a decision variable.

An aim of this study is to strengthen the understanding of the optimal behaviour of rolling bodies moving under the influence of gravity. This is considered a first step towards an analysis of the optimal control of road vehicles where fuel expenditure is important. This work is also being used to probe the capabilities of the direct collocation method for optimal control.

The dynamic equations for a disc rolling on a level surface are classical; see for example [25, 54, 59, 72, 80]. While this work is interesting background, it does not accommodate

a hemispherical rolling surface, which complicates the kinematics of the rolling contact.

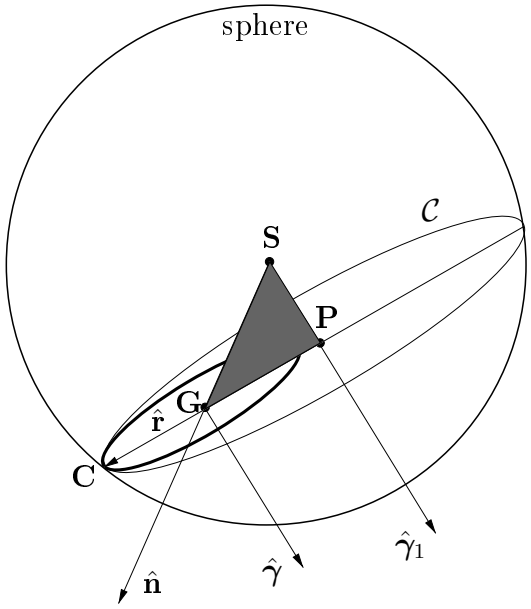

Figure 3.9: Contact geometry for a disc rolling on the inner surface of a sphere.

The contact point location is computed considering Figure 3.9, in which the disc is in an arbitrary configuration in contact with the spherical surface. The circle $\mathcal{C}$ is generated by the intersection of the disc plane and the spherical surface. Next, three vectors are introduced as follows:

1. the vector $\hat{\boldsymbol{\gamma}}_1$ in the direction **S**-**P**, with **S** the sphere's centre and **P** the centre of the circle of intersection $\mathcal{C}$;
2. the vector $\hat{\boldsymbol{n}}$ in the direction **S**-**G**;
3. the vector $\hat{\boldsymbol{r}}$ in the direction **G**-**C**.

It is observed that the disc's spin vector $\hat{\boldsymbol{\gamma}}$ is parallel to $\hat{\boldsymbol{\gamma}}_1$, and that all four vectors lie in the plane generated by the triangle **G**-**P**-**S** and that this plane is perpendicular to the plane of the disc. Now observe that $\hat{\boldsymbol{\gamma}} \times \hat{\boldsymbol{n}}$ is perpendicular to the plane **G**-**P**-**S**. Finally, it can be seen that $(\hat{\boldsymbol{\gamma}} \times \hat{\boldsymbol{n}}) \times \hat{\boldsymbol{\gamma}}$ is in the **G**-**P**-**S** plane and in the direction of the radial contact vector $\hat{\boldsymbol{r}}$. The contact point is thus given by the vector

$$\mathbf{C} = \mathbf{G} + r\hat{\boldsymbol{r}}, \tag{3.75}$$

in which $\hat{\boldsymbol{r}}$ is the unit vector given by

$$\hat{\boldsymbol{r}} = \frac{(\hat{\boldsymbol{\gamma}} \times \hat{\boldsymbol{n}}) \times \hat{\boldsymbol{\gamma}}}{\|(\hat{\boldsymbol{\gamma}} \times \hat{\boldsymbol{n}}) \times \hat{\boldsymbol{\gamma}}\|}. \tag{3.76}$$

This contact-point description can be used in a multibody analysis package to generate the equations of motion (EOMs) for the disc rolling on the interior surface of the sphere. In the next section these EOMs are used to solve a number of time-optimal control problems.

The numerical study can be extended to arbitrary smooth surfaces that guarantee the existence of a single contact point. The particular choice of a hemispherical (or spherical) surface allows for a definition of the equations of motion in the form of ODEs instead of the more general DAE formulation.

3.2.1 Results

In this section a number of optimal-control problems are studied, in which a thin disc is required to roll on a hemispherical surface between prescribed boundary points in minimum time. The effect of changes to the initial conditions on the roll angle, the yaw angle and the spin velocity will be examined, as well as the influence of external steering. In the study presented here it is assumed that the initial roll and yaw rates are zero. In common with the classical brachistochrone problem and its variants, the interchange between the disc's kinetic and potential energy is crucially important. Differences include the fact that the rolling surface is prescribed and three-dimensional, with the optimised trajectory lying on the rolling surface. Another difference relates to the fact that the total stored energy is not necessarily constant if the disc is externally steered. The optimisation problems are solved numerically. In all the examples considered here, it is assumed that the disc travels in minimum time from $(x, y) = (4\,\mathrm{m},\ 0\,\mathrm{m})$ to $(x, y) = (0\,\mathrm{m},\ 4\,\mathrm{m})$ in a sphere of radius 5 m centred at $(0\,\mathrm{m},\ 0\,\mathrm{m},\ -5\,\mathrm{m})$.

In the simplest version of the problem the disc has a given initial spin rate (zero initial yaw and roll rates), it is unsteered and it is initially upright (zero initial roll angle). The only free decision variable is the initial yaw angle. Since gravity does not enter the yaw and roll EOMs if ϕ is 0, gravitational effects on the yaw and roll dynamics are absent

in this case and the disc remains upright oriented in the same direction throughout the motion. Constancy of the yaw and roll angles also means that the self-steering moment $M_s = \boldsymbol{\omega} \times I\boldsymbol{\omega}$ is zero in this case; I is the disc's moment of inertia and $\boldsymbol{\omega}$ its angular speed. In the absence of an ability to steer, the disc's initial yaw angle must be selected so that the disc rolls in a straight line towards the prescribed terminal point.

Figure 3.10 considers the case of an unsteered upright disc with zero initial speed. As expected, the disc rolls upright along a path that has a linear ground-plane trajectory that connects the initial and terminal points.

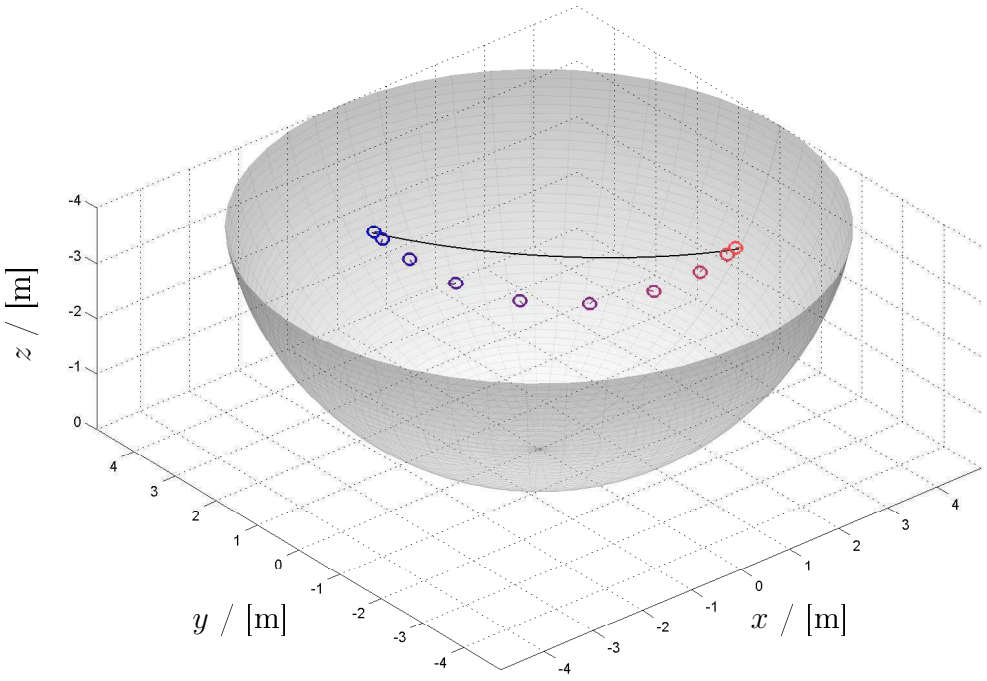

Figure 3.10: Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m, 0 m) to (x, y)=(0 m, 4 m). The initial roll angle is zero with the initial yaw angle a free decision variable. There is no externally applied steering torque. The optimal transfer time is 2.5614 s. The coloured cartoons show the disc at equispaced instants in time (from light red to dark blue). The solid black line is the segment of the great circle that joins the starting and terminal points. The roll and yaw angles are constant and equal to, respectively, 0 rad and $-\pi/4$ rad.

The next two results, given in Figures 3.11 and 3.12, show the effect of introducing the

initial roll angle as a free decision variable. As soon as the disc develops a non-zero roll rate, self-steering occurs and oscillatory behaviour is possible. The first of these figures corresponds to the case of zero initial spin rate (speed), while the spin rate is 100 rad/s in the second case. As already mentioned, the x- and y-axis coordinates of the disc's mass centre are imposed as initial and final position constraints, with the boundary values of the z-axis component of the mass centre determined by the boundary roll and yaw angles in combination with the holonomic rolling constraint. The first of these figures shows the disc 'wobbling' along its trajectory; this behaviour is due to self-steering. As with the classical brachistochrone and its variants, the disc does not follow the shortest path (geodesic) in order to achieve the minimum-time transition. Instead, the disc steers a path that exploits the energy transfer between kinetic and potential in order to minimise the transition time. Freedom in the roll angle allows for different initial and final disc orientations that can be used in the energy management process. In this example this freedom results in a lower final potential energy as compared with its initial value. In essence, this potential energy reduction is used to 'buy' additional translational speed.

In common with the classical brachistochrone, when the initial speed is high, and when there is less scope to exploit the potential-kinetic energy interplay, the trajectory follows more and more nearly a geodesic; this is what occurs in Figure 3.12, where the disc follows closely the appropriate great circle.

Figure 3.13 shows the ground-plane trajectory of the disc's mass centre for a number of initial velocities (disc spin rates). This diagram also shows the corresponding approximate geodesic for the given terminal positions[3], which represents the shortest-length path between the start and end points. The zero- and high-speed cases are considered first. In the case of Figure 3.11, the ground-plane projection of the disc's path follows approximately a straight line that joins the ground-plane projections of the terminal points. In the case of Figure 3.12, the ground-plane projection of the disc's trajectory follows more

[3]The approximate geodesic is the arc of a great circle (on the surface of the sphere) that passes through the intersections of the spherical surface and the lines joining the centre of the sphere and the terminal (start and end) positions of the mass centre of the disc; for this definition the contact radius is imposed parallel to the sphere radius.

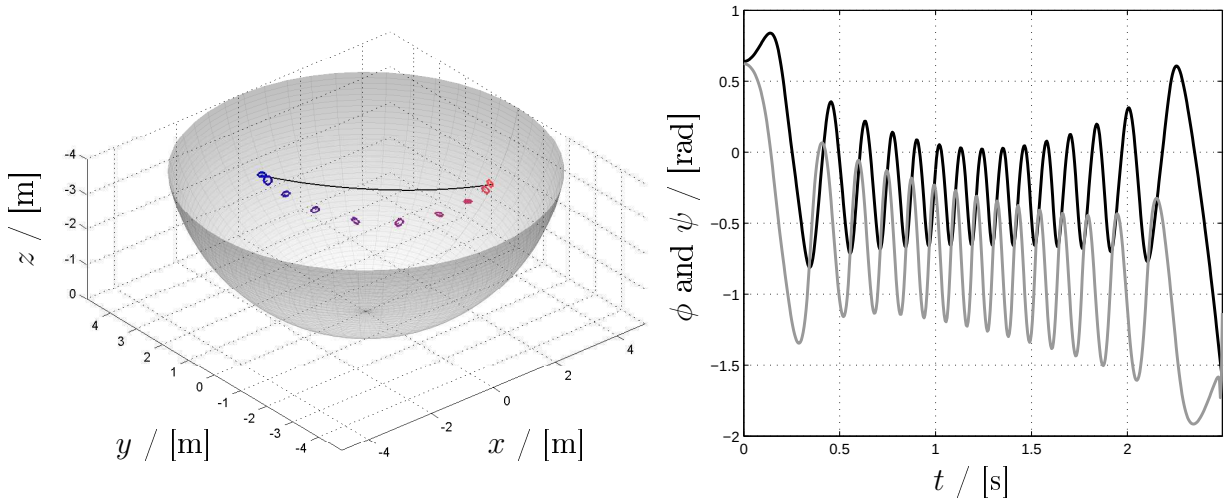

Figure 3.11: Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial yaw and roll angles are free decision variables. The initial spin rate is zero and there is no external steering. The resulting transfer time is 2.4163 s. On the right, the roll and yaw angles are plotted against time, in the black and grey curves respectively.

closely the ground-plane projection of the great circle that passes through the terminal points. As the disc's initial speed increases, one observes an increase in the disc's ability to self-steer, and as a result, the projection of the trajectory of the disc's mass centre onto the ground plane becomes curved. For an initial spin rate of 20 rad/s the ground plane projection bends towards the bottom of the hemisphere, exploiting the potential-kinetic energy exchange. When the disc's initial spin rate increases to 50 rad/s the trajectory moves towards the geodesic.

In Figure 3.14 the introduction of an external steering moment $M(t)$ is considered. This moment acts on the disc about the radial contact vector $\hat{\boldsymbol{r}}$ in Figures 3.8 and 3.9. The control action appears to be used as a way of injecting additional energy into the system. To restrict this effect it was necessary to assign a stage cost to the control action. As the control cost is reduced, there is an expected increase in the steering torque and a consequential reduction in the transition time.

Figure 3.15 shows the effect of the steering torque on the disc's stored energy. Not unexpectedly, the total stored energy increases as more external steering is allowed. What

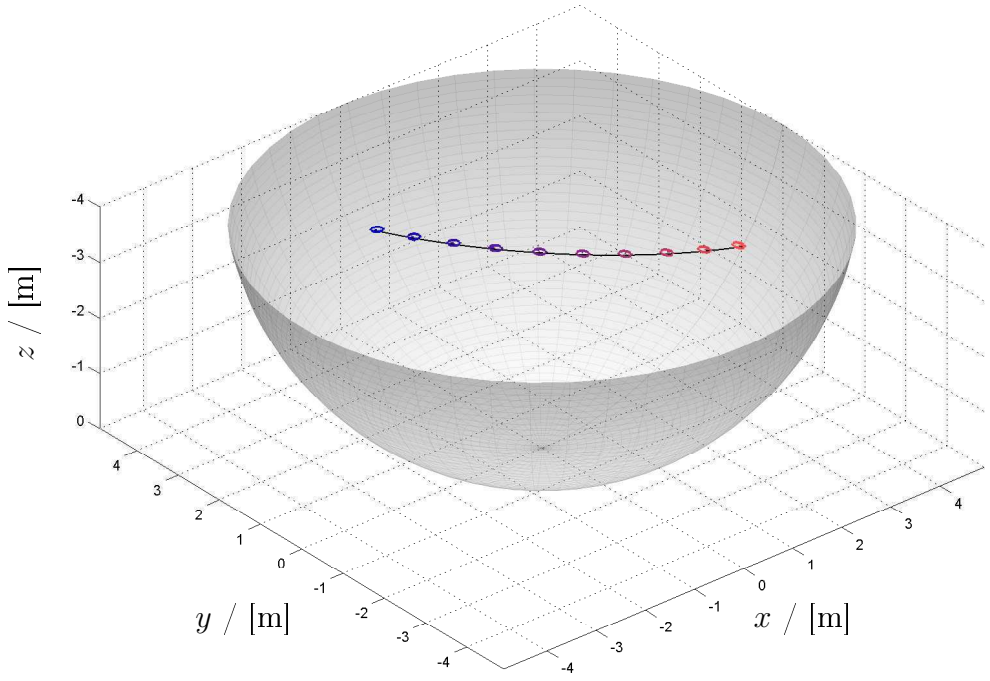

Figure 3.12: Three-dimensional cartoons of a disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial yaw and roll angles are free with an initial spin rate of 100 rad/s. There is no external control and the transfer time is 0.6152 s.

is possibly more interesting, is the way in which external steering is used to increase not only the total energy, but also the transfer of potential energy into kinetic energy throughout the motion thereby reducing the transfer time. The disc also reaches its final destination with higher terminal speeds as more active steering is allowed.

3.3 Summary

A number of minimum-time optimal-control problems for rolling bodies are studied. An optimal-control analysis framework is used in a brief review of the classical brachistochrone problem. Parametric equations for the well-known cycloidal trajectory and the optimal transit time are derived in terms of the initial speed and terminal position of the point mass.

A first generalisation studies the so-called unrestrained brachistochrone problem (see

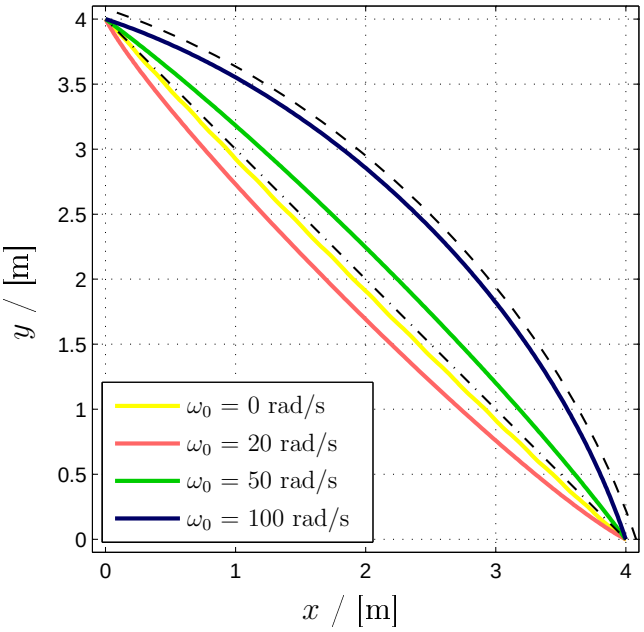

Figure 3.13: Projections on the ground plane of the optimal trajectories of the disc's mass centre computed for different initial spin rates. The line segment (dot-dashed black line) and the approximated geodesic (dashed black curve) are shown as limit references in the figure.

[76]), in which a point mass slides on the surface of a two-dimensional track that lies in a uniform gravitational field. The mass must remain in contact with the track throughout — this condition is embedded in the optimal control problem as a path constraint. The track has a pre-specified terminal point as well as pre-specified initial and terminal inclination angles. The initial velocity of the mass is pre-specified. Closed-form solutions for the trajectory and the transfer times are derived. When solutions exist the trajectory is composed of ballistic parabolic sections, where the track-contact constraint is active, and a cycloidal portion, where it is inactive.

A further generalisation studies the unrestrained brachistochrone problem for a rolling rigid body with rotational inertia. As before, the rolling body must remain in contact with a track that lies in a uniform gravitational field. The problem boundary conditions include the terminal point, the body's initial speed and the initial and terminal track inclination angles. Closed-form solutions for the trajectory and the transit time are derived. When solutions exist the trajectory is composed of a central cycloidal section akin to the classical

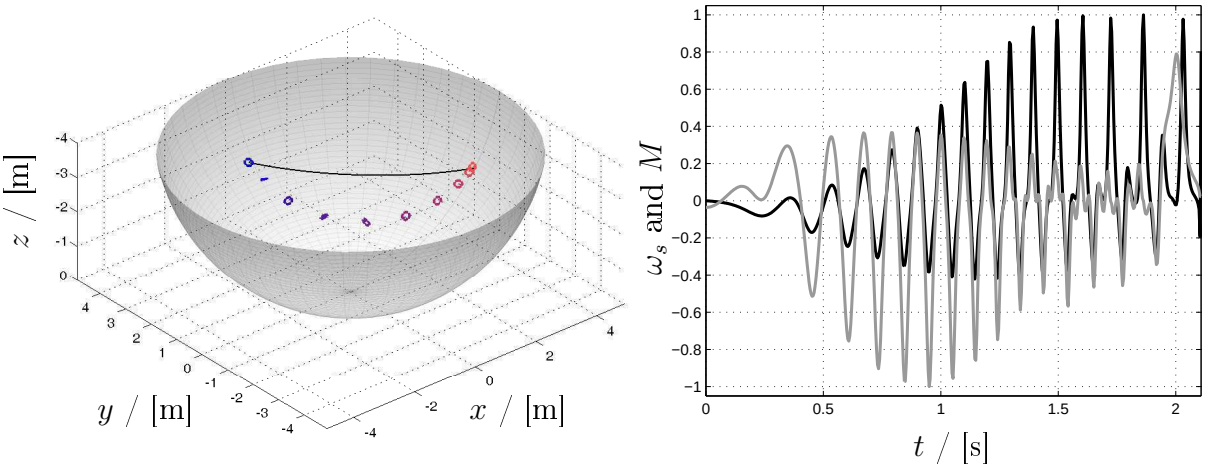

Figure 3.14: Disc travelling in minimum time from (x, y)=(4 m,0 m) to (x, y)=(0 m,4 m). The initial angular rates are zero, with the initial yaw and roll angles free decision variables. A stage cost $L(t) = 10^{-2} \cdot M(t)^2$ results in a transition time of 2.1018 s. A three-dimensional plot of the disc's trajectory is shown on the left. On the right, the normalised steering rate ω_s (projection of the disc's angular speed onto the contact radius, with scaling factor 120.5 rad/s) and the normalised steering moment M (scaling factor 3.307 Nm) are plotted against time in the black and grey curves respectively.

brachistochrone problem solution. The initial and final subarcs of the trajectory, as well as the transit time, are described in terms of Gaussian hypergeometric functions. These results correct and generalise those given in [90], providing an expression for the mass position in terms of a Gaussian hypergeometric function and considering problems with arbitrary final height and arbitrary initial and final angles.

The final part of the chapter considers a three-dimensional minimum-time transfer problem involving a possibly steerable disc rolling on the interior surface of a hemisphere. This problem is solved using a numerical optimal-control procedure. The disc is 'forced' to remain in contact by imposing a holonomic constraint, while perfect (non-slip) rolling is ensured through the imposition of two nonholonomic constraints. If the disc is initially inclined and/or subject to an externally imposed steering moment it will 'wobble' towards the terminal point along a curved path. This 'wobbling' motion is caused by a gyroscopic moment that is produced by the interaction of the disc's angular momentum and its roll rate. As with all the brachistochrone-like problems described here, the path approaches a geodesic as the disc's initial speed increases. In the case of the steerable disc, the optimal

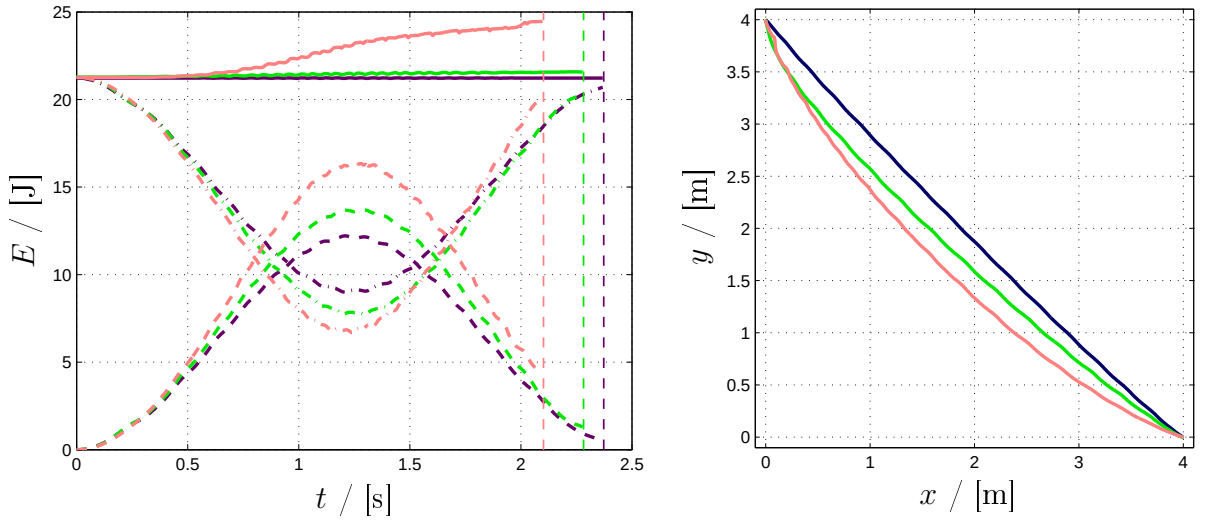

Figure 3.15: The disc's energy (left) and the ground-plane projection of the mass centre (right). The dark blue plots correspond to no external steering, the green plots correspond to a stage cost of $M^2(t)$, and the light red plots correspond to a stage cost of $10^{-2} \cdot M^2(t)$. In the energy diagram, the solid curves represent the total stored energy of the disc, while the dashed and dashed-dotted curves are, respectively, the kinetic and the potential energy. The vertical dashed lines identify the transition times.

steering control employs an energy-pumping mechanism that is used to boost the total stored energy as well as the rolling speed. The steering process is also used to convert more of the disc's stored potential energy into kinetic energy.

CHAPTER 4

Track Modelling and Identification

"The road was a ribbon of moonlight over the purple moor", Alfred Noyes.
"He saw nothing except the endless ribbon of road unfolding in twisting narrow curves.", Lois Lowry.

ROAD models are required in a number of applications areas including virtual reality environments, transportation management and planning systems, automated vehicle navigation and geographic information systems. Road models are also required in vehicular optimal-control problems so that road-related path constraints can be incorporated into the problem. In the context of road racing these optimal-control problems are usually related to minimum-lap-time control problems. Road-related path constraints are enforced by ensuring that the vehicle's mass centre, or its wheels, remain within the track boundaries. In [33] a two-dimensional (flat) road model is considered, and any deviations from the road are penalised by a performance index term involving the distance between the car's mass centre and the centre of the road. These ideas are further developed in [18], where a flat road is again used with state constraints applied to a finite number of check-points that are distributed along the periphery of the track. It is assumed that if these boundary constraints are properly located, they can be used to represent the track boundaries everywhere. Much the same ideas are used in [35], where a flat track is again assumed. By building on the work in [19], the study presented in [64] makes use of a two-dimensional curvilinear-coordinate-based track description. In this work the perpen-

dicular distance between the track centre line and the car's mass centre is a state that can be constrained in optimal-control studies.

As was observed in [18], measurement noise makes it difficult to deduce track models by numerically integrating measured telemetry data. Key difficulties relate to errors in the kinematic models used, and drift in the integration processes. In combination, these errors can result in a displacement mismatch between the track start and end points of several tens (and even hundreds) of metres. These problems were solved in [64], where the track curvature is deduced from the solution of an optimal-control problem that includes a path-closure constraint. An intrinsic coordinate description of the road is also adopted in [84], where the absolute orientation of the road centre-line tangent is used instead of its curvature. The road model is employed for minimum-manoeuvre-time calculations using convex optimisation in a linear model-predictive-control framework. The disadvantages associated with absolute coordinates are also recognised in [19], [64] and [84], none of whom use absolute coordinates in their vehicle models. The main purpose of the work described in this chapter is to generalise the use of curvilinear coordinates and the differential-geometric ideas used in [64] to three-dimensional track modelling. When working in three dimensions, three rather than one curvature functions must be deduced from the measurement data.

Central to road models are *ribbons*, or *strips*, that are used to represent the roadway and provide a differential-geometric description of navigable road surface. As is explained in [38], general surfaces can be studied in terms of strips or ribbons — think of the ribbon of peel one discards after removing it from the surface of an apple. In the same way that surfaces can be studied in terms of ribbons, ribbons can be studied in terms of curves; one such curve could be the strip's centre line. These curves are also referred to as the *spine* of the ribbon; the spine need not be the centre line. The general idea is to describe the surface, or ribbon, in terms of tangent planes computed along the spine. Some of these modelling ideas are illustrated in Figure 4.1, which will later be returned to when formalising these geometric concepts.

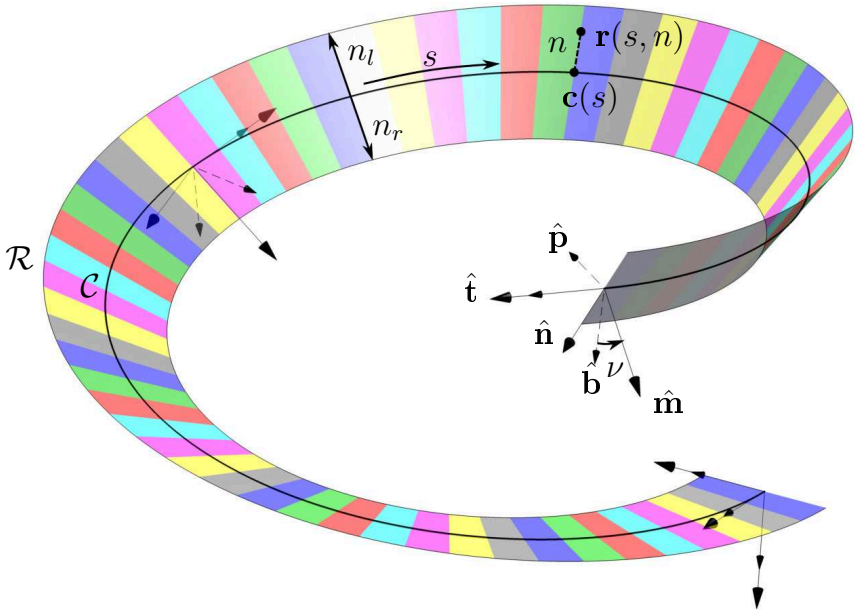

Figure 4.1: A ribbon $\mathcal{R}$ supported by a spine curve $\mathcal{C}$.

Ribbon models have been used to describe DNA, polymers and other chain-like molecules and some of their properties [36, 56, 71, 88]. For the current purposes these papers play a useful role by describing some of the differential-geometric ideas that will be used here; the account given in [71] is particularly useful.

Ribbon-based models also occur in the context of road modelling from satellite imagery; recent overviews of the literature in this field are given in [48, 52]. In these papers the road is modelled as a ribbon using conventional curvilinear-coordinate ideas based on arc length and curvature. A Kalman filter, based on a kinematic road model, is used to predict future road centre-line positions from observations and the current state of the road. It was observed that most public roads tend to be 'smooth' and that the curvature of their centre line can usually be bounded in proportion to the road width. This has the useful side effect of excluding possible singularities from the ribbon-based road models. Three-dimensional road modelling is considered in [89], where arc-length parametrised splines are used to model the road.

The differential geometry of curves and ribbons is summarised in Section 4.1. Section 4.1.1 offers a summary of some of the standard properties of cross products and

skew-symmetric matrices. The background theory relating to curves is summarised in Section 4.1.2, with the differential-geometric properties of ribbons given in Section 4.1.3. The relationship between these differential-geometric curvature ideas and Euler angles is summarised in Section 4.1.4. The optimal-estimation problem is described in Section 4.2; this includes a finite-difference estimate of the track boundaries which appears in Section 4.2.1. A summary of the influences of measurement noise is given in Section 4.2.2. The estimation problem is formulated as an optimal-control problem in Section 4.2.3. The results are given in Section 4.3: a ribbon based on the Lemniscate of Bernoulli is studied in Section 4.3.1, and the Circuit de Catalunya is modelled in Section 4.3.2. The chapter is summarised in Section 4.4. The contents of this chapter are being considered for publication in [65].

4.1 Geometry of Ribbons

The purpose of this section is to assemble a number of geometric ideas that will be used to describe three-dimensional roads. All of these concepts can be found in the classical differential geometry literature [39] and [77]. In order to minimise notational clutter, independent variables will be suppressed. These variables will only be given explicitly when defining new quantities, or when confusion or ambiguity could result from their omission.

4.1.1 Rotations and Cross Products

A position vector $\mathbf{r}_B$ that is expressed in the coordinate system of a rotating body B can be expressed in an inertial frame using the transformation

$$\mathbf{r}_I = R\,\mathbf{r}_B, \tag{4.1}$$

where R is a rotation matrix that maps the body-fixed axis system into the inertial coordinate system.

All rotation matrices satisfy $RR^T = I_{3\times 3}$ with the added constraint that $\det(R) = 1$; rotation matrices preserve the orientation and are therefore a subset of the orthogonal

matrices. Every rotation matrix can be expressed in terms of an axis-of-rotation unit-vector $\hat{\mathbf{n}}$ and a rotation angle φ. For any rotation matrix R there exists a pair $\hat{\mathbf{n}}$ and φ such that $R = R(\hat{\mathbf{n}}, \varphi)$, in which

$$R(\hat{\mathbf{n}}, \varphi) = \cos(\varphi) I_{3\times 3} + (1 - \cos(\varphi))\hat{\mathbf{n}}\hat{\mathbf{n}}^T + \sin(\varphi) S(\hat{\mathbf{n}}). \tag{4.2}$$

The skew-symmetric matrix $S(\hat{\mathbf{n}})$ is defined by

$$S(\hat{\mathbf{n}}) = \begin{bmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{bmatrix}, \tag{4.3}$$

where n_x, n_y and n_z are the x-, y- and z-axis components of $\hat{\mathbf{n}}$. Equation (4.2) is often referred to as the Rodrigues' rotation formula; $R(\hat{\mathbf{n}}, \varphi)$ performs a rotation about an arbitrary unit vector $\hat{\mathbf{n}}$ through angle φ. Since $\hat{\mathbf{n}}$ is of unit length, $R(\hat{\mathbf{n}}, \varphi)$ is determined by three parameters.

If in some frame of reference the vectors $\mathbf{a}$ and $\mathbf{b}$ have component-wise representations $[a_1\, a_2\, a_3]^T$ and $[b_1\, b_2\, b_3]^T$, then their cross product is given by

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix}.$$

Since the cross product (by $\mathbf{a}$) is a linear operation, it has a matrix representation $S(\mathbf{a})$, where $S(\cdot)$ is defined in (4.3); $\mathbf{a} \times \mathbf{b} = S(\mathbf{a})\mathbf{b}$. It is important to note that rotations are distributive over the cross product; that is $R(\mathbf{a} \times \mathbf{b}) = (R\mathbf{a}) \times (R\mathbf{b})$[1]. A standard calculation shows that

$$RS(\mathbf{a})R^T = S(R\mathbf{a}). \tag{4.4}$$

The properties of cross products will be used to derive a number of formulae relating to motions in rotating reference frames. The arc-length s is now introduced as an independent variable, and geometric entities are defined as functions of s. For any rotation matrix $R(s)$, there holds $d(RR^T)/ds = \dot{R}R^T + (\dot{R}R^T)^T = 0$[2], which implies $\dot{R}R^T = -(\dot{R}R^T)^T$, whence $\dot{R}R^T$ is skew symmetric. Since $R^T\dot{R} = R^T(\dot{R}R^T)R$, $R^T\dot{R}$ is skew-symmetric too.

[1] If $\mathbf{a}\times\mathbf{b} = \mathbf{c}$, then $R(\mathbf{a}\times\mathbf{b}) = R\mathbf{c}$. If $\mathbf{a}' = R\mathbf{a}$, $\mathbf{b}' = R\mathbf{b}$ and $\mathbf{c}' = R\mathbf{c}$, then $\mathbf{a}'\times\mathbf{b}' = \mathbf{c}'$, since the relative orientations and lengths of these vectors have not changed. Consequently $(R\mathbf{a}) \times (R\mathbf{b}) = R\mathbf{c} = R(\mathbf{a} \times \mathbf{b})$.

[2] The dot notation denotes total derivatives with respect to time t. Unit-speed curves will be considered throughout this chapter, so that derivatives with respect to time t and arc length s are equivalent; $\frac{ds}{dt} = 1$. A dot will thus be used to denote derivatives with respect to either variable.

In the case that $\mathbf{r}_B$ is constant, it follows from (4.1) that $\dot{\mathbf{r}}_I = \dot{R}\mathbf{r}_B = \dot{R}R^T\mathbf{r}_I = S(\boldsymbol{\Omega}_I)\mathbf{r}_I = \boldsymbol{\Omega}_I \times \mathbf{r}_I$, where $\boldsymbol{\Omega}_I$ is the angular velocity of the moving frame (expressed in the inertial frame). Alternatively, $\dot{\mathbf{r}}_I = R(R^T\dot{R})R^T\mathbf{r}_I = S(R\boldsymbol{\Omega}_B)\mathbf{r}_I$, where $R^T\dot{R} = S(\boldsymbol{\Omega}_B)$ with $\boldsymbol{\Omega}_B$ is the angular velocity of the moving frame expressed in its own coordinate system. Finally, suppose that the columns of R are the basis vectors of a rotating trihedral coordinate system (expressed in an inertial frame). Then the way that this trihedron moves relative to the inertial frame is described by $\dot{R} = (\dot{R}R^T)R = R(R^T\dot{R})$, and so $\dot{R} = S(\boldsymbol{\Omega}_I)R$, or alternatively $\dot{R} = RS(\boldsymbol{\Omega}_B)$.

4.1.2 Curves

Consider the parametric representation of the *arc of a curve*

$$\mathcal{C} = \left\{\mathbf{c}(s) = [x(s)\ y(s)\ z(s)]^T \in \mathbb{R}^3 : s \in [s_0, s_f]\right\}, \tag{4.5}$$

in which the parameter s is the *arc length* of $\mathcal{C}$. It is assumed that each element of $d^3\mathbf{c}/ds^3$ is continuous and that $d^2\mathbf{c}/ds^2$ is not identically 0 for any s. The curve is deemed *simple*, if any point on the curve corresponds to a single value of s.

A *moving trihedron* associated with $\mathcal{C}$ is defined by a triple of unit vectors $\hat{\mathbf{t}}(s)$, $\hat{\mathbf{p}}(s)$ and $\hat{\mathbf{b}}(s)$ that form a right-handed coordinate system with origin $\mathbf{c}$. The unit vector $\hat{\mathbf{t}}$ is tangent to $\mathcal{C}$ and is defined by $\hat{\mathbf{t}} = \dot{\mathbf{c}}$. The unit vector $\hat{\mathbf{p}}$ is the *principal normal* to $\mathcal{C}$ and is defined as $\hat{\mathbf{p}} = \dot{\hat{\mathbf{t}}}/|\dot{\hat{\mathbf{t}}}|$. The unit vector $\hat{\mathbf{b}}$ is called the *binormal* to the curve, and is defined by the vector product $\hat{\mathbf{t}} \times \hat{\mathbf{p}}$. A pictorial representation of $\mathcal{C}$ and its moving trihedron is shown in Figure 4.1.

The *curvature vector* of $\mathcal{C}$ is defined as the rate of change of the tangent vector, $\boldsymbol{\kappa}(s) = \dot{\hat{\mathbf{t}}}$. The *curvature* is defined as the scalar product $\kappa(s) = \boldsymbol{\kappa} \cdot \hat{\mathbf{p}}$, which is a measure of the deviation of $\mathcal{C}$ from a straight line; it follows that $\boldsymbol{\kappa} = \kappa\hat{\mathbf{p}}$, where

$$\kappa = |\ddot{\mathbf{c}}|\,. \tag{4.6}$$

The *radius of curvature* is the inverse of the curvature $\rho(s) = 1/\kappa$. The *centre of curvature* is given by $\mathbf{C}(s) = \mathbf{c} + \rho\hat{\mathbf{p}}$. The *torsion* of $\mathcal{C}$ measures the direction and extent to which

the curve deviates from its *osculating plane* $\hat{\mathbf{t}}\hat{\mathbf{p}}$. The torsion is defined as $\tau(s) = -\hat{\mathbf{p}} \cdot \dot{\hat{\mathbf{b}}}$; the curvature and torsion are also known as the *first curvature* and *second curvature* respectively. Finally:

$$\tau = \dot{\mathbf{c}} \cdot (\ddot{\mathbf{c}} \times \dddot{\mathbf{c}}) \, |\ddot{\mathbf{c}}|^{-2}, \tag{4.7}$$

which is why $\mathcal{C}$ is assumed 2-smooth (see [39]).

The orientation of the moving trihedron is defined by the three-dimensional rotation matrix

$$R(s) = \begin{bmatrix} \hat{\mathbf{t}} & \hat{\mathbf{p}} & \hat{\mathbf{b}} \end{bmatrix}. \tag{4.8}$$

The curvature and torsion define the *angular velocity* of the moving trihedron through the kinematic equations known as the *Frenet-Serret formulae*:

$$\dot{R} = R \begin{bmatrix} 0 & -\kappa & 0 \\ \kappa & 0 & -\tau \\ 0 & \tau & 0 \end{bmatrix} = RS(\boldsymbol{\Omega}_B). \tag{4.9}$$

It is immediate from (4.3) that each column of (4.9) has a kinematic interpretation of the form $\mathbf{v} = \boldsymbol{\Omega} \times \mathbf{r}$; $\mathbf{v}$ is a velocity vector, $\mathbf{r}$ is a position vector and $\boldsymbol{\Omega}$ is an angular velocity vector of the type familiar in classical mechanics. More particularly, $\dot{\hat{\mathbf{t}}} = \kappa\hat{\mathbf{p}}$, $\dot{\hat{\mathbf{p}}} = \tau\hat{\mathbf{b}} - \kappa\hat{\mathbf{t}}$ and $\dot{\hat{\mathbf{b}}} = -\tau\hat{\mathbf{p}}$. The angular velocity vector of the moving trihedron is known as the *Darboux* vector and is given by $\boldsymbol{\Omega}_B = \tau\hat{\mathbf{t}} + \kappa\hat{\mathbf{b}}$.

By imposing initial conditions on the position $\mathbf{c}(s_0)$ and the orientation $R(s_0)$ of the body-fixed reference frame, the curve $\mathcal{C}$ is generated by

$$\left.\begin{aligned} R(s) &= \int_{s_0}^{s} \dot{R}\, ds, \\ \mathbf{c}(s) &= \int_{s_0}^{s} \hat{\mathbf{t}}\, ds, \end{aligned}\right\} \tag{4.10}$$

in which $\hat{\mathbf{t}}$ is the first column of R in (4.8). The Frenet-Serret equations can be used as a parametrisation of a curve in terms of its curvature and torsion.

4.1.3 Ribbons

A ribbon $\mathcal{R}$ can be created by augmenting $\mathcal{C}$ with notions of 'width' and 'twist'. Both of these quantities will be defined in terms of a unit *camber* vector $\hat{\mathbf{n}}(s)$, which lies in

the plane of the ribbon and is orthogonal to the tangent vector $\mathbf{t}$. The ribbon will be represented parametrically as:

$$\mathcal{R} = \left\{ \mathbf{r}(s,n) = \mathbf{c}(s) + \hat{\mathbf{n}}(s)n \in \mathbb{R}^3 : s \in [s_0, s_f], n \in [n_l(s), n_r(s)] \right\}, \tag{4.11}$$

in which s is the arc length and n is a lateral offset in the ribbon plane. Lateral displacements on the ribbon surface are measured from the spine in the direction of $\hat{\mathbf{n}}$. The width of the ribbon at each s is given by $|n_r(s) - n_l(s)|$. This parametrisation will accommodate variable-width ribbons that need not be symmetric around the spine curve $\mathcal{C}$. If $\mathbf{r}(s,n)$ is a point in $\mathcal{R}$, then it will be assumed that there is a one-to-one mapping between this point and the parameter domain. The curves on $\mathcal{R}$ corresponding to constant s, or constant n, are called *coordinate curves*; the coordinate curve corresponding to $n = 0$ is the spine of the ribbon. The spine is modelled as a simple curve $\mathcal{C}$ and can be described in terms of the geometric entities introduced in Section 4.1.2. The camber vector lies in the plane normal to the spine's tangent vector $\hat{\mathbf{t}}$ and can therefore be described in terms of $\hat{\mathbf{p}}$, $\hat{\mathbf{b}}$ and a *twist angle* $\nu(s)$ as follows

$$\hat{\mathbf{n}} = \hat{\mathbf{p}} \cos\nu - \hat{\mathbf{b}} \sin\nu. \tag{4.12}$$

The *twist rate* is $\dot{\nu}$. A unit vector $\hat{\mathbf{m}}(s)$ that is normal to the ribbon surface is defined to complete a right-handed axis system with $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$, whence $\hat{\mathbf{m}} = \hat{\mathbf{t}} \times \hat{\mathbf{n}}$. See Figure 4.1 for an illustration of a ribbon and these quantities.

The Frenet-Serret formulae can be used to describe the spine, but they are not sufficient to describe the ribbon itself. Developing again a kinematic interpretation of the problem, a moving triad $\hat{\mathbf{t}}\hat{\mathbf{n}}\hat{\mathbf{m}}$, which is called the *ribbon trihedron*, is used to describe the evolution of the spine and the ribbon. The ribbon trihedron is sometimes called the *Darboux frame* of $\mathcal{C}$ on $\mathcal{R}$. The angular velocity of the reference frame fixed to the ribbon trihedron is given by the sum of the Darboux vector and the twist rate:

$$\boldsymbol{\Omega}(s) = [\tau + \dot{\nu}]\, \hat{\mathbf{t}} + \kappa \hat{\mathbf{b}}. \tag{4.13}$$

The ribbon angular velocity vector $\boldsymbol{\Omega}(s)$ can be expressed in the ribbon trihedron

through a coordinate transformation from the curve trihedron (see [56], [36] and [71]):

$$\mathbf{\Omega}_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\nu & \sin\nu \\ 0 & -\sin\nu & \cos\nu \end{bmatrix} \begin{bmatrix} \tau + \dot{\nu} \\ 0 \\ \kappa \end{bmatrix} = \begin{bmatrix} \tau + \dot{\nu} \\ \kappa \sin\nu \\ \kappa \cos\nu \end{bmatrix}. \tag{4.14}$$

The angular velocity vector $\mathbf{\Omega}_B$ is thus a vector in $\mathbb{R}^3$, which has a skew-symmetric matrix representation $S(\mathbf{\Omega}_B)$:

$$\mathbf{\Omega}_B = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix}, \qquad S\left(\mathbf{\Omega}_B\right) = \begin{bmatrix} 0 & -\Omega_z & \Omega_y \\ \Omega_z & 0 & -\Omega_x \\ -\Omega_y & \Omega_x & 0 \end{bmatrix}, \tag{4.15}$$

where Ω_x, Ω_y and Ω_z are known as the *relative torsion*, the *normal curvature* and the *geodesic curvature*, respectively. The matrix $S\left(\mathbf{\Omega}_B\right)$, also known as the *Cartan matrix* (see [38]), is used to generalise the Frenet-Serret formulae to the case of ribbons and

$$\dot{R} = RS\left(\mathbf{\Omega}_B\right). \tag{4.16}$$

As before, the body-fixed reference frame translates at unit speed in the tangent axis direction, starting from the initial conditions $\mathbf{c}(s_0)$ and $R(s_0)$. The ribbon $\mathcal{R}$ is thus given in integral form by

$$\left.\begin{aligned} R(s) &= \int_{s_0}^{s} \dot{R}\, ds, \\ \mathbf{r}(s,n) &= \int_{s_0}^{s} \hat{\mathbf{t}}\, ds + n\hat{\mathbf{n}}, \end{aligned}\right\} \tag{4.17}$$

in which $\dot{R}$ is given by (4.16). The generalised Frenet-Serret equations (4.16) are used within a differential-geometric description of ribbons in terms of their generalised curvature vector $\mathbf{\Omega}_B$ and their width parameters.

4.1.4 Euler Angles

The ribbon model of the road is generated by (4.17), with the evolution of the ribbon trihedron as a function of arc length coming from integrating a 3×3 rotation matrix.

Since $RR^T = I_{3\times3}$ for any rotation matrix R, there are six constraints on its nine entries so that all rotation matrices can be described in terms of three parameters. One such parametrisation is based on *Euler angles*, whereby a general rotation matrix is decomposed into a sequence of three *elementary rotations* about body-fixed axes. Each elementary

rotation is described by a single Euler angle. Discounting degenerate cases corresponding to consecutive rotations about the same axis, twelve possibilities remain, such as xyx, xyz, yxy and so on. As is well known, the three elementary rotations are defined by (4.2):

$$\begin{aligned} R_x(\phi) = R(\hat{\mathbf{e}}_x, \phi) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}, \\ R_y(\mu) = R(\hat{\mathbf{e}}_y, \mu) &= \begin{bmatrix} \cos\mu & 0 & \sin\mu \\ 0 & 1 & 0 \\ -\sin\mu & 0 & \cos\mu \end{bmatrix}, \\ R_z(\theta) = R(\hat{\mathbf{e}}_z, \theta) &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{aligned} \tag{4.18}$$

in which the subscripts represent the axes around which rotations occur and the arguments the associated angles; $\hat{\mathbf{e}}_x$, $\hat{\mathbf{e}}_y$ and $\hat{\mathbf{e}}_z$ are unit vectors associated with the x-, y- and z-axes. The sequence of rotations must be chosen in a way best suited to the problem. One of the key issues relates to the fact that every sequence of rotations has an associated subspace of singular configurations in which the values of the Euler angles are not uniquely defined for the given rotation. This happens when the second rotation aligns the first and the third rotation axes, and the first and third Euler angles can be chosen arbitrarily. Singular configurations may be an issue when the problem of finding the Euler angles corresponding to a given rotation matrix is considered. These are known as *inverse kinematics problems*.

In this road-modelling work a zyx Euler angle sequence is used, in which the first rotation θ (yaw) is associated with the direction of the road in the xy inertial plane, the second angle μ (pitch) describes the slope in the direction of travel, while the third angle ϕ (roll) describes the road camber. Singular configurations for this rotation sequence correspond to a road with a pitch angle of $\pm\pi/2$, which does not exist in usable driving environments. For this choice of Euler angles the rotation matrix is parametrised as

$$R = R_z(\theta)R_y(\mu)R_x(\phi) = \begin{bmatrix} \mathrm{c}_\theta\mathrm{c}_\mu & \mathrm{c}_\theta\mathrm{s}_\mu\mathrm{s}_\phi - \mathrm{s}_\theta\mathrm{c}_\phi & \mathrm{c}_\theta\mathrm{s}_\mu\mathrm{c}_\phi + \mathrm{s}_\theta\mathrm{s}_\phi \\ \mathrm{s}_\theta\mathrm{c}_\mu & \mathrm{s}_\theta\mathrm{s}_\mu\mathrm{s}_\phi + \mathrm{c}_\theta\mathrm{c}_\phi & \mathrm{s}_\theta\mathrm{s}_\mu\mathrm{c}_\phi - \mathrm{c}_\theta\mathrm{s}_\phi \\ -\mathrm{s}_\mu & \mathrm{c}_\mu\mathrm{s}_\phi & \mathrm{c}_\mu\mathrm{c}_\phi \end{bmatrix}, \tag{4.19}$$

in which 's' and 'c' correspond to the sine and cosine of the subscript angles.

Suppose $\mathbf{P}$ is a point stationary in a moving reference frame. The coordinates of

the position vector of this point, when expressed in the moving frame, are constant and denoted $\mathbf{r}_B$. In an inertial reference frame this point is described by the position vector $\mathbf{r}_I = R\mathbf{r}_B$. The velocity of this point (expressed in the inertial frame) is found as follows:

$$\dot{\mathbf{r}}_I = \dot{R}\mathbf{r}_B = \dot{R}R^T\mathbf{r}_I = S\left(\boldsymbol{\Omega}_I\right)\mathbf{r}_I = \boldsymbol{\Omega}_I \times \mathbf{r}_I. \tag{4.20}$$

The angular velocity vector $\boldsymbol{\Omega}_I$ is expressed in the inertial frame; $\boldsymbol{\Omega}_B = R^T\boldsymbol{\Omega}_I$ for the moving frame. If these ideas are applied to the rotation matrix given in (4.20), an equivalent expression for the speed of $\mathbf{P}$ can be derived as follows:

$$\begin{aligned}
\dot{\mathbf{r}}_I &= \dot{R}R^T\mathbf{r}_I \\
&= \left(\dot{R}_z R_y R_x + R_z \dot{R}_y R_x + R_z R_y \dot{R}_x\right) R_x^T R_y^T R_z^T \mathbf{r}_I \\
&= \left(\dot{R}_z R_z^T + R_z \dot{R}_y R_y^T R_z^T + R_z R_y \dot{R}_x R_x^T R_y^T R_z^T\right) \mathbf{r}_I \\
&= \left(S(\dot{\theta}\hat{\mathbf{e}}_z) + R_z S(\dot{\mu}\hat{\mathbf{e}}_y) R_z^T + R_z R_y S(\dot{\phi}\hat{\mathbf{e}}_x) R_y^T R_z^T\right) \mathbf{r}_I \\
&= \left(\dot{\theta}\hat{\mathbf{e}}_z + \dot{\mu} R_z \hat{\mathbf{e}}_y + \dot{\phi} R_z R_y \hat{\mathbf{e}}_x\right) \times \mathbf{r}_I.
\end{aligned} \tag{4.21}$$

Since the equalities (4.20) and (4.21) must hold for all points $\mathbf{P}$, one obtains

$$\boldsymbol{\Omega}_B = R^T(\dot{\theta}\hat{\mathbf{e}}_z + \dot{\mu} R_z \hat{\mathbf{e}}_y + \dot{\phi} R_z R_y \hat{\mathbf{e}}_x) = K[\dot{\theta}\,\dot{\mu}\,\dot{\phi}]^T, \tag{4.22}$$

with

$$K = \begin{bmatrix} R_x^T R_y^T \hat{\mathbf{e}}_z & R_x^T \hat{\mathbf{e}}_y & \hat{\mathbf{e}}_x \end{bmatrix}. \tag{4.23}$$

As in all mechanics problems, the idea is to replace the 3×3 matrix in (4.16) with three scalar differential equations in the Euler angles. In this case the kinematic parametrisation of the ribbon (4.17) becomes

$$\left.\begin{aligned}
\theta(s) &= \int_{s_0}^{s} \dot{\theta}(s)\, ds, \\
\mu(s) &= \int_{s_0}^{s} \dot{\mu}(s)\, ds, \\
\phi(s) &= \int_{s_0}^{s} \dot{\phi}(s)\, ds, \\
\mathbf{r}(s,n) &= \int_{s_0}^{s} \hat{\mathbf{t}}(s)\, ds + n\hat{\mathbf{n}}(s),
\end{aligned}\right\} \tag{4.24}$$

where $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$ are, respectively, the first and second columns of the rotation matrix R as given in (4.19), and the initial conditions $\theta(s_0)$, $\mu(s_0)$ and $\phi(s_0)$ are found by solving the

inverse problem for $R(s_0)$. The generalised curvature vector components are determined using (4.22), in which the Jacobian matrix K for the selected Euler-angle convention is

$$K(s) = \begin{bmatrix} -\mathrm{s}_\mu & 0 & 1 \\ \mathrm{c}_\mu \mathrm{s}_\phi & \mathrm{c}_\phi & 0 \\ \mathrm{c}_\mu \mathrm{c}_\phi & -\mathrm{s}_\phi & 0 \end{bmatrix}, \tag{4.25}$$

as derived from (4.23).

4.2 Numerical Optimal Control

The overarching aim of the work presented in this chapter is to use measured data to find a continuous curvilinear-coordinate-based road model that can be used in the solution of vehicular optimal-control problems. It is assumed that the road-related measurement data comprises discrete GPS coordinate measurements of the track boundaries. These measurements will typically be noisy and need not be equispaced. Let us suppose that the left-hand boundary measurements are given by $\mathcal{B}_l = \{\mathbf{b}_{li} \in \mathbb{R}^3 : i = 1 \dots N\}$. The right-hand boundary measurements are given by $\mathcal{B}_r = \{\mathbf{b}_{ri} \in \mathbb{R}^3 : i = 1 \dots N\}$; equal numbers of right- and left-hand boundary points are assumed[3].

The track centre line can be found by a simple nearest-neighbour search and averaging calculation. Suppose the first right-hand boundary point $\bar{\mathbf{b}}_{r1} = \mathbf{b}_{r1}$ is selected. One then searches the left-hand boundary measurement set for its closest neighbour, which is referred to as the first left-hand boundary point $\bar{\mathbf{b}}_{l1}$. The first track centre point is $\bar{\mathbf{c}}_1 := \left(\bar{\mathbf{b}}_{l1} + \bar{\mathbf{b}}_{r1}\right)/2$. By repeating this process for all the right-hand boundary points, estimates for the track centre line $\bar{\mathcal{C}} := \{\bar{\mathbf{c}}_i \in \mathbb{R}^3 : 1 \dots N\}$ can be found. The arc lengths associated with each point are found by summing the incremental distances between successive measurement points. Suppose that the GPS coordinates of $\bar{\mathbf{c}}_i$ are known and that $\Delta s_i := |\bar{\mathbf{c}}_{i+1} - \bar{\mathbf{c}}_i|$ for $i = 1, \dots N-1$, then the arc lengths associated with each point are given by $s_j := \sum_{i=1}^{j-1} \Delta s_i$ for $j = 1, \dots N-1$. If the track is closed one can define $\Delta s_N := |\bar{\mathbf{c}}_1 - \bar{\mathbf{c}}_N|$, with s_N the track length in this case.

[3]Interpolation can be used to guarantee this condition if the number of measurement points for the two boundaries is different, and to populate segments of the boundaries where the data is sparse.

This simple heuristic procedure provides a sets of values for the arc length $\mathcal{S} := \{s_1, \ldots, s_N\}$ with associated Cartesian coordinates for the centre line $\bar{\mathbf{c}}(s_i)$ and boundaries $\bar{\mathbf{b}}_l(s_i)$ and $\bar{\mathbf{b}}_r(s_i)$; here these coordinates are defined as vector-valued functions on the set $\mathcal{S}$: $\bar{\mathbf{c}}(s_i) := \bar{\mathbf{c}}_i$, $\bar{\mathbf{b}}_l(s_i) := \bar{\mathbf{b}}_{li}$, $\bar{\mathbf{c}}_r(s_i) := \bar{\mathbf{c}}_{ri}$. Alternative methods for the generation of these functions may be used when deemed appropriate. Interpolation is used to extrapolate coordinates for continuous values of s.

4.2.1 Finite-Differencing

In the absence of noise, the track might be described in terms of curvature variables determined by numerical differentiation. If Euler-forward differencing is used, an approximate ribbon parametrisation can be found using

$$\begin{aligned}
\hat{\mathbf{t}}(s_i) &= \frac{\bar{\mathbf{c}}(s_{i+1}) - \bar{\mathbf{c}}(s_i)}{|\bar{\mathbf{c}}(s_{i+1}) - \bar{\mathbf{c}}(s_i)|}, \\
\hat{\mathbf{n}}(s_i) &= \frac{\bar{\mathbf{b}}_r(s_i) - \bar{\mathbf{c}}(s_i)}{|\bar{\mathbf{b}}_r(s_i) - \bar{\mathbf{c}}(s_i)|}, \\
\hat{\mathbf{m}}(s_i) &= \hat{\mathbf{t}}(s_i) \times \hat{\mathbf{n}}(s_i), \\
R(s_i) &= \left[\hat{\mathbf{t}}(s_i)\ \hat{\mathbf{n}}(s_i)\ \hat{\mathbf{m}}(s_i)\right], \\
\dot{R}(s_i) &= \frac{1}{\Delta s_i}\left(R(s_{i+1}) - R(s_i)\right).
\end{aligned} \tag{4.26}$$

The generalised curvature vector $\mathbf{\Omega}_B(s_i)$ is determined from $S(\mathbf{\Omega}_B(s_i)) = R^T(s_i)\dot{R}(s_i)$. Finally, the road width on either side of the centre line is approximated by

$$\begin{aligned}
n_l(s_i) &= \hat{\mathbf{n}}(s_i) \cdot (\bar{\mathbf{b}}_l(s_i) - \bar{\mathbf{c}}(s_i)), \\
n_r(s_i) &= \hat{\mathbf{n}}(s_i) \cdot (\bar{\mathbf{b}}_r(s_i) - \bar{\mathbf{c}}(s_i)).
\end{aligned} \tag{4.27}$$

The danger with this approach is explicit in (4.6) and (4.7), which show that the determination of curvature-type variables require that position variables be differentiated at least twice; this process is not viable when noise is present in the underlying measurements.

4.2.2 Noise Influences

Noisy road-boundary measurements can have a negative impact on the solution of related vehicular simulation and optimal-control problems. As is evident from equations such

as (4.6) and (4.7), boundary noise will re-appear (amplified by multiple differentiation operations) in road models such as those described in Section 4.1.4.

When numerical methods are used to solve the equations of motion, particularly if adaptive-step-size algorithms are used, modelling noise can take on the guise of 'fast dynamics' resulting in foreshortened integration step lengths within numerical integration schemes (see [11], [24] and [28]). Reduced integration step lengths will increase the computation times, and this effect must be managed. In optimal control, mesh-refinement schemes are used to improve the speed and accuracy (see [8]). In pseudospectral methods for optimal control, variable order orthogonal polynomials, which are used in Gaussian quadrature schemes, are used for the same purpose (see [20]). Again, noise in the road model will drive erroneously the meshing and polynomial order adaptation algorithms. Since noise disrupts both the numerical-integration and the optimal-control solution processes it should be removed where possible.

Another issue relates to the introduction of drift phenomena in road modelling. As has been demonstrated in [18], classical filtering techniques can be used to remove noise, but these processes tend to introduce drift that changes the road geometry and prevents circuit closure. In this case the circuit's beginning and end are no longer at the same point in space.

As it is now shown, an appropriately formulated optimal control problem can be used to reject measurement noise from the road model, while simultaneously guaranteeing track closure. In essence, the use of an optimal filtering process with boundary constraints is proposed.

4.2.3 Problem Formulation

The track-smoothing optimal-control problem is embedded in the following general framework:

$$\begin{aligned} &\min_{\{\boldsymbol{u}(\cdot),\boldsymbol{x}(s_0)\}} \int_{s_0}^{s_f} L(s,\boldsymbol{x}(s),\boldsymbol{u}(s))ds, \\ &\text{s.t.} \begin{cases} \dot{\boldsymbol{x}} &= \mathbf{f}(s,\boldsymbol{x}(s),\boldsymbol{u}(s)) \\ 0 &\geq \mathbf{h}(s,\boldsymbol{x}(s),\boldsymbol{u}(s)) \\ 0 &= \mathbf{g}_b(\boldsymbol{x}(s_0),\boldsymbol{x}(s_f)). \end{cases} \end{aligned} \tag{4.28}$$

The integration variable is the arc length s. The state and the control vectors are denoted $\boldsymbol{x}(s)$ and $\boldsymbol{u}(s)$ respectively. The system dynamics are described by the vector-valued equation $\dot{\boldsymbol{x}} = \mathbf{f}(s, \boldsymbol{x}, \boldsymbol{u})$. The vector-valued function $\mathbf{h}(s, \boldsymbol{x}, \boldsymbol{u})$ and the scalar function $L(s, \boldsymbol{x}, \boldsymbol{u})$ define the path inequality constraints and the cost integrand, respectively, with the boundary equality constraints described by the vector-valued function $\mathbf{g}_b(\boldsymbol{x}(s_0), \boldsymbol{x}(s_f))$.

4.2.3.1 Dynamics

A dynamic model based on the Euler angles θ, μ and ϕ is used to represent the ribbon. The coordinates of the origin of the moving ribbon trihedron are given by x, y and z. The right- and left-hand track widths are given by n_l and n_r. The state equations are based on (4.24), with the state vector given by

$$\boldsymbol{x} = [x \;\; y \;\; z \;\; \theta \;\; \mu \;\; \phi \;\; \dot{\theta} \;\; \dot{\mu} \;\; \dot{\phi} \;\; n_l \;\; n_r]^T. \tag{4.29}$$

The control vector is

$$\boldsymbol{u} = [\ddot{\theta} \;\; \ddot{\mu} \;\; \ddot{\phi} \;\; \dot{n}_l \;\; \dot{n}_r]^T. \tag{4.30}$$

The ribbon is generated by the control variables and the system dynamics

$$\mathbf{f}(s,\boldsymbol{x},\boldsymbol{u}) = \begin{bmatrix} \cos\theta\cos\mu & \sin\theta\cos\mu & -\sin(\mu) & \dot{\theta} & \dot{\mu} & \dot{\phi} & \ddot{\theta} & \ddot{\mu} & \ddot{\phi} & \dot{n}_l & \dot{n}_r \end{bmatrix}^T. \tag{4.31}$$

When the model is in this form, state and input bounds can be used to limit the velocities and accelerations of the Euler angles, and the velocity and magnitude variations in the track width.

4.2.3.2 Cost

The aim of the optimal-control problem is to minimise the Euclidean distances between the ribbon model given in (4.29), (4.30) and (4.31), and the measurement-related data $\bar{\mathbf{b}}_l$, $\bar{\mathbf{b}}_r$ and $\bar{\mathbf{c}}$. The coordinates of the ribbon spine are given by the arc-length-dependent states x, y and z, while the coordinates of the boundaries are the *system outputs* $\mathbf{b}_l$ and $\mathbf{b}_r$ as described by

$$\begin{aligned} \mathbf{c} &= [x\ y\ z]^T, \\ \mathbf{b}_l &= \mathbf{c} + \hat{\mathbf{n}} n_l, \\ \mathbf{b}_r &= \mathbf{c} + \hat{\mathbf{n}} n_r, \end{aligned} \tag{4.32}$$

with $\hat{\mathbf{n}}$ given by the second column of (4.19).

The cost integrand function $L(s, \boldsymbol{x}, \boldsymbol{u})$ in (4.28) is made up of three terms:

$$L(s, \boldsymbol{x}, \boldsymbol{u}) = e(s, \boldsymbol{x}, \boldsymbol{u}) + r_c(s, \boldsymbol{u}) + r_w(s, \boldsymbol{u}). \tag{4.33}$$

The first term in $L(s, \boldsymbol{x}, \boldsymbol{u})$ describes the *tracking error*, which also contains three terms as follows:

$$e(s, \boldsymbol{x}, \boldsymbol{u}) = w_c ||\mathbf{c} - \bar{\mathbf{c}}||^2 + w_{b_l} ||\mathbf{b}_l - \bar{\mathbf{b}}_l||^2 + w_{b_r} ||\mathbf{b}_r - \bar{\mathbf{b}}_r||^2, \tag{4.34}$$

in which w_c, w_{b_l} and w_{b_r} are weighting coefficients.

The first term in (4.34) describes the error between the spine curve $\mathbf{c}$ and the road centre line $\bar{\mathbf{c}}$, which is a curve that interpolates the discrete measurement data. The second and third terms relate to the errors between the ribbon edges and the curves $\bar{\mathbf{b}}_l$ and $\bar{\mathbf{b}}_r$, which interpolate the track boundary data.

The second term in (4.33) is used to attenuate the high-frequency content in the ribbon curvatures. This is achieved through the *curvature rate* term $r_c(s, \boldsymbol{u})$, which is defined as the weighted sum of the squares of the second derivatives of the Euler angles

$$r_c(s, \boldsymbol{u}) = w_\theta \ddot{\theta}^2(s) + w_\mu \ddot{\mu}^2(s) + w_\phi \ddot{\phi}^2(s), \tag{4.35}$$

in which w_θ, w_μ and w_ϕ are again weighting terms.

The third term in (4.33) is used to remove high-frequency noise from the track boundaries. The *width rate* is defined as the weighted sum of the squared derivatives of the left

and right offsets

$$r_w(s, \boldsymbol{u}) = w_{n_l}\dot{n}_l^2(s) + w_{n_r}\dot{n}_r^2(s); \tag{4.36}$$

w_{n_l} and w_{n_r} are weighting terms.

The eight weighting factors are used to quantify the relative importance of the various influences in the optimal-control problem. In broad terms, more accurate tracking is obtained with larger weighting factors in (4.34), whereas larger weights in (4.35) and (4.36) lead to smoother results. These weights are design parameters that have to be tuned in every application.

4.2.3.3 Constraints

As mentioned in Section 4.1.4 singular configurations must be avoided. In order to exclude Euler-angle-related singularities from the optimal-control problem, the following box constraint is placed on the pitch angle: $-\pi/2 < \mu < \pi/2$, which limits the absolute value of the modelled road slope.

Boundary constraints on the initial and final positions and orientations of the ribbon trihedron will be applied. Such constraints are of the form $\boldsymbol{x}(s_0) = \boldsymbol{x}_0$ and $\boldsymbol{x}(s_f) = \boldsymbol{x}_f$, where the initial and final references $\boldsymbol{x}_0$ and $\boldsymbol{x}_f$ are found[4] using the procedure in Section 4.2.1. In some cases the initial and final values of the states may be left unconstrained, relying on the optimisation process to establish optimal values for these variables. When modelling closed road circuits, such as race tracks, cyclic constraints are required. In this case one must enforce $\boldsymbol{x}(s_f) = \boldsymbol{x}(s_0)$, which ensures continuity of the state across the start-finish line.

Note that the periodicities in the Euler angles may have to be accommodated, which allow the boundary angles to differ by multiples of π. For 'real' closed circuits the pitch and roll angles are identical at the boundaries. However, if an integer number of complete clockwise rotations (k say) is imposed by the shape of the circuit, a constraint of the form $\theta(s_f) - \theta(s_0) - 2k\pi = 0$ may be required.

[4] The forward Euler scheme for finite differences can be used for x_0, whereas x_f has to be defined using a backward finite-differentiation rule.

4.3 Results

The optimal-control problem formulated in Section 4.2.3 is used to parametrise two ribbon models. The first is illustrative, and demonstrates the optimal-control solution method for a three-dimensional ribbon with a known analytical parametrisation. One purpose of this application is to validate the identification method using a known track model as a reference. The second example involves the parametrisation of the Circuit de Catalunya, in Barcelona, Spain. The boundary points for the racing track are in form of GPS data. The generalised curvature vector and the distances of the boundaries from the ribbon spine are derived using an optimal-control calculation. The continuous-time[5] optimal-control problems are solved using a direct collocation method. In each case the solution of the resulting discrete-time optimal-control problem is then found using NLP techniques. The toolbox GPOPS-II (see [20, 60, 61]) was used in conjunction with IPOPT (see [87]) for the solutions presented in this chapter. GPOPS-II employs an *hp*-adaptive Gaussian quadrature integration scheme with the collocation performed at Legendre-Gauss-Radau quadrature points. A sparse finite-differencing process is employed to estimate all the first- and second-order derivatives required by the NLP solver IPOPT.

4.3.1 Lemniscate Ribbon

The illustrative ribbon considered here is based on the lemniscate of Bernoulli (see [44]), which has been augmented with an elevation dimension in order to generate a spacial spine curve. A camber angle is also introduced that describes the direction in which the ribbon's width evolves.

The original lemniscate is a plane curve defined as the locus of points for which the product of the distances from two fixed foci at $x = \pm c$ is constant:

$$\left((x-c)^2 + y^2\right)\left((x+c)^2 + y^2\right) = c^4. \tag{4.37}$$

[5] The independent variable is the arc length.

In the case that $c = 1/\sqrt{2}$, the lemniscate is described parametrically as

$$\begin{aligned} x(p) &= \frac{\cos(p)}{1+\sin(p)^2}, \\ y(p) &= \frac{\sin(p)\cos(p)}{1+\sin(p)^2}, \end{aligned} \tag{4.38}$$

for $p \in [0, 2\pi]$. It is well known that the parameter p and the lemniscate arc length s are related by an incomplete elliptic integral of the first kind. Such integrals must be evaluated numerically and so the differential-geometric properties of lamniscate-related curves are also best computed numerically.

If the following z-coordinate is introduced

$$z(p) = \frac{1}{10}\sin(p), \tag{4.39}$$

one obtains a three-dimensional curve in the form of a figure of eight. In the first example this space curve is used as the spine of a ribbon that is then augmented with a symmetric width of the form

$$n_r(p) = -n_l(p) = \frac{1}{20}\left(1 + \frac{1}{2}\cos(2p)\right). \tag{4.40}$$

The direction of the camber vector $\hat{\mathbf{n}}(p)$, which is orthogonal to the curve tangent $\hat{\mathbf{t}}(p)$, is deemed to lie at an angle of

$$\phi(p) = \cos(p) \tag{4.41}$$

relative to the inertial xy-plane. This camber is measured around $\hat{\mathbf{t}}$ with a right-hand rule. Figure 4.2 shows the ribbon surface as a grey-shaded membrane. The ribbon spine and its boundaries are given by

$$\begin{aligned} \mathbf{c} &= \begin{bmatrix} x & y & z \end{bmatrix}^T, \\ \mathbf{b}_l &= \mathbf{c} + n_l\hat{\mathbf{n}}, \\ \mathbf{b}_r &= \mathbf{c} + n_r\hat{\mathbf{n}}. \end{aligned} \tag{4.42}$$

The optimal estimation problem is solved for the simulated data sets

$$\begin{aligned} \bar{\mathbf{b}}_l(s_i) &= \mathbf{b}_l(s_i) + \Delta_l, \\ \bar{\mathbf{b}}_r(s_i) &= \mathbf{b}_r(s_i) + \Delta_r, \end{aligned} \tag{4.43}$$

in which $\mathbf{b}_l(s_i)$ and $\mathbf{b}_r(s_i)$ are sampled values of $\mathbf{b}_l$ and $\mathbf{b}_r$, respectively, at each sample point s_i. The sequences Δ_l and Δ_r are realisations of additive measurement noise processes that will be described below.

The data associated with the optimal control problem is a finite set of p-equispaced boundary samples computed from (4.43), in which the components of Δ_l and Δ_r are independent realisations of $\mathcal{U}(-.01, .01)$; $\mathcal{U}(\cdot, \cdot)$ is a uniform distribution with the designated prescribed limits. The maximum noise-related displacement of each boundary point is thus $\sqrt{3}/100$, with the minimum track width given by $5/100$.

In this application the weights in the optimal control cost (see Section 4.2.3.2) were selected as follows: $w_c = w_{b_l} = w_{b_r} = 10^2$, $w_\theta = 10^{-3}$, $w_\mu = 5 \cdot 10^{-2}$, $w_\phi = 10^{-3}$, and $w_{n_l} = w_{n_r} = 1$.

The solution to the optimal-estimation problem is shown in Figure 4.2. It is evident that the spine and boundary curves are 'smooth' in comparison with the raw reference data on which the optimal parameter estimates are based. The three differential-geometric pa-

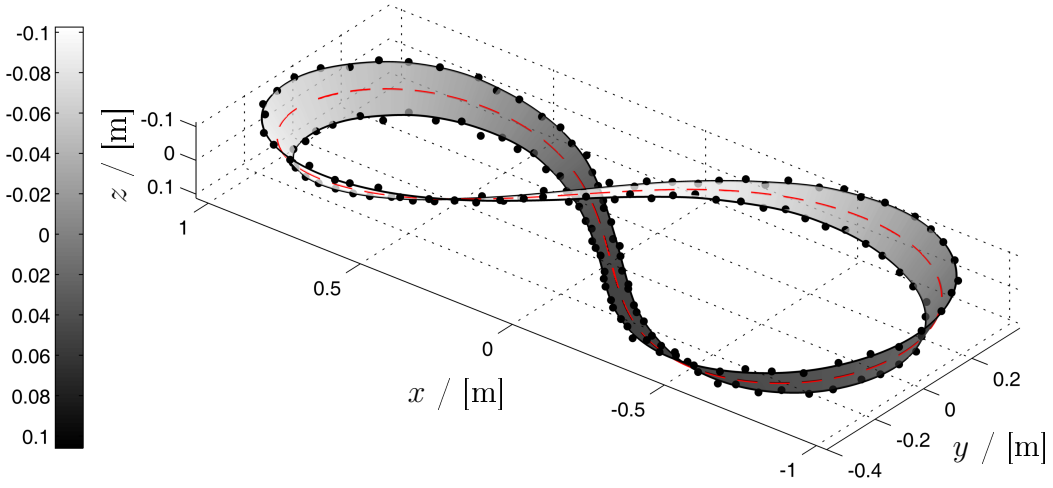

Figure 4.2: Three-dimensional ribbon for the extended lemniscate of Bernoulli. The solid and dashed lines are the optimised boundaries and centre line, respectively. The dots represent the noisy simulated boundary-point measurements on which the solution is based. The ribbon grey scale represents the z-coordinate of the optimised centre line and is calibrated according to the colour bar given.

rameters (generalised curvatures) for the optimised ribbon spine are shown in Figure 4.3. In each case the theoretical values and the optimal estimates are displayed, with good

agreement between the two demonstrated.

The track width and its optimal estimates are also shown in Figure 4.3. The oscillations in the estimates around the true values are a consequence of the additive noise in the simulated boundary points; see (4.43). To further reduce these oscillations one could increase the w_{n_l} and w_{n_r} weighting coefficient (as defined in Section 4.2.3.2), but this would come at the cost of increased errors in the estimates. It is obviously desirable to have some frequency separation between measurement noise and the characteristic frequencies of the track itself. It may be necessary to obtain this track-noise separation by collecting finer measurements (in terms of both distance between measurement points and measuring accuracy), especially where the track contains important higher-frequency features.

4.3.2 Circuit de Catalunya

The second application of the optimal ribbon parametrisation aims to model the Circuit de Catalunya, which is the current Spanish Formula 1 Grand Prix race track. The track boundary data were collected manually from satellite pictures combined with GPS coordinate data available from *earth.google.com*. The measured data points are not evenly spaced along the track edges. The elevation of each boundary point was found using altitude data from *www.gpsvisualizer.com*; the altitude variation around the track was found to be approximately 30 m. This data was transformed into local SAE Cartesian coordinates using the MATLAB Geodetic Toolbox in compliance with the WGS84 geodetic system[6]. The inertial reference frame has its x- and y-axes tangent to the Earth's surface, centred at a point enclosed by the track, and with its x-axis parallel to the equatorial plane. The lowest point on the track was deemed to have an altitude of 0 m. This process produced two three-dimensional data sets describing the track's left- and right-hand boundaries. The circuit length is 4655 m, and approximately three hundred data points were used for each track boundary.

[6]The World Geodetic System is a standard for use in cartography and geodesy. It comprises a standard coordinate frame for the Earth, a datum or reference ellipsoid for altitude data and a gravitational equipotential surface (the geoid) that defines nominal sea level.

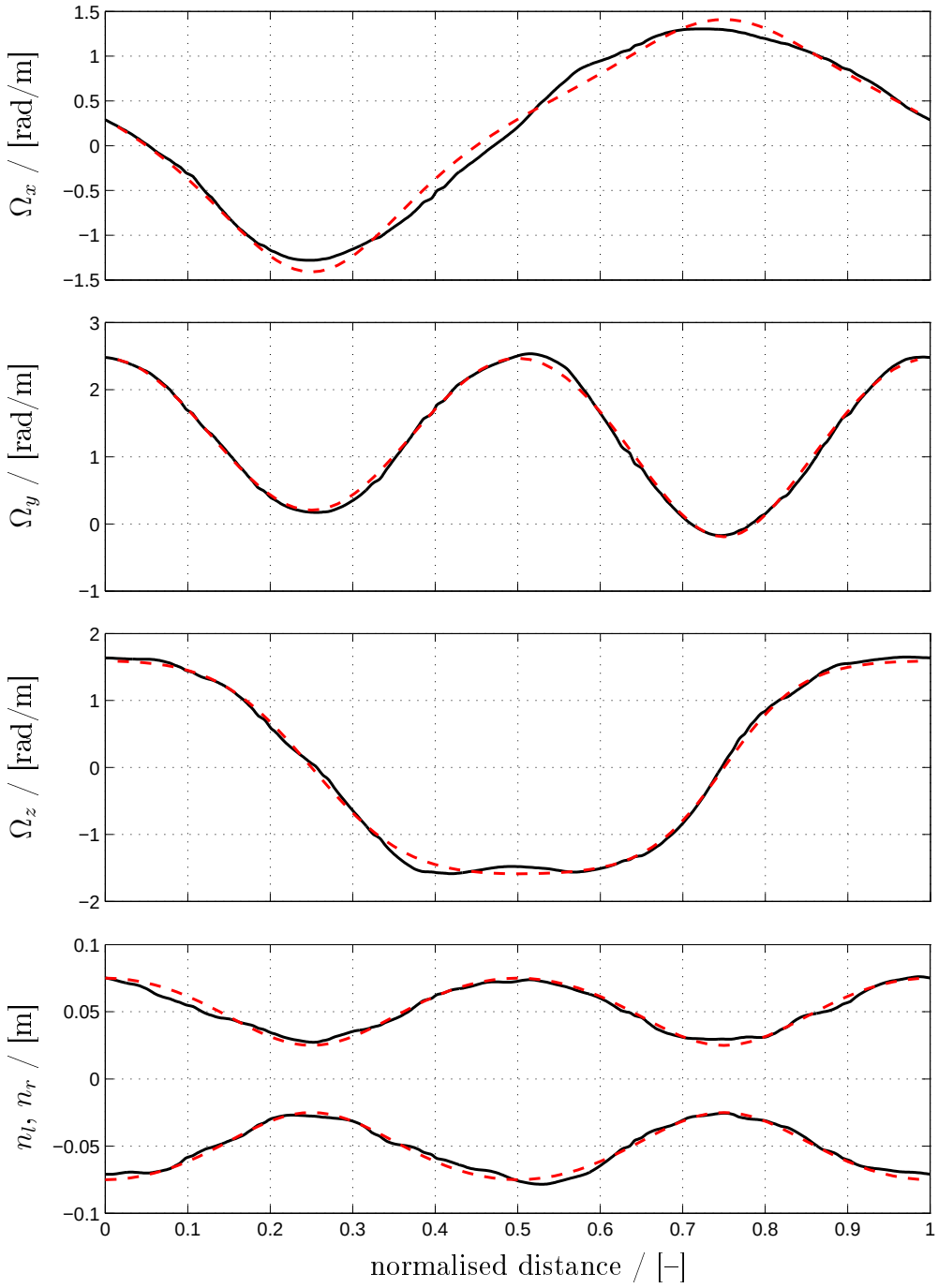

Figure 4.3: Parameters for the extended lemniscate ribbon. The theoretically correct values are the red dashed curves, while the optimal estimates are the solid black curves. The top plot shows the relative torsion Ω_x, the second the normal curvature Ω_y, and the third the geodesic curvature Ω_z. The distances between the ribbon boundaries and its spine are shown in the bottom diagram. The theoretical boundaries come from (4.40).

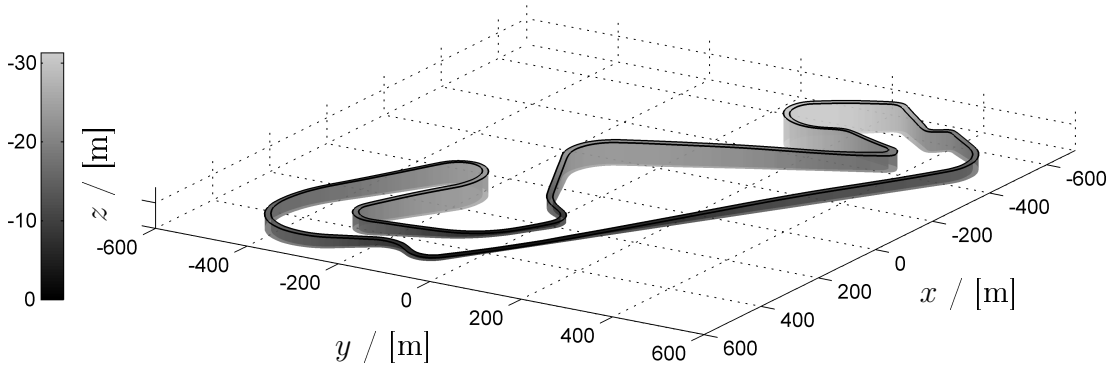

Figure 4.4: Ribbon-based model of the Circuit de Catalunya. The solid lines are the optimised track boundaries. The track spine's z-coordinate is shown using a grey scale.

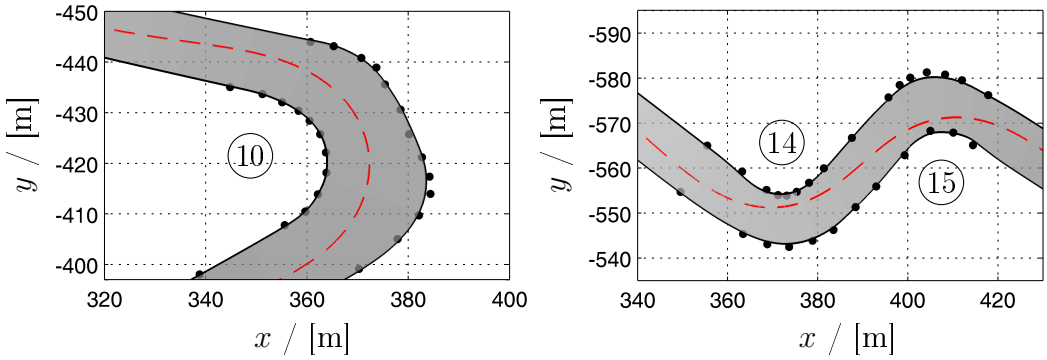

Figure 4.5: Magnified views of corner (10) (left-hand side), and corners (14) and (15) (right-hand side). The measured boundary points are shown as black dots, with the dashed red curve the spine estimate.

For this problem the weights in the optimal-control cost function (see Section 4.2.3.2) were chosen as: $w_c = w_{b_l} = w_{b_r} = 10^{-3}$, $w_\theta = 3 \cdot 10^3$, $w_\mu = w_\phi = 10^9$, and $w_{n_l} = w_{n_r} = 10^{-2}$. These weights were chosen to best reflect the problem requirements and the units used. The weight selection is a design choice and will almost always require some form of trial-and-error tuning process. The optimal ribbon found via the solution of the optimal-control problem is displayed in Figure 4.4. The corners where the maximum boundary tracking errors occur are shown in Figure 4.5. This figure illustrates clearly the effects of the smoothing process: the optimised boundary lines follow closely the measured boundary points, but without the introduction of sharp kinks or rapid changes of curvature.

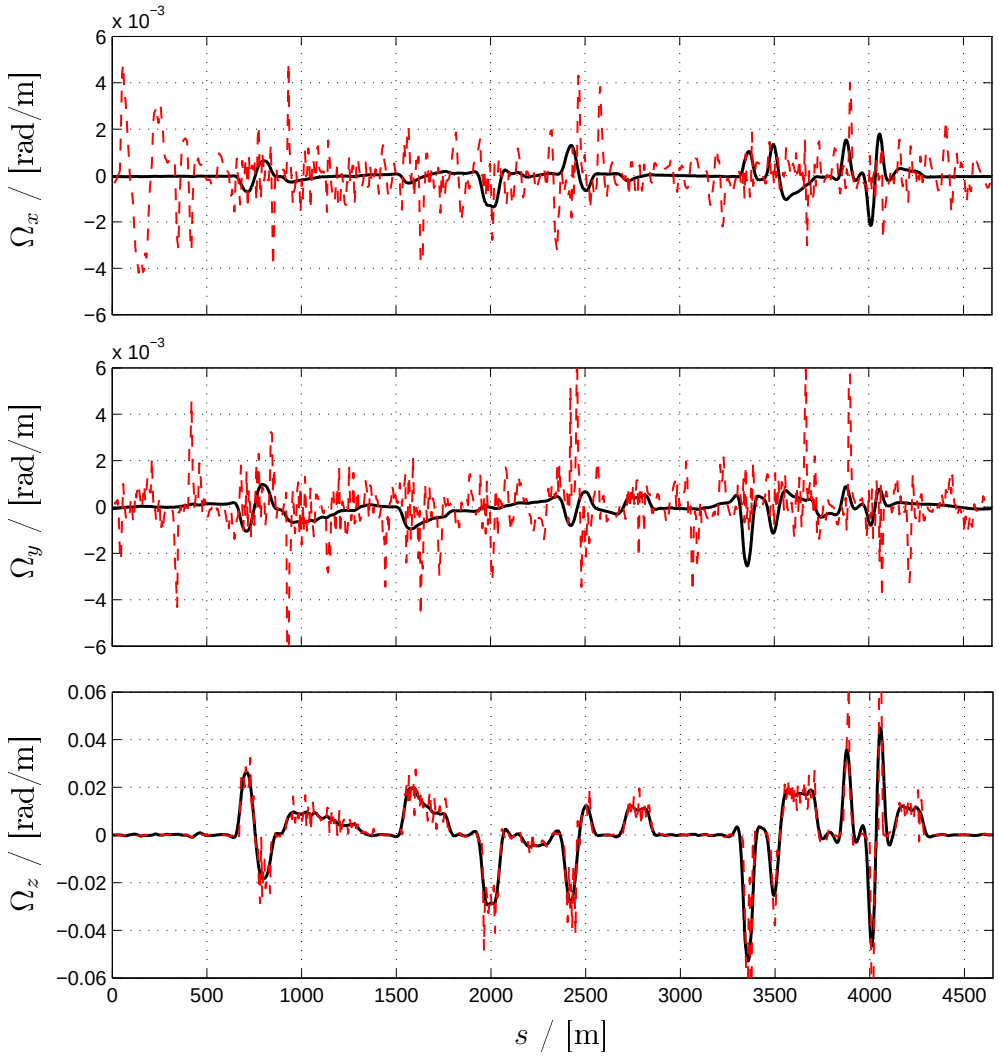

Figure 4.6: Estimated curvature of the Circuit de Catalunya. The plots show, respectively, the relative torsion Ω_x, the normal curvature Ω_y and the geodesic curvature Ω_z. The optimal model-based estimates are shown as the solid black curves, while the finite-difference approximations are the red dashed curves.

The three diagrams in Figure 4.6 provide a comparison between the differential parameters obtained through optimal control and the finite-difference approximation described in Section 4.2.1. It is clear that the optimally-smoothed track parameters are more suitable for subsequent vehicular simulation and/or optimal-control studies. The relative torsion Ω_x, the normal curvature Ω_y and the geodesic curvature Ω_z are shown in order in Figure 4.6.

The width of the optimised ribbon is represented in Figure 4.7. It appears that the track has an approximate width of twelve metres, with the exception of turn (10), where the

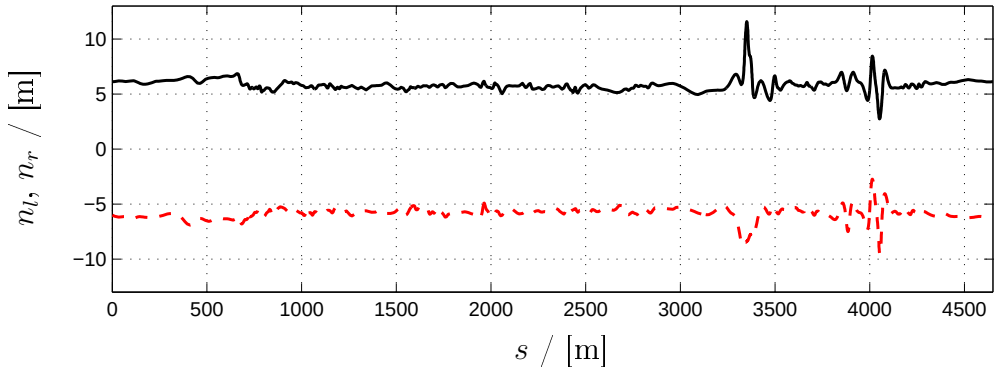

Figure 4.7: Distances to the right-hand (solid black) and left-hand (dashed red) boundaries of the track as measured from the track's spine.

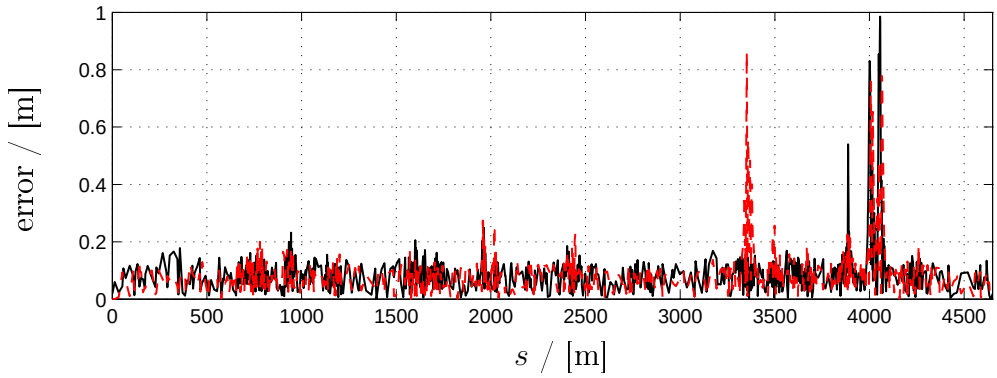

Figure 4.8: Error estimates for the track boundaries (as calculated in the ribbon tangent plane). The solid black curve corresponds to the right-hand-side boundary, while the dashed red curve corresponds to the left-hand boundary.

width increases due to the corner's particular profile. Although the track width is almost constant through the corners (14) and (15), some oscillations appear in the estimates. This is due to the high geodesic curvature associated with these corners, where large curvature variations are smoothed by the optimal estimation process. In order to achieve this smoothing the ribbon axis is 'taking a racing line' through these corners causing oscillatory variations in the boundary-to-spine distances.

Distance errors between the optimised track boundaries and the noisy measurement points are given in Figure 4.8. For the most part the estimated errors are less than 0.2 m, which is comparable with the accuracy expected of the process used to generate the original data. Not unexpectedly, peaks in the boundary tracking errors correspond to corners (10), (14) and (15). This is again a consequence of the high geodesic curvature

through these corners. If higher accuracy were required for the boundary tracking, it might be beneficial to use larger weights w_{n_l} and w_{n_r}, at the price of noisier width signals. The definition of variable weights (for instance depending on the estimated curvature) is another option to be considered. For the purpose of the following calculations, the position errors obtained with the proposed choice of weights is considered acceptable.

4.4 Summary

This chapter describes a method for modelling and estimating the geometric properties of roads and race tracks from discrete noisy data sets. Models of this type are useful in a wide variety of vehicular simulation and optimal-control problems, such a minimum-lap-time studies typically associated with closed-circuit racing. The proposed method is an optimal estimation process in which a generic geometric model of the roadway is assumed, with the unknown track curvature and boundary parameters deduced from discrete measured data. Unlike standard white-box identification problems in which fixed parameters are sought, the unknown parameters in this case are curves.

The first part of the chapter shows how known ideas from differential geometry can be used to develop a generic road model, which is described in terms of three curvature functions and two track-width functions. It is then shown how the estimation of these five functions can be transformed into a numerical optimal-control problem, which can be solved by known means. The procedure is demonstrated on the illustrative problem of finding the parameters of a ribbon based on the lemniscate of Bernoulli. The chapter concludes with the more substantial problem of optimising the ribbon parameters of the Circuit de Catalunya. In this case the noisy raw data is derived from publicly available resources.

CHAPTER 5

Minimum-Lap-Time Problem

"I want to build a car that's faster than all of them ...",
Enzo Ferrari.

OPTIMAL control calculations for road cars go back to the 1950s, when simple heuristic arguments were employed to estimate lap times and optimise setup parameters (see [73]). By the early 1960s, numerical optimal-control methods were used routinely to solve a variety of aerospace flight path optimization problems (see [12]). In the context of motor racing the objective is to complete a given set of manoeuvres (laps) in the minimum possible time. This is commonly referred to as lap-time simulation, or minimum-time manoeuvring. In these studies the driver is replaced by a control system that 'drives' a car model as fast as possible within the mechanical constraints of the vehicle.

Self-evident is the fact that minimum-time control problems of this type become computationally burdensome as the dynamic model and the constraints increase in complexity. There are several overview observations one can make with regards to the possible resolution of these complexity-related maladies:

1. various approaches to the solution of the optimal-control problem might be contemplated (see Sections 2.3.3 and 2.3.4);

2. one could consider reformulating the problem and/or decomposing it into smaller more manageable sub-problems;

3. if the state space is limited to a region close to a nominal optimal path, the optimisation problem may be approximated by a linear-quadratic (LQ) problem; a problem with linear (possibly time-varying) dynamics and a quadratic performance index;

4. further progress might be facilitated by appropriately simplifying the dynamic vehicle model and/or the boundary constraints;

5. one could contemplate the use of steady-state or quasi-steady-state models — these terms will be explained in the sequel; and

6. ensure that the problem is properly scaled and that the numerical solution algorithms are implemented efficiently — in a numerical optimisation setting mathematically equivalent statements are not computationally equivalent.

To some extent all of these avenues of investigation have been considered with varying degrees of success.

The reader steeped in the classical theory of optimal control and the calculus of variations might think that the most natural approach to the solution of these problems is to use first-order necessary conditions to formulate and solve a two-point BVP (see Section 2.2.2). This is the approach taken in [19], where two motorcycle-manoeuvring problems are solved for the Luco-Poggiosecco curves on the Mugello raceway. This paper introduces an elegant curvilinear-coordinate-based description of the position and orientation of the vehicle on the track that is of general utility. An indirect method is also used in [4], which presents a numerical procedure for solving optimal-control problems related to multibody systems. A motorcycle minimum-lap-time problem for the Adria Raceway in the Veneto region of Northern Italy is solved, which compares favourably with data acquired from an on-board inertial measurement device. In later developments of this line of research an indirect method is used to examine minimum-time manoeuvres for a single-track car model through a U-turn (see [81]). Here a series of nonlinear Pacejka-type

tyre models (see [55]), for varying road surfaces, and different transmission layouts are considered.

An early example of a direct numerical method for time-optimal control being applied to road vehicles is given in [33]. This paper considers a lane-change manoeuvre for a simple nonlinear single-mass car model with yaw, lateral, and longitudinal freedoms. The tyre lateral force model is based on classical magic formulae (see [55]). The optimal control problem was solved using a gradient projection algorithm (see [37]). A key advantage of direct methods is that they do not require the adjoint equations, which become more and more difficult to derive as the complexity of the vehicle model and of the constraints increase.

Recognising the disadvantages associated with conventional indirect methods, Casanova invoked a direct-multiple-shooting technique in [18], which he deemed a sensible compromise between the sensitivity problems associated with conventional single-shooting methods and the size problems associated with collocation methods (see [8], [12] and Section 2.3.3). In conformity with multiple-shooting, the vehicle trajectory is divided into (decoupled) segments whose lengths are chosen in order to achieve good solution robustness. In addition to a coarse multiple-shooting segmentation grid, Casanova uses a finer control grid and a still finer time-marching integration grid. The control inputs are found using a linear interpolation process that is based on the initial values of the control at the start of each control interval. A disappointing feature of this approach is the long computation time required. Full-lap optimal-control calculations for the Barcelona track using a simple four-wheeled car model required between 28 and 60 hours to converge on a Sun Spark workstation (see [18]). Some related work is described in [35], which uses a vehicle model that is similar to the one used in this chapter, although the track modelling is done in a different way. The method finds simultaneously the optimal racing line and the controls required to follow this line in minimum time. A full lap of the Jerez circuit with 10 m control spacing requires approximately 8 hours of computing time on a single core 2.4 GHz processor (see [35]).

The Modelica modelling and optimisation environment (see [70]) was used in [29] to solve various minimum-lap-time problems. In order to solve the optimisation problem on closed tracks, the circuit is partitioned into a number of overlapping segments. The Modelica environment was also used in this work to optimize the car setup parameters, which are considered as constant controls. However, the vehicle optimisation is not achieved for a whole circuit, but is limited to one track segment at the time.

The apparently unusable computation times associated with high-fidelity optimal-control calculations motivated a search for faster optimisation methods. One hypothesis was that the computational time can be reduced by supplying the racing line rather than computing it as part of the lap-time optimisation. Another was the use of quasi-steady-state models, which are now widespread within the Formula 1 racing community. Both of these ideas were investigated in [10] and, not surprisingly, much shorter computation times were achieved. The term 'quasi-steady state', as it is applied here, contemplates the vehicle being modelled as traversing the race track in segments each with its own dynamic equilibrium. In order that the quasi-steady-state assumption remains valid, changes in vehicle states between adjacent segments must occur slowly. This can usually be achieved by making the segment spacing 'sufficiently small'; the transient effect of changing the state of the vehicle is then neglected in the simulation. Under this regime, a step change in steer angle, and/or throttle position between segments, results in the vehicle assuming its new state instantly. Even with fine track segmentation this is only an approximation to the car's real dynamic behaviour (see [74]). The difference between the predicted lap times using transient-optimal and quasi-steady-state analysis for the Barcelona circuit, investigated in [10], was 2.19 s, which is a significant discrepancy. That said, the close agreement of the acceleration magnitudes during cornering suggests that the vehicle is operating on, or very close to, its mechanical limits. This method may be suitable for a comparative analysis of vehicle setup and configuration changes for a pre-defined trajectory.

Another approach to minimising the burden associated with minimum-lap-time studies

is the use of LQ preview to follow a prescribed driving line at a fixed speed. The method proposed in [82] makes use of multiple linear models and a gain scheduling scheme, with its operation demonstrated on three track segments. This work provides valuable insights into the information required by drivers operating their vehicles at high speed, but is hampered by the use of linear models, a prescribed racing line and a constant-speed assumption. When car setup changes are being contemplated, one must confront the need for nonlinear dynamic models and the need to re-consider the racing line, which might change with car setup. The same is true of the vehicle's (unknown) optimal speed profile.

Ideas based on linear approximations have been further developed in [83] and [84], where (linear-time-varying) Model-Predictive Control (MPC), rather than linear preview control, is exploited. An important contribution of this work comes from the formulation of an MPC scheme as a quadratic-programming problem. These methods allow one to define the problem of finding a racing line (at constant speed in [83]) and a speed profile (both trajectory and speed are optimised in [84]) as a sequence of convex optimisation problems. The computed trajectory is sub-optimal, because linear models are used (as opposed to nonlinear ones) in conjunction with control problems that are defined over a sequence of receding track segments (as opposed to the whole track).

Recent applications of direct collocation methods to the solution of minimum-time vehicle control are reported in [62] and [66]. These papers study the control of non-holonomic bicycle models, considering problems that range from obstacle avoidance and dynamic parameter optimisation, to the manoeuvring of a two-mass-skate bicycle without the application of steering action. The vehicle motion is restricted to a plane, and the models are based on a Cartesian-coordinate description of the track and of the vehicle's position.

The aim of this chapter is to study the minimum-lap-time optimal control of a Formula 1 car with simultaneous parameter optimisation. Calculations of this type include the simultaneous optimisation of the steering angle (as a function of the distance travelled

from the start-finish line), the drive and braking of the vehicle, and one or more setup parameters, such as the location of the car's centre of mass and/or its centre of pressure; see Figure 5.1. Other optimisation parameters are considered, that include the roll balance and the differential viscosity constant. By way of a theme example, the Circuit de Catalunya will be considered for the minimisation of lap time. The optimal control calculations are carried out using a direct collocation scheme and an NLP algorithm.

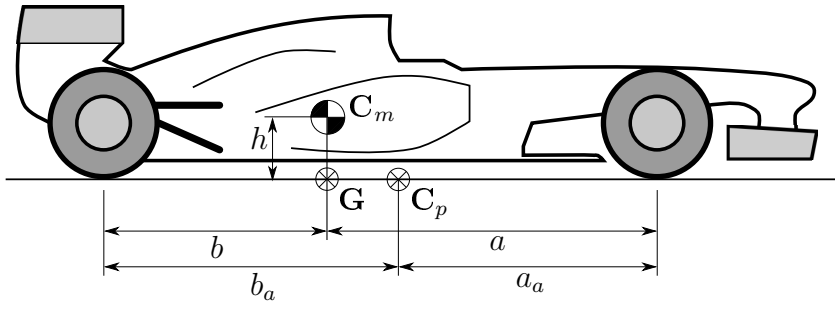

Figure 5.1: Side view of a Formula 1 car showing its mass centre $\mathbf{C}_m$, its centre of pressure $\mathbf{C}_p$ and its geometric centre $\mathbf{G}$. The other symbols refer to vehicle parameters described in Table B.2

Section 5.1 describes the model of the track and car used in this study. The track is modelled using a curvilinear-coordinate-based description of the centre line. In order to control the noise level in the track model, its geometric properties are computed by solving an auxiliary optimal control problem based on measured GPS data (see Section 4.2). The car model is described in Section 5.1.2, and has longitudinal, lateral and yaw freedoms. The model has a combined-slip magic-formula-like tyre description, longitudinal and lateral load transfer effects, four-wheel braking and a limited-slip differential. The optimal-control problem formulation is described in Section 5.2. A brief account of the scaling procedures is also given. The results are given in Section 5.3. Minimum-lap-time results for a nominal vehicle are given in Section 5.3.1. A technique for including stability constraints in the optimal control problem formulation is considered in Section 5.3.2, with a parameter optimisation study given in Section 5.3.3. The possible treatment of neglected and slow dynamics is remarked on in Section 5.3.4 with a summary of the chapter

given in Section 5.4. The contents of this chapter are published in [64].

5.1 Track and Vehicle Models

The optimal-control calculations presented in this chapter require mathematical models of both the track and the vehicle. Since the track modelling is purely kinematic it will be dealt with first.

5.1.1 Track Model

The track will be modelled using the differential geometric description proposed in Chapter 4. The curvilinear coordinate system associated with a track ribbon is used to express the vehicle position. This approach provides a compact way of describing the vehicle's progress and of constraining it to remain within the track boundaries (see [19]).

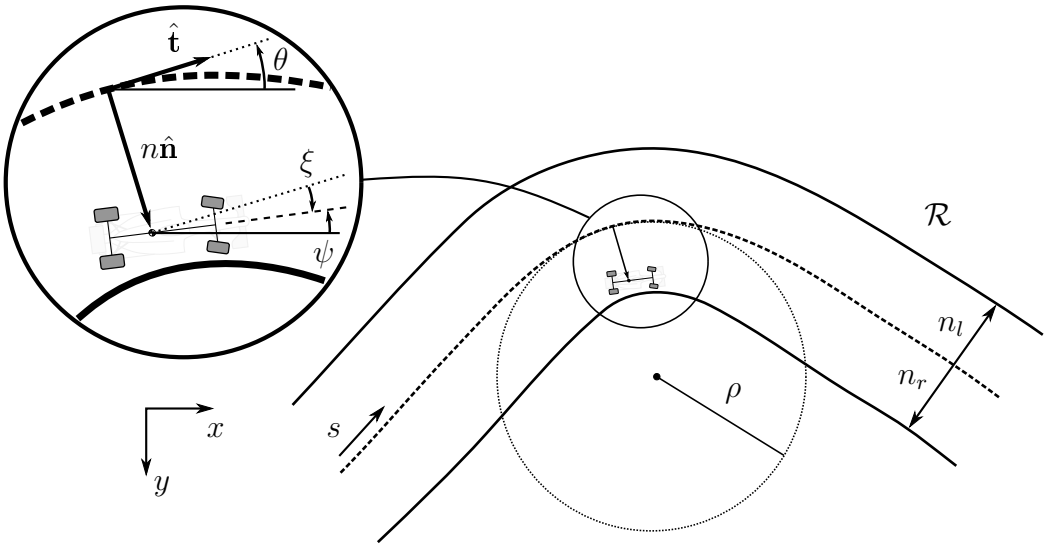

Figure 5.2: Curvilinear-coordinate-based description of a track segment $\mathcal{R}$. The independent variable s represents the elapsed centre-line distance, with ρ the radius of curvature and n_l and n_r defining the track width; x and y represent the inertial-frame axes. The arrows used for angles start from the corresponding reference direction. Angles are positive when measured clockwise.

This chapter considers horizontal flat tracks only, which correspond to ribbon parametrisations with zero relative torsion, normal curvature, and absolute pitch and roll angles. Under these assumptions gravity is always normal to the track surface. Referring to

Figure 5.2, the location of the vehicle's geometric centre is described in terms of the curvilinear abscissa $s(t)$ and the lateral offset $n(s(t))$. The first quantity is the distance travelled along the track centre line from the start-finish line, while the second gives the position of the vehicle's geometric centre in a direction (defined by the unit camber vector $\hat{\mathbf{n}}(s(t))$ in Section 4.1.3) perpendicular to the track centre-line tangent vector $\hat{\mathbf{t}}(s(t))$. Both quantities are defined as functions of time, and their values in time depend on the vehicle motion. It is understood that the travelled distance $s(t)$ is realistically a strictly increasing function of time, and that 'time' and 'distance' can be thought of as alternative independent variables. In order to simplify the notation, the track-related quantities will be described without explicit reference to an independent variable, although this could be interchangeably time or arc length. The standard dot notation will be used to signify derivatives with respect to time. At any arc length s the track's radius of curvature is given by ρ and its geodesic curvature by Ω_z. The track centre-line tangent vector $\hat{\mathbf{t}}$ will be described in terms of the track orientation angle θ, with the width given by n_l and n_r.

The absolute yaw angle of the vehicle is given by ψ and the angle between the vehicle and the track by ξ, thus $\psi = \theta + \xi$. It is clear that in this coordinate system each increment of progress ds along the track centre line produces increments dx and dy in the conventional rectangular inertial system as follows:

$$dx = ds\cos\theta \tag{5.1}$$

$$dy = ds\sin\theta. \tag{5.2}$$

This gives

$$\frac{dy}{dx} = \tan\theta. \tag{5.3}$$

It follows from elementary geometry that the track curvature is given by

$$\Omega_z = \frac{d\theta}{ds} = \frac{d}{ds}\left(\arctan\frac{dy}{dx}\right), \tag{5.4}$$

which provides a conceptual way of finding the track curvature from GPS data expressed in rectangular coordinates.

Resolving the vehicle's velocity in the $\hat{\mathbf{t}}$-direction (see Figure 5.2) gives

$$\dot{s} - n\dot{\theta} = u\cos\xi - v\sin\xi, \tag{5.5}$$

where u and v are, respectively, the longitudinal and lateral components of the vehicle velocity. Note that $\frac{d(n\hat{\mathbf{n}})}{dt} = \dot{n}\hat{\mathbf{n}} + n\dot{\hat{\mathbf{n}}} = \dot{n}\hat{\mathbf{n}} - n\hat{\mathbf{t}}\dot{\theta}$ with $\dot{n}\hat{\mathbf{n}}$ orthogonal to $\hat{\mathbf{t}}$. Identity (5.5) gives

$$\dot{s} = \frac{u\cos\xi - v\sin\xi}{1 - n\Omega_z}. \tag{5.6}$$

using (5.4). The rate of change of n is given by

$$\dot{n} = u\sin\xi + v\cos\xi. \tag{5.7}$$

Finally, differentiating $\psi = \xi + \theta$ with respect to time results in

$$\dot{\xi} = \dot{\psi} - \Omega_z\dot{s} \tag{5.8}$$

using (5.4).

In this coordinate system, constraints on the track width are easily expressed in terms of constraints on the lateral offset n. In fact one can constrain the vehicle to drive on the track by simply enforcing $n_l \leq n \leq n_r$. The real track boundary constraints apply to the positions of the vehicle's tyres with respect to the track edges. In order to simplify the problem, the boundary constraints are enforced on the position of the vehicle's geometric centre, with the track width reduced on each side by the vehicle half-width. A simplified formulation of these constraints is based on the assumption that the vehicle is approximately parallel to the constraining boundary when a constraint is active. This assumption is motivated by the following arguments:

1. track racing experience and the simulation results produced within this work show that in the Formula 1 context the vehicle side-slip angle, defined as $\beta = \tan^{-1}(-v/u)$, is typically small (usually not larger than 10° in absolute value);

2. since the vehicle is bound to remain inside the track, the trajectory of the wheel touching the track boundary must be tangent to the boundary itself at any point of contact;

3. the geometry of the track boundaries includes measurement errors which are comparable to the maximum expected lateral displacement of each wheel from the vehicle half width; an upper bound for this quantity can be estimated as $\max(a, b)\sin(\beta_{max}) \approx 0.3\,\mathrm{m}$, where $\beta_{max} \approx 10°$ is the absolute value of the maximum expected vehicle side-slip angle.

If any of these conditions is not valid, the track boundary constraints might need to be reformulated in order to account for the position of each wheel. For example, assumption 1 is usually invalid for off-road rally driving, where due to particular tyre friction characteristics the vehicle side-slip angle can be large.

5.1.1.1 Change of Independent Variable

Since 'time' and the 'distance travelled' are closely related independent variables, one would naturally assume that either could be dispensed with in the formulation of optimal control problems. Using the 'distance travelled' as the independent variable has the advantage of maintaining an explicit connection with the track position, as well as reducing (by one state) the number of problem variables. In order to make this change of variables one has to ensure that there is a one-to-one correspondence between s and t. Since $ds = \dot{s}dt$, one need only ensure that the vehicle velocity in the track-tangent direction is positive; this will be assumed from now on. Next, it is observed

$$dt = \frac{dt}{ds}ds = s_t(s)ds,$$

where s_t comes from (5.6) as follows

$$s_t = \left(\frac{ds}{dt}\right)^{-1} = \frac{1 - n\Omega_z}{u\cos\xi - v\sin\xi}. \tag{5.9}$$

The quantity s_t is the reciprocal of the component of the vehicle velocity in the track-tangent direction, which defines a bijective map between s and t when the numerator and denominator are both positive. Selecting the 'distance travelled' as the independent variable, one obtains

$$\frac{dn}{ds} = s_t\left(u\sin\xi + v\cos\xi\right) \tag{5.10}$$

from (5.7), and

$$\frac{d\xi}{ds} = s_t\omega - \Omega_z \tag{5.11}$$

from (5.8); $\omega = \dot{\psi}$ is the vehicle yaw rate. Expressing the vehicle control and states in terms of the distance travelled allows one to assess the performance anywhere on the track.

5.1.2 Dynamic Model

The car model used here is standard, and is based on a rigid-body representation of a chassis with longitudinal, lateral and yaw freedoms. The important geometric modelling quantities are shown in plan view in Figure 5.3.

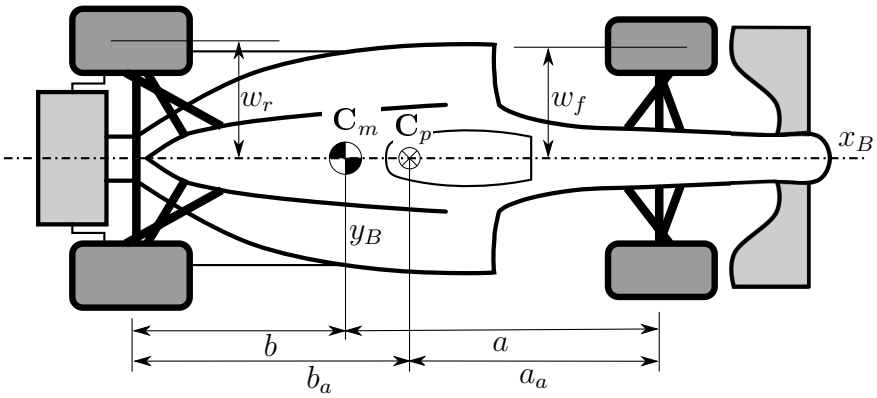

Figure 5.3: Plan view of a Formula 1 car with its basic geometric parameters. The body-fixed axes x_B and y_B are in the road plane.

Each tyre produces longitudinal and lateral forces in response to slip quantities (see Appendix A). These forces together with the steer (δ) and yaw (ψ) angle definitions are given in Figure 5.4; the inertial reference frame is shown as x and y.

Balancing forces in the longitudinal and lateral directions, while also balancing the yaw moments, gives

$$\begin{aligned}
m\tfrac{d}{dt}u(t) &= m\omega v + F_x \\
m\tfrac{d}{dt}v(t) &= -m\omega u + F_y \\
I_z\tfrac{d}{dt}\omega(t) &= a\left(\cos\delta(F_{fry} + F_{fly}) + \sin\delta(F_{frx} + F_{flx})\right) + \\
&\quad w_f\left(\sin\delta F_{fry} - \cos\delta F_{frx}\right) - w_r F_{rrx} + \\
&\quad w_f\left(\cos\delta F_{flx} - \sin\delta F_{fly}\right) + w_r F_{rlx} - b\left(F_{rry} + F_{rly}\right),
\end{aligned} \tag{5.12}$$

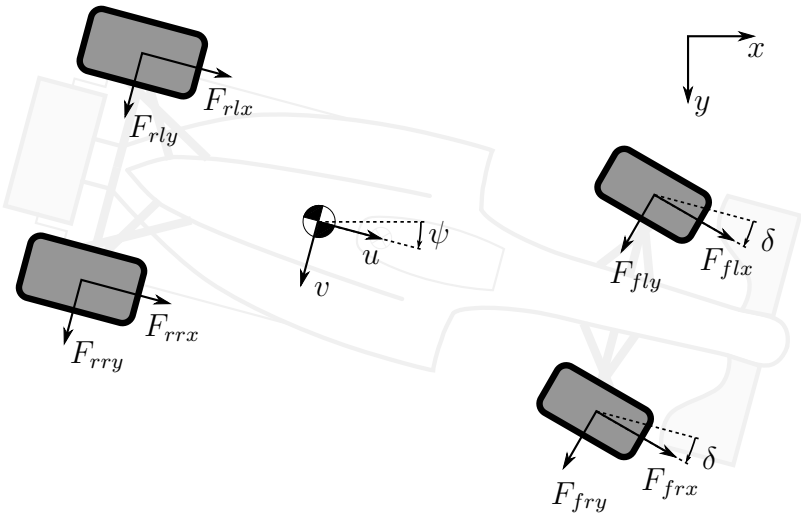

Figure 5.4: Tyre force system.

in which F_x and F_y are, respectively, the total longitudinal and lateral forces acting on the car. These forces are given by

$$F_x = \cos\delta(F_{frx} + F_{flx}) - \sin\delta(F_{fry} + F_{fly}) + (F_{rrx} + F_{rlx}) + F_{ax} \tag{5.13}$$

$$F_y = \cos\delta(F_{fry} + F_{fly}) + \sin\delta(F_{frx} + F_{flx}) + (F_{rry} + F_{rly}), \tag{5.14}$$

in which F_{ax} is the aerodynamic drag force, defined later in Section 5.1.4. The dynamic equations (5.12) can be expressed in terms of the independent variable s as follows

$$\frac{du}{ds} = s_t(s)\dot{u}, \qquad \frac{dv}{ds} = s_t(s)\dot{v}, \qquad \frac{d\omega}{ds} = s_t(s)\dot{\omega}. \tag{5.15}$$

5.1.3 Tyre Forces

The tyre forces have normal, longitudinal and lateral components that act on the vehicle's chassis at the tyre-ground contact points and react on the inertial frame. The rear-wheel tyre forces are expressed in the vehicle's body-fixed reference frame, while the front tyre forces are expressed in a steered reference frame; refer again to Figure 5.4. In each case these forces are functions of the normal loads and the tyre's longitudinal and lateral slips.

5.1.3.1 Load Transfer

The time-varying tyre loads normal to the ground plane are computed balancing the forces acting on the car in the z-direction and balancing moments around the body-fixed x_B-

and y_B-axes (see Figure 5.3). These calculations must recognise the gravitational, inertial and aerodynamic forces acting on the car. The vertical force balance gives

$$0 = F_{rrz} + F_{rlz} + F_{frz} + F_{flz} + mg + F_{az}, \tag{5.16}$$

in which the $F_{..z}$'s are the vertical tyre forces for each of the four wheels, g is the acceleration due to gravity and F_{az} is the aerodynamic downforce acting on the car, defined later in Section 5.1.4. Balancing moments around the car's body-fixed x_B-axis gives

$$0 = w_r(F_{rlz} - F_{rrz}) + w_f(F_{flz} - F_{frz}) + hF_y, \tag{5.17}$$

in which F_y is the lateral inertial force acting on the car's mass centre; see (5.14). Balancing moments around the car's body-fixed y_B-axis gives

$$0 = b(F_{rrz} + F_{rlz}) - a(F_{frz} + F_{flz}) + hF_x + (a_a - a)F_{az}, \tag{5.18}$$

where F_x is the total longitudinal force acting on the car (see (5.13)).

Equations (5.16), (5.17) and (5.18) are a set of linear equations in four unknowns. A unique solution for the tyre loads can be obtained by adding a suspension-related roll-balance relationship, in which the lateral load difference across the front axle is some fraction of the total lateral load difference

$$F_{frz} - F_{flz} = d_r(F_{frz} + F_{rrz} - F_{flz} - F_{rlz}), \tag{5.19}$$

where $d_r \in [0, 1]$. If this fourth equation is recognised, and equations (5.16), (5.17), (5.18) and (5.19) are linearly independent, they have a unique solution for the tyre normal load distribution.

5.1.3.2 Non-Positive Tyre Loads

The forces satisfying equations (5.16), (5.17), (5.18) and (5.19) are potentially both positive and negative. If these forces are negative, the corresponding wheels are subject to vertical reaction forces. On the other hand, positive forces represent non-physical 'forces of attraction', which cannot be accommodated in a realistic car model. Since the model

being used here has no pitch, roll or heave freedoms, none of the wheels is free to leave the road, while simultaneously keeping faith with (5.16) to (5.19). In order to cater for the possible 'positive-force' situation within a nonlinear programming environment, a vector of 'fictitious' forces is introduced

$$\bar{\mathbf{F}}_z = \begin{bmatrix} \bar{F}_{frz} \\ \bar{F}_{flz} \\ \bar{F}_{rrz} \\ \bar{F}_{rlz} \end{bmatrix}, \tag{5.20}$$

and the vector of non-positive loads is defined

$$\mathbf{F}_z = -u_{-2}(-\bar{\mathbf{F}}_z); \tag{5.21}$$

the ramp function

$$u_{-2}(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{5.22}$$

is interpreted element-wise. It is clear that $\bar{\mathbf{F}}_z$ and $\mathbf{F}_z$ will be equal unless at least one entry of $\bar{\mathbf{F}}_z$ is positive (i.e. non-physical). It is now argued that the model must respect the laws of mechanics at all times and so equations (5.16), (5.17) and (5.18) must be enforced. In contrast, it is assumed that the solution to (5.19), which is only an approximate representation of the suspension system, can be 'relaxed' in the event of a wheel load sign reversal.

Equations (5.16), (5.17) and (5.18) can be assembled in matrix form as

$$A_1 \mathbf{F}_z = \mathbf{q}, \tag{5.23}$$

while (5.19) can be written as

$$A_2 \mathbf{F}_z = 0. \tag{5.24}$$

The entries in A_1 and A_2, and the vector $\mathbf{q}$ are assembled from (5.16), (5.17), (5.18) and (5.19).

In order to deal with the 'light-wheel' situation, equations (5.23) and (5.24) are combined as

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} \mathbf{F}_z \\ \bar{\mathbf{F}}_z \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ 0 \end{bmatrix}; \tag{5.25}$$

with $\mathbf{F}_z$ replaced with $\bar{\mathbf{F}}_z$ in the second block row. In the situation where all the wheels are normally loaded, $\bar{\mathbf{F}}_z = \mathbf{F}_z$ and (5.25) reduces to (5.16) to (5.19). On the other hand, if there is a 'light wheel', the mechanics equations (5.16), (5.17) and (5.18) will be satisfied by the non-positive forces $\mathbf{F}_z$, while the roll-balance equation is satisfied by the fictitious forces $\bar{\mathbf{F}}_z$ that contain a force of attraction. The non-positive forces $\mathbf{F}_z$ are used to represent the normal tyre loads. It is clear that the four components of $\bar{\mathbf{F}}_z$ have to satisfy the nonlinear relationship (5.25), which will be solved by a NLP algorithm.

There is a circular dependency between tyre loads and friction forces inherent in the tyre description. This circularity is treated as a nonlinear set of algebraic equations that is solved as part of the optimising NLP; the tyre friction model is given in Appendix A.

5.1.4 Aerodynamics

The external forces acting on the car come from the tyres and from aerodynamic influences. The aerodynamic model used here is limited to speed-dependent downforce and drag influences acting in the car's plane of symmetry; aerodynamic side forces, yawing moments and pitching effects have been neglected. As is standard in models of this type, the aerodynamic force is assumed to be quadratic in the vehicle speed and is applied at the centre of pressure, which is located in the vehicle's plane of symmetry (see Figure 5.1). The downforce and drag are modelled as

$$F_{az} = 0.5\, c_l\, \rho_a\, a_f\, u^2 \tag{5.26}$$

and

$$F_{ax} = -0.5\, c_d\, \rho_a\, a_f\, u^2 \tag{5.27}$$

respectively, which act in the car's body-fixed z- and x-axis directions respectively. The constants c_l and c_d are the aerodynamic downforce and drag coefficients, ρ_a is the air density and a_f the frontal area of the vehicle.

5.1.5 Wheel Torque Distribution

In order to optimise the vehicle's performance, one needs to control the torques applied to the individual road wheels. In some applications, including current F1 racing, the braking system has to be designed so that equal pressure is applied to the brake callipers of each axle. The drive torques applied to the rear wheels are controlled by a differential mechanism.

5.1.5.1 Brakes

Equal brake calliper pressures is equated with equal braking torques when neither wheel on a particular axle is locked. If one wheel is locked, the braking torque applied to the locked wheel may be lower than that applied to the rolling wheel. In the case of the front wheels this constraint is modelled as

$$0 = u_{-2}(-\omega_{fr})u_{-2}(-\omega_{fl})(F_{frx} - F_{flx}), \tag{5.28}$$

in which ω_{fr} and ω_{fl} are the angular velocities of the front-right and front-left wheels, respectively (negative values correspond to the forward motion of the car). If either road wheel 'locks up', the corresponding angular velocity will be non-positive and the braking torque constraint (5.28) becomes inactive.

5.1.5.2 Differential

It is assumed that the drive torque is delivered to the rear wheels through a limited-slip differential that is modelled as

$$r(F_{rlx} - F_{rrx}) = -k_d(\omega_{rl} - \omega_{rr}), \tag{5.29}$$

in which ω_{rl} and ω_{rr} are the rear-wheel angular velocities, r is the wheel radius and k_d is a torsional damping coefficient. The special cases of an open- and a locked-differential correspond to $k_d = 0$ and k_d arbitrarily large respectively.

5.2 Numerical Optimal Control

The minimum-lap-time optimal-control problem includes nonlinear and non-smooth dynamics, as well as state and control constraints. Care is required when formulating these problems in order that a reliable solution is obtained in a reasonable time. Key steps include: selecting the optimisation strategy to be used; selecting a simple vehicle model that captures all the important problem features; finding an efficient and numerically stable track centre-line description; scaling the problem correctly and developing techniques for 'smoothing out' non-smooth features of the problem, so that the gradients required by the optimisation algorithm are properly defined.

5.2.1 Problem Formulation

The optimal-control problem formulation used here involves minimising a Lagrange cost over a finite horizon, subject to position-dependent DAEs with equality and inequality constraints on the problem parameters, state and control. An optimal-control problem general enough for the present purposes is

$$\min_{\{\boldsymbol{x}(\cdot),\boldsymbol{u}(\cdot),\boldsymbol{p}\}} \int_{s_0}^{s_f} L\big(s,\boldsymbol{x}(s),\boldsymbol{u}(s),\boldsymbol{p}\big)ds, \tag{5.30}$$

subject to the constraints

$$\begin{cases} \dfrac{d\boldsymbol{x}}{ds} &= \boldsymbol{f}\big(s,\boldsymbol{x}(s),\boldsymbol{u}(s),\boldsymbol{p}\big) \\ 0 &= \mathbf{g}\big(s,\boldsymbol{x}(s),\boldsymbol{u}(s),\boldsymbol{p}\big) \\ 0 &\geq \mathbf{h}\big(s,\boldsymbol{x}(s),\boldsymbol{u}(s),\boldsymbol{p}\big) \\ 0 &= \mathbf{g}_b\big(\boldsymbol{x}(s_0),\boldsymbol{x}(s_f),\boldsymbol{p}\big), \end{cases} \tag{5.31}$$

where s is the distance travelled along the track centre line; in this case s_0 and s_f are on the start-finish line. The vector $\boldsymbol{p} \in \mathbb{R}^{\underline{p}}$ contains the vehicle parameters to be optimised, and $\boldsymbol{x}(s) \in \mathbb{R}^{\underline{x}}$ and $\boldsymbol{u}(s) \in \mathbb{R}^{\underline{u}}$ are the vehicle state and control vectors respectively. The vector function $\boldsymbol{f}$ describes the vehicle dynamics. The vector functions $\mathbf{g}$ and $\mathbf{h}$ define the equality and inequality constraints for the system (e.g. the track topology and box constraints on the state, control and parameters); the subscript b refers to boundary constraints. The scalar function L is the stage cost.

In the case of the work presented here, $\boldsymbol{f}$ contains equations (5.10), (5.11) and (5.15), with the associated states given by n, ξ, u, v and ω. The normal tyre loads appear in $\mathbf{g}$ and are constrained by (5.21) and (5.25), with the tyre longitudinal slips constrained by conditions of the form (5.28), through the definition of the friction forces given in Appendix A. The rear tyre slips are constrained by the differential according to (5.29). The track width constraint $n_l \leq n \leq n_r$ appears in $\mathbf{h}$, with the constraint on the maximum engine power approximated by $(F_{rrx} + F_{rlx})u \leq P_{max}$. The trajectory closure condition $\boldsymbol{x}(s_f) - \boldsymbol{x}(s_0) = 0$ is given in $\mathbf{g}_b$. The Lagrange cost integrand is $L = s_t$.

5.2.2 Transcription

As mentioned in Section 2.3, a range of numerical procedures for the solution of optimal-control problems can be called upon. Indirect strategies convert the original problem into a BVP that is then solved to determine candidate optimal trajectories. Each of these computed extremals is then examined to verify whether it is a local minimum. Direct transcription methods make use of a discretisation scheme to 'transcribe' the original problem into a discrete problem with a finite set of variables to be determined by an NLP algorithm. A second classification is based on the finite-dimensionalisation technique, which recognises single-shooting, multiple-shooting and collocation methods (see Section 2.3.3).

There are a number of issues in favour of using a direct collocation method that will be summarised briefly. Firstly, direct methods do not require one to find the necessary conditions for optimality. Secondly, direct methods are relatively easy to implement and a wide variety of 'ready-to-use' software is available. Thirdly, parameter optimisation is easily accommodated by simply adding new decision variables to the transcribed optimisation problem. In the work presented in this chapter the MATLAB-based optimal control transcription toolbox ICLOCS (see [22]) is used with the trapezoidal rule to convert the OC problem into an NLP problem. The ICLOCS toolbox also constructs data structures containing the derivative information required by the NLP algorithm IPOPT

(see [87]). The required derivatives include first- and second-order partial derivatives of the stage cost, the path constraints, the system dynamics and the boundary constraints with respect to the decision variables, that include the control, the state and the parameters. The interested reader can consult the ICLOCS manual [22] (downloadable), or [8] for a full account of the derivatives required in transcription methods.

In order to negate the influences of the truncation errors associated with computing derivatives through finite-differencing, analytical derivatives were used, which enhance the convergence of the NLP solution. Since analytical expressions are available for the stage cost, the path constraints, the system dynamics and the boundary constraints, they can be differentiated symbolically using a symbolic mathematical computation engine such as Maple. These symbolic expressions were automatically coded in C and compiled into executable functions that are used to generate numerical values of the derivatives for any choice of the decision variables. A level of parallelisation was obtained for the function and derivative calculations. The sampling points were considered in sub-sets of equal size, with the function and derivative computations assigned to an independent thread for each sub-set of the samples.

The transcription of the OC problem given in (5.30) and (5.31) into an NLP problem was achieved using the direct collocation methodology described in Section 2.3.3.3 and in [8], based on the trapezoidal basic rule. To begin, the control horizon is discretised in terms of arc length increments, while the continuous state and control functions are quantised into a finite set of decision variables. First, let us introduce a strictly increasing finite sequence of arc length samples[1] $\boldsymbol{s} := [s_0\, s_1\, \ldots\, s_N]^T \in \mathbb{R}^{N+1}$, with $s_N := s_f$ and all the entries in strictly increasing order. The sequence $\boldsymbol{s}$ defines the collocation points for the trapezoidal approximation, where N is the number of sampling intervals. The arc length difference sequence is then defined as $\Delta\boldsymbol{s} := [s_1 - s_0\ s_2 - s_1\ \ldots\ s_N - s_{N-1}]^T \in \mathbb{R}^N$.

[1] Here the vectors $\boldsymbol{s}$ and $\Delta\boldsymbol{s}$ correspond to the vectors $\boldsymbol{t}$ and $\Delta\boldsymbol{t}$ defined in Section 2.3.3.3, respectively.

If

$$T_1 := \begin{bmatrix} -1 & +1 & & \\ & \ddots & \ddots & \\ & & -1 & +1 \end{bmatrix}_{N\times(N+1)}, \tag{5.32}$$

then

$$\Delta \boldsymbol{s} = T_1 \boldsymbol{s}. \tag{5.33}$$

Next, two parameter vectors are defined that represent the values of the state and the control at the collocation points, $\bar{\boldsymbol{u}} := [\boldsymbol{u}_0^T \ \dots \ \boldsymbol{u}_N^T]^T \in \mathbb{R}^{(N+1)\underline{u}}$ and $\bar{\boldsymbol{x}} := [\boldsymbol{x}_0^T \ \dots \ \boldsymbol{x}_N^T]^T \in \mathbb{R}^{(N+1)\underline{x}}$. Sampled versions of L, $\boldsymbol{f}$, $\mathbf{g}$, $\mathbf{h}$ and $\mathbf{g}_b$ are required too, which will replace their continuous counterparts in (5.31). These are defined in vector form as

$$\bar{L}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) := \begin{bmatrix} L(s_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{p}) \\ \vdots \\ L(s_N, \boldsymbol{x}_N, \boldsymbol{u}_N, \boldsymbol{p}) \end{bmatrix} \in \mathbb{R}^{N+1}, \tag{5.34}$$

$$\bar{\boldsymbol{f}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) := \begin{bmatrix} \boldsymbol{f}(s_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{p}) \\ \vdots \\ \boldsymbol{f}(s_N, \boldsymbol{x}_N, \boldsymbol{u}_N, \boldsymbol{p}) \end{bmatrix} \in \mathbb{R}^{(N+1)\underline{f}}, \tag{5.35}$$

$$\bar{\mathbf{g}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) := \begin{bmatrix} \mathbf{g}(s_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{p}) \\ \vdots \\ \mathbf{g}(s_N, \boldsymbol{x}_N, \boldsymbol{u}_N, \boldsymbol{p}) \end{bmatrix} \in \mathbb{R}^{(N+1)\underline{\mathrm{g}}}, \tag{5.36}$$

$$\bar{\mathbf{h}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) := \begin{bmatrix} \mathbf{h}(s_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{p}) \\ \vdots \\ \mathbf{h}(s_N, \boldsymbol{x}_N, \boldsymbol{u}_N, \boldsymbol{p}) \end{bmatrix} \in \mathbb{R}^{(N+1)\underline{\mathrm{h}}}. \tag{5.37}$$

The integral cost in (5.30) is approximated using the trapezoidal rule as follows

$$\begin{aligned} &\int_{s_0}^{s_f} L\big(s, \boldsymbol{x}(s), \boldsymbol{u}(s), \boldsymbol{p}\big) ds \\ &\quad \simeq \sum_{i=1}^{N} (s_i - s_{i-1}) \, \frac{L(s_i, \boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{p}) + L(s_{i-1}, \boldsymbol{x}_{i-1}, \boldsymbol{u}_{i-1}, \boldsymbol{p})}{2} \\ &\quad = \Delta \boldsymbol{s}^T \, T_2 \, \bar{L}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}), \end{aligned} \tag{5.38}$$

with

$$T_2 = \begin{bmatrix} 1/2 & 1/2 & & \\ & \ddots & \ddots & \\ & & 1/2 & 1/2 \end{bmatrix}_{N\times(N+1)}. \tag{5.39}$$

The constraints associated with the system dynamics are transcribed using a trapezoidal-rule approximation of the differential operator $\frac{d}{ds}(\cdot)$:

$$(T_1 \otimes I_{\underline{x}\times\underline{x}})\bar{\boldsymbol{x}} = \big(\mathrm{diag}(\Delta\boldsymbol{s})T_2 \otimes I_{\underline{x}\times\underline{x}}\big)\bar{\boldsymbol{f}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}), \tag{5.40}$$

in which $\otimes$ is the Kronecker product (see [34]). After transcription, the optimal-control problem given in (5.30) and (5.31) is re-cast as the NLP problem

$$\begin{aligned} &\min_{\{\bar{\boldsymbol{x}},\bar{\boldsymbol{u}},\boldsymbol{p}\}} \ \Delta\boldsymbol{s}^T T_2 \bar{L}\left(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}\right) \\ &\text{s.t.} \ \begin{cases} 0 &= (T_1 \otimes I_{\underline{x}\times\underline{x}})\bar{\boldsymbol{x}} - \big(\mathrm{diag}(\Delta\boldsymbol{s})T_2 \otimes I_{\underline{x}\times\underline{x}}\big)\bar{\boldsymbol{f}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) \\ 0 &= \bar{\mathbf{g}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) \\ 0 &\geq \bar{\mathbf{h}}(\boldsymbol{s}, \bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \boldsymbol{p}) \\ 0 &= \mathbf{g}_b\left(\boldsymbol{x}_0, \boldsymbol{x}_N, \boldsymbol{p}\right) \end{cases}, \end{aligned} \tag{5.41}$$

in which the decision variables appear in $\bar{\boldsymbol{x}}$, $\bar{\boldsymbol{u}}$ and $\boldsymbol{p}$.

5.2.3 State and Control Scaling

Scaling can have a significant influence on the performance of optimisation algorithms. Since convergence tolerances and other criteria are necessarily based on some notion of 'small' and 'large' quantities, problems with unusual or unbalanced scaling may cause difficulty. One notion of scaling is to transform the variables from their original representation, which may reflect the physical nature of the problem, into variables that have desirable properties in terms of optimisation.

5.2.3.1 Non-Dimensionalisation

In problems that involve physical quantities poor scaling can be a direct consequence of the choice of the problem units. Using SI units, the vehicle has a mass of approximately 600 kg and the acceleration due to gravity is approximately $9.8\,\mathrm{ms}^{-2}$. When the effect of aerodynamic downforce is included, the normal tyre loads and the tyre friction forces can be of the order of 10^4 N in some cases. The forces acting on the vehicle are thus 4 orders of magnitude larger than length-related decision variables, angles (e.g. steering) and/or tyre slip coefficients. For these reasons the model is non-dimensionalised by normalising the basic physical units of length, mass and time. All the mass- and length-related quantities

are scaled so that the vehicle has unit mass and unit length. Time is scaled so that the acceleration due to gravity is unity too; after non-dimensionalisation angles are still expressed in radians.

5.2.3.2 Affine Transformations

A second layer of scaling can be introduced so that the decision variables of the transcribed optimal-control problem lie approximately in the range $[-1, +1]$. When introducing scaling of this kind one normally restricts oneself to a linear transformation of variables, although nonlinear ones can be used. The simplest linear transformation is direct scaling, but it is sometimes convenient to remove offsets in the decision variables by using affine transformations of the form $\boldsymbol{x} = D\tilde{\boldsymbol{x}} + \boldsymbol{d}$ in which $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ are the new and old variables, and D is a diagonal matrix of scale factors with $\boldsymbol{d}$ a vector of offsets.

For example, affine transformations are applied to the vehicle parameters, using the nominal values as their offsets.

5.2.4 Smoothing Functions

Interior-point NLP algorithms such as IPOPT need first- and second-order derivative information for the functions defining the cost and constraints. For this reason non-smooth problem features have to be approximated in a way that does not change significantly the problem's solution. The ramp function $u_{-2}(x)$ has undefined derivatives at $x = 0$, and is therefore approximated using

$$u_{-2}(x) \approx \frac{x + \sqrt{x^2 + \epsilon}}{2}, \tag{5.42}$$

in which ϵ is a 'small' constant. Since

$$\frac{\partial}{\partial x}\left(\frac{1}{2}\left(x + \sqrt{x^2 + \epsilon}\right)\right) = \frac{1}{2}\left(\frac{x}{\sqrt{x^2 + \epsilon}} + 1\right), \tag{5.43}$$

one sees that the derivative approximation for $u_{-2}(x)$ at $x = 0$ is well defined as $1/2$. These approximations are more accurate for smaller values of ϵ, with values in the range $10^{-4} \leq \epsilon \leq 10^{-2}$ used for the computation of the results given in this thesis.

5.3 Results

This section presents results relating to minimum-lap-time and parameter-optimisation studies. The track and vehicle models described in Section 5.1 are limited to 2-dimensional problems. For this reason the following calculations will consider a 'flat' version of the Circuit de Catalunya, for which the differential parametrisation is found projecting all the boundary data points (as defined in Section 4.2) onto the xy-plane, simply setting their z-coordinate to 0.

The circuit plan view is in Figure 5.5, where the corners are numbered starting from the start-finish line; the corner numbering will be used throughout the remainder of this thesis following the notation in Figure 5.5. Figure 5.6 gives the geodesic curvature identified for

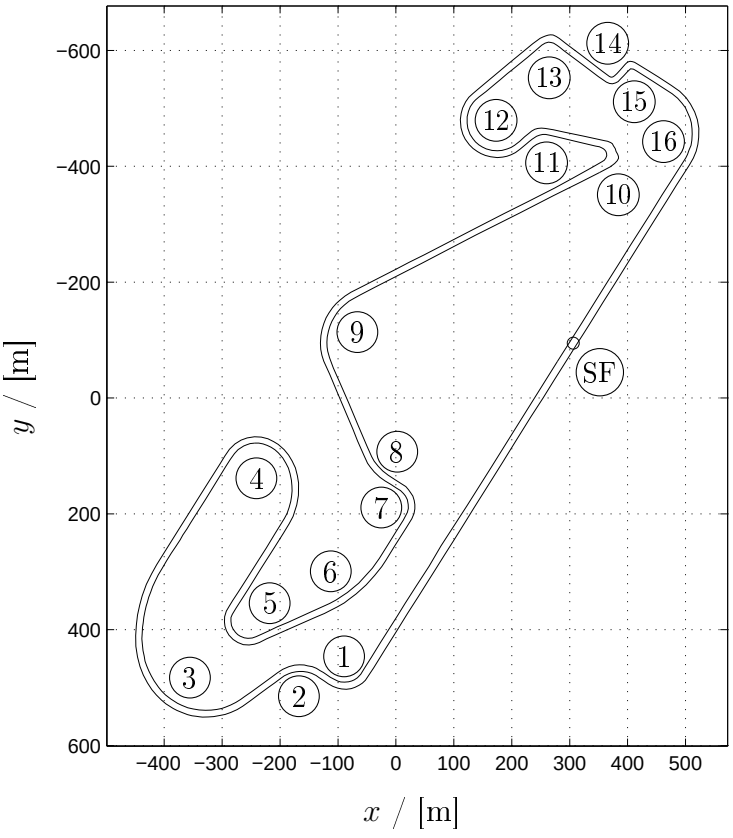

Figure 5.5: Plan view of the Circuit de Catalunya (current Spanish Formula 1 Grand Prix). The start-finish line is labelled 'SF'; the corners are numbered from start to finish (this is a clockwise circuit).

Corner	Distance	Corner	Distance
1	730 m	9	2750 m
2	800 m	10	3350 m
3	1050 m	11	3500 m
4	1650 m	12	3650 m
5	2000 m	13	3900 m
6	2250 m	14	4000 m
7	2420 m	15	4050 m
8	2500 m	16	4200 m

Table 5.1: Approximate distances to mid-corner on the Circuit de Catalunya (from the start-finish line).

the flat track. Approximate distances from the start-finish line to each of the corners are

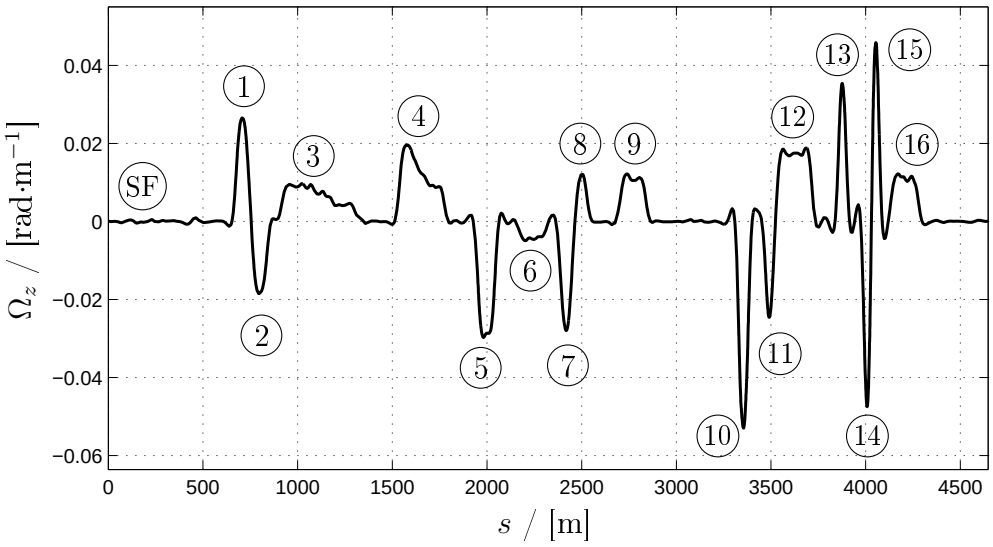

Figure 5.6: Geodesic curvature for the plane projection of the Circuit de Catalunya. The labels are consistent with Figure 5.5.

given in Table 5.1.

After presenting the minimum-lap-time results for a nominal vehicle, some preliminaries relating to possible dynamic instabilities will be dealt with. The optimisation of the vehicle parameters will follow. The vehicle's geometric, mechanical, tyre and aerodynamic data are given in Appendix B.

5.3.1 Optimal Lap

In this section the behaviour of the optimally-controlled car is studied using the nominal parameter set given in Appendix B. The speed profile for a computed minimum-time lap of the Barcelona track is given in Figure 5.7. The computed lap time is 80.643 s, which was obtained using a fixed uniform grid, with a 1 m sampling distance, that will be used for all the minimum-lap-time calculations analysed in this thesis. The NLP problem associated with the optimal-control transcription was initialised setting all the state and control variables to 0, with the exception of the longitudinal speed, and the vector of fictitious tyre loads; these were respectively initialised to $u = 8\,\mathrm{m/s}$ and $\bar{F}_{..z} = -1.5\,\mathrm{kN}$, all constant throughout the lap. This trajectory is actually infeasible in respect of the vehicle dynamics and algebraic constraints. However, the NLP algorithm did not incur in any convergence-related issue, reaching a solution that could be qualified as a local minimum. This behaviour is recognised as an improvement from prior minimum-lap-time solution methods, which were highly reliant on the quality of the initial trajectory guesses (see [18, 29]). Nonlinear problems of this kind can be in general characterised by multiple local minima. Repeated solutions of the transcribed problem with alternative initial guesses produced consistent results, which backed the hypothesis that a global minimum had been achieved.

The optimal racing line for four track sections is shown in Figure 5.8. In each case the cornering manoeuvre starts with a ◯ and ends with a ☆. As one would expect, the optimal racing line seeks to 'smooth out' the corners by maximising the radius of curvature of the trajectory throughout the manoeuvre.

Figure 5.9 focuses on the tyre force system behaviour in turn (1) at the six points highlighted in Figure 5.8. At point ◯ the front tyres are essentially free-rolling, while they are both under equal normal loads of just over 6 kN. At this point the rear tyres are supporting equal longitudinal drive forces of just under 3 kN and normal loads of $\approx 7.5\,\mathrm{kN}$. During the braking phase, between △ and □, large longitudinal braking forces are produced by all four tyres with a simultaneous reduction in the normal loads resulting

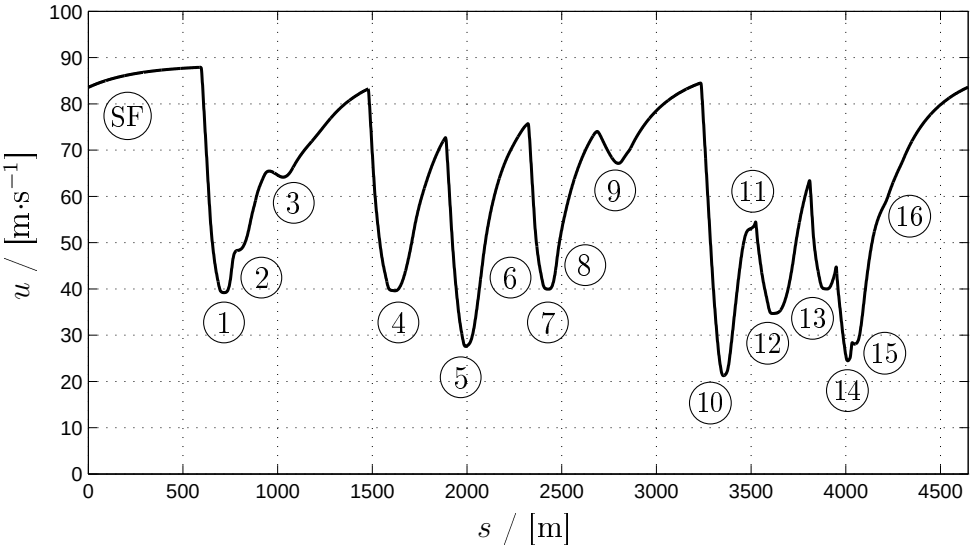

Figure 5.7: Car speed as a function of the distance travelled from the start-finish line on the Barcelona Formula 1 track. The track model is the one given in Figures 5.5 and 5.6. The corner numbers correspond to those given in the same figures.

from reduced aerodynamic downforce. There is also a load transfer from the rear to the front axle during braking. During the mid-corner at ◊ the tyres are experiencing almost no longitudinal force, but large lateral forces on the left-hand (outer) tyres, that are also experiencing high normal loads due to a lateral load transfer. Once the car starts to exit the corner between ▽ and ☆ the lateral tyre forces reduce, the drive forces on the rear tyres increase, and the lateral load transfer is reversed in preparation for the following left-hand turn (2).

Figure 5.10 shows the effect of the mesh size variations on the minimum lap time and the NLP solution time. The important message from this figure is that the predicted optimal lap time approaches an asymptotic value of 80.65 s. A subsidiary point is that the solution time increases significantly for finer meshes, but without any apparent benefit in terms of prediction accuracy. Once the mesh length falls below 0.5 m, solution times of the order of hours can be expected. This figure also shows that for the considered computation meshes the solution time grows approximately according to $t_{NLP} \approx 2 \cdot 10^{-3} N^{1.65}$, in which N is the number of mesh points. This provides evidence that the NLP software is exploiting the sparsity of the derivative matrices (see Sections 2.3.1 and 2.3.4 for details);

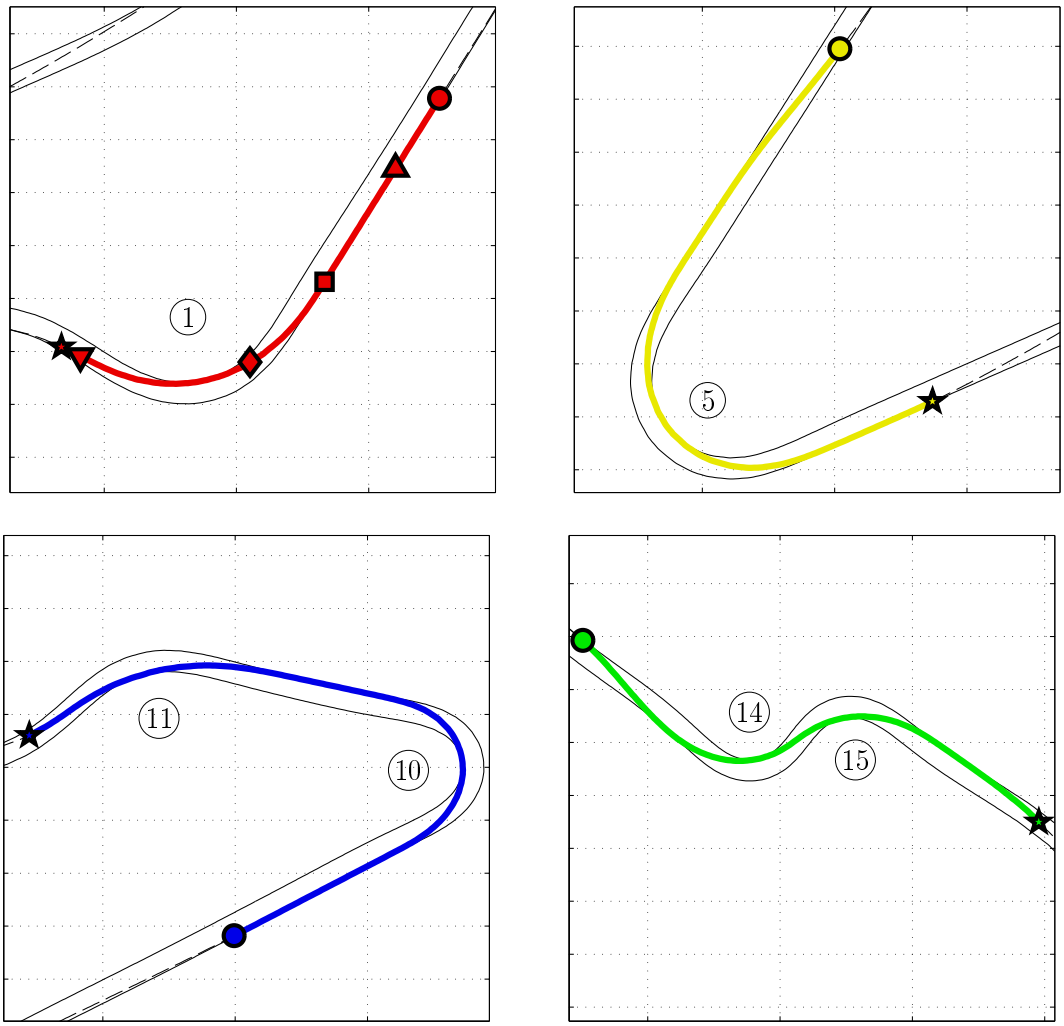

Figure 5.8: Optimal trajectory of the car's geometric centre through turns (1), (5), (10) - (11) and (14) - (15). The markers associated with different points in the corners are as follows: The start of the manoeuvre is marked with a ◯; the start of the braking phase is marked with a △; the start of the cornering phase is marked with a □; the ◊ marks the beginning of the corner-exit acceleration phase; the ▽ symbol marks a reversal in the direction of the tyre side forces, while the ☆ is the end of the manoeuvre. The marker points defined for turn (1) correspond to specific transition points in the tyre forces shown in Figure 5.9.

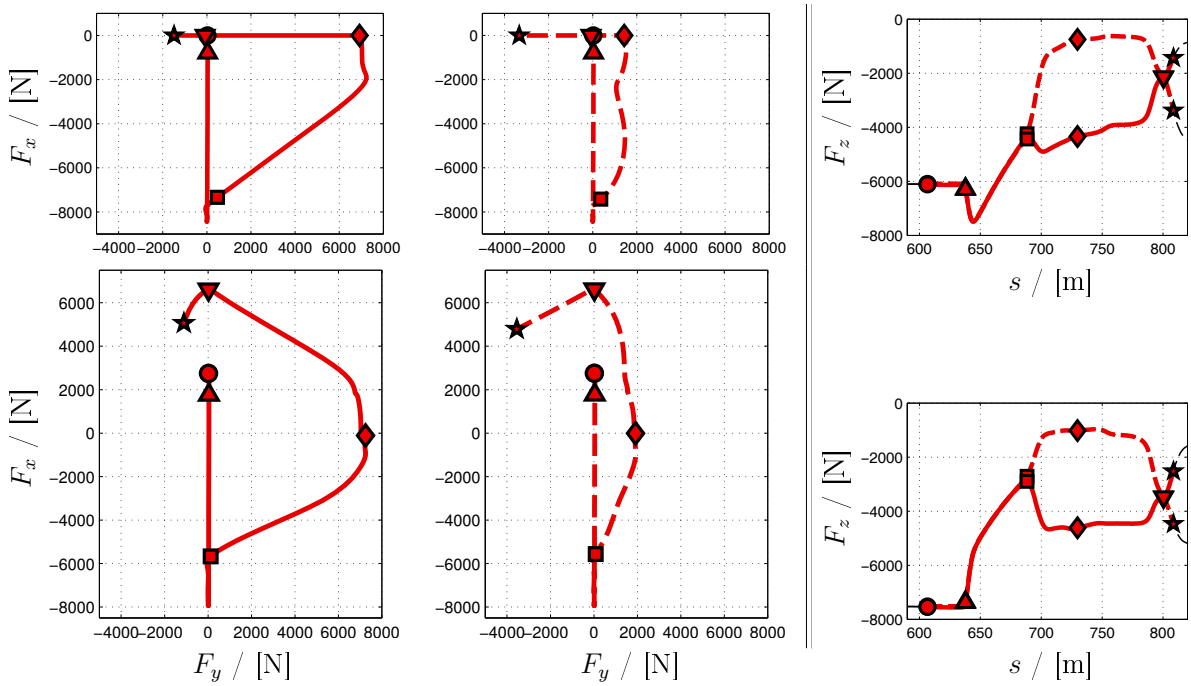

Figure 5.9: Tyre forces in turn (1). The first column gives the longitudinal and lateral tyre forces for the left-hand wheels, with the front-left wheel at the top. The longitudinal forces are on the vertical axes, while the side forces are on the horizontal axes. The second column gives the longitudinal and lateral tyre forces for the right-hand wheels, with the front-right wheel at the top. The third column gives the normal loads as functions of the distance travelled from the start-finish line, with the front wheels at the top; the left-hand wheels are the solid lines, while the right-hand wheels are the dashed lines.

otherwise the computational complexity would be at least cubic in the number of samples, due to the solution of a linear system of size proportional to N. The NLP computation times in Figure 5.10 refer to calculations run on a desktop PC with an Intel Core i7-2600K CPU (4 cores @ 3.40 GHz) and 16 GB of RAM, running a 64-bit Windows 7 operating system. These details are provided as a basis for performance comparisons.

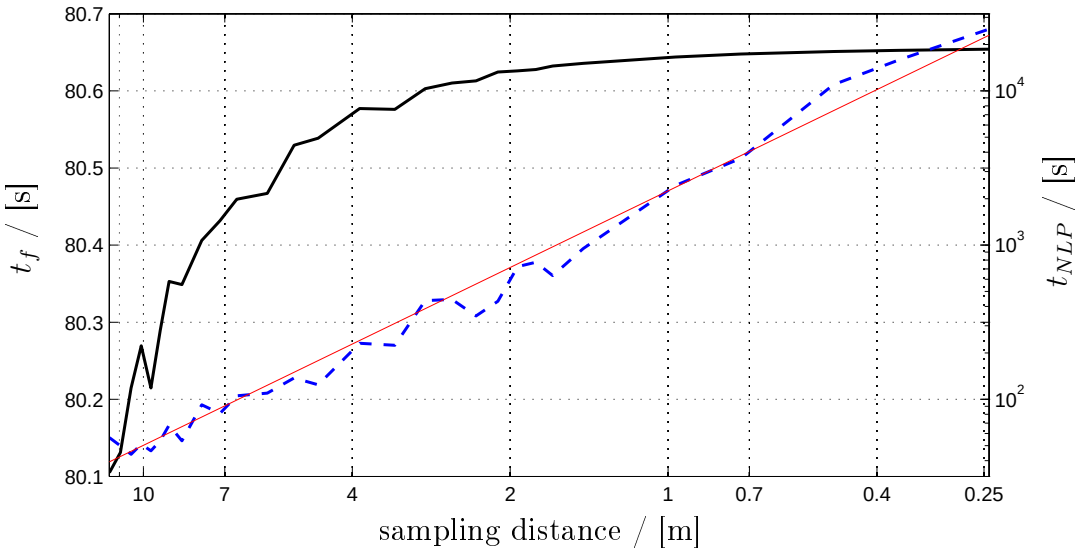

Figure 5.10: Sensitivity of the minimum lap time to arc length mesh size. The solid black line shows the optimal lap time as a function of the arc length sampling distance (on a semi-log scale). The dashed blue line shows the NLP solution time (on a log-log scale) as a function of the arc length sampling distance. The red line is the approximate computation time given by $t_{NLP} \approx 2 \cdot 10^{-3} N^{1.65}$.

5.3.2 Vehicle Stability

Under over-steering conditions four-wheeled vehicles can be unstable in yaw (see [55]). While stability-related constraints have not been included explicitly in the optimal-control problem formulation described in Section 5.2.1, it is possible to include them implicitly by restricting the range over which parameters can be optimised. To show how this might be accomplished one can consider the impact of the location of the aerodynamic centre of pressure on the yaw-mode stability of the car under high-speed cornering. The conditions examined are representative of the racing conditions in turn (3) of the Barcelona track, which is approximated as a 65 m/s corner with a constant radius of curvature of 150 m.

This analysis is conducted using a vehicle model that includes a parallel spring-damper double-wishbone suspension system, and the gyroscopic effects related to the road wheels. The car's parameters are based on those given in Appendix B, with the suspension parameters used in the sprung model consistent with the nominal roll balance used for the optimal control calculations. In order to establish the vehicle's stability properties, the sprung vehicle was driven at constant speed (65 m/s) around a circular trajectory of constant radius (150 m). The equilibrium states thus obtained were used to find a small-perturbation linear model that is continuously updated as the centre of pressure is moved slowly towards the front axle. The vehicle's yaw-mode stability is analysed as a function of the location of the centre of pressure by plotting the centre-of-pressure dependent eigenvalues of the linearised model in Figure 5.11. For the cornering conditions considered, when the aerodynamic centre of pressure is behind $a_a = 1.64$ m the vehicle is stable, but the vehicle becomes unstable in yaw when the centre of pressure is forward of this point. Stability information of this type can be used to restrict the search range over which parameters might be optimised. The nominal location of the aerodynamic centre of pressure is at a_a =1.9 m and one could constrain any parameter-optimisation problem so that $a_a \geq$1.7 m for example. Once potential instabilities of this type have been

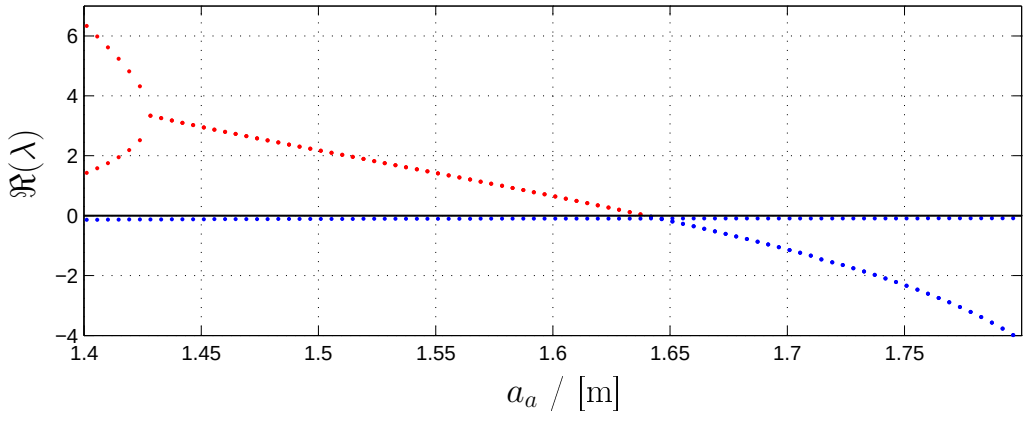

Figure 5.11: Yaw-mode stability as a function of the location of the aerodynamic centre of pressure. The plot shows the real parts of the yaw-mode eigenvalues as functions of the distance between the front axle and the centre of pressure (see Figure 5.1); negative real parts are displayed in blue, while positive ones are red. The model is linearised at a constant speed of 65 m/s in a corner with radius of curvature 150 m, with the aerodynamic centre of pressure moved towards the front axle at 0.0033 m/s.

identified, it is a straightforward matter to constrain parameter-optimisation studies to avoid dynamic instabilities in parameter space.

In general, the displacement of the centre of pressure for drag and downforce is not the only possible source of instability. Other effects, whether included or not in this model, may affect the open-loop stability of the vehicle; possible examples are: changes in the tyre friction, or aerodynamic side forces. Considering the motion of a flying dart one can easily imagine that the position of the aerodynamic centre of pressure relative to the centre of mass may considerably affect the vehicle stability when lateral aerodynamic influences are not negligible. Moving the centre of pressure ahead of the centre of mass is likely to degrade the open-loop stability, following the same principles that impede a stable motion of a dart thrown tail first. When additional sources of instability are recognized and measurable, these can be introduced in the optimal control formulation as constraints on the design parameters.

5.3.3 Parameter Optimisation

One of the important attributes of direct methods for optimal control is the ease with which they optimise simultaneously parameter values and control and state trajectories. This facility will be exploited by optimising a number of setup parameters. The goal is to assess the sensitivity and importance of these parameters in minimising optimal lap times. Figure 5.12 considers the optimisation of the location of the vehicle's mass centre, its aerodynamic centre of pressure, its roll balance and the differential viscosity constant. Each of these parameters is first adjusted on a one-parameter-at-a-time basis with multi-parameter optimisation considered after that.

The first plot in Figure 5.12 shows that the nominal location of the vehicle's mass centre is almost optimal, with little to be gained from a small 39 mm forward adjustment. The optimisation of the aerodynamic centre of pressure location requires a rearward shift of 14.4 cm, which has the effect of reducing the lap time by approximately 0.25 s. The third plot shows that the roll stiffness (see (5.19)) should be increased from its nominal

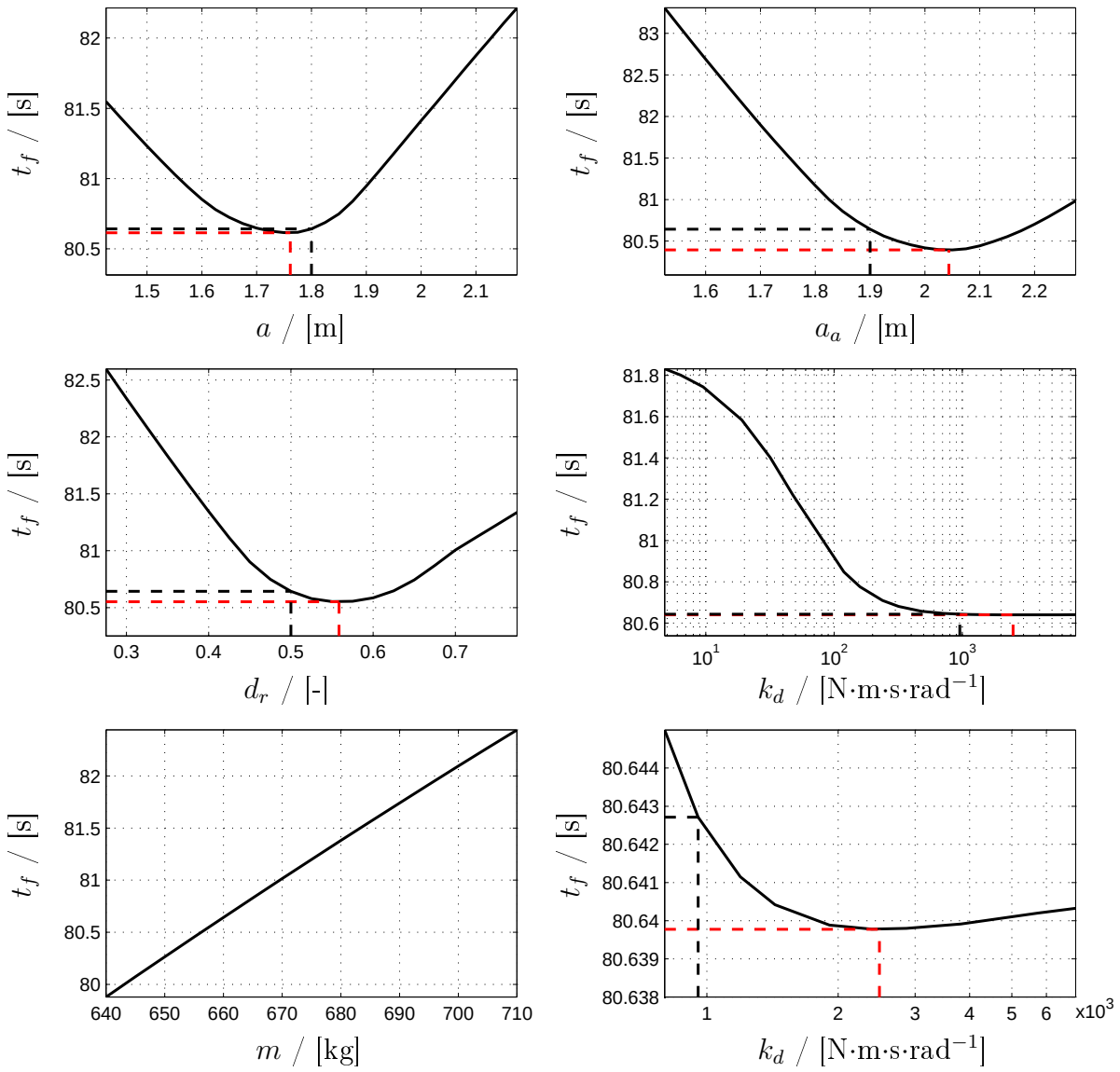

Figure 5.12: Minimum lap time as a function of the centre of mass position a, the position of the centre of pressure a_a, the roll balance d_r, the differential viscosity k_d and the vehicle mass m. The bottom-right plot is a magnified view of the one just above, representing variations in k_d. These curves were computed by varying each parameter in isolation. The black dashed lines correspond to the nominal configuration and its corresponding lap time, while the dashed red lines correspond to optimised parameter values.

Parameter	Optimal Value	Lap Time
a	1.761 m	80.615 s
a_a	2.044 m	80.393 s
d_r	0.559	80.551 s
k_d	2485.7 N·m·s·rad^{-1}	80.640 s
m	–	$\sim$ 0.036 s·kg^{-1}

Table 5.2: Independently optimised parameter values with corresponding lap times.

value of 0.5 to 0.559 thereby reducing the lap time by 0.09 s in this case. The parameter in the group studied that appears to be closest to its optimal value is the viscosity constant of the differential (see (5.29)). By running the car with a viscosity constant of $k_d = 2485.7\,\mathrm{N\,m\,s/rad}$, the minimum lap time is reduced by 3 ms only. The minimum lap time increases when the differential viscosity is reduced because the maximum driving force becomes more and more correlated with the tyre-road friction available at the lighter driving wheel. Interestingly, the lap time attains a minimum for a finite value of the differential viscosity, with a minor performance degradation for values above the optimum[2]. In another single-parameter study the change in lap time was found to vary almost linearly with fuel load at a rate of 0.036 s/kg, which is in close agreement with the commonly employed 'rule of thumb' (0.03 s/kg) that is based on years of track-side measurements. The results of this study are summarised in Table 5.2.

Figure 5.13 shows that if the four parameters considered in Figure 5.12 are optimised in concert, the nominal minimum lap time can be reduced by 0.467 s, which is significantly larger than the improvement achieved in any of the one-parameter-at-a-time studies. In the context of competitive motor sport a lap time reduction of this magnitude is substantial. The results of the multiple-parameter optimisation study are given in Table 5.3. Interestingly, the position of the centre of pressure is the only parameter for which the change from its nominal value is in the same direction observed in the single-parameter

[2]Since tyre wear is not modelled, the differential serves only to maximise the drive torque. In practice a fully locked differential will result in different longitudinal rear-tyre slips through corners, that will give rise to tyre wear and energy dissipation. If tyre wear were recognised in the optimal-control problem setup, a lower value for the differential viscosity constant would probably result.

Parameter	Optimal Value
a	2.047 m
a_a	2.275 m
d_r	0.466
k_d	833.3 N·m·s·rad^{-1}

Table 5.3: Parameter values from a multi-parameter optimisation study. The resulting lap time is 80.178 s.

studies. The solid black curve in Figure 5.13 shows, at each point on the track, the time difference between the nominal-parameter study and the results achieved with optimised parameters. Although the parameter-optimised car is faster than the nominal vehicle in terms of lap time, there are isolated sections on the track, for instance turns (1), (4), (5) and (12) where the nominal car is quicker. The most substantial performance gain appears to result from a higher exit speed from turn (16), which is then carried down the long high-speed straight towards turn (1); there is a 0.2 s gain in this section alone. Further gains are achieved between turns (10) and (15) where a lot of braking, accelerating and steering is required.

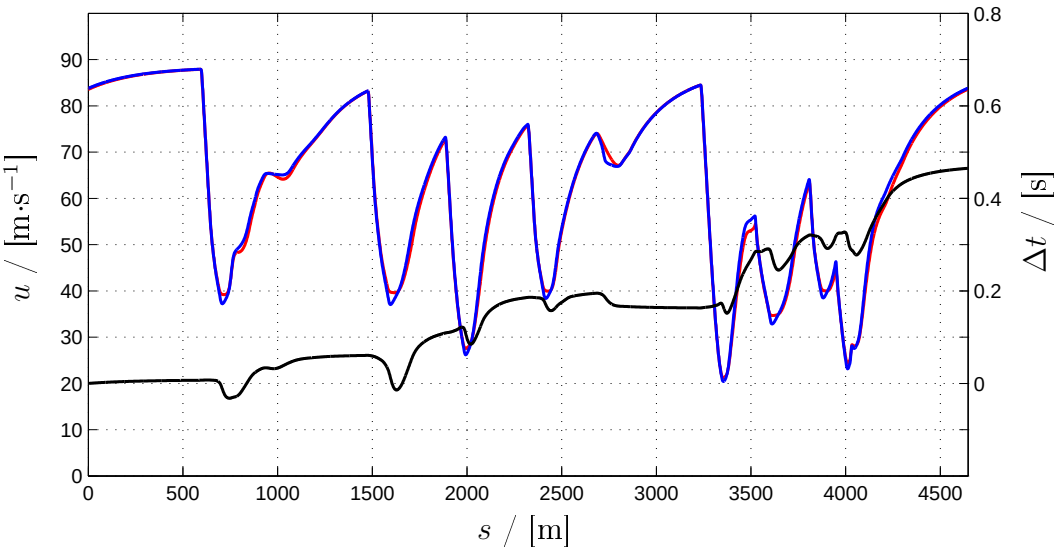

Figure 5.13: Speed profiles for the nominal (red line) and optimised (blue line) parameter sets. The black line shows the time gain for the optimised vehicle against distance travelled from the start-finish line. The left-hand y-axis gives the speed, while the right-hand y-axis is the time gained by the optimised car.

It is common lore that the racing line is relatively insensitive to the dynamic properties

of individual cars. While this belief is generally borne out by this study, it is not always true. Figure 5.14 shows that the racing line for the optimised car in turn (4) can vary by a few metres across the track as compared with the vehicle using nominal parameter settings.

If one studies Figures 5.13 and 5.14 in combination, it is evident that the braking/driving strategy is adjusted to take full advantage of the vehicle's enhanced performance. In the case of the parameter-optimised car, the braking phase in turn (4) starts slightly later and lasts longer than with the nominal car, which results in a significantly lower minimum speed in the corner; the optimised car loses time during the braking phase. In compensation, the optimised vehicle undergoes a more rapid transition from braking to driving that provides a higher exit speed from the corner, which is carried on to the straight to turn (5). The higher exit speed and shorter trajectory (see Figure 5.14) gain the optimised vehicle almost 0.1 s between the braking points preceding turns (4) and (5). Similar behaviours appear elsewhere, with even more pronounced time gains for the slow-speed corners.

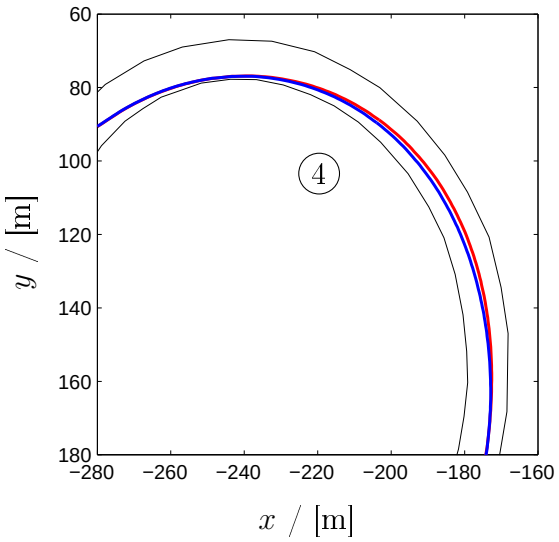

Figure 5.14: Change in the racing line in turn (4) due to parameter variations. The trajectory for the nominal car is shown in red, while the trajectory for the optimised car is shown in blue.

The calculations herein reported are based on approximate models that are not val-

idated and do not reproduce the dynamics of any specific real vehicle. Like in any simulation and model-based design environment, the accuracy of the numerical results is highly reliant on the experimental validation of the dynamic models. However, the proposed models are fully parameterised, and can be tuned to match available measurements. These reference models are used to prove that minimum-time-manoeuvring problems can be solved retaining dynamics and nonlinearities, and to offer a qualitative analysis of interesting simulation results. Considering concurrent parameter optimisation, a realistic vehicle model should be validated in the whole parameter space. Such a task requires the collection of a large amount of experimental data and may be impractical in some circumstances. However, for the purpose of vehicle design, incremental variations in lap time are more important than accurate absolute lap-time references. A performance comparison between alternative vehicle parameterisations can support the choice of setup parameters.

5.3.4 Neglected and Slow Dynamics

The vehicle model used here captures the essential behaviour of a racing car, but it also neglects a number of dynamic features that may be of interest in some studies. These include, for example, the suspension dynamics, vehicle mass variations due to reducing fuel loads, and the effects of tyre wear and tyre temperature changes. Prior minimum-lap-time calculations (see [35]) considered the wheel spin velocities as part of the system state, and modelled the associated spin dynamics. In such a case the wheel spin is controlled by a torque produced either by the engine or by the brakes. However, considering the ratio between the inertial parameters of the wheels and of the vehicle, the wheel spin dynamics are comparatively 'fast', and their inclusion in the optimal-control transcription may require a significantly finer sampling mesh, with a consequential expansion of the NLP problem. The spin dynamics are avoided here, and the tyre slip coefficients are controlled instead; these are connected with the wheel spin velocities by a one-to-one kinematic mapping (see Appendix A). Moreover, gyroscopic and inertial effects associated with wheel spin are relatively small compared to forces and moments entering the dynamics of the

whole vehicle, and are neglected in the vehicle models.

The inclusion of slow dynamics, associated with system states such as the amount of fuel on board, or the temperature of the tyres, would require only routine changes to the model and the optimal-control problem setup. There is no apparent need to change the way in which the model is transcribed, although any new dynamics may affect the convergence of the NLP solution, or its sensitivity to initial conditions. As mentioned earlier, the inclusion of faster (and therefore 'stiffer') dynamics, such as those associated with suspension systems, or tyre carcass deformation, may require a change in the transcription method used. At this point in time the difficulties that these changes may produce are largely unknown. Another challenge relates to the use of tables that may be associated with such things as engine and fuel consumption maps, the thermal properties of the tyres, and the heave- and pitch-dependent aerodynamic drag and downforce properties.

5.4 Summary

This chapter focuses on minimum-lap-time control problems for Formula 1 cars. Full-lap vehicular optimal-control problems can be solved quickly using direct collocation methods without recourse to expedients such as linearisation, quasi-steady-state simplifications or track segmentation. While it is difficult to make quantitative statements about performance gains of this type, prior-art calculations that took many hours can now be completed in less than fifteen minutes on standard desktop computing facilities. These gains come from a combination of several refinements including careful car and track modelling, model non-dimensionalisation, problem scaling, the approximation of non-smooth modelling features with differentiable functions, and the use of computer-generated analytic derivatives in combination with a sparse large-scale NLP solution algorithm.

It is shown that the simultaneous optimisation of the driver controls (therefore the racing line), and several setup parameters, can be easily achieved using standard optimal-control techniques. The setup parameters studied here include the location of the vehicle's mass centre, the location of the aerodynamic centre of pressure, the roll balance and the

differential viscosity constant.

CHAPTER 6

Vehicle Model Extensions

"When I see a car which uses one litre or 1.2 litres of fuel for every kilometre it travels, I ask myself, 'Is this a racing car or is it a tank wagon?' ", Enzo Ferrari.

THE dynamic models considered in this chapter are extensions of the vehicle model presented in Section 5.1.2. The aerodynamic model is updated to include the dependency of the drag and downforce coefficients, and of the position of the centre of pressure, on the longitudinal speed of the car. Modelling an Energy-Recovery System (ERS) requires the addition of auxiliary state and control variables: new system states must be defined in connection with energy stored in the batteries, or to measure the energy exchanged between components throughout any one lap; new control variables are associated with the power absorbed or generated by the powertrain components, which must be optimised in order to guarantee efficiency, while retaining high performance.

The speed-dependent aerodynamics are modelled in Section 6.1. Section 6.2 describes the model updates required in the case of a three-dimensional track. It is assumed that the vehicle travels on a moving 'tangent plane' that is orthogonal to the Darboux-frame normal vector $\hat{\mathbf{m}}$ as defined in Section 4.1.3. The kinematics of this plane are studied in Section 6.2.1.1. This analysis is followed by a study of the vehicular motion on the tangent plane. The car's dynamic equations are derived in Section 6.2.1.2, with the vehicle data available in Appendix B. The algebraic equations for the tyre load transfer are considered

in Section 6.2.1.3, with the tyre model given in Appendix A. The cost and constraints for the minimum-lap-time optimal-control problem are outlined in Section 6.2.2. The numerical results are discussed in Section 6.2.3. The contents of Sections 6.1 and 6.2 are currently being considered for publication in [46].

Section 6.3 studies the inclusion of energy-recovery systems. The model extensions for the 2013 kinetic energy-recovery system and the 2014 kinetic-thermal energy-recovery system are proposed in Sections 6.3.1 and 6.3.2, with the corresponding results discussed in Sections 6.3.1.1 and 6.3.2.1. The modelling techniques in Section 6.3 are covered in [47], which has been accepted for publication; the results in Sections 6.3.1.1 and 6.3.2.1 extend the ones presented in the paper to the case of a three-dimensional track.

6.1 Speed-Dependent Aerodynamic Coefficients

The overall performance of current Formula 1 cars is largely dependent on the aerodynamic design, and teams invest a considerable amount of resources in the continuing research and testing of new and improved aerodynamic components. The overarching goal is to pinpoint the optimal trade-off between downforce and drag, and the optimal distribution of the downforce across the wheels. This optimisation aims to maximise cornering capabilities, braking and traction. It is self-evident that aerodynamic forces are heavily dependent on the speed of the car, as well as other factors such as its orientation with respect to the air flow and ground effects. An accurate description of the vehicle aerodynamics is beyond the scope of this thesis. However, in this section it will be proved that the inclusion of a slightly more detailed aerodynamic model, as opposed to the one in Section 5.1.4, is possible, and that it does not degrade the effectiveness of the numerical optimal-control methods.

In Section 5.1.4 the aerodynamic drag and downforce were modelled by a quadratic rule with constant coefficients; the centre of pressure was considered fixed in the vehicle frame. Consistently with Section 5.1.4 the following model neglects any aerodynamic force acting outside the vehicle's plane of symmetry, and any effect associated with the

orientation of the car. The aerodynamic force is applied at the centre of pressure, which is located in the vehicle's plane of symmetry (see Figure 5.1). In this case the position of the centre of pressure varies with the longitudinal speed of the car:

$$a_a(u) = c_{p2}u^2 + c_{p1}u + c_{p0}. \tag{6.1}$$

The drag and lift forces are given by

$$F_{ax} = -0.5\, c_d(u)\, \rho_a\, a_f\, u^2, \tag{6.2}$$

and

$$F_{az} = 0.5\, c_l(u)\, \rho_a\, a_f\, u^2, \tag{6.3}$$

respectively. The speed-dependent drag and downforce coefficients are approximated by polynomials:

$$\begin{aligned} c_d(u) &= c_{d2}u^2 + c_{d1}u + c_{d0} \\ c_l(u) &= c_{l2}u^2 + c_{l1}u + c_{l0} \end{aligned} \tag{6.4}$$

The polynomial coefficients adopted for the speed-dependent aerodynamic features are reported in Table B.3; Figure 6.1 gives the numerical values of a_a, c_d and c_l for varying longitudinal speed.

6.1.1 Nominal Car Performance

The solution of the minimum-lap-time problem for the car with speed-dependent aerodynamic coefficients results in the vehicular speed profile given in Figure 6.2. Comparing the results for constant (see Section 5.3.1) and speed-dependent parameters shows that an advantage can be gained from a higher value of the downforce coefficient at lower speeds. Although the top speed is higher for the constant-parameter car, the extended vehicle benefits from improved aerodynamic properties through low-speed corners and driven sections. The minimum lap time for the updated vehicle is 78.691 s, which is 1.951 s lower than the estimate for the constant-parameter vehicle from Section 5.3.1. Changes in the aerodynamic model entail slight changes in the vehicle optimal trajectory too. Figure 6.3 shows how the driven line changes at the exit of turn (4).

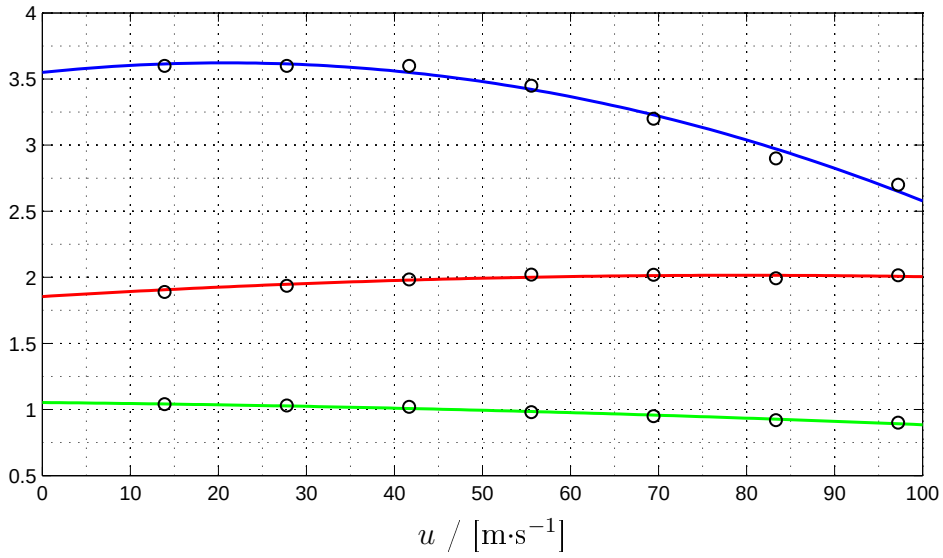

Figure 6.1: Car aerodynamic maps: the drag coefficient c_d is the green curve; the downforce coefficient c_l is the blue curve; the distance of the aerodynamic centre of pressure from the front axle a_a is given by the red curve (in metres). The '$\circ$' symbols represent reference data points.

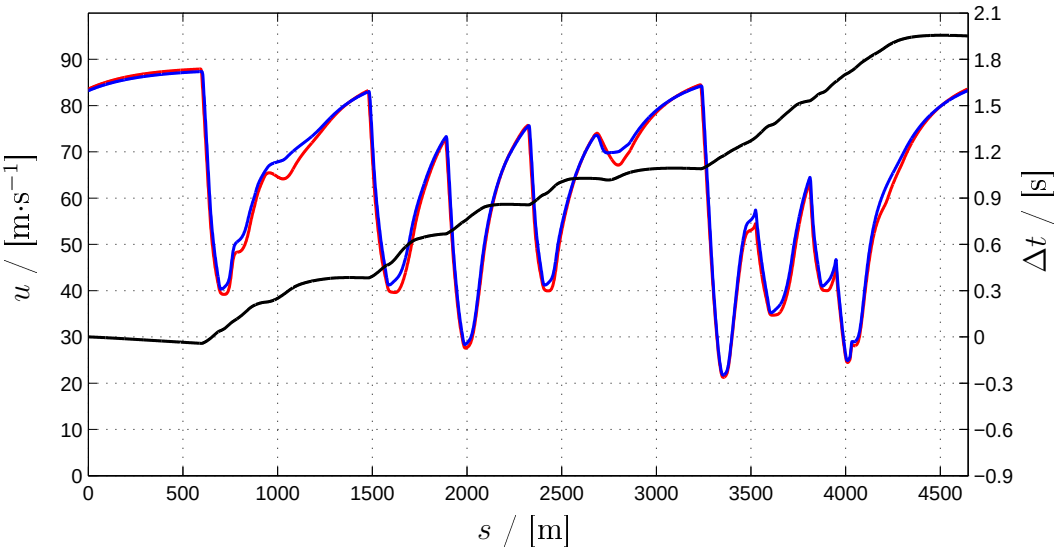

Figure 6.2: Speed profiles for the vehicle with constant (red line) and speed-dependent (blue line) aerodynamic coefficients. The black line shows the time gain for the vehicle with speed-dependent parameters against distance travelled from the start-finish line. The left-hand y-axis gives the speed, while the right-hand y-axis is the time gained.

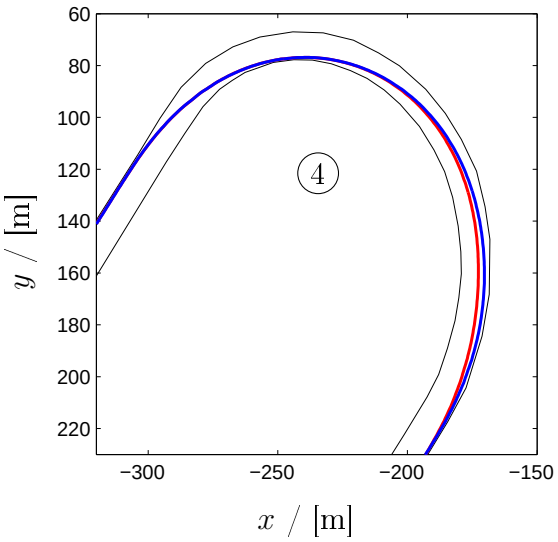

Figure 6.3: Change in the racing line in turn (4) due to speed-dependent aerodynamic properties. The constant-parameter trajectory is shown in red, while the trajectory for the car with speed-dependent parameters is shown in blue.

6.2 3D Track

Minimum-lap-time problems for Formula 1 cars have thus far been studied in two dimensions only. While the 'flat-track' assumption is often a valid and useful idealisation, it neglects a number of influences that cumulatively may become important on some race tracks.

It is well known from everyday experience that a vehicle's top speed increases on downhill sections, while the opposite is true on ascending inclines. It is also known from this everyday experience that extended braking distances are required on descending gradients, while the opposite is true on ascending ones. High-speed corners on modern roads are often banked (superelevated) in order to improve drainage and road holding, on the other hand adverse cambering undermines road holding and reduces maximum achievable cornering speed. Slightly less obvious is the fact that a vehicle tends to become 'light' if it is driven fast over the brow of a hill, while it will 'squat' down on the suspension system if driven fast through a dip in the road. This apparent variation in weight is the result of a centripetal force, and has an effect similar to that produced by aerodynamic

lift or downforce. As with all changes in downforce, this centripetal force can impact the speed of the vehicle. Three-dimensionality also introduces gyroscopic moments that change the lateral and longitudinal load transfer between the tyres. The aim of this work is to study the significant influences on the minimum achievable lap time of a Formula 1 car resulting from 3D road geometry. The focus will be on general-purpose analysis techniques and conceptual understanding, rather than on arriving at specific conclusions that relate to particular tracks.

The three-dimensional character of the track is modelled using three curvatures as explained in Section 4.1.3. In the case of a flat (horizontal) track the geodesic curvature only is used to describe the tightness of the bends. Once the track takes on three-dimensional features, such as banked corners, elevation changes and camber variations, the geometric and dynamic behaviour of the vehicle becomes significantly more complicated. If a car is driving along a straight road that then goes over the brow of a hill a second curvature variable is required. This is the normal curvature, which describes the curvature of the car's path in the vertical plane. Normal curvature produces centripetal forces that change the car's apparent weight. The third curvature variable, the relative torsion, describes the way in which the road changes camber, or how it can adopt a 'cork-screw' motion, while still going straight ahead. This cork-screwing motion also produces forces and moments that may influence the dynamics and operating envelope of the car. It is self evident that the importance, or otherwise, of 3D influences will depend on the geometry of the track under study.

6.2.1 Car Model

The track and car kinematics are modelled using the differential-geometric framework described in Chapter 4. In the analysis given here it is assumed[1] that the road surface is, at any time, a plane orthogonal to the normal vector $\hat{\mathbf{m}}$ as defined in Section 4.1.3. This

[1] As an alternative, and potentially more accurate approximation, the road-approximating tangent plane can be considered tangent to the ribbon at the vehicle's geometric centre. This modelling approach is explained in Appendix C. This abstraction is more complicated to analyse and, as in the case of the Barcelona track, is unnecessary when the track does not have a large relative torsion.

plane moves with the car along the track spine, thereby generating a road surface that heaves, pitches and rolls under the vehicle.

The vehicle model is extended from the standard description in Section 5.1.2 represented in Figures 5.1, 5.3 and 5.4, and accounts for the speed-dependent aerodynamic properties modelled in Section 6.1. The vehicle's position and orientation on the track are expressed in terms of three coordinates, according to the method given in Section 5.1.1, summarised in Figure 5.2. The position $\mathbf{r}(s, n)$ of the vehicle's geometric centre $\mathbf{G}$ is determined by its curvilinear abscissa s and the lateral position relative to the ribbon spine n. The angle ξ is the yaw angle of the car relative to the spine of the track. In Section 5.1.1 the reference frame attached to the ribbon spine could only translate in the x- and y- directions and rotate around the z-axis. Here the spine frame is allowed to translate and rotate in all directions in space. The geometric notation introduced in Section 4.1 provides the framework to fully describe the track in three dimensions. As was explained in Section 4.2, the track model is based on measured data, with its curvature representation found as the solution of a subsidiary optimal-control problem. The track's generalised curvature vector is given by $\boldsymbol{\Omega}_B = [\Omega_x \ \Omega_y \ \Omega_z]^T$ in which Ω_x is the relative torsion, Ω_y is the normal curvature and Ω_z the geodesic curvature. The track's angular velocity vector is given by $\boldsymbol{\omega}_B = \boldsymbol{\Omega}_B \frac{ds}{dt}$, where $\boldsymbol{\omega}_B = [\omega_x \ \omega_y \ \omega_z]^T$ and $\frac{ds}{dt}$ is the vehicle's speed in the curvilinear abscissa s. In the unit-speed case, where $\frac{ds}{dt} = 1$, it follows that $\boldsymbol{\omega}_B = \boldsymbol{\Omega}_B$.

6.2.1.1 Kinematics

The Euler angles underlying the track description will be needed to orient the gravity vector. Using equation (4.19), and $S(\boldsymbol{\omega}_B) = R^T \dot{R}$, one can find differential equations that describe the evolution of the Euler angles controlled by the track's angular velocity

$$\dot{\phi} = \omega_x + (\sin\phi\,\omega_y + \cos\phi\,\omega_z)\tan\mu \tag{6.5}$$

$$\dot{\mu} = \cos\phi\,\omega_y - \sin\phi\,\omega_z \tag{6.6}$$

$$\dot{\theta} = (\sin\phi\,\omega_y + \cos\phi\,\omega_z)\sec\mu. \tag{6.7}$$

These road-related Euler angles are a by-product of the optimal-control problem used to determine the track curvature as described in Section 4.2. These equations become an integral part of the optimal-control problem solution if the Euler angles are not already available.

The next kinematic relationships relate to the way in which the car progresses down the road. In order to obtain the required equations, the velocity of the car will be expressed in the Darboux frame, following the notation introduced in Section 4.1.3 and illustrated in Figure 4.1. The velocity of the car's geometric centre **G** expressed in its body-fixed coordinate system is $\mathbf{v}_B = [u\ v\ w]^T$; the longitudinal component u is determined by the throttle control, the lateral component v is determined by steering, while the vertical component w is determined by the track geometry. The position of the car's geometric centre expressed in the Darboux frame is $n\hat{\mathbf{n}} = [0\ n\ 0]^T$, while the angular velocity of the Darboux frame is $\boldsymbol{\omega}_B = [\omega_x\ \omega_y\ \omega_z]^T$. The velocity of the car in the Darboux frame is:

$$\dot{s}\hat{\mathbf{e}}_x + \dot{n}\hat{\mathbf{e}}_y + \boldsymbol{\omega}_B \times (n\hat{\mathbf{n}}) = R_z(\xi)\mathbf{v}_B \quad \longleftrightarrow \quad \begin{bmatrix} \dot{s} - n\omega_z \\ \dot{n} \\ n\omega_x \end{bmatrix} = \begin{bmatrix} u\cos\xi - v\sin\xi \\ v\cos\xi + u\cos\xi \\ w \end{bmatrix}, \quad (6.8)$$

in which

$$R_z(\xi) = R(\hat{\mathbf{e}}_z, \xi) = \begin{bmatrix} \cos\xi & -\sin\xi & 0 \\ \sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6.9)$$

accounts for the yawing of the car relative to the spine of the track. The terms on the left-hand side of the equalities in (6.8) express the velocity of the vehicle's geometric centre in terms of the curvilinear coordinates and their rate of change. The right-hand terms give the velocity expressed in the vehicle's body-fixed reference frame. The vector-product term in (6.8) derives from the angular velocity of the Darboux frame. The first two rows of (6.8) are equivalent to the kinematic equations (5.6) and (5.7) introduced for the flat track; the third row describes the absolute vertical velocity of the car's geometric centre as a result of road camber changes, and follows from the fact that the car cannot have a vertical velocity component relative to the Darboux frame in the $\hat{\mathbf{m}}$-direction. This gives

$$w = n\,\omega_x. \quad (6.10)$$

If the distance travelled is selected as the independent variable, then a complete set of kinematic equations includes (5.9) – that generates $s_t(s)$ – (5.7) and

$$\frac{d\phi}{ds} = s_t\dot{\phi}, \tag{6.11}$$
$$\frac{d\mu}{ds} = s_t\dot{\mu}, \tag{6.12}$$
$$\frac{d\theta}{ds} = s_t\dot{\theta}, \tag{6.13}$$

which come from (6.5), (6.6) and (6.7). Suppose $\bar{R} = RR_z(\xi)$, where R is defined in Section 4.1.3, then $S(\bar{\boldsymbol{\omega}}_B) = \bar{R}^T\dot{\bar{R}}$ gives

$$\bar{\boldsymbol{\omega}}_B = \begin{bmatrix} \bar{\omega}_x \\ \bar{\omega}_y \\ \bar{\omega}_z \end{bmatrix} = \begin{bmatrix} \cos\xi\omega_x + \sin\xi\omega_y \\ \cos\xi\omega_y - \sin\xi\omega_x \\ \omega_z + \dot{\xi} \end{bmatrix}, \tag{6.14}$$

which is the angular velocity of the car in its body-fixed reference frame. In the sequel the vehicle dynamic equations will be used to find $\bar{\omega}_z$, with the car yaw angle ξ deduced from (6.14) by integrating

$$\frac{d\xi}{ds} = s_t(s)\bar{\omega}_z - \Omega_z, \tag{6.15}$$

which corresponds to (5.11) when the track is flat.

6.2.1.2 Dynamics

The equations describing the dynamics of the car will be derived using the Lagrangian approach (see [43, Ch.1] or Section 2.2.1). The kinetic energy of the vehicle is

$$T = \frac{1}{2}\Big(m\mathbf{v}_{mB}^T\mathbf{v}_{mB} + \bar{\boldsymbol{\omega}}_B^T I_B\bar{\boldsymbol{\omega}}_B,\Big) \tag{6.16}$$

where $\mathbf{v}_m$ is the absolute speed of the vehicle's mass centre, the matrix I is the vehicle's inertia tensor. The absolute velocity of the car's mass centre is given by

$$\mathbf{v}_{mB} = \mathbf{v}_B + \bar{\boldsymbol{\omega}}_B \times (-\hat{\mathbf{e}}_z h). \tag{6.17}$$

The potential energy due to gravity is

$$U = -mgz. \tag{6.18}$$

The equations of motion are thus given by the Euler-Lagrange equations, defined in the inertial frame:

$$\begin{array}{rcl} \dfrac{d}{dt}\left(\dfrac{\partial L}{\partial \mathbf{v}_{mB}}\right)\Big|_I & = & F_B + \dfrac{\partial L}{\partial \mathbf{x}_{mB}} \\ \dfrac{d}{dt}\left(\dfrac{\partial L}{\partial \bar{\boldsymbol{\omega}}_B}\right)\Big|_I & = & M_B \end{array}, \tag{6.19}$$

which include the action of external forces F_B and moments M_B, expressed in the car's body-fixed reference frame, and where

$$L = T - U \tag{6.20}$$

is the system Lagrangian. The time derivatives in (6.19) make use of the time-derivative operator

$$\frac{d(\cdot)}{dt}\Big|_I = \frac{d(\cdot)}{dt}\Big|_B + \bar{\boldsymbol{\omega}} \times (\cdot), \tag{6.21}$$

which describes the transformation of the time derivative between inertial and body-fixed reference frames (see [42, Ch.4]). Since $\mathbf{x}_{mI} = \mathbf{x}_I + R\mathbf{x}_{mB}$ there holds $\partial z/\partial \mathbf{x}_{mB} = R^T\hat{\mathbf{e}}_z$. Solving (6.19) gives

$$\begin{array}{rcl} m\Big\{\dot{\mathbf{v}}_B + \dot{\bar{\boldsymbol{\omega}}}_B \times (-\hat{\mathbf{e}}_z h) + \bar{\boldsymbol{\omega}}_B \times \big[\mathbf{v}_B + \bar{\boldsymbol{\omega}}_B \times (-\hat{\mathbf{e}}_z h)\big]\Big\} & = & F_B + mgR^T\hat{\mathbf{e}}_z \\ I_B\dot{\bar{\boldsymbol{\omega}}}_B + \bar{\boldsymbol{\omega}}_B \times (I_B\bar{\boldsymbol{\omega}}_B) & = & M_B. \end{array} \tag{6.22}$$

The term $\dot{\mathbf{v}}_B$ is the acceleration of the vehicle's geometric centre in its body-fixed reference frame:

$$\dot{\mathbf{v}}_B = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}. \tag{6.23}$$

The term $m\dot{\bar{\boldsymbol{\omega}}}_B \times (-\hat{\mathbf{e}}_z h)$ represents an apparent force perceived by the vehicle due to the non-uniform rotation of the track surface:

$$\dot{\bar{\boldsymbol{\omega}}}_B \times (-\hat{\mathbf{e}}_z h) = \begin{bmatrix} -h\dot{\bar{\omega}}_y \\ +h\dot{\bar{\omega}}_x \\ 0 \end{bmatrix}. \tag{6.24}$$

The term $m\bar{\boldsymbol{\omega}}_B \times \big[\mathbf{v}_B + \bar{\boldsymbol{\omega}}_B \times (-\hat{\mathbf{e}}_z h)\big]$ represents an apparent centrifugal force:

$$\begin{bmatrix} \bar{\omega}_x \\ \bar{\omega}_y \\ \bar{\omega}_z \end{bmatrix} \times \begin{bmatrix} u - h\bar{\omega}_y \\ v + h\bar{\omega}_x \\ n\omega_x \end{bmatrix} = \begin{bmatrix} \bar{\omega}_y n\omega_x - \bar{\omega}_z(v + h\bar{\omega}_x) \\ \bar{\omega}_z(u - h\bar{\omega}_y) - \bar{\omega}_x n\omega_x \\ \bar{\omega}_x(v + h\bar{\omega}_x) - \bar{\omega}_y(u - h\bar{\omega}_y) \end{bmatrix}. \tag{6.25}$$

The term $mg\bar{R}^T\hat{\mathbf{e}}_z$ is the gravitational force expressed in the car body-fixed reference frame:

$$mgR^T\hat{\mathbf{e}}_z = mg\begin{bmatrix}\sin\xi\sin\phi\cos\mu-\cos\xi\sin\mu\\ \sin\xi\sin\mu+\cos\xi\sin\phi\cos\mu\\ \cos\phi\cos\mu\end{bmatrix}. \tag{6.26}$$

The term $\bar{\boldsymbol{\omega}}_B\times(I_B\bar{\boldsymbol{\omega}}_B)$ captures the gyroscopic moment acting on the car; if the vehicle's principal axes are assumed parallel to the vehicle's body-fixed coordinate system, this moment is given by

$$\bar{\boldsymbol{\omega}}_B\times\left(\begin{bmatrix}I_x&0&0\\0&I_y&0\\0&0&I_z\end{bmatrix}\bar{\boldsymbol{\omega}}_B\right)=\begin{bmatrix}(I_z-I_y)\bar{\omega}_z\bar{\omega}_y\\(I_x-I_z)\bar{\omega}_z\bar{\omega}_x\\(I_y-I_x)\bar{\omega}_y\bar{\omega}_x\end{bmatrix}. \tag{6.27}$$

The term $\dot{\bar{\boldsymbol{\omega}}}_B$ is the angular acceleration of the vehicle.

The vector F_B represents the external forces acting on the vehicle:

$$F_B=\begin{bmatrix}F_x\\F_y\\F_z\end{bmatrix} \tag{6.28}$$

in which F_x, F_y and F_z are, respectively, the longitudinal, lateral and vertical components of all the tyre and aerodynamic forces acting on the car that are given by

$$F_x = \cos\delta(F_{frx}+F_{flx})-\sin\delta(F_{fry}+F_{fly})+(F_{rrx}+F_{rlx})+F_{ax} \tag{6.29}$$

$$F_y = \cos\delta(F_{fry}+F_{fly})+\sin\delta(F_{frx}+F_{flx})+(F_{rry}+F_{rly}) \tag{6.30}$$

$$F_z = F_{rrz}+F_{rlz}+F_{frz}+F_{flz}+F_{az}. \tag{6.31}$$

The aerodynamic drag force F_{ax} and downforce F_{az} are given in (6.2) and (6.3), with the tyre friction forces illustrated in Figure 5.4 and described in detail in Appendix A; the $F_{..z}$'s are the vertical tyre forces for each of the four wheels.

The vector M_B represents the external moments acting on the vehicle:

$$M_B=\begin{bmatrix}M_x\\M_y\\M_z\end{bmatrix} \tag{6.32}$$

with

$$M_x = w_r(F_{rrz} - F_{rlz}) + w_f(F_{frz} - F_{flz}) - hF_y \tag{6.33}$$

$$M_y = b(F_{rrz} + F_{rlz}) - a(F_{frz} + F_{flz}) + hF_x + (a_a - a)F_{az} \tag{6.34}$$

$$M_z = a\left(\cos\delta(F_{fry} + F_{fly}) + \sin\delta(F_{frx} + F_{flx})\right) - b(F_{rry} + F_{rly}) + w_f\left(\sin\delta F_{fry} - \cos\delta F_{frx} + \cos\delta F_{flx} - \sin\delta F_{fly}\right) + w_r(F_{rlx} - F_{rrx}). \tag{6.35}$$

The first two terms on the right-hand side of equation (6.33) are the roll moments produced by the vertical tyre forces, the third term is the roll moment produced by the lateral tyre forces F_y (see (6.30)). The first two terms on the right-hand side of equation (6.34) represent the pitching moments produced by the vertical tyre forces, the third term is the pitching moment produced by the longitudinal force F_x (see (6.29)), the fourth term comes from the aerodynamic downforce.

The car's equations of motion can now be assembled by recognising the forces of inertia, the centripetal forces, the gravitational forces, the road-related gyroscopic moments, the tyre forces and the aerodynamic forces:

$$\dot{u} = \bar{\omega}_z(v + h\bar{\omega}_x) - \bar{\omega}_y n\omega_x + g\left(\sin\xi\sin\phi\cos\mu - \cos\xi\sin\mu\right) + F_x/m + h\dot{\bar{\omega}}_y, \tag{6.36}$$

$$\dot{v} = \bar{\omega}_x n\omega_x - \bar{\omega}_z(u - h\bar{\omega}_y) + g\left(\sin\xi\sin\mu + \cos\xi\sin\phi\cos\mu\right)) + F_y/m - h\dot{\bar{\omega}}_x, \tag{6.37}$$

$$\dot{\bar{\omega}}_z = \big((I_x - I_y)\bar{\omega}_x\bar{\omega}_y + M_z\big)/I_z. \tag{6.38}$$

The equations (6.36), (6.37) and (6.38) expressed in terms of the elapsed arc length are

$$\frac{du}{ds} = s_t\dot{u}, \tag{6.39}$$

$$\frac{dv}{ds} = s_t\dot{v}, \tag{6.40}$$

$$\frac{d\bar{\omega}_z}{ds} = s_t\dot{\bar{\omega}}_z. \tag{6.41}$$

6.2.1.3 Load Transfer

The algebraic equations governing the s-dependent tyre loads are computed balancing the forces normal to the road and balancing moments around the body-fixed x_B- and y_B-axes; see Figure 5.3. The vertical force balance gives

$$\dot{w} = F_z/m + g\cos\phi\cos\mu + \bar{\omega}_y(u - h\bar{\omega}_y) - \bar{\omega}_x(v + h\bar{\omega}_x). \tag{6.42}$$

Balancing moments around the car's mass centre in the x_B- and y_B-directions gives

$$I_x\dot{\bar{\omega}}_x = M_x + (I_y - I_z)\bar{\omega}_z\bar{\omega}_y, \tag{6.43}$$

$$I_y\dot{\bar{\omega}}_y = M_y + (I_z - I_x)\bar{\omega}_z\bar{\omega}_x. \tag{6.44}$$

The inertial terms on the left-hand side of equations (6.43) and (6.44) relate to the angular acceleration of the road expressed in the car's coordinate system. The last term in equation (6.43) is a gyroscopic moment acting in the x_B-direction. The same can be said for equation (6.44) considering moments in the y_B-direction.

Equations (6.42), (6.43) and (6.44) are a set of linear equations in four unknowns considering the tyre normal loads as decision variables. The roll balance relationship (5.19) is included to complete the set of equations and provide a unique solution for the load distribution across the four wheels. The nonlinear techniques described in Section 5.1.3.2 are again employed to guarantee that each load is non-positive.

6.2.2 Optimal-Control Problem Formulation

In order to solve the minimum-lap-time optimal-control problem one has to assemble a performance index, which is the lap time in this case, the constraints associated with the problem dynamics, the constraints describing the track geometry and a constraint on the engine power.

The kinematic equations required are: (5.9) that describes the relationship between time and distance; (6.11) to (6.13) that describe the evolution of the Darboux frame Euler angles; (5.10) that describes the evolution of the car's position relative to the spine of the

track; and (6.15) that describes the evolution of the vehicle's yaw angle relative to the spine.

The dynamic equations include the longitudinal and lateral force balance equations (6.39) and (6.40), which require (6.29) and (6.30), and the yaw moment balance equation (6.41). The tyre force constraint equations include: the normal load balance condition (6.42); the roll and pitch load transfer conditions (6.43) and (6.44); and the roll balance constraint (5.19). To complete the tyre force description one also requires the normal load constraint (5.25) and the tyre force description given in Appendix A. The aerodynamic forces are given by (6.2) and (6.3). One also needs to include longitudinal tyre force constraints associated with the brakes and the differential — one possible set of constraint equations can be found in Section 5.1.5.

6.2.3 Results

In order to gain a proper insight into the three-dimensional behaviour of the optimally-controlled car it is necessary to understand the relevant features of the track. A 3D view of the Barcelona track is shown in Figure 4.4. The corner numbering scheme and the approximate distances to each corner from the start-finish line are consistent with Figure 5.5 and Table 5.1.

Figure 6.4 shows the track elevation and geodesic curvature as functions of the distance from the start-finish line. In the sign convention used here positive geodesic curvature is associated with right-hand corners. The z-axis points downwards and so points of high elevation have large negative values. In order to determine if a corner has a positive or negative camber, the product of the yaw rate $d\psi/ds$ and the camber angle ϕ is computed and plotted in Figure 6.5. Since positive values of yaw rate are associated with right-handed corners, and positive values of camber angle indicate cambering towards the right, a positive product indicates positive camber, while a negative value for the product indicates adverse cambering.

The speed profiles of the car on the three-dimensional track and a two-dimensional

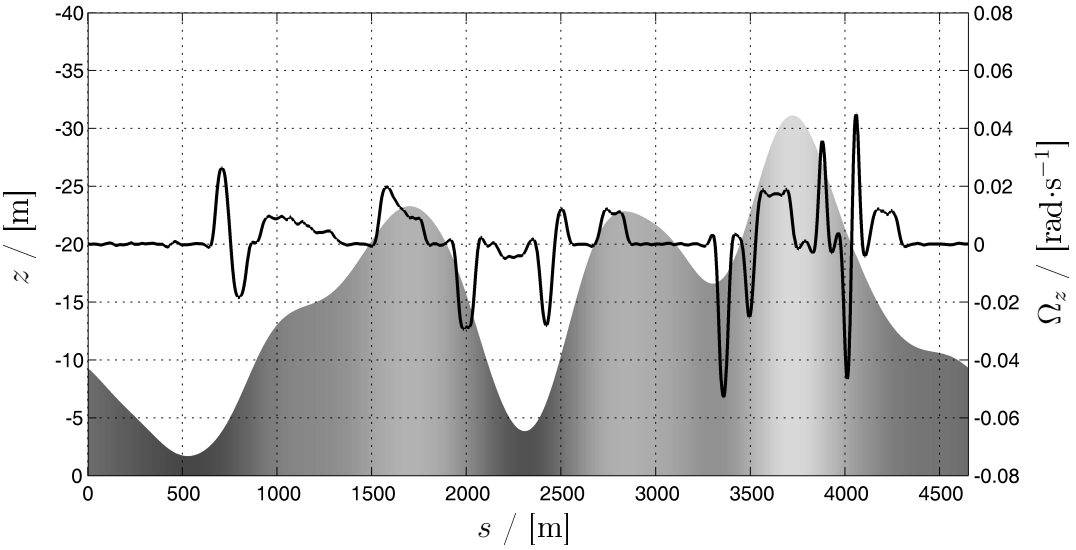

Figure 6.4: Elevation z and geodesic curvature Ω_z of the track against distance travelled. The elevation is given by the shaded area; the shading is consistent with Figure 4.4. The black line represents the geodesic curvature.

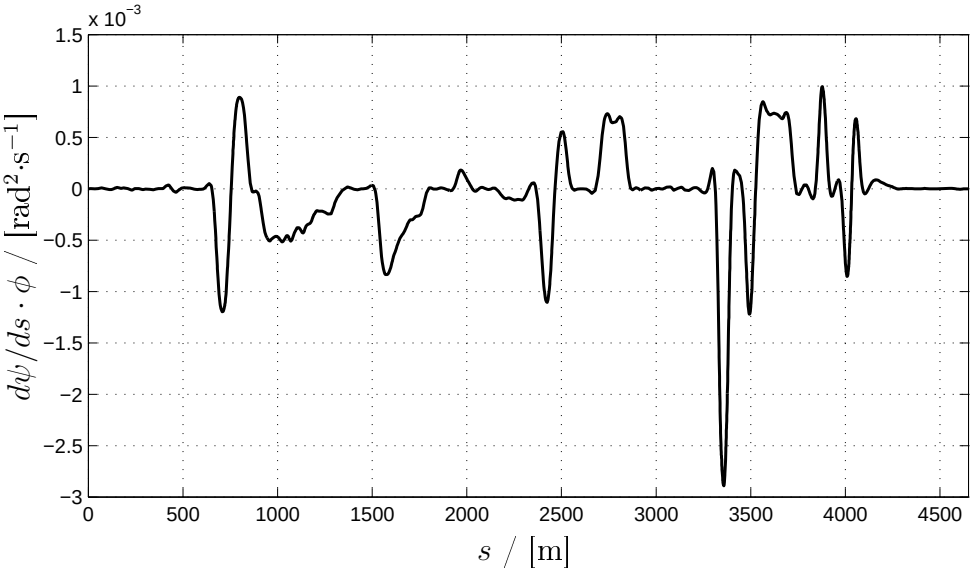

Figure 6.5: Product of the track yaw rate and camber angle against distance travelled. Positive values are indicative of positive cambering, while negative values are associated with adverse cambering.

projection of the track are shown in Figure 6.6. This figure also shows the differences in the times required to reach each point on the track. Positive values indicate that the car on the three-dimensional track is faster. In this case the optimal lap time for the three-dimensional track is 79.300 s, while that for the two-dimensional track is 78.691 s (see Section 6.1.1), giving a difference of −0.609 s. A small fraction of this time difference is due to a slightly longer track in the 3D case; the 3.7 m increment in distance accounts for approximately 0.063 s considering an average speed of about 59 m/s obtained for the 2D optimal lap. Figure 6.6 also shows that there are speed differences of up to ± 11 km/h at various points on the circuit. These differences come primarily from two influences: a centripetal force component normal to the road surface and changes in the direction of the gravitational force with respect to the road surface. A vertical component of the centripetal force changes the tyre loading and thus the ability to generate side forces; this effect is very similar to that associated with changes in aerodynamic downforce. Well-known road features that produce vertical centripetal forces are cambered turns and the presence of dips or humps: dips and positive cambered turns increase the tyre load, while the opposite happens for adverse cambered turns and humps. On an inclined road the gravitational force acting on a vehicle has components parallel to the road surface, which can either push the vehicle forward, when travelling downhill, or slow it down, on an ascent; at 90 m/s, a 100 N gravitational force component in the direction of travel is equivalent to a 9 kW variation in engine power. Gravitational force components in the vehicle's lateral direction can enhance or degrade the cornering ability too.

As shown in Figure 6.7, three-dimensional track features can cause the car to change its racing line in order to find the best compromise between the many factors limiting the car's top speed. In turns (2) and (3) the vehicle experiences positive and then adverse camber, and it is running uphill throughout. This combination of effects is causing the car to take a significantly different racing line in the two cases. In fact, the car racing the 3D track dives through turns (2) and (3) in the interior of the road profile, whereas the vehicle trajectory is restricted by the track boundaries in the 2D case. In turn (4) the

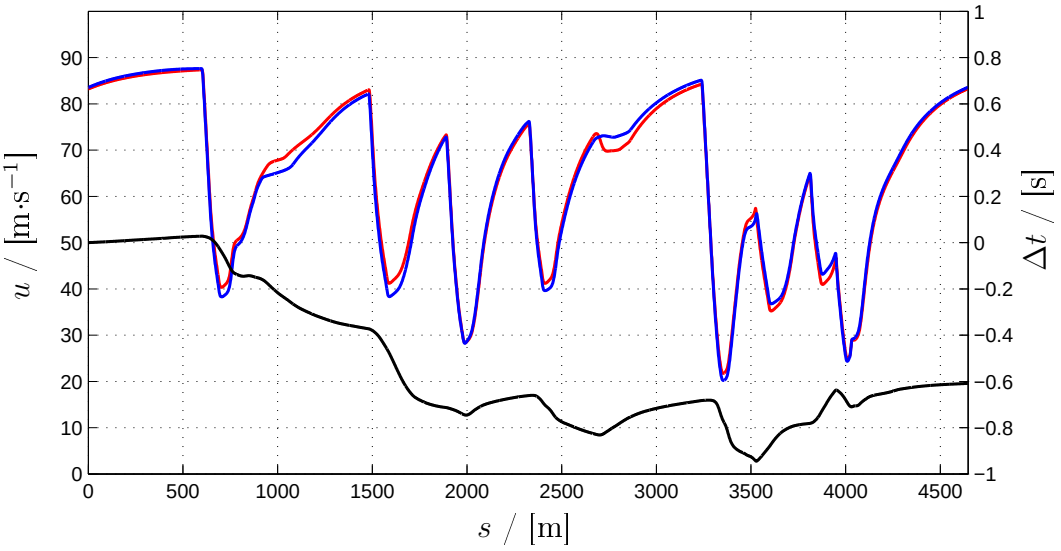

Figure 6.6: Speeds and time difference for the 2D and 3D track. The red curve gives the speed of the car on the flat track, while the blue curve is the speed on the 3D track. The black curve is the time difference; positive values indicate that the car is faster on the 3D track. The left-hand y-axis gives the speed, while the right-hand y-axis refers to the time gained.

car is going over the brow of a hill, it experiences adverse camber for the majority of the corner and the centripetal force acts to undermine the aerodynamic downforce. In this case, however, this combination of influences has little effect on the racing line taken.

This study ends with a more detailed analysis of the effect of gravitational, centripetal and camber changes on the vehicle's top speed. Figure 6.8 shows the speed variations due to three-dimensional influences and the longitudinal gravitational force as a function of distance, while the 'centripetal downforce' is illustrated in Figure 6.9. Three track sections are now analysed, with others open to a similar study should the reader wish to undertake one.

The first section is the down-hill straight from turn (16) to turn (1). The average speed difference on this section is approximately 0.5 m/s. Camber and centripetal force influences appear relatively unimportant on this section of the track. Since the longitudinal gravitation force component is of the order 100 N, which at the end of the straight is acting on the car at approximately 85 m/s, this gradient is equivalent to an increase in the engine power of up to approximately 8.5 kW. An easy calculation using (5.27) shows

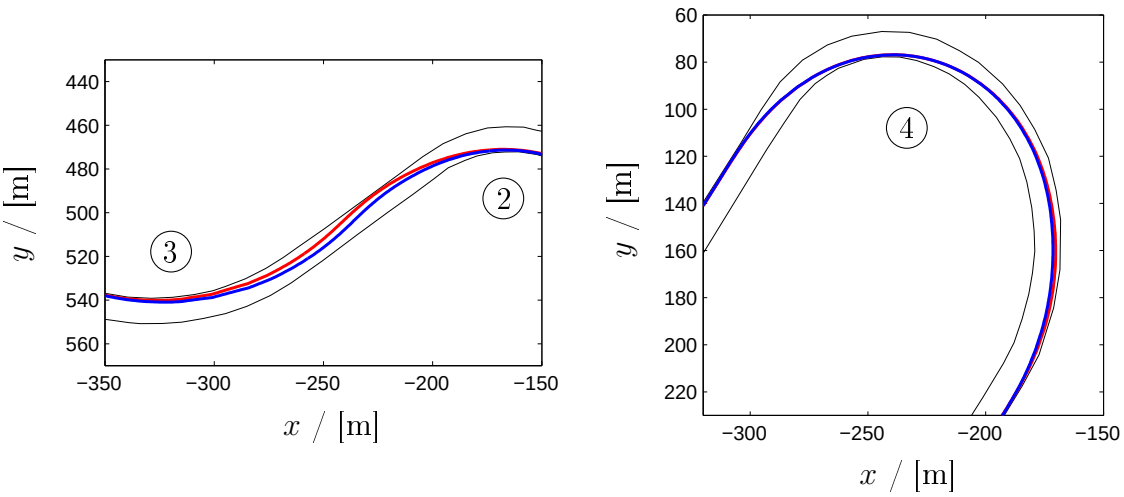

Figure 6.7: Changes in the racing line between turns (2) and (3) (on the left) and turn (4) (on the right) due to three-dimensional track features. The trajectory for the flat track is shown in red, while the trajectory for the 3D track is shown in blue.

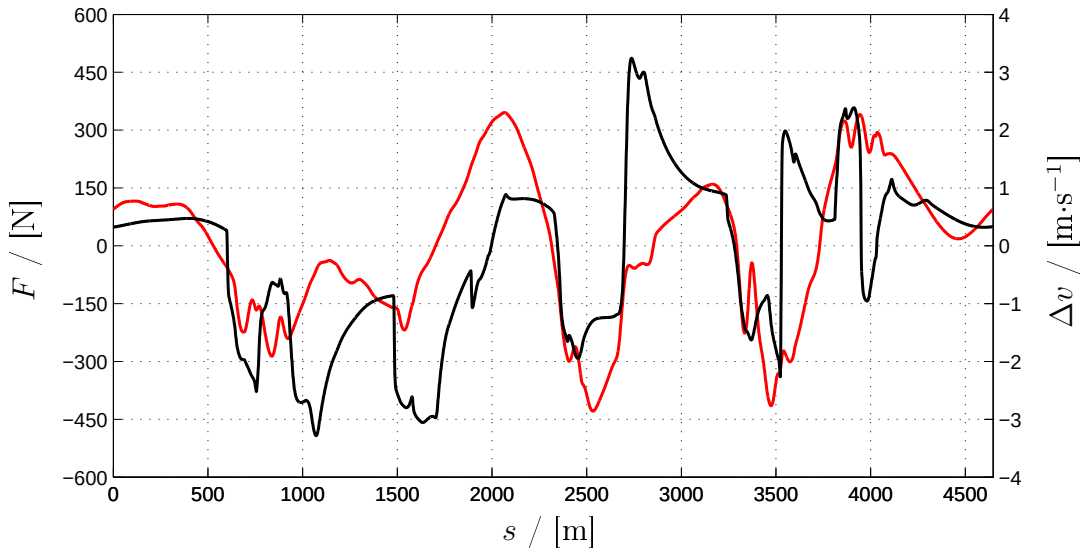

Figure 6.8: Speed difference Δv (in black) and the gravitational force component in the car's x_B-direction F (in red). The speed difference is positive when the car is faster on the 3D track. The car is travelling down hill when the gravitational force component is positive.

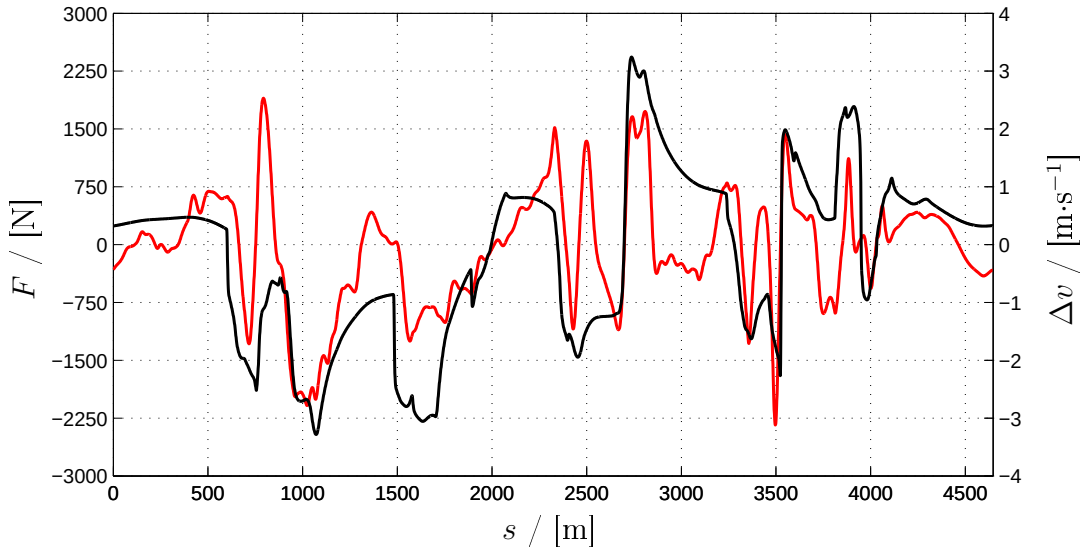

Figure 6.9: Speed difference Δv (in black) and the centripetal force in the car's z-direction F (in red). The speed difference is positive when the car is faster on the 3D track. The centripetal force provides additional downforce when it is positive.

that the peak engine power of 560 Kw is capable of supporting a top speed of 85.4 m/s against aerodynamic drag. If the engine power is increased to 569 kW the top speed of the car is limited by aerodynamic drag to 85.9 m/s, which explains this speed difference. The second track section studied is the long turn (3) from 900 m to 1300 m. The track elevation increases by approximately 10 m over this section and so lower speeds can be expected. In addition to this, there is a significant stretch of adverse camber, which causes a downforce reduction due to centripetal forces (see Figure 6.9). Track three-dimensionality therefore causes the car to lose 0.2 s in this section. As is evident from Figure 6.8, there are significant differences between the three-dimensional and flat-road car speeds between corners (7) and (10). These speed differences are both positive and negative over this section. Before analysing the reasons for these variabilities, it is worth recalling that since $\Delta v \propto \int \Delta F dt$, and so speed changes are a lagging indicator of force changes. It is clear that the car on the 3D track performs relatively poorly on the exit of turn (7). Figure 6.8 shows that the track has a positive gradient here, which is a first reason for the slower exit speed from (7). As can be seen in Figure 6.5 the track also has an adverse camber over this section; a further reason for a reduced exit speed from turn (7). The downforce

reduction due to adverse camber is evident from Figure 6.9. This reduced exit speed is carried all the way to turn (9). Fortunes are reversed at the exit of turn (9), where gravitational effects appear to be neutral. Figure 6.9 shows that significantly improved centripetal downforce occurs for approximately 100 m through the corner. This is again a consequence of sustained positive camber (see Figure 6.5). This combination of advantageous circumstances leads to an increased exit speed from turn (9), which is then carried to turn (10), thereby reversing the earlier setback.

The interchange between potential and kinetic energy is noticeable considering variations in the vehicle speed through inclined sections; however, it cannot be directly exploited to minimise the manoeuvring time. In the brachistochrone-like problems, in Chapter 3, the rate of the energy exchange was controlled by varying the descent angle; in lap-time simulations such an angle is restrained by the geometry of the track and can be varied only marginally, limiting the ability to convert gravitational potential energy into speed.

6.3 Energy-Recovery Systems

The prohibitively high cost of engine development, in combination with a drive towards a 'greener' sport, resulted in the introduction of kinetic and thermal energy-recovery devices into Formula 1. The underlying concept is that energy recovered during braking, and/or from the engine exhaust gases, should be redeployed to improve both the fuel efficiency and the lap times. Hopefully the continued development of energy-recovery systems of this type will produce cost-effective gains that are also relevant to road cars. The optimal control of these systems is studied here by recognising the power transfer and energy storage restrictions imposed by the F1 technical regulations in [50, 51].

6.3.1 Kinetic Energy Recovery

Contemporary Kinetic Energy-Recovery Systems (KERS) facilitate the capture of kinetic energy that derives from braking the moving vehicle. The recovered energy can be stored

for later use in propelling the car. Referring to Figure 6.10, one can appreciate that a

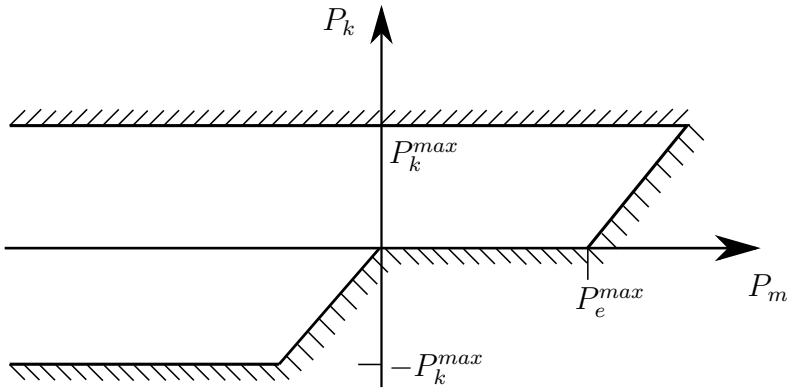

Figure 6.10: Operating regime of a KERS. The ordinate represents the KERS power flow P_k, while the abscissa represents the power P_m delivered to the rear wheels.

vehicle without KERS can operate only along the horizontal axis on the left of P_e^{max}, the engine's peak power limit. The left-hand limit will be dictated by the braking capacity of the tyres. With the introduction of KERS the vehicle can operate within the cross-hatched region; according to the Formula 1 2013 Technical Regulations [50] the KERS power limits are given by $P_k^{max} = 60\,\text{kW}$, while the energy released from the energy storage system may not exceed 400 kJ in any one lap. In order to enforce these restrictions the following constraints are introduced:

$$P_m - P_k \leq P_e^{max}, \tag{6.45}$$

$$u_{-2}(-P_m) + P_k \geq 0, \tag{6.46}$$

in which $u_{-2}(\cdot)$ is the ramp function. It is also necessary to enforce the KERS power limit $-P_k^{max} \leq P_k \leq P_k^{max}$ and the per-lap energy-store discharge limit $\int_{lap} u_{-2}(P_k)ds \leq 400\,\text{kJ}$ required in the regulations [50]; $\int_{lap}(\cdot)ds$ are line integrals carried out around a lap of the circuit. In a standard race lap it must be ensured that all the energy released from the KERS is recovered during braking, so that the lap is repeatable. A cyclic constraint on the stored energy $\int_0^s P_k ds = 0\,\text{kJ}$ is imposed to this end.

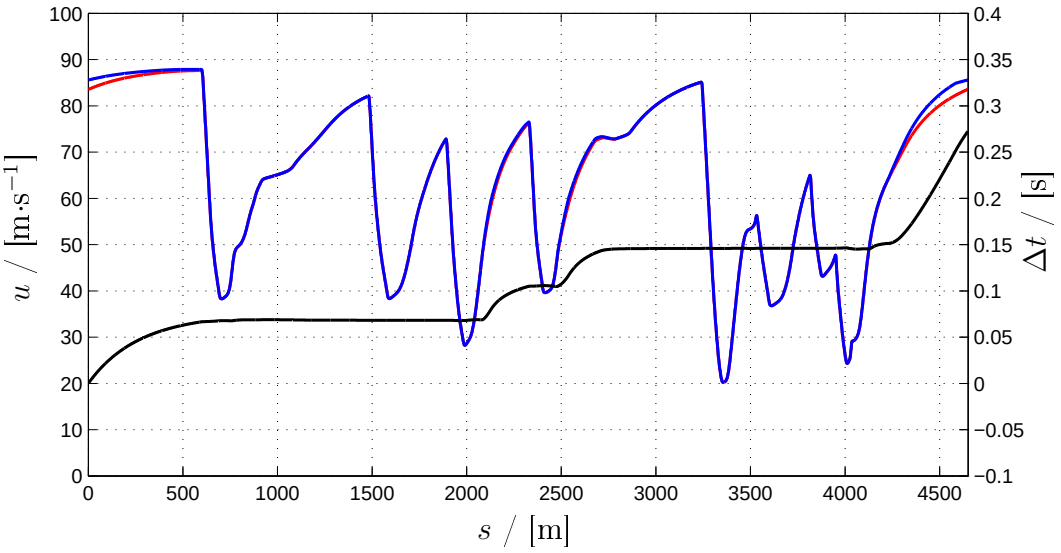

Figure 6.11: Speeds and time difference for the vehicle with and without KERS. The red curve gives the speed of the car with no KERS, while the blue curve is the speed using KERS. The (solid) black curve is the time difference; positive values indicate that the car is faster with KERS. The left-hand y-axis gives the speed, while the right-hand y-axis refers to the time gained.

6.3.1.1 Results

The minimum-lap-time problem was solved with an optimal lap time of 78.423 s achieved. This is 0.877 s faster than the time achieved without KERS. The KERS power and energy usage is shown in Figure 6.12. It is clear from this figure that the KERS operates against the allowable power limits using a bang-bang type strategy[2]. Figure 6.12 shows that the KERS is being deployed after turns (5) and (7), which are relatively slow corners followed by sustained high-acceleration phases. Most of the KERS energy is used on the entry to the long main straight, where a speed advantage gained early in this section is carried all the way to turn (1), resulting in a significant lap time reduction. During periods of rapid

[2] Optimal-control problems with cost and dynamics that are linear in the control often involve switching strategies. In these problems the control takes on limit values dictated by the sign of $\partial H/\partial u$, where H is the control Hamiltonian (see Section 2.2); this is the well-known *bang-bang principle* [37]. In the case that $\partial H/\partial u = 0$ for a finite time, the problem has a *singular arc*, during which the value of the control may be arbitrary, and a bang-singular-bang strategy may result. When solutions are sought numerically, a singular arc can generate oscillations in the iterative update of the solution estimate. These oscillations usually degrade the NLP convergence. In order to avoid these issues one can redefine the optimal-control problem for a slightly modified cost, that negates the existence of singular arcs without significantly changing the problem structure. Modifications of this type proved to be useful to enhance the convergence for the problems in Section 6.3.2, where the terms $10^{-5} \cdot w_g$ and $10^{-5} \cdot r_f$ are added to the cost integrand.

deceleration the KERS is recharged, which is evident on the entry to most corners. The optimal strategy is to maximize the stored energy on the entry to turn (14) in preparation for a period of prolonged KERS usage out of turn (16). The KERS per-lap discharge limit is an active constraint which is met at the start-finish line; this constraint restricts the KERS power usage to the three full-power pulses shown in Figure 6.12.

Figure 6.11 illustrates the speed differences and the time gained by the vehicle provided with KERS. Comparing Figures 6.11 and 6.12 shows how stored energy is traded for lap time. The time gained through the acceleration sections following KERS deployment is each time proportional to the energy spent.

Influences of three-dimensional track modelling appear again considering the optimal control of a vehicle equipped with KERS. Figure 6.13 shows the optimal power and energy usage on the plane projection of the Circuit de Catalunya; in this case the use of KERS is extended through the main straight and reduced after turn (7). The reason for this change in strategy comes from the lower speed through turn (9) in the 2D optimal lap, which would cancel out the speed advantage provided by deploying the KERS after turn (7).

The controlled interchange between potential and kinetic energy, which was a recurring theme in Chapter 3, appears again for a car equipped with an energy store. A key difference is that the potential energy is now extracted from (and stored into) an on-board device (typically a battery), as opposed to elevation changes in a gravitational field. Kinetic energy is converted into stored potential energy using a motor-generator unit when the vehicle speed is limited by the geometry of the track and thereafter extracted from the store when opportune.

6.3.2 Kinetic and Thermal Energy Recovery

The energy-recovery rules for the 2014 Formula 1 season are more complex than those used previously, with the optimal-control problem correspondingly more intricate. The key differences are hereby listed.

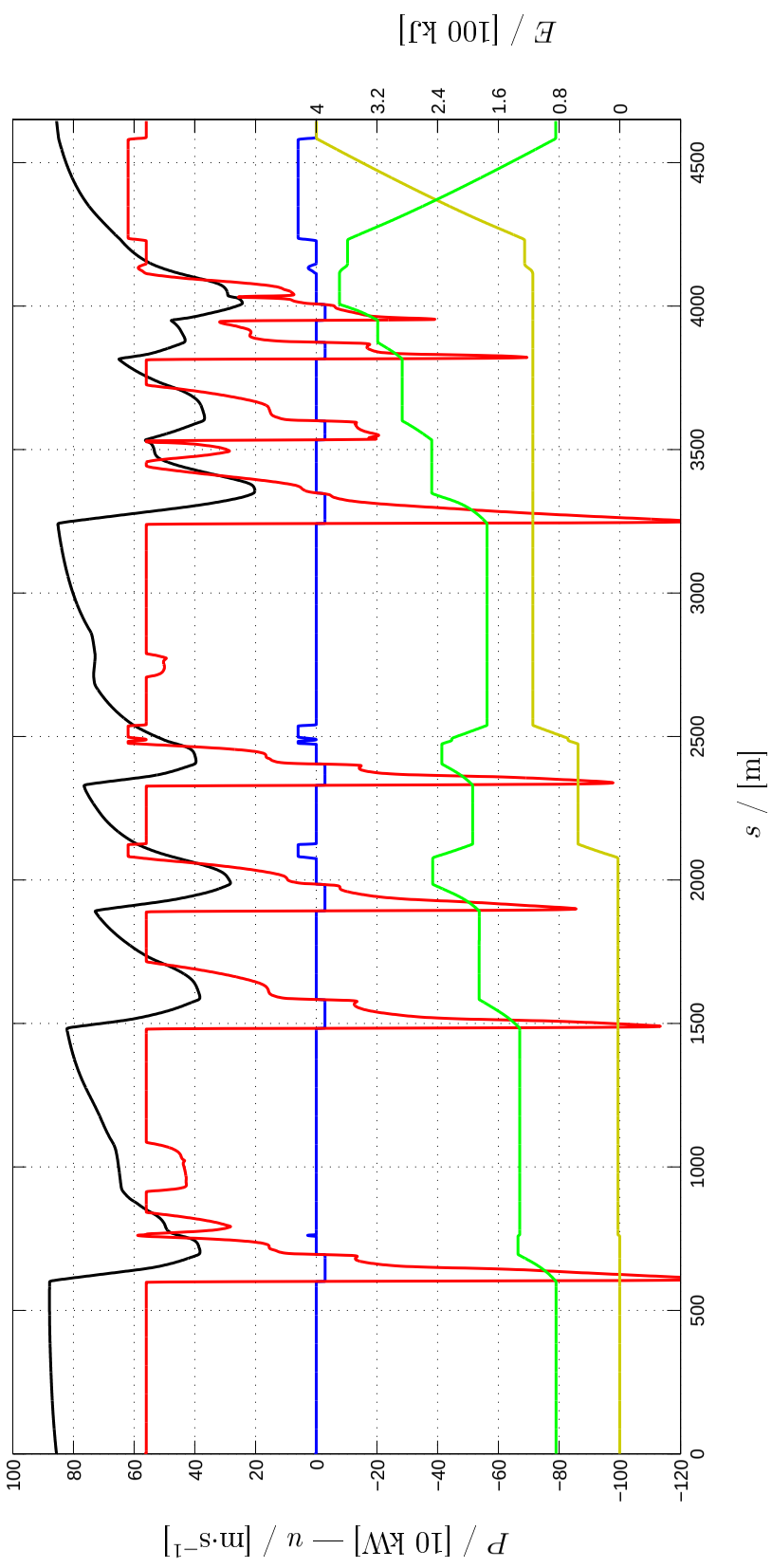

Figure 6.12: KERS optimal control for a race lap on the three-dimensional Circuit de Catalunya. The black curve shows the vehicle's speed; the red curve is the power delivered by the rear wheels; the blue curve is the power to/from the KERS (the KERS is propelling for positive values of power); the green curve is the stored energy; the yellow curve is the energy released from the KERS. The units on the left-hand side y-axis refer to speed and power; energy values appear on the right-hand side.

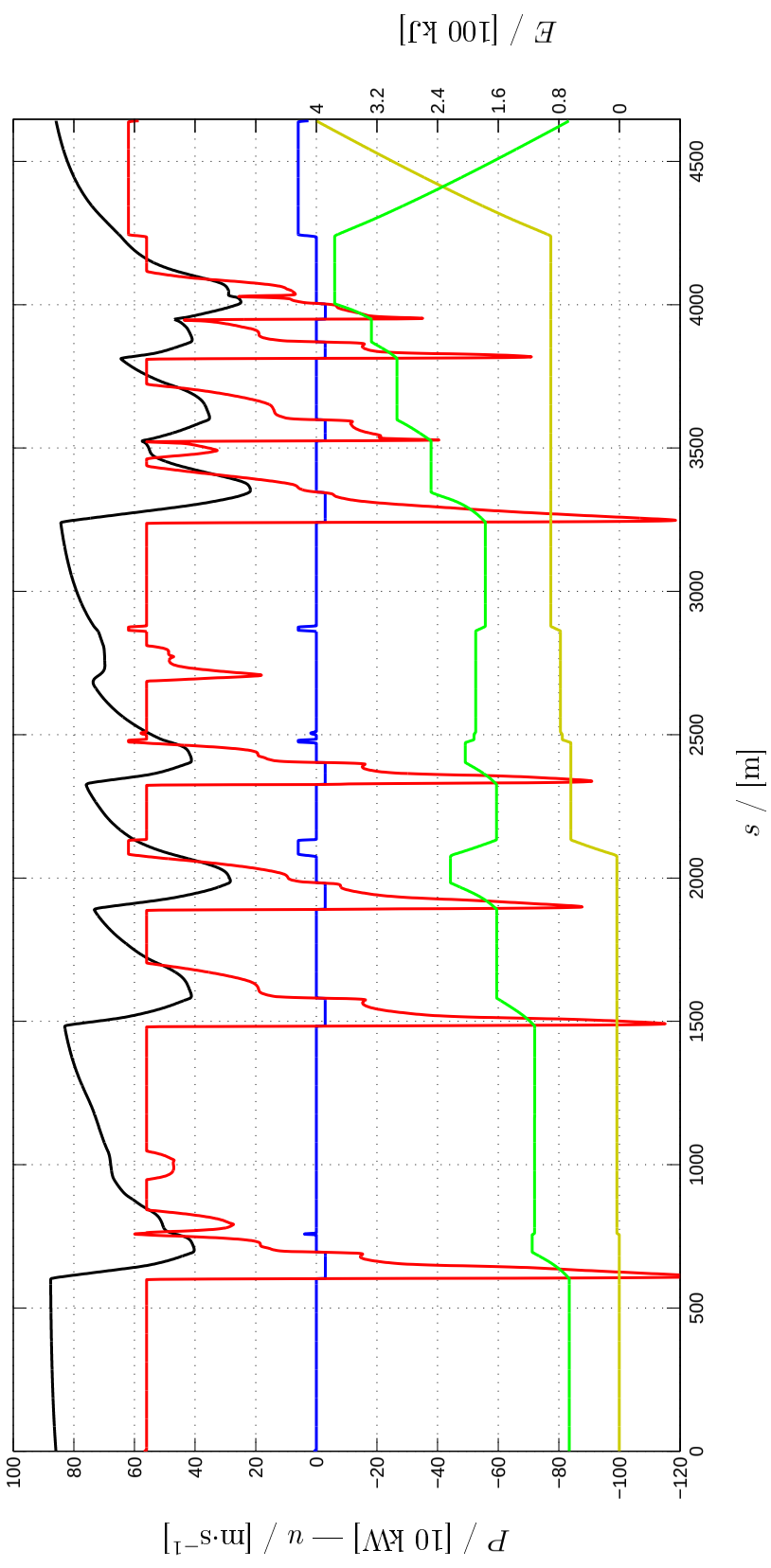

Figure 6.13: KERS optimal control for a race lap on the plane projection of the Circuit de Catalunya. The notation is consistent with Figure 6.12

1. The car is restricted to a maximum 100 kg of fuel per race.

2. The fuel mass flow rate must not exceed 0.028 kg/s (100 kg/hour).

3. Turbocompounded engines (see [49]) have been introduced that include a turbine, compressor and motor-generator on a common drive shaft; this motor-generator is referred to as the Motor-Generator-Unit-Heat (MGU-H). The turbine is bypassed by a controllable waste gate; a closed waste gate increases the power generated by the MGU-H, but decreases the power generated by the engine. There is no limit on the amount of energy than can be recovered from the exhaust gases. The operation of the thermal ERS is summarised in Figure 6.14 that shows a plot of power from the Internal-Combustion Engine (ICE) P_e versus power supplied by the MGU-H, denoted P_h. Here it is assumed that the inlet manifold pressure is set, and thus the use of the compressor is dictated by the waste gate opening. When the waste gate is open the MGU-H requires 60 kW to operate the compressor and the ICE output power is boosted by 20 kW. When the waste gate is closed 40 kW is recovered from the MGU-H and the ICE output power drops to approximately 440 kW in this case. All the figures given here are representative values only.

4. An electric motor/generator unit (MGU-K) is coupled directly to the engine crankshaft. As before, the MGU-K can be used to recover vehicular kinetic energy during braking, or boost the power to the rear wheels during high-speed operation.

5. The car's Energy Store (ES) can store up to 4 MJ.

6. The ES can accept up to 2 MJ per lap from the MGU-K.

7. The MGU-K can draw up to 4 MJ per lap from the ES.

8. Power flow to and from the MGU-K is restricted to ± 120 kW.

9. The power and energy to and from the MGU-H are unrestricted (see Appendix 3 in [51]).

It is clear that operating the engine with an open waste gate is very inefficient from an energy-conservation point of view. The engine control variables are the fuel delivery rate and the waste-gate opening.

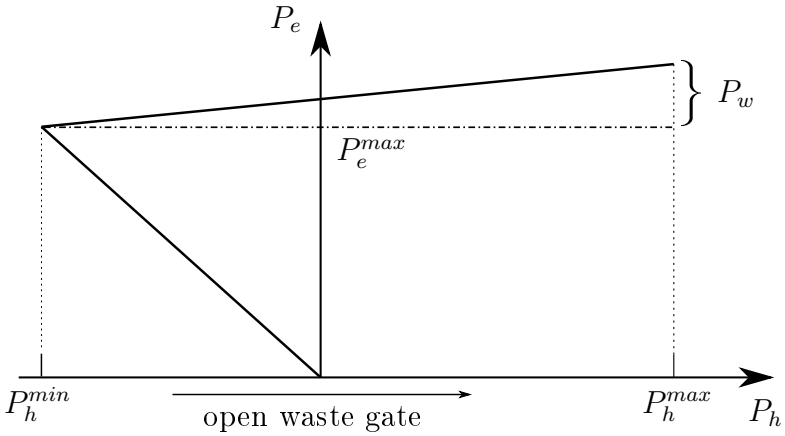

Figure 6.14: Operating regime of the 2014 engine and thermal ERS at full power; power absorbed by the MGU-H is deemed positive.

It is now explained how these various constraints are set up in the optimal-control calculation. The power available to the rear wheels is constrained by the following inequality:

$$P_m - (P_e^{max} + P_w w_g) r_f - P_k \leq 0. \tag{6.47}$$

The P_e^{max} term in equation (6.47) represents the power generated by the ICE under full fuelling when the waste gate is closed ($P_e^{max} = 440\,\text{kW}$); $r_f \in [0, 1]$ is the normalized fuel mass flow rate. The $P_w w_g$ term in (6.47) represents the power boost resulting from opening the waste gate; $w_g \in [0, 1]$ is the waste-gate control, with $P_w = 20\,\text{kW}$. P_k is the power delivered to the MGU-K (P_k is positive when the MGU-K is motoring) and P_m represents the mechanical power delivered to the rear wheels (positive values accelerate the car). The power generated/absorbed by the MGU-H is given by

$$P_h = \big((P_h^{max} - P_h^{min}) w_g + P_h^{min}\big) r_f, \tag{6.48}$$

and is a function of the waste-gate opening and the fuel flow rate; $P_h^{min} = -40$ kW and $P_h^{max} = 60$ kW.

In order to monitor the resource constraints associated with fuel usage, ES usage and MGU-K usage, four auxiliary state variables are introduced. The state

$$m_f = \int r_f^{max} r_f ds \tag{6.49}$$

is used to monitor the fuel consumption and is constrained by $m_f \leq 100/n_{laps}\,\text{kg}$, in which n_{laps} is the number of laps in the race. As pointed out above, r_f is normalised to the interval [0, 1], with $r_f^{max} = 0.028\,\text{kg/s}$ the maximum fuel rate allowed by the regulations. The dynamics of the second auxiliary state are given by

$$\dot{E}_s = -\left(P_h + P_k\right). \tag{6.50}$$

This state is used to monitor the stored energy and is constrained by $0 \leq E_s \leq 4\,\text{MJ}$ throughout the whole lap. The dynamics of the third auxiliary state are described by

$$\dot{E}_{ES2K} = \begin{cases} P_k & \text{if } P_k > 0,\ P_h > 0 \\ P_k + P_h & \text{if } P_k > -P_h > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{6.51}$$

This state is used to monitor the energy supplied to the MGU-K from the energy store and is constrained by $0 \leq E_{ES2K} \leq 4$ MJ. The first alternative corresponds to the case when both electrical machines are running as motors. In this case it is only the MGU-K energy usage that is 'taxed'. The second alternative corresponds to the case in which the MGU-K is motoring, but the MGU-H is generating. In this case the power generated by the MGU-H is off set against the MGU-K power requirements, with the difference attracting a penalty. The fourth auxiliary state is used to tax energy flows from the MGU-K into the energy store and is described by

$$\dot{E}_{K2ES} = \begin{cases} P_k & \text{if } P_k < 0,\ P_h < 0 \\ P_k + P_h & \text{if } 0 > -P_h > P_k \\ 0 & \text{otherwise.} \end{cases} \tag{6.52}$$

This state is constrained by $-2\,\text{MJ} \leq E_{ES2K} \leq 0$. The first alternative corresponds to the case when both electrical machines are operating as generators. In this case it is only the MGU-K energy generation that is penalised. The second alternative corresponds to the case in which the MGU-K is generating, while the MGU-H is motoring. In this case

the power required by the MGU-H is off set against the power generated by the MGU-K, with the difference penalised.

The conditional operators in constraints (6.51) and (6.52) are implemented using Heaviside step functions and ramp functions as follows:

$$\dot{E}_{ES2K} = u_{-2}(P_k)u_{-1}(P_h) + u_{-2}(P_k + P_h)u_{-1}(-P_h), \tag{6.53}$$

$$\dot{E}_{K2ES} = -u_{-2}(-P_k)u_{-1}(-P_h) - u_{-2}(-P_k - P_h)u_{-1}(P_h). \tag{6.54}$$

As before, differentiable approximations to non-smooth functions are used to ensure differentiability of (6.53) and (6.54). Inequality constraint (6.47), in combination with the four auxiliary states (6.49) to (6.52) and their associated box constraints are used to model the power and energy constraints required by the 2014 Formula 1 rules [51].

6.3.2.1 Results

The minimum-lap-time problem for the 2014 car is solved, resulting in an optimal lap time of 80.521 s, which is 2.098 s slower than the lap time achieved with the car studied in Section 6.3.1. Contributing to this downgraded performance is the fact that the 2014 car is 30 kg heavier than its predecessor and its internal combustion engine is 120 kW less powerful. The sensitivity with respect to variations in vehicle mass, of 0.036 s/kg estimated in Section 5.3.3, would cut the lap time to 79.441 s if the minimum mass allowed for a car were 30 kg lower. These results suggest that contemporary car performances can almost be achieved on two-thirds of the current fuel usage.

The performance analysis of the 2014 car must take into account the power allocation amongst the three alternative power units, which include the ICE and the electric machines, as well as the management of recoverable energy and fuel budget. These quantities are illustrated in Figure 6.15 for an optimal race lap.

The first thing to observe from the dashed-black, orange and yellow traces is that an optimal lap is fully utilising the fuel (100 kg / 66 laps = 1.515 kg) and the MGU-K generation allowances (2 MJ), but not the MGU-K motoring allowance (4 MJ). Not surprisingly there are braking periods when no fuel is being used; this occurs for example

on the entries to turns (1), (4), (5) and (10). As expected, the MGU-K is used as a motor to increase the vehicle speed on the fast straights between turns (16) and (1), turns (3) and (4), and between turns (9) and (10). The MGU-K is instead recovering kinetic energy to recharge the ES during braking sections, that overlap with the periods of zero fuel usage; see for example the sections entering turns (1), (4) and (5). There is also intermittent energy recovery occurring on the slower sections, where the available engine power is larger than what can be delivered to the wheels, due to limited tyre friction; this happens for example in turn (3) or between turns (10) and (15). Notable is the fact that the ES is under-utilised on an optimal lap of the Barcelona circuit; indeed only 1.12 MJ of its 4 MJ capacity is being used. This is a result of restrictions on the total amount of MGU-K generation and motoring allowed, and by the fact that the state of charge is constrained to be equal at the beginning and at the end of the lap. It is likely that greater use of the ES will occur on sub-optimal laps involving the overtaking of other vehicles. The blue trace in Figure 6.15 reveals a liberal use of the MGU-K to boost the total drive power available. A bang-bang-type strategy is again being employed by the MGU-K during motoring. The MGU-K is being used up to its power limits to recharge the energy store on the entry to corners; several particular instances of this are discussed above. The waste gate is essentially unused due to its poor energy efficiency. A new behaviour appears in the optimal-control strategy which was not seen for the vehicle equipped with smaller ERS units (see Section 6.3.1): the MGU-K is used as a generator for a short time interval before the main braking phase at the entry of any corner. With this aggressive energy-recovery strategy three phases can be identified when the vehicle decelerates before a corner:

1. while the engine is still at full throttle, the MGU-K switches from full to no power (MGU-K drop);

2. the fuel rate is set to zero and the MGU-K starts regenerating kinetic energy at the maximum allowed power (regenerative braking);

3. the MGU-K keeps regenerating energy at full power and the friction brakes are engaged (friction braking).

A secondary effect of this new braking policy is a 'smoother' speed profile compared to prior calculations.

Figure 6.16 considers the case of a qualifying lap. The fundamental difference between this case and the one considered in Figure 6.15 is the freedom to start the lap with a fully charged energy store, and to then end it with the energy store fully discharged. The qualifying lap is completed in 80.161 s, which is 0.360 s faster than an optimal race lap (considering the same vehicle mass). If the difference in vehicle mass is taken into account applying the 0.036 s/kg rule, the qualifying lap for the 2014 vehicle is only 0.658 s slower than the 2013 vehicle. In this case the fuel and the MGU-K motoring-energy quotas are fully utilised, and the energy regenerated by the MGU-K is close to the allowance. In its qualifying configuration the engine is run with the waste gate open for sustained periods of time when maximum engine power is needed. During these intervals the energy store is supplying both the MGU-K and the MGU-H, with the latter used to drive the engine boost compressor. The main benefit of using the MGU-K in combination with an open waste gate is a 20 kW power gain, that improves the vehicle acceleration especially through the fast sections between turns (16) and (1), and between turns (9) and (10). Unlike the racing lap, the car is simultaneously using the MGU-K and a fully open waste gate. This extravagant use of the waste gate derives from the ability to simply 'run down' the energy store on a qualifying lap. In contrast to the racing lap, the waste gate is typically open when the engine is being fully fuelled. On the entry to turns (1), (4), (7) and (10) the waste gate is being closed before cutting the MGU-K and the fuel. This strategy is necessary in order to save power, as the ES is fully drained at the end of the lap. During phases of full throttle and closed waste gate, the MGU-K can drive the car with 40 kW of power that are being regenerated by the MGU-H at the price of 20 kW that could be potentially obtained opening the waste gate; this 20 kW difference is in a sense 'free' power that is not accounted for in the energy-flow-limit calculations, as the power flows

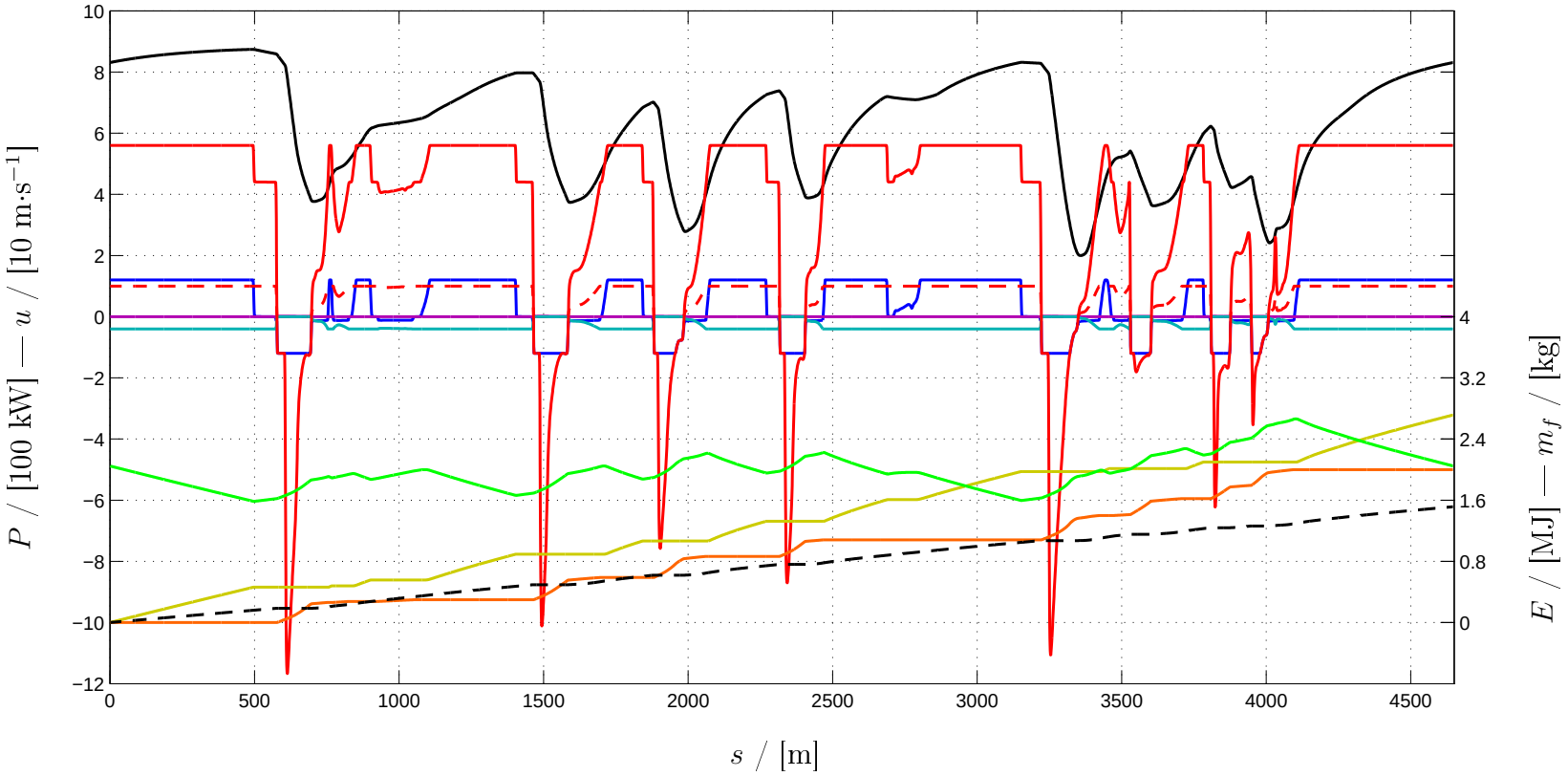

Figure 6.15: Kinetic and Thermal ERS optimal control for a race lap on the three-dimensional Circuit de Catalunya. The solid black curve shows the vehicle's speed; the solid red curve is the power delivered by the rear wheels; the blue curve is the power to/from the MGU-K (the MGU-K is propelling for positive values of power); the cyan curve is the power to/from the MGU-H (the MGU-H generates energy for negative values of power); the purple line represents the state of the waste gate (1 is open); the green curve is the stored energy; the yellow curve is the energy transferred from the energy store to the MGU-K, while the orange curve is the energy transferred from the MGU-K to the energy store; the dashed black curve is the mass of fuel consumed; the dashed red curve is the normalised fuel rate. The units on the left-hand side y-axis refer to speed, power, waste-gate opening and normalised fuel rate; energy values and fuel mass appear on the right-hand side.

Figure 6.16: Kinetic and Thermal ERS optimal control for a qualifying lap on the three-dimensional Circuit de Catalunya. The notation is equivalent to the one in Figure 6.15

from the MGU-H directly to the MGU-K. As usual the MGU-K is used regeneratively on the entry to corners when possible, although its generation quota is not fully utilised. It is noteworthy that the fuel burnt through a qualifying lap is approximately equivalent to the race case; in fact the qualifying vehicle uses less fuel: 1.504 kg instead of 1.512 kg. This is possible because 4 MJ of energy are drawn at a high efficiency from the ES in this case, whereas a cyclic constraint did not allow any net energy extraction from the ES throughout a racing lap. The take-home message of this last section is that current vehicles, propelled by ICEs only, can be replaced by vehicles equipped with downsized engines and well-designed hybrid powertrains, without incurring in a significant loss of performance. The remarkable benefit is a reduction in fuel consumption of approximately one-third of the current levels.

6.4 Summary

Previous work on the minimum-lap-time optimal control of a Formula 1 car has been generalised to the case of speed-dependent aerodynamic coefficients, three-dimensional tracks and energy-recovery systems. The vehicle dynamics are modelled under the assumption that the car moves on a plane that is tangent to the road at its spine. Since the tangent plane under the car can pitch, roll and heave, several atypical dynamic influences result, that include:

1. a gravitational force vector that moves in the car's reference frame; this motion produces lateral and longitudinal gravitational force components that can either supplement, or oppose the engine thrust, and that can either augment or undermine the tyre side forces;

2. new centripetal forces that act along all three of the car's body-fixed axes, produced as a consequence of road humps, dips and camber; the most important influence appears in the large vertical component that can either augment or oppose the aerodynamic downforce;

3. road-related gyroscopic moments that affect the lateral and longitudinal load balances; under most circumstances these gyroscopic effects are found to be small.

As an illustrative example, the minimum-lap-time performance of a Formula 1 car on the Barcelona circuit is studied. The study shows that three-dimensional effects can significantly alter the car's maximum speed. The effects of three-dimensional features may vary considerably between tracks. Given that three-dimensional influences are relatively easy to include in lap-time-simulation studies, the errors associated with the 'flat-track' assumption can now be addressed.

The last vehicle extensions in this chapter focus on the control of energy-recovery systems in Formula 1 cars, where the emphasis is on minimising the lap time. This minimum-time nature of the performance index leads to the use of bang-bang strategies during KERS-assisted driving. A strategy reminiscent of bang-bang control is also employed during braking, although regenerative braking is limited by the braking power available. Following the analysis of contemporary KERS, attention is devoted to the control issues relating to the 2014 ERS, that admits both kinetic and thermal energy recovery. Unlike the conventional KERS, the upcoming ERS has an explicit fuel-saving role. It is demonstrated that the ERS in combination with lower-powered turbo-boosted engines can provide contemporary levels of lap-time performance with approximately 100 kg (rather than the current 150 kg) of fuel per race. It is also shown that the ICE and the ERS have to be operated in a very different manner during qualifying, when the profligate use of stored electrical energy is allowed. The potential for redeploying these technologies in every-day road cars is self-evident.

CHAPTER 7

Conclusions

THIS thesis discusses the application of numerical optimal-control methods to a range of vehicular systems. A selected review of the optimal-control theory and of related numerical methods motivated the choice of direct collocation for vehicle-control applications. A simple calculation shows that the optimality conditions in direct and indirect collocation are equivalent for a given numerical discretisation rule.

Elementary mechanical systems were studied in order to evaluate the importance of fundamental dynamic effects in problems with a more complicated structure. In particular, the interchange between kinetic and potential energy in a constant gravitational field was found to be crucial. A newly formulated extension of the brachistochrone problem was solved exactly using arguments from optimal-control theory and considered as a validation benchmark for numerical methods.

A variety of minimum-lap-time problems were solved for a Formula 1 car. The vehicle model includes: saturated tyre friction, with load dependency and anisotropic properties for combined longitudinal and lateral slip; aerodynamic forces in the longitudinal and vertical directions, quadratic in the vehicle speed and with a specified application point; a viscous differential, and other nonlinear features. An important aspect of this work is that the full nonlinear dynamic model is used for all the optimal-control calculations, without resorting to model approximation techniques such as quasi-steady-state assumptions or linearisation.

A series of numerical optimal-control simulations were carried out considering a real F1 racing circuit with varying sampling distances (down to less than a quarter of a metre), which showed that these calculations can be executed in acceptable time scales on standard desktop computers. These results dispelled the common belief that minimum-lap-time calculations of this type are not practical due to unreasonably long execution times. Equally under-appreciated was the possible optimisation of vehicle parameters based on minimum-lap-time calculations; the well-established iterative-search approach to vehicle tuning diverted the mainstream interest from these techniques, favouring the adoption of simplifying approximations including quasi-steady-state analysis or model linearisation. It was proved instead that vehicle parameters can be optimised either independently, or in concert with a small price to be paid in terms of problem size and solution speed of the minimum-lap-time calculations; significant lap-time gains are obtained in return.

In practice, a numerical optimal-control algorithm can be compared to the learning process that occurs when a driver tries to minimise the lap time by repeatedly racing around a track; the driven line, the braking strategy and possibly the vehicle setup are incrementally adjusted, often involving a discussion with the race engineers. Every iteration of the nonlinear-programming algorithm applied to the transcribed optimal-control problem is analogous to a driven lap. Computer simulations are cheaper and less prone to human error than running standard practice sessions. The potential for savings in terms of time and resources dedicated to vehicle tuning is evident.

The minimum-lap-time problem was extended to account for additional modelling features not yet considered in the literature available to-date: these include three-dimensional tracks with consistently updated vehicle's equations of motion, and energy-recovery-systems accompanied by the dynamics of the energy storage. The extension to 3D tracks required the generalisation of a pre-existing road modelling framework based on curvilinear coordinates, from the plane to three-dimensional Euclidean space. This generalisation was neatly achieved using basic concepts from differential geometry, which allowed for the definition of a three-dimensional road model based on the geometrical concept of a rib-

bon. A full track description in three dimensions was made independent of the absolute characterisation in Cartesian coordinates, and was reduced to a differential description based on three generalised curvatures and two values for the width. This track modelling approach is key to the successful application of optimal control to vehicle-racing problems.

Numerical investigations referred to an existing Formula 1 racing track: the Circuit de Catalunya, in Spain. The circuit parametrisation was extracted from GPS data relating to the road boundaries. Obtaining an accurate and smooth model of the track from noisy data represented a technical challenge. Optimal-control techniques were employed to solve the track-smoothing problem, avoiding issues such as circuit closure and guaranteeing the differentiability of all the track attributes appearing in the minimum-lap-time calculations. The numerical simulations showed that the effects of track three-dimensionality are not negligible, especially considering inertial forces acting in the vertical direction of the car, caused by centripetal accelerations, and gravitational effects in the longitudinal and lateral vehicle's directions, due to the inclination of the road surface.

New models were proposed for the energy-recovery systems prescribed by the 2013 and 2014 Formula 1 technical regulations. Formulations suitable for optimal-control purposes were obtained by employing smooth approximations of the non-smooth functions necessary to characterise the switching nature of the new systems. Minimum-lap-time calculations showed that future Formula 1 cars with downsized engines and improved energy-recovery systems can attain performances comparable to current vehicles, while consuming two-thirds of the fuel.

Although the calculations presented in this thesis are specialised to Formula 1 control, they are directly relevant to standard road vehicles. It was proven that hybrid technologies represent a feasible alternative to vehicles powered by combustion engines only, without degrading the performance perceived by users.

CHAPTER 8

Proposed Developments

8.1 Parallel Computing

The biggest obstacle to the use of nonlinear optimal-control techniques in vehicular applications is the high computational burden associated with numerical solutions. A combination of poorly conceived models and inappropriately applied multi-purpose numerical techniques can easily become the ingredients of inefficient and lengthy computer simulations. This thesis proved that an adequate choice of methods and models can dramatically reduce solution times, or even allow for the solution of problems previously considered insoluble. There is certainly a margin for improvement in the implementation of the numerical optimal-control methods. The review in Chapter 2 described multiple-shooting and collocation techniques used to partition the optimisation horizon into sub-intervals; a careful examination of these techniques reveals that dynamics and constraints within different sub-intervals are independent, and that the derivative matrices required by gradient-descent NLP techniques are sparse. Furthermore, full state and control discretisation allows one to evaluate dynamic, constraint and cost functions independently at each discrete sample. State-of-the-art general-purpose software can exploit the sparsity, but a lot of work can still be done in order to fully exploit the independence between samples; where parallel-computing facilities are available, the functions associated with each sample can be computed on an independent processing unit. A careful

implementation of this policy would cut down the function evaluation time, by a factor proportional to the minimum between the number of parallel processing units and the number of discretisation samples. A small level of parallelisation was achieved for the optimal-control calculations presented in this thesis, with the function evaluation speed trebled exploiting a 4-core CPU. The use of relatively cheap Graphics Processing Units (GPU), equipped with hundreds of parallel cores, represents a promising perspective for future developments of numerical optimal-control software.

8.2 Model Extensions

The vehicle models described in this thesis are relatively simple compared to high-fidelity models used for vehicle simulation in the automotive industry. The model development started from a simple steerable sleigh and was incrementally improved up to the current stage, with consistent care to compatibility issues with optimal-control techniques. Although some limitations were recognised, such as the presence of non-smooth modelling features, the eventual boundaries to the application of numerical optimal control are still unknown. There is certainly room for further model extensions in several directions, such as:

1. the inclusion of empirical maps, which may be efficiency or aerodynamic tables;
2. the addition of new dynamics, such as tyre temperature and wear, or tyre carcass and suspension dynamics;
3. the improvement of road models in terms of frictional or geometric features.

The use of tables in numerical optimal control represents a technical challenge. Table data is defined only at discrete points of the vehicle's operating workspace, and intermediate values are often generated by means of interpolation techniques. Standard linear interpolation does not provide second-order differentiability at the discrete points, which would degrade the NLP convergence. Higher-order polynomial interpolation methods might guarantee sufficient smoothness, but their extension to multi-dimensional tables

is not trivial. Attention must be given to the choice of interpolation methods that concurrently guarantee smoothness, accuracy and computational efficiency. One promising approach is based on radial-basis functions.

As discussed in Section 5.3.4, a series of dynamic effects are neglected by the current models. Before considering any of them for inclusion in the model, there is an important distinction to be made between 'slow' and 'fast' dynamics. Slow dynamics are the ones characterised by time constants much longer than any of the sampling intervals: these may include, but are not limited to, tyre thermal and wear dynamics, or fuel mass consumption. In general these dynamics can be added to the model without endangering the accuracy of the numerical integration schemes. In this context, dynamics are considered 'fast' if their characteristic time constants are shorter or comparable to the sampling time intervals: typical examples are the suspension dynamics, the tyre carcass dynamics, or the dynamics of a turbocharger. The accurate integration of these dynamics would require an increased sampling resolution and in some cases the use of higher-order numerical integration schemes. The effects of additional fast dynamics on the accuracy of the solution are currently unknown, and represent an interesting research direction. In both cases there is a risk of degrading the convergence of the solution, and any new model feature should be considered with care.

Optimal-control calculations applied to more accurate vehicle models are only meaningful if the road model is comparatively accurate. Changes in the road-tyre frictional interaction and road unevenness might be crucial to the performance of a vehicle. Future improvements in the track modelling should include variable friction coefficients to account for different conditions of the road surface. If the suspension dynamics were to be considered it would also be interesting to model the road surface profile on a short-distance scale, which would in turn require a finer identification of the track curvature parameters (see Section 4.1.3). However, all improvements to the track modelling will depend on the availability of road-related data. The identification procedure based on satellite imagery and GPS data may have to be supplemented with finer data collection

procedures.

8.3 Optimisation Studies

Starting from the models contained in this thesis there are still a range of vehicle-optimal-control problems to be considered. For instance it is possible to define problems where quantities currently assumed constant can be actively controlled. These calculations would assist the design of optimised vehicle sub-systems, such as controllable linked-suspension systems, variable brake force distribution systems or differentials with adjustable viscosity. It would be also interesting to study how to optimise the profiles of the aerodynamic parameters as functions of the vehicle state, e.g. its speed (see Section 6.1).

The results reported in Section 6.3.2.1 showed that only a fraction of the maximum allowed energy storage is required to achieve an optimal lap time. It is also shown that exploiting previously stored energy permits a reduced fuel usage with improved performance (considering the qualifying-lap results). Optimal-control calculations can be used to optimise the size of the energy storage, keeping in mind that current batteries cannot yet achieve the same specific energy as fossil fuel, and that the lap time is highly sensitive to changes in vehicle mass. In a similar direction of investigation, one can study the possibility to improve a race strategy by reducing the initial fuel load and optimally exploiting the hybrid powertrain.

8.4 Robustness and Feedback Control

The trajectories returned by the numerical optimal-control calculations represent optimal open-loop references. They are obtained under several modelling assumptions and simplifications, without considering any disturbance possibly coming from external agents. Typical sources of uncertainty include:

1. a vehicle model with a reduced number of degrees of freedom;

2. simplified aerodynamic and tyre-friction models;

3. neglected vehicle features, such as the suspension systems, gears or a flexible chassis;
4. approximate values for the vehicle parameters;
5. a smooth approximation of the road surface;
6. variations in the tyre-road friction characteristics throughout one lap;
7. environmental influences, such as wind, temperature or competing vehicles;
8. the driver's action.

It is extremely unlikely that the optimal trajectories generated with this deterministic approach can be exactly reproduced by real-world vehicles. In real racing conditions, a primary concern of the drivers is to avoid catastrophic accidents, and to achieve consistently low lap times for the duration of a race; in essence, their control strategy must be 'robust' with respect to model uncertainties and external disturbances.

The proposed formulation addresses the open-loop optimal control problem (generation of a reference trajectory), but does not explicitly consider issues relating to the robustness of a feedback control strategy (trajectory tracking). A method to enforce the open-loop vehicle stability for pre-specified operating conditions was proposed in Section 5.3.2. Such a method is however insufficient to guarantee stability around the whole computed optimal trajectory.

In order to enhance the correlation between numerical-control calculations and real driving experience, it would be useful to devise methods that 'robustify' the optimal trajectories. At this stage a few alternative approaches can be suggested:

1. a first possibility is to 'tighten' a selection of inequality constraints involved in the definition of the minimum-lap-time problem; in this way a 'conservative' optimal solution will guarantee a margin of error, which would allow for small diversions from the nominal trajectory where uncertainties or disturbances were to appear;
2. an alternative approach is based on the interpretation of the costates and Lagrange multipliers associated with optimal-control problem constraints; it was mentioned in

Section 2.1.3 that "*the Lagrange multipliers provide the sensitivity of the optimal cost with respect to variations of the associated constraints*"; if the sensitivity is high, a small variation in the constraint, caused for instance by model uncertainties, might result in a large change in the lap time; controlling the values of the multipliers might thus reduce the sensitivity of the lap time with respect to uncertainties;

3. a third method considers the loss of controllability of the vehicle that happens when the tyre friction forces approach their saturation limits; if the vehicle 'controllability' is maintained above a certain threshold, one can ensure that the vehicle operates robustly with respect to variations in the friction coefficients; however, defining a simple and meaningful measure for vehicle controllability is an interesting challenge.

Once a robust open-loop trajectory is available, it can be used as a reference to run feedback-controlled vehicle simulations of higher-fidelity models. Ideally, feedback controllers should be designed to work in real-time and to minimise the loss of performance from the reference trajectory. Control laws based on complicated nonlinear models seem still far from real-time applications, due to the significant computational burden associated. In alternative, techniques based on preview and model-predictive control applied to linear time-varying approximations appear promising.

8.5 Validation

The work in this thesis focussed on vehicle models derived from first principles in mechanics. All efforts were made to use realistic parameters, although the actual values might not be representative of any existing vehicle. However, the modelling approach makes all the calculations easily repeatable for any desired parameter set. A future development of this work involves the identification of real vehicle models and the validation of the optimal-control results.

A first validation platform is the direct comparison of the simulated vehicle trajectories with telemetry data collected during racing. What is more important than an accurate lap-

time prediction is the validation of the sensitivity results concerning lap-time variations with respect to parameter changes.

A 'grand challenge' is the real implementation of automatic driving-control systems based on optimal-control calculations. The techniques and results proposed in this thesis set the ground for the realisation of driving-assistance systems capable of operating at the boundary of the vehicle's performance envelope.

APPENDIX A

Tyre Friction

Tyre frictional forces are modelled using empirical formulae that are responsive to the tyre normal loads and combined slip. The longitudinal slip is described by a longitudinal slip coefficient κ, while the lateral slip is described by a slip angle α (see [55]). Following standard conventions

$$\kappa = -\left(1 + \frac{r\omega_w}{u_w}\right) \tag{A.1}$$

and

$$\tan\alpha = -\frac{v_w}{u_w}, \tag{A.2}$$

where r is the wheel radius and ω_w the wheel spin velocity. The terms u_w and v_w are the longitudinal and lateral components of the absolute velocity of the wheel centre expressed in the wheel's body-fixed frame.

For each wheel, the following formulae give the peak values and locations of the lateral and longitudinal friction coefficients using linear interpolation in the tyre normal load F_z (from [35]):

$$\mu_{x\,max} = (F_z - F_{z1})\frac{\mu_{x\,max2} - \mu_{x\,max1}}{F_{z2} - F_{z1}} + \mu_{x\,max1}, \tag{A.3}$$

$$\mu_{y\,max} = (F_z - F_{z1})\frac{\mu_{y\,max2} - \mu_{y\,max1}}{F_{z2} - F_{z1}} + \mu_{y\,max1}, \tag{A.4}$$

$$\kappa_{max} = (F_z - F_{z1})\frac{\kappa_{max2} - \kappa_{max1}}{F_{z2} - F_{z1}} + \kappa_{max1}, \tag{A.5}$$

$$\alpha_{max} = (F_z - F_{z1})\frac{\alpha_{max2} - \alpha_{max1}}{F_{z2} - F_{z1}} + \alpha_{max1}, \tag{A.6}$$

where the quantities containing a '1' or a '2' in the subscript are measured tyre parameters (see Table B.1). Next, the tyre slip is normalised with respect to the peak slip:

$$\kappa_n = \kappa/\kappa_{max}, \tag{A.7}$$

$$\alpha_n = \alpha/\alpha_{max}. \tag{A.8}$$

Following normalisation, the slip is characterised by a combined-slip coefficient

$$\rho = \sqrt{\alpha_n^2 + \kappa_n^2}. \tag{A.9}$$

The friction coefficients in the longitudinal and lateral directions are described by

$$\mu_x = \mu_{x\,max} \sin\big(Q_x \arctan(S_x\rho)\big), \tag{A.10}$$

$$\mu_y = \mu_{y\,max} \sin\big(Q_y \arctan(S_y\rho)\big). \tag{A.11}$$

with

$$S_x = \frac{\pi}{2\arctan(Q_x)}, \tag{A.12}$$

$$S_y = \frac{\pi}{2\arctan(Q_y)}. \tag{A.13}$$

Finally, the longitudinal and lateral components of the tyre forces are given by

$$F_x = \mu_x F_z \frac{\kappa_n}{\rho}, \tag{A.14}$$

$$F_y = \mu_y F_z \frac{\alpha_n}{\rho}. \tag{A.15}$$

In the car model used here the normal loads F_z are determined by solving the load balance equations given in Sections 5.1.3.1 and 5.1.3.2, or in Section 6.2.1.3. The four wheel slip angles are given by

$$\begin{aligned}
\alpha_{rr} &= \arctan\left(\tfrac{v-\bar{\omega}_z b}{u-\bar{\omega}_z w_r}\right), \\
\alpha_{rl} &= \arctan\left(\tfrac{v-\bar{\omega}_z b}{u+\bar{\omega}_z w_r}\right), \\
\alpha_{fr} &= \arctan\left(\tfrac{\sin\delta(\bar{\omega}_z w_f-u)+\cos\delta(\bar{\omega}_z a+v)}{\cos\delta(u-\bar{\omega}_z w_f)+\sin\delta(\bar{\omega}_z a+v)}\right), \\
\alpha_{fl} &= \arctan\left(\tfrac{\cos\delta(\bar{\omega}_z a+v)-\sin\delta(\bar{\omega}_z w_f+u)}{\cos\delta(\bar{\omega}_z w_f+u)+\sin\delta(\bar{\omega}_z a+v)}\right),
\end{aligned} \tag{A.16}$$

with the longitudinal slip coefficients given by

$$\begin{aligned}
\kappa_{rr} &= -\left(1+\tfrac{r\omega_{rr}}{u-\bar{\omega}_z w_r}\right), \\
\kappa_{rl} &= -\left(1+\tfrac{r\omega_{rl}}{u+\bar{\omega}_z w_r}\right), \\
\kappa_{fr} &= -\left(1+\tfrac{r\omega_{fr}}{\cos\delta(u-\bar{\omega}_z w_f)+\sin\delta(\bar{\omega}_z a+v)}\right), \\
\kappa_{fl} &= -\left(1+\tfrac{r\omega_{fl}}{\cos\delta(u+\bar{\omega}_z w_f)+\sin\delta(\bar{\omega}_z a+v)}\right),
\end{aligned} \tag{A.17}$$

where $\bar{\omega}_z$ is the z-component of the vehicle's absolute angular speed expressed in its body-fixed reference frame.

As is evident from equations (A.14) and (A.15) the tyre model is responsive to normal load variations and combined slip effects. It also allows one to specify locations and peak values for the coefficients of friction in the longitudinal and lateral directions, which are treated as linear functions of the normal load. Longitudinal and lateral 'shape factors' are used to determine the cornering stiffness and the asymptotic 'roll off' characteristics of the tyre. Due to a paucity of publicly available Formula 1 tyre data, the same tyre model was used for the front and rear wheels.

The left-hand part of Figure A.1 shows the load dependence of the longitudinal tyre force under pure longitudinal slip conditions; larger longitudinal forces are possible when the normal load is increased. The main reason for introducing aerodynamic downforce-generating systems on Formula 1 cars is to exploit this effect. The right-hand part of Figure A.1 shows how the longitudinal force is compromised when the tyre is side-slipping. As the side-slip angle increases, the longitudinal peak force reduces and moves towards higher slip values.

The normal-load-dependent inverted witch's hat in Figure A.2 gives a three-dimensional representation of the combined slip characteristics of the tyre model at a fixed normal load of 2 kN. The tyre's pure-slip characteristics are obtained by taking vertical cuts through the lines $\alpha = 0$ and $\kappa = 0$.

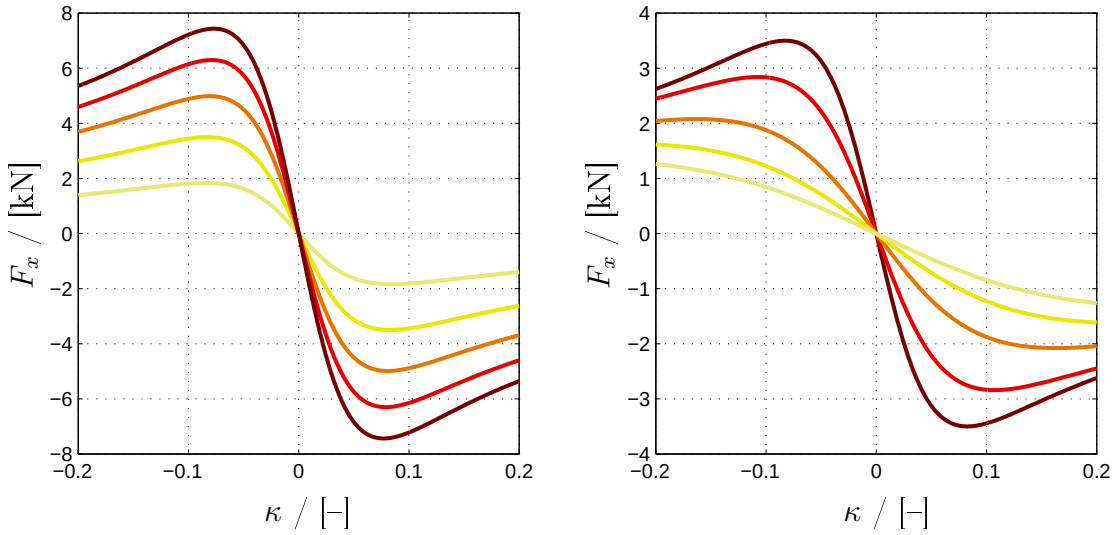

Figure A.1: Longitudinal tyre force versus longitudinal slip. The left-hand diagram considers five normal loads (at $\alpha = 0°$); the lightest yellow curve corresponds to 1 kN with the normal load then increased in steps of 1 kN. The right-hand diagram considers combined slip for five values of side-slip angle at a normal load of 2 kN; the darkest red curve corresponds to a slip angle of 0° with the slip angle increased in steps of 5°.

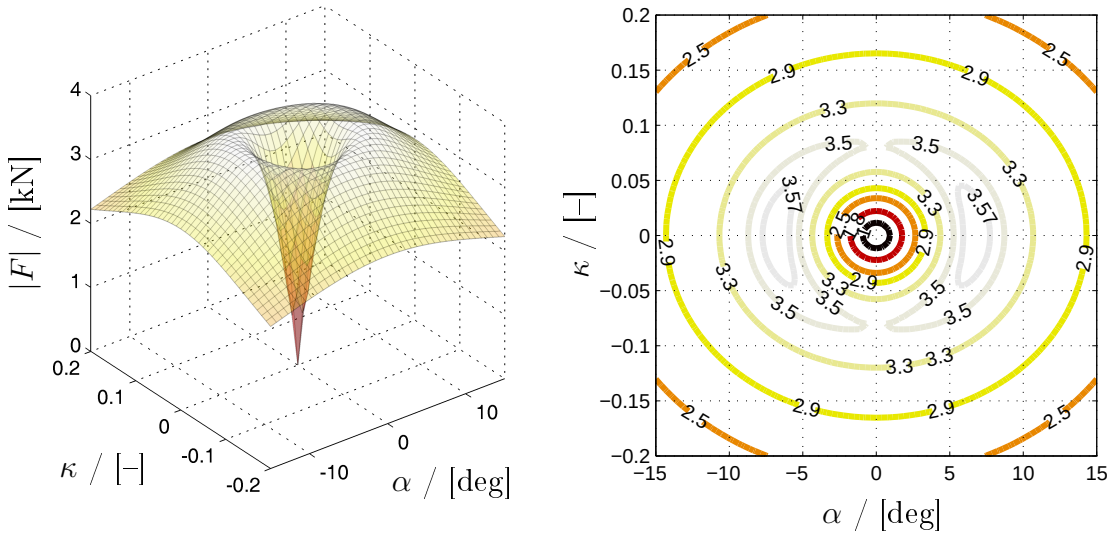

Figure A.2: Modulus of the tyre force as a function of the longitudinal slip κ and the slip angle α for a normal load of 2 kN. The right-hand diagram shows contours of equal tyre-force (in kN).

APPENDIX B

Vehicle and Tyre Data

This appendix contains the nominal values for the tyre and vehicle parameters used in this study: the tyre parameters are in Table B.1; the vehicle parameters are in Table B.2; the polynomial coefficients for the speed-dependent aerodynamic model (see Section 6.1) are in Table B.3.

Symbol	Description	Value
F_{z1}	reference load 1	2 kN
F_{z2}	reference load 2	6 kN
μ_{x1}	peak longitudinal friction coefficient at load 1	1.75
μ_{x2}	peak longitudinal friction coefficient at load 2	1.40
κ_1	slip coefficient for the friction peak at load 1	0.11
κ_2	slip coefficient for the friction peak at load 2	0.10
μ_{y1}	peak lateral friction coefficient at load 1	1.80
μ_{y2}	peak lateral friction coefficient at load 2	1.45
α_1	slip angle for the friction peak at load 1	9°
α_2	slip angle for the friction peak at load 2	8°
Q_x	longitudinal shape factor	1.9
Q_y	lateral shape factor	1.9

Table B.1: Tyre friction parameters.

Symbol	Description	Value
m	vehicle mass	660 kg
I_x	moment of inertia about the $x-$axis	112.5 kg·m^2
I_y	moment of inertia about the $y-$axis	450 kg·m^2
I_z	moment of inertia about the $z-$axis	450 kg·m^2
w_b	wheelbase	3.4 m
a	distance of the mass centre from the front axle	1.8 m
b	distance of the mass centre from the rear axle	$w_b - a$
h	centre of mass height	0.3 m
d_r	roll moment distribution (fraction at the front axle)	0.5
w_f	front-wheel to car-centre-line distance	0.73 m
w_r	rear-wheel to car-centre-line distance	0.73 m
r	wheel radius	0.33 m
k_d	differential friction coefficient	954.9 N·m·s·rad^{-1}
c_d	drag coefficient	0.9
c_l	downforce coefficient	3.0
a_f	frontal area	1.5 m^2
ρ_a	air density	1.2 kg·m^{-3}
a_a	centre-of-pressure to front-axle distance	1.9 m
b_a	centre-of-pressure to rear-axle distance	$w_b - a_a$
P_{max}	peak engine power	560/440* kW
P_{kers}^{max}	maximum MGU-K power	60/120* kW
r_f^{max}	maximum fuel rate	0.028 kg·s^{-1}
$P_h^{min/max}$	minimum/maximum MGU-H power	−40/60 kW
P_w	power boost obtained opening the waste gate	20 kW

Table B.2: Vehicle parameters; values marked with an asterisk relate to the 2014 car.

Symbol	Description	Value
c_{d0}	0^{th} order coeff. for the variable drag	1.0529
c_{d1}	1^{st} order coeff. for the variable drag	$-6.6429 \cdot 10^{-4}$ s·m^{-1}
c_{d2}	2^{nd} order coeff. for the variable drag	$-1.0183 \cdot 10^{-5}$ s^2·m^{-2}
c_{l0}	0^{th} order coeff. for the variable downforce	3.5500
c_{l1}	1^{st} order coeff. for the variable downforce	$6.9429 \cdot 10^{-3}$ s·m^{-1}
c_{l2}	2^{nd} order coeff. for the variable downforce	$-1.6663 \cdot 10^{-4}$ s^2·m^{-2}
c_{p0}	0^{th} order coeff. for the moving centre of pressure	1.8545 m
c_{p1}	1^{st} order coeff. for the moving centre of pressure	$4.0454 \cdot 10^{-3}$ s
c_{p2}	2^{nd} order coeff. for the moving centre of pressure	$-2.5419 \cdot 10^{-5}$ s^2·m^{-1}

Table B.3: Polynomial coefficients for the speed-dependent aerodynamic model.

APPENDIX C

Ribbon Tangent Plane

The vehicle model in Section 6.2.1 is based on the assumption that the road surface is, at any time, a plane orthogonal to the vector $\hat{\mathbf{m}}$. The analysis in this appendix recognises that the plane tangent to the ribbon surface at the curvilinear coordinates (s, n) is not in general parallel to the tangent plane at the corresponding point on the spine, with coordinates $(s, 0)$. The car's kinematics and dynamics are thus derived for an extended model that accounts for this phenomenon.

The fundamental concepts and notation introduced for the differential parametrisation of a ribbon in Section 4.1.3 are briefly summarised. The orientation of the ribbon trihedron attached to the spine is given by the rotation matrix

$$R(s) = \left[\begin{array}{ccc} \hat{\mathbf{t}} & \hat{\mathbf{n}} & \hat{\mathbf{m}} \end{array} \right]. \tag{C.1}$$

The *generalised curvature* vector $\mathbf{\Omega}_B$ is a vector in $\mathbb{R}^3$ with the associated skew-symmetric matrix $S(\mathbf{\Omega}_B)$:

$$\mathbf{\Omega} = \left[\begin{array}{c} \Omega_x \\ \Omega_y \\ \Omega_z \end{array} \right], \qquad S\left(\mathbf{\Omega}\right) = \left[\begin{array}{ccc} 0 & -\Omega_z & \Omega_y \\ \Omega_z & 0 & -\Omega_x \\ -\Omega_y & \Omega_x & 0 \end{array} \right]. \tag{C.2}$$

The matrix $S\left(\mathbf{\Omega}_B\right)$ satisfies the differential relation

$$\frac{dR}{ds} = RS\left(\mathbf{\Omega}_B\right). \tag{C.3}$$

The ribbon frame translates in the spine-tangent direction for increasing s, from the initial conditions $\mathbf{c}(s_0)$ and $R(s_0)$. The integral of the generalised curvature $\mathbf{\Omega}_B$ along the

curvilinear abscissa s determines the orientation of the frame. The ribbon surface $\mathcal{R}$ is thus defined in integral form using the equations

$$\begin{aligned} R(s) &= \int_{s_0}^{s} \frac{dR}{ds}\, ds, \\ \mathbf{r}(s,n) &= \int_{s_0}^{s} \hat{\mathbf{t}}\, ds + n\hat{\mathbf{n}}, \end{aligned} \tag{C.4}$$

in which $\frac{dR}{ds}$ comes from (C.3).

The following definitions are introduced to simplify the notation:

$$\mathbf{r}_s(s,n) := \frac{\partial \mathbf{r}(s,n)}{\partial s}, \qquad \mathbf{r}_n(s,n) := \frac{\partial \mathbf{r}(s,n)}{\partial n}, \qquad \hat{\mathbf{m}}_s(s,n) := \frac{\partial \hat{\mathbf{m}}(s)}{\partial s}. \tag{C.5}$$

C.1 Tangent Frame

At the coordinates $\mathbf{r}(s,n)$ the plane tangent to the ribbon surface is defined by the directional derivatives

$$\begin{aligned} \mathbf{r}_s &= \hat{\mathbf{t}} + n\frac{d\hat{\mathbf{n}}}{ds} \\ &= \hat{\mathbf{t}} + n\big[-\Omega_z \hat{\mathbf{t}} + \Omega_x \hat{\mathbf{m}}\big] \\ &= \hat{\mathbf{t}}\big[1 - n\Omega_z\big] + n\Omega_x \hat{\mathbf{m}}, \\ \mathbf{r}_n &= \hat{\mathbf{n}}. \end{aligned} \tag{C.6}$$

The *tangent frame* is defined as the reference frame having the x- and y-axes parallel to $\mathbf{r}_s$ and $\mathbf{r}_n$, respectively. A complete base for the tangent frame is given by the three unit vectors (identified by the subscript r)

$$\begin{aligned} \hat{\mathbf{t}}_r(s,n) &= \frac{\mathbf{r}_s}{|\mathbf{r}_s|}, \\ \hat{\mathbf{n}}_r(s,n) &= \mathbf{r}_n = \hat{\mathbf{n}}(s), \\ \hat{\mathbf{m}}_r(s,n) &= \hat{\mathbf{t}}_r \times \hat{\mathbf{n}}_r. \end{aligned} \tag{C.7}$$

Note that the ribbon and the tangent frames have a common y-axis $\hat{\mathbf{n}}(s) = \hat{\mathbf{n}}_r(s,n)$. The rotation matrix associated with the tangent frame is

$$R_r(s,n) = \begin{bmatrix} \hat{\mathbf{t}}_r & \hat{\mathbf{n}}_r & \hat{\mathbf{m}}_r \end{bmatrix}. \tag{C.8}$$

It can be observed from (C.6) and (C.7) that the tangent frame at $\mathbf{r}(s,n)$ is obtained through a rotation of the ribbon frame at s. The ribbon frame is rotated around the

vector $\hat{\mathbf{n}}(s)$ of an angle $\eta(s,n)$, with

$$\eta(s,n) = \arctan \frac{-n\Omega_x}{1-n\Omega_z}. \tag{C.9}$$

The following relation between rotation matrices hold:

$$R_r(s,n) = R(s)R_y\big(\eta(s,n)\big), \quad \text{with} \quad R_y(\eta) = \begin{bmatrix} \cos\eta & 0 & \sin\eta \\ 0 & 1 & 0 \\ -\sin\eta & 0 & \cos\eta \end{bmatrix}. \tag{C.10}$$

The orientation of the tangent frame is a function of both the curvilinear coordinates s and n.

C.1.1 Embedded Curve

Consider a curve embedded in the ribbon surface (e.g. the trajectory of the vehicle's geometric centre), and assume that such curve can be parametrised in the ribbon abscissa s only. The n coordinate associated with each point of the said curve is a function of s, dictated by the vehicle motion. The total derivative of the tangent rotation matrix with respect to s along the embedded curve is

$$\frac{dR_r(s,n(s))}{ds} = \frac{\partial R_r(s,n(s))}{\partial s} + \frac{\partial R_r(s,n(s))}{\partial n(s)}\frac{dn(s)}{ds}. \tag{C.11}$$

The generalised curvature vector for a frame moving along an arbitrary curve embedded in the ribbon surface is found rearranging equation (C.3):

$$S(\boldsymbol{\Omega}_r(s)) = R_r^T\frac{dR_r}{ds} = R_r^T\begin{bmatrix} \dfrac{d\hat{\mathbf{t}}_r}{ds} & \dfrac{d\hat{\mathbf{n}}_r}{ds} & \dfrac{d\hat{\mathbf{m}}_r}{ds} \end{bmatrix}. \tag{C.12}$$

The partial derivatives of $\mathbf{r}_s$ and $\hat{\mathbf{m}}_s$ with respect to s and n are

$$\begin{aligned}
\frac{\partial \mathbf{r}_s}{\partial s} &= (\Omega_z\hat{\mathbf{n}} - \Omega_y\hat{\mathbf{m}})(1-n\Omega_z) - n\tfrac{d\Omega_z}{ds}\hat{\mathbf{t}} + \\
&+ n\tfrac{d\Omega_x}{ds}\hat{\mathbf{m}} + n\Omega_x\left(\Omega_y\hat{\mathbf{t}} - \Omega_x\hat{\mathbf{n}}\right), \\
\frac{\partial \mathbf{r}_s}{\partial n} &= -\Omega_z\hat{\mathbf{t}} + \Omega_x\hat{\mathbf{m}}, \\
\frac{\partial \hat{\mathbf{m}}_s}{\partial s} &= \left(\Omega_y\hat{\mathbf{t}} - \Omega_x\hat{\mathbf{n}}\right)(1-n\Omega_z) - n\tfrac{d\Omega_z}{ds}\hat{\mathbf{m}} + \\
&- n\tfrac{d\Omega_x}{ds}\hat{\mathbf{t}} - n\Omega_x\left(\Omega_z\hat{\mathbf{n}} - \Omega_y\hat{\mathbf{m}}\right), \\
\frac{\partial \hat{\mathbf{m}}_s}{\partial n} &= -\Omega_z\hat{\mathbf{m}} - \Omega_x\hat{\mathbf{t}}.
\end{aligned} \tag{C.13}$$

The total derivatives with respect to s of the unit vectors defining the tangent frame are

$$\begin{aligned}
\frac{d\hat{\mathbf{t}}_r(s)}{ds} &= \frac{\partial\hat{\mathbf{t}}_r}{\partial s}+\frac{\partial\hat{\mathbf{t}}_r}{\partial n}\frac{dn}{ds}\\
&= \left[\frac{\frac{\partial\mathbf{r}_s}{\partial s}}{|\mathbf{r}_s|}+\frac{\mathbf{r}_s}{|\mathbf{r}_s|}\frac{\mathbf{r}_s^T\frac{\partial\mathbf{r}_s}{\partial s}}{|\mathbf{r}_s|^2}\right]+\left[\frac{\frac{\partial\mathbf{r}_s}{\partial n}}{|\mathbf{r}_s|}+\frac{\mathbf{r}_s}{|\mathbf{r}_s|}\frac{\mathbf{r}_s^T\frac{\partial\mathbf{r}_s}{\partial n}}{|\mathbf{r}_s|^2}\right]\frac{dn}{ds},\\
\frac{d\hat{\mathbf{n}}_r(s)}{ds} &= \frac{d\hat{\mathbf{n}}}{ds},\\
\frac{d\hat{\mathbf{m}}_r(s)}{ds} &= \frac{\partial\hat{\mathbf{m}}_r}{\partial s}+\frac{\partial\hat{\mathbf{m}}_r}{\partial n}\frac{dn}{ds}\\
&= \left[\frac{\frac{\partial\hat{\mathbf{m}}_s}{\partial s}}{|\mathbf{r}_s|}+\frac{\hat{\mathbf{m}}_s}{|\mathbf{r}_s|}\frac{\mathbf{r}_s^T\frac{\partial\mathbf{r}_s}{\partial s}}{|\mathbf{r}_s|^2}\right]+\left[\frac{\frac{\partial\hat{\mathbf{m}}_s}{\partial n}}{|\mathbf{r}_s|}+\frac{\hat{\mathbf{m}}_s}{|\mathbf{r}_s|}\frac{\mathbf{r}_s^T\frac{\partial\mathbf{r}_s}{\partial n}}{|\mathbf{r}_s|^2}\right]\frac{dn}{ds}.
\end{aligned} \tag{C.14}$$

C.2 Kinematics

The vehicle is allowed to yaw on the ribbon surface. The yaw freedom allows for a rotation around the axis normal to the ribbon passing through the vehicle's geometric centre[1] (the $\hat{\mathbf{m}}_r$ axis). The yaw angle ξ is measured around $\hat{\mathbf{m}}_r$ between the vector $\hat{\mathbf{t}}_r$ and the vehicle's body-fixed x-axis. The rotation matrix associated with the vehicle's frame is thus

$$\bar{R}_r(s,n,\xi)=R_rR_z(\xi),\qquad \text{with } R_z(\xi)=\begin{bmatrix}\cos\xi & -\sin\xi & 0\\ \sin\xi & \cos\xi & 0\\ 0 & 0 & 1\end{bmatrix}; \tag{C.15}$$

applying $S(\bar{\boldsymbol{\omega}}_{rB})=\bar{R}_r^T\dot{\bar{R}}_r$ gives the angular velocity of the car in its body-fixed reference frame

$$\bar{\boldsymbol{\omega}}_{rB}=\begin{bmatrix}\bar{\omega}_{rx}\\ \bar{\omega}_{ry}\\ \bar{\omega}_{rz}\end{bmatrix}=\begin{bmatrix}\cos\xi\omega_{rx}+\sin\xi\omega_{ry}\\ \cos\xi\omega_{ry}-\sin\xi\omega_{rx}\\ \omega_{rz}+\dot{\xi}\end{bmatrix}, \tag{C.16}$$

with

$$\boldsymbol{\omega}_{rB}=\begin{bmatrix}\omega_{rx}\\ \omega_{ry}\\ \omega_{rz}\end{bmatrix}=\frac{ds}{dt}\boldsymbol{\Omega}_{rB}=\frac{ds}{dt}\begin{bmatrix}\Omega_{rx}\\ \Omega_{ry}\\ \Omega_{rz}\end{bmatrix}. \tag{C.17}$$

[1] According to the vehicle model used in Chapters 5 and 6 the road-wheel contact points belong to a plane; it is thus necessary to approximate the ribbon surface under the moving vehicle with a plane; this was chosen in this appendix as the plane tangent to the ribbon at the vehicle's geometric centre.

Resolving the vehicle's velocity in the $\hat{\mathbf{t}}$-direction gives

$$\begin{aligned} &\frac{ds}{dt}\hat{\mathbf{t}} + n\frac{d}{dt}\hat{\mathbf{n}} \cdot \hat{\mathbf{t}} = (u\cos\xi - v\sin\xi)\hat{\mathbf{t}}_r \cdot \hat{\mathbf{t}} \\ &\quad \rightarrow \frac{ds}{dt}(1 - \Omega_z n) = (u\cos\xi - v\sin\xi)\cos\eta \\ &\quad \rightarrow \frac{ds}{dt} = \frac{(u\cos\xi - v\sin\xi)\cos\eta}{1 - \Omega_z n}, \end{aligned} \tag{C.18}$$

where u and v are, respectively, the longitudinal and lateral components of the vehicle's velocity expressed in its body-fixed reference frame. The rate of change of n is

$$\frac{dn}{dt} = u\sin\xi + v\cos\xi. \tag{C.19}$$

Finally, summing the angular speeds of the vehicle reference frame one obtains

$$\frac{d\xi}{dt} = \bar{\omega}_{rz} - \Omega_{rz}\frac{ds}{dt}, \tag{C.20}$$

where $\bar{\omega}_{rz}$ is the angular speed of the vehicle around its z-axiz (in the direction of $\hat{\mathbf{m}}_r$), which will come from the integration of the dynamics to be defined in Section C.3.

C.3 Dynamics

As described in Chapters 5 and 6, the vehicle used in the optimal-control calculations is dynamically modelled as a rigid body with mass m and moment of inertia I. The mass is located at coordinates $(0, 0, -h)$ in the car's body-fixed reference frame. In the following equations it is assumed that the principal axes of inertia are aligned with the vehicle's body-fixed frame, thus the inertia matrix is diagonal, with entries I_x, I_y and I_z in order.

The vehicle's geometric centre is allowed to move on the ribbon surface, while the z-axes of the vehicle's and tangent frames must stay parallel. The fact that the vehicle cannot translate in the body-fixed z-direction represents a non-holonomic constraint that must be enforced within the z-direction force balance (see (5.16) or (6.42)). Two non-holonomic constraints determine the orientation of the vehicle's z-axis, which are enforced by moment balances around the body-fixed x- and y-directions (see (5.17,5.18) or (6.43,6.44)). The procedure used in Section 6.2.1.2 for the derivation of the equations

of motion through Lagrangian mechanics is applied to define the three non-holonomic constraints:

$$0 = F_z/m + \bar{\omega}_{ry}(u - h\bar{\omega}_{ry}) - \bar{\omega}_{rx}(v + h\bar{\omega}_{rx}) + g\hat{\mathbf{e}}_z \cdot \hat{\mathbf{m}}_r, \tag{C.21}$$

$$I_x \frac{d}{dt}\bar{\omega}_{rx} = M_x + (I_y - I_z)\bar{\omega}_{rz}\bar{\omega}_{ry}, \tag{C.22}$$

$$I_y \frac{d}{dt}\bar{\omega}_{ry} = M_y + (I_z - I_x)\bar{\omega}_{rz}\bar{\omega}_{rx}; \tag{C.23}$$

in which F_x, F_y and F_z are, respectively, the longitudinal, lateral and vertical components of the external forces, and M_z is the yawing moment acting on the car due to tyre friction and aerodynamic effects (see Section 6.2.1.2); $\hat{\mathbf{e}}_z$ is the unit vector in the direction of the inertial z-axis. These equations differ from (6.42, 6.43, 6.44) in that $\bar{\boldsymbol{\omega}}_r$ accounts for the rotation of the tangent plane with respect to the ribbon frame; as a consequence the vertical speed and acceleration of the car expressed in its body-fixed reference frame are identically 0. However, it is readily proven that the vehicle's speed in the $\hat{\mathbf{m}}$ direction has value $n\omega_x$ as given in (6.10). The vertical component of the vehicle speed in the ribbon frame is derived projecting the absolute speed of the vehicle's geometric centre on $\hat{\mathbf{m}}$ and using the directional derivatives (C.6)

$$\begin{aligned} \frac{d}{dt}\mathbf{r} \cdot \hat{\mathbf{m}} &= \left(\mathbf{r}_s + \mathbf{r}_n \frac{\partial n}{\partial s}\right)\frac{ds}{dt} \cdot \hat{\mathbf{m}} \\ &= \left(\left[1 - n\Omega_z\right]\hat{\mathbf{t}} + n\Omega_x\hat{\mathbf{m}} + \hat{\mathbf{n}}\right)\frac{ds}{dt} \cdot \hat{\mathbf{m}} \\ &= n\Omega_x \frac{ds}{dt} \\ &= n\omega_x. \end{aligned} \tag{C.24}$$

The longitudinal and lateral components of the vehicle's acceleration and the yaw angular acceleration are again derived through the Lagrangian approach:

$$\frac{d}{dt}u = \bar{\omega}_{rz}(v + h\bar{\omega}_{rx}) + g\hat{\mathbf{e}}_z \cdot \hat{\mathbf{t}}_r + F_x/m + h\frac{d}{dt}\bar{\omega}_{ry}, \tag{C.25}$$

$$\frac{d}{dt}v = -\bar{\omega}_{rz}(u - h\bar{\omega}_{ry}) + g\hat{\mathbf{e}}_z \cdot \hat{\mathbf{n}}_r + F_y/m - h\frac{d}{dt}\bar{\omega}_{rx}, \tag{C.26}$$

$$I_z \frac{d}{dt}\bar{\omega}_{rz} = (I_x - I_y)\bar{\omega}_{rx}\bar{\omega}_{ry} + M_z. \tag{C.27}$$

Minimum-lap-time problems for the model described in this appendix were solved using direct collocation methods. Considering the Circuit de Catalunya (see Section 4.3.2), the

differences between the results obtained using a frame tangent to the surface as opposed to the ribbon frame at the spine (see Section 6.2.1) are negligible.

The mathematical model of the vehicle in this appendix is significantly more complicated than the model in Section 6.2.1 due to the introduction of additional geometric relations in the definition of the equations of motion. In the context of minimum-time-manoeuvring calculations, the small gain in modelling accuracy is not worth the growth in computational complexity, and the 'ribbon-frame' assumption of Section 6.2.1 is considered satisfactory. Nevertheless, more accurate modelling of the vehicle-road interaction might be useful in simulation applications.

Bibliography

[1] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell. *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. SIAM, Philadelphia, siam's cla edition, 1995. (cited on Pages 36, 39, 40, and 44)

[2] R. E. Bellman and S. E. Dreyfus. *Applied Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1962. (cited on Page 19)

[3] J. Bernoulli. Problema novum ad cujus solutionem Mathematici invitantur. *Acta Eruditorum*, 1696. (cited on Pages xii and 58)

[4] E. Bertolazzi, F. Biral, and M. Da Lio. Symbolic-numeric indirect method for solving optimal control problems for large multibody systems. *Multibody System Dynamics*, 13:233 — 252, 2005. (cited on Pages 5 and 116)

[5] D. P. Bertsekas. *Dynamic Programming and Optimal Control Volume I*. Athena Scientific, 1995. (cited on Pages 19 and 25)

[6] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, second edition, 1999. (cited on Pages 11, 12, 15, 16, and 18)

[7] J. T. Betts. Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207, 1998. (cited on Pages 32 and 53)

[8] J. T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. SIAM, second edition, 2010. (cited on Pages 31, 32, 44, 45, 49, 52, 77, 78, 102, 117, and 133)

[9] G. A. Bliss. *Lectures on the Calculus of Variations*. The University of Chicago Press, 1946. (cited on Pages 21 and 22)

[10] D. L. Brayshaw. *The use of numerical optimisation to determine on-limit handling behaviour of race cars*. Cranfield University, 2004. PhD Thesis. (cited on Page 118)

[11] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. SIAM's Classics in Applied Mathematics. SIAM, Philadelphia, 1996. (cited on Pages 34, 35, 41, and 102)

[12] A. E. Bryson. Optimal control —1950 to 1985. *IEEE Control Systems Magazine*, 13(3):26–33, 1996. (cited on Pages 8, 25, 27, 28, 50, 115, and 117)

[13] A. E. Bryson and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control*. Taylor & Francis, revised printing edition, 1975. (cited on Pages 9, 20, 46, 51, 61, and 66)

[14] C. Büskens. Whorp large-scale sparse nonlinear optimization, http://www.worhp.de, September 2013. (cited on Page 27)

[15] J. C. Butcher. Implicit runge-kutta processes. *Mathematics of Computation*, pages 50–64, 1964. (cited on Pages 37 and 39)

[16] M. D. Canon, C. D. Cullum Jr., and E. Polak. *Theory of Optimal Control and Mathematical Programming*. McGraw-Hill series in systems science. McGraw-Hill, 1970. (cited on Pages 8 and 42)

[17] A. Carlini, A. Hosoya, T. Koike, and Y. Okudaira. Time-optimal quantum evolution. *Phys. Rev. Lett.*, 96(6):060503 [4], February 2006. (cited on Page 59)

[18] D. Casanova. *On Minimum Time Vehicle Manoeuvring: The Theoretical Optimal Lap*. Cranfield University School of Engineering, 2000. PhD Thesis. (cited on Pages 5, 89, 90, 102, 117, and 139)

[19] V. Cossalter, M. Da Lio, R. Lot, and L. Fabbri. A general method for the evaluation of vehicle manoeuvrability with special emphasis on motorcycles. *Vehicle System Dynamics*, 31:113–135, 1999. (cited on Pages 5, 89, 90, 116, and 121)

[20] C. L. Darby, W. W. Hager, and A. V. Rao. An hp-adaptive pseudospectral method for solving optimal control problems. *Optimal Control Applications and Methods*, 32(4):476–502, 2011. (cited on Pages 102 and 106)

[21] S. Dreyfus. The numerical solution of variational problems. *J. Math. Anal. and Appl.*, 5:30–45, 1962. (cited on Page 59)

[22] P. Falugi, E. Kerrigan, and E. van Wyk. *Imperial College London Optimal Control Software User Guide (ICLOCS)*. Imperial College London, London UK, May 2010. (cited on Pages xi, 132, and 133)

[23] C. W. Gear. *Numerical initial value problems in ordinary differential equations*. Prentice-Hall series in automatic computation. Prentice-Hall, Englewood Cliffs, New Jersey, 1971. (cited on Pages 34, 35, and 40)

[24] C. W. Gear. Simultaneous numerical solution of differential-algebraic equations. *Circuit Theory, IEEE Transactions on*, C:89–95, 1971. (cited on Page 102)

[25] H. Goldstein, C. P. Poole, and J. Safko. *Classical Mechanics*. Addison-Wesley, Reading, Massachusetts; London, England, 2nd edition, 1980. (cited on Page 79)

[26] H. F. Goldstein and C. M. Bender. Relativistic brachistochrone. *J.Math.Phys.*, 27:507, 1986. (cited on Page 59)

[27] H. H. Goldstine. *A History of the Calculus of Variations from the 17th through the 19th Century*. Springer-Verlag, 1980. (cited on Page 8)

[28] D. F. Griffiths and D. J. Higham. *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*. Springer, 2010. (cited on Pages 34, 35, 37, and 102)

[29] T. Gustafsson. Computing the ideal racing line using optimal control. Master's thesis, Linköping University, 2008. (cited on Pages 118 and 139)

[30] E. Hairer, G. Wanner, and C. Lubich. *Geometric Numerical Integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, corrected edition, 2002. (cited on Pages 37 and 38)

[31] P. C. Hammer and J. W. Hollingsworth. Trapezoidal Methods of Approximating Solutions of Differential Equations. *Mathematical Tables and Other Aids to Computation*, 9(51):92–96, 1955. (cited on Page 27)

[32] J. C. Hayen. Brachistochrone with Coulomb friction. *International Journal of Non-Linear Mechanics*, 40:1057–1075, 2005. (cited on Page 59)

[33] J. P. M. Hendrikx, T. J. J. Meijlink, and R. F. C. Kriens. Application of optimal control theory to inverse simulation of car handling. *Vehicle System Dynamics*, 26:449–461, 1996. (cited on Pages 89 and 117)

[34] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, New York, 1991. (cited on Page 135)

[35] D. P. Kelly. *Lap Time Simulation with Transient Vehicle and Tyre Dynamics*. Cranfield University School of Engineering, 2008. PhD Thesis. (cited on Pages 4, 5, 89, 117, 150, and 198)

[36] D. A. Kessler and Y. Rabin. Effect of curvature and twist on the conformations of a fluctuating ribbon. *The Journal of Chemical Physics*, 118(2):897–904, 2003. (cited on Pages 91 and 97)

[37] D. E. Kirk. *Optimal Control Theory: An Introduction*. Prentice-Hall, Inc., Englewood Cliffs, N. J., 1970. (cited on Pages 117 and 174)

[38] J. J. Koenderink. *Solid Shape*. MIT Press, 1990. (cited on Pages 90 and 97)

[39] E. Kreyszig. *Differential Geometry*. University of Toronto Press, 1991. (cited on Pages 92 and 95)

[40] H. W. Kuhn and A. W. Tucker. Nonlinear Programming. In *Second Berkeley Symposium on Mathematical Statistics and probability*, pages 481–492. University of California Press, 1951. (cited on Page 15)

[41] C. Lanczos. *Applied Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey, 1956. (cited on Page 36)

[42] C. Lanczos. *The Variational Principles of Mechanics*. Dover Publications, fourth edition, 1986. (cited on Page 162)

[43] L. D. Landau and E. M. Lifshitz. *Mechanics*. Elsevier, third edition, 1976. (cited on Pages 2 and 161)

[44] J. D. Lawrence. *A Catalog of Special Plane Curves*. Dover Publications, 1972. (cited on Page 106)

[45] D. Liberzon. *Calculus of Variations and Optimal Control Theory*. Princeton University Press, 2012. (cited on Pages 20 and 25)

[46] D. J. N. Limebeer and G. Perantoni. Optimal control of a formula one car on a three-dimensional track part 2: Optimal control. *Journal of Dynamic Systems, Measurement and Control*. under review, 2013. (cited on Pages vi, 6, and 154)

[47] D. J. N. Limebeer, G. Perantoni, and A. V. Rao. Optimal control of formula one car energy recovery systems. *International Journal of Control*. under review, 2013. (cited on Pages vi, 6, and 154)

[48] X. Lin, J. Zhang, Z. Liu, J. Shen, and M. Duan. Semi-automatic extraction of road networks by least squares interlaced template matching in urban areas. *International Journal of Remote Sensing*, 32(17):4943–4959, 2011. (cited on Page 91)

[49] A. M. I. Mamat, A. Romagnoli, and R. F. Martinez-Botas. Design and development of a low-pressure turbine for turbocompounding applications. *International Journal of Gas Turbine, Propulsion and Power Systems*, 4(3):1–8, 2012. (cited on Page 178)

[50] Fédération Internationale de l'Automobile. 2013 formula one technical regulations. Technical report, 2012. (cited on Pages 6, 172, and 173)

[51] Fédération Internationale de l'Automobile. 2014 formula one technical regulations. Technical report, 2013. (cited on Pages 1, 6, 172, 178, and 181)

[52] J. B. Mena. State of the art on automatic road extraction for GIS update: a novel classification. *Pattern Recognition Letters*, 24(16):3037 – 3058, 2003. (cited on Page 91)

[53] C. B. Moler. 7.12 The Matlab ODE Solvers. In *Numerical Computing with MATLAB*. SIAM, 2008. (cited on Page 40)

[54] J. I. Neĭmark and N. A. Fufaev. *Dynamics of Nonholonomic Systems*, volume 33. American Math. Soc. Translations of Mathematical Monographs, 1972. (cited on Page 79)

[55] H. B. Pacejka. *Tyre and Vehicle Dynamics*. Butterworth-Heinemann, second edition edition, 2008. (cited on Pages 4, 117, 143, and 198)

[56] S. Panyukov and Y. Rabin. Fluctuating filaments: statistical mechanics of helices. *Physical review. E*, 62(5 Pt B):7135–46, 2000. (cited on Pages 91 and 97)

[57] A. S. Parnovsky. Some generalisations of brachistochrone problem. *Acta Physica Polonica A*, 93, Supplement S-55, 1998. (cited on Page 59)

[58] L. A. Pars. *An Introduction to the Calculus of Variations.* Heinemann Educational Books, 1962. (cited on Pages 20, 21, and 22)

[59] L. A. Pars. *A Treatise on Analytical Mechanics.* Heinemann, London, 1965. (cited on Page 79)

[60] M. A. Patterson and A. V. Rao. Exploiting sparsity in direct collocation pseudospectral methods for solving optimal control problems. *Journal of Spacecraft and Rockets*, 49(2):364–377, 2011. (cited on Page 106)

[61] M. A. Patterson and A. V. Rao. GPOPS - II: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software,* under review, 2013. (cited on Page 106)

[62] G. Perantoni and D. J. N. Limebeer. A direct method for optimal control and optimal design of two-wheeled vehicles. *The 11th International Symposium on Advanced Vehicle Control*, September 2012. (cited on Pages vi and 119)

[63] G. Perantoni and D. J. N. Limebeer. A dynamic motorcycle tyre model based on lugre friction. *The 11th International Symposium on Advanced Vehicle Control*, September 2012. (cited on Pages vi and 4)

[64] G. Perantoni and D. J. N. Limebeer. Minimum-lap-time optimal control for a formula one car with variable parameters. *Vehicle System Dynamics.* under review, 2013. (cited on Pages vi, 6, 89, 90, and 121)

[65] G. Perantoni and D. J. N. Limebeer. Optimal control of a formula one car on a three-dimensional track part 1: Track modelling and identfication. *Journal of Dynamic Systems, Measurement and Control.* under review, 2013. (cited on Pages vi, 5, and 92)

[66] G. Perantoni and D. J. N. Limebeer. Optimal control of a two-mass skate bicycle without steering. *IEEE International Conference on Industrial Technology*, February 2013. (cited on Pages vi and 119)

[67] G. Perantoni and D. J. N. Limebeer. Time-optimal control of rolling bodies. *International Journal of Control*, 86, 2013. (cited on Pages vi, 4, and 59)

[68] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko. *The Mathematical Theory of Optimal Processes.* Interscience Publishers, 1962. (cited on Page 25)

[69] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *NUMERICAL RECIPES - The Art of Scientific Computing.* Cambridge University Press, third edit edition, 2007. (cited on Page 30)

[70] J. Åkesson, K. E. Årzén, M. Gäfvert, T. Bergdahl, and H. Tummescheit. Modeling and optimization with optimica and jmodelica.org-languages and tools for solving large-scale dynamic optimization problems. *Computers and Chemical Engineering*, 34:1737–1749, 2010. (cited on Page 118)

[71] S. M. Rappaport and Y. Rabin. Differential geometry of polymer models: worm-like chains, ribbons and fourier knots. *J. Phys. A: Math. Theor.*, 40:4455 – 4466, 2007. (cited on Pages 91 and 97)

[72] E. J. Routh. *Advanced Part of a Treatise on the Dynamics of a System of Rigid Bodies.* McMillan Co., London, sixth edition, revised and enlarged edition, 1905. (cited on Page 79)

[73] H. Scherenberg. Mercedes-Benz racing design and cars experience. *SAE Transactions*, (580042):414–420, 1958. (cited on Page 115)

[74] B. Siegler, A. Deakin, and D. Crolla. Lap time simulation: Comparison of steady state, quasi-static and transient racing car cornering strategies. In *SAE Technical Paper 2000-01-3563*, 2000. (cited on Page 118)

[75] D. G. Stork and J. Yang. The general unrestrained brachistochrone. *Am. J. Phys.*, 56(1):22–26, 1988. (cited on Pages xii, 59, 64, 68, and 69)

[76] D. G. Stork, J. Yang, and C. Stover. The unrestrained brachistochrone. *Am. J. Phys.*, 54(11):992–997, 1986. (cited on Pages 59 and 86)

[77] D. J. Struik. *Lectures on Classical Differential Geometry.* Dover, New York, second edition, 1988. (cited on Page 92)

[78] H. J. Sussmann and J. C. Willems. 300 years of optimal control: from the brachystochrone to the maximum principle. *IEEE Control Systems Magazine*, 17(3):32–44, June 1997. (cited on Page 8)

[79] H. J. Sussmann and J. C. Willems. The brachistochrone problem and modern control theory. *Contemporary Trends In Nonlinear Geometric Control Theory And Its Applications*, 113-166, 2002. (cited on Page 58)

[80] J. Synge and B. Griffith. *Principles of Mechanics.* McGraw-Hill Book Company, 3rd edition, 1959. (cited on Page 79)

[81] D. Tavernini, M. Massaro, E. Velenis, D. I. Katzourakis, and R. Lot. Minimum time cornering: the effect of road surface and car transmission layout. *Vehicle System Dynamics*, 2013. (cited on Page 116)

[82] M. Thommyppillai, S. A. Evangelou, and R. S. Sharp. Car driving at the limit by adaptive linear optimal preview control. *Vehicle System Dynamics*, 47(12):1535 — 1550, 2009. (cited on Page 119)

[83] J. P. Timings and D. J. Cole. Minimum manoeuvre time of a nonlinear vehicle at constant forward speed using convex optimisation. In *Proceedings of The 10th International Symposium on Advanced Vehicle Control*, Loughborough, UK, 2010. (cited on Page 119)

[84] J. P. Timings and D. J. Cole. Minimum maneuver time calculation using convex optimization. *Journal of Dynamic Systems, Measurement, and Control*, 135:0310151–1 to 0310151–9, 2013. (cited on Pages 90 and 119)

[85] B. Vratanar and M. Saje. On the analytical solution of the brachistochrone problem in a non-conservative field. *Int. J. Non-Linear Mechanics*, 33(3):489–505, 1998. (cited on Page 59)

[86] A. Wächter. *An interior point algorithm for large-scale nonlinear optimization with applications in process engineering.* PhD thesis, Carnegie Mellon University, 2002. (cited on Pages 32 and 33)

[87] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006. 10.1007/s10107-004-0559-y. (cited on Pages xi, 106, and 133)

[88] J. H. White and W. R. Bauer. Calculation of the twist and the writhe for representative models of DNA. *Journal of Molecular Biology*, 189(2):329 – 341, 1986. (cited on Page 91)

[89] P. Willemsen, J. K. Kearney, and H. Wang. Ribbon networks for modeling navigable paths of autonomous agents in virtual urban environments. In *Virtual Reality, 2003. Proceedings. IEEE*, pages 79–86, 2003. (cited on Page 91)

[90] J. Yang, D. G. Stork, and D. Galloway. The rolling unrestrained brachistochrone. *Am. J. Phys.*, 55(9):844–847, 1986. (cited on Pages 59, 70, 74, 75, and 87)