

Optimal Motion Cueing for Race-Car Simulators



Ingrid Gael Salisbury

New College

Department of Engineering Science

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Hilary Term 2016

Abstract

Optimal Motion Cueing for Race-Car Simulators

Ingrid Gael Salisbury

New College

Submitted for the degree of *Doctor of Philosophy* in Hilary Term 2016

This thesis is a study of motion cueing algorithms in race-car simulators. Motion cues are accelerations that are used by a driver to infer particular vehicle behaviours. Motion cueing algorithms use vehicle accelerations to generate simulator accelerations. Typically, these algorithms aim to produce motion cues that indicate the vehicle acceleration and behaviour, without reproducing exactly the vehicle motion. The cueing algorithms also prevent the simulator motions from hitting the hardware kinematic and dynamic limits.

Existing simulator platforms and cueing algorithms, most of which have been developed for passenger car simulators, are discussed to determine how these can be applied to race-cars. In the process, the distinguishing features of the race-car application are identified. The large accelerations and high-frequency motions are a challenging combination to render in a power and workspace limited simulator. Furthermore, the driver is required to operate the vehicle on the performance limit, which is a difficult control task requiring particular vehicle information. The study of fundamental vehicle stability and handling across the range of operating conditions is conducted to understand better the vehicle motion cues required by the simulator driver to perform the race-driving task.

With a knowledge of the driver cueing requirements, passenger car cueing strategies are modified and implemented in a Formula 1 simulator. Alternative cueing methods are also developed that address the racing cueing problem. Tests are conducted and on the basis of the driver feedback the algorithms are retuned to develop a cueing system that enables the drivers to execute their tasks successfully.

The implemented cueing algorithms do not consider explicitly the optimal usage of simulator workspace. The final element of this thesis is the use of numerical optimal control to determine the optimal platform workspace exploitation, while minimising cueing acceleration and velocity errors. The solutions of open-loop optimal control calculations are used as a baseline against which to compare the tested cueing techniques. The limitations of, and areas in which to improve the implemented methods are identified.

Acknowledgements

This thesis would not have been possible without the support, encouragement and contributions of many people.

Firstly and foremost, I would like to thank Prof. David Limebeer, who has been an invaluable research collaborator and mentor. His constant reminder that the only way to eat an elephant is one bite at time has been the best advice for tackling any problem, not least the writing of this thesis. Both his encouragement and criticism (and occasional insistence that I “trust him – he is a Professor”) have made it an honour and such fun to work under his supervision.

For their advice, help and testing facilities I would like to thank Scuderia Ferrari. I have been lucky to work with a great Formula 1 team, and to be exposed to the world of motorsport. In particular I would like to thank Giacomo Tortora, Nikhil and Franco for their time, insightful discussions and for giving me the benefit of their experience.

Studying at Oxford would not have been possible without the scholarship from the Emma Smith Foundation. I am truly grateful to the organisation for funding me and helping me pursue what can only be described as a dream.

I would like to thank the whole of the Oxford control group for providing a stimulating academic environment in which to work. Among my colleagues I would like to thank my office-mates Tom, Giacomo, Akis and Moritz for both their help and teatime discussions.

One of the highlights of my time in Oxford has been rowing. The freezing early mornings were frequently accompanied by beautiful sunrises and a chance to see Oxford at its best. To have been part of a club that is as old as my home town, Pietermaritzburg, has been an unbelievable experience. NCBC, I thank you, and GDBM.

The love and support of my parents, sisters and friends cannot be overstated. I would not have made it this far without them. In particular, I would like to thank Ashleigh Reid. She has always been ready to listen to my troubles, encourage me, and her sense of fun has brightened countless days.

I reserve my final thanks for Anthony Potter, whose love and strength are invaluable. He and Pringle have kept me laughing and have made these last few years an exciting and amazing adventure.

I.G. Salisbury
Oxford, February 2016

Contents

List of Figures	vii
List of Tables	xvi
Abbreviations	xvii
Publications	xviii
1 Introduction	1
1.1 Simulator Components	2
1.2 Race Car Applications	3
1.3 Thesis Outline	5
1.4 Contributions of Thesis	6
2 Simulator Motion Platforms	8
2.1 Motion Platform Designs	9
2.1.1 Stewart Platform	11
2.1.2 Advanced Hexapod Motion Platforms	14
2.1.3 Desdemona	17
2.1.4 CyberMotion Simulator	17
2.1.5 Driver in Motion Simulator	19
2.1.6 Vehicle Dynamics Simulator	20
2.2 Simulator Kinematics	22
2.2.1 Reference Frames	23
2.2.2 Euler Angles and Angular Velocity	23
2.2.3 Hexapod Inverse Kinematics	25
2.2.3.1 Actuator Length	25
2.2.3.2 Actuator Velocity	27
2.2.3.3 Actuator Acceleration	28
2.2.4 Hexapod Workspace	28
2.3 Summary	30
3 Motion Cueing Algorithms	32

CONTENTS

3.1	Vestibular Models	34
3.1.1	Semi-Circular Canals	34
3.1.1.1	Mathematical Model	34
3.1.1.2	Rotational Velocity Threshold Values	36
3.1.2	Otoliths	36
3.1.2.1	Mathematical Model	37
3.1.2.2	Linear Acceleration Threshold Values	38
3.2	Classical Filtering Technique	39
3.3	Nonlinear Coordinated Adaptive Algorithm	41
3.4	Optimal Cueing	42
3.5	MPC Cueing	45
3.6	Spherical Algorithms	48
3.7	Prepositioning	49
3.8	Summary	50
4	Vehicle Stability Analysis	53
4.1	Introduction	53
4.2	Stability Theory	56
4.2.1	Stability of Linear, Time-varying systems	57
4.3	Straight Line Motion	59
4.3.1	Vehicle Model	59
4.3.2	Constant Velocity Stability	62
4.3.3	Accelerating and Braking Stability	65
4.3.3.1	Closed-Form Solution	66
4.3.3.2	Results	71
4.3.3.3	Validation of the Eigenvalues	71
4.3.3.4	Real Eigenvalues	72
4.3.3.5	Complex Eigenvalues	74
4.4	Cornering Stability	75
4.4.1	Constant Velocity	76
4.4.2	Accelerating and Braking Under Cornering	80
4.5	Model Extensions	84
4.6	Conclusion	87
5	Motion Cueing Implementation	90
5.1	Longitudinal Cueing Filter Design	91
5.1.1	Optimal Filter Design	92
5.1.1.1	Basic Optimal Algorithm	92
5.1.1.2	Frequency Weighted Cost Function	94
5.1.1.3	False Cue Penalty	95
5.1.1.4	Removal of Vestibular Model	96
5.1.2	Non-Linear Modifications	97

CONTENTS

5.1.3	Anti-backlash Filter	97
5.1.4	Miscue Reshaping Module	99
5.2	Lateral and Yaw Cueing Filter Design	100
5.2.1	Centre of Turning	101
5.2.2	Lateral Filter Design	102
5.2.3	Yaw Filter Design	103
5.3	MPC	104
5.4	Summary	105
6	Experimental Results	107
6.1	Longitudinal Testing	109
6.1.1	Optimal Filter Designs	110
6.1.1.1	Basic Optimal Filter	111
6.1.1.2	Frequency Weighting Functions	115
6.1.1.3	False Cue Penalty Filter	116
6.1.1.4	Cueing Filters without Vestibular Model	118
6.1.2	Non-Linear Filter Testing	119
6.2	Lateral and Yaw Testing	123
6.2.1	Lateral Filter Tuning	123
6.2.2	Yaw Filter Testing	128
6.3	MPC Cueing Strategy	131
6.4	Conclusion	134
7	Numerical Optimal Control	138
7.1	Optimal Control Theory	140
7.2	Optimal Control Problem Formulation	141
7.2.1	Longitudinal Cost Development	143
7.2.2	Lateral Cost Development	151
7.2.3	Yaw Cost Development	155
7.3	Optimal Cueing for a Racing Lap	155
7.3.1	Longitudinal Cueing	156
7.3.2	Lateral Cueing	158
7.3.3	Yaw Cueing	159
7.3.4	Longitudinal and Lateral Combined Cueing	159
7.3.5	Full 3 DOF Cueing	163
7.4	Workspace Extensions	167
7.5	Comparison of Cueing Techniques	170
7.5.1	Longitudinal Comparison	170
7.5.2	Lateral Comparison	172
7.5.3	Yaw Comparison	173
7.6	Summary	174

8 Conclusion	177
9 Future Work	181
9.1 Further Experimental Testing	181
9.2 Platform Design Optimisation	183
9.3 Passenger Car Applications	184
A Vehicle and Tyre Parameters	186
B 4-Wheeled Vehicle Model	188
B.1 Vehicle body	188
B.2 Suspension System	189
B.3 Aerodynamic Forces	190
B.4 Tyres	191
B.5 Speed and Steering Controllers	191
C Linear Cueing Filters	193
C.1 Longitudinal Weights and Filters	193
C.2 Lateral Weights and Filters	194
C.3 Yaw Weights and Filter	195

List of Figures

1.1	Components of a generic vehicle simulator and their relationship to the driver.	2
1.2	The left- and right-hand figure contain the frequency responses of the translational and rotational acceleration signals respectively.	4
2.1	An example of a typical Stewart platform. This large, high-fidelity hexapod is installed at the Technical University of Delft in the Netherlands [49].	10
2.2	A smaller hexapod system with simple controls and graphics, installed at the University of Loughborough [25].	13
2.3	An inverted hexapod motion platform, at the DLR in Germany [71].	13
2.4	A graphic reproduction of the Citroen simulator platform with 8 DOF [10]. This is an example of a large workspace, advanced hexapod centred system.	15
2.5	The Desdemona simulator platform [19], which is characterised by a cylindrical workspace with large yaw capabilities.	16
2.6	The CyberMotion platform [53], which is a novel simulator design, equipped with unlimited yaw, large longitudinal workspace and capable of sustaining lateral accelerations.	18
2.7	The Driver in motion platform produced by VI-Grade [7].	19
2.8	Simulator platform designed specifically for vehicle dynamics and race-car applications [29].	20
2.9	A schematic of a Formula 1 car, showing the body-fixed and head-fixed reference frames.	23
2.10	Hexapod motion platform with the body-fixed and inertial reference frames, and leg, base-joint and platform-joint vectors.	26
2.11	Body-fixed lateral and longitudinal hexapod workspace boundaries for increasing yaw angles from 0° (black) to 25° (yellow) in steps of 5°	29
3.1	The Bode plot of the semi-circular canal model used in this work. The highlighted region corresponds to the frequency range of the rotational motions experienced in a race-car.	36

3.2	The Bode plot of the otolith model used in this work, corresponding to the parameters given in right-most column of Table 3.2. The highlighted range corresponds to the frequency of motions experienced in a race-car.	38
3.3	Structure of the classical cueing system with tilt-coordination. The input signals a_{ref}^x and $\dot{\phi}_{ref}$ are the vehicle longitudinal acceleration and pitch velocity, respectively. The output signals, p_{sim}^x and ϕ_{sim} are the platform longitudinal and pitch displacement demands. . .	40
3.4	Formulation of the optimal motion cueing problem.	43
4.1	Sample G-G diagram of a Formula One car that shows the vehicle's lateral and longitudinal acceleration on a qualifying lap at Silverstone. The measured vehicle data has been filtered to remove noise. The track contains 18 corners, of which 10 are right-handed, that accounts for the vehicle's positive-dominated lateral acceleration. In addition, the high-speed corners, out of which the driver accelerates, are also right-handed, which is consistent with vehicle's prolonged operation in the top-right quadrant of the diagram. The highlighted regions correspond with simultaneous braking and cornering, a condition in which the vehicle is typically unstable. .	54
4.2	Kinematics of a single-track vehicle model showing its basic geometric parameters.	59
4.3	The left-hand figure shows the real part of the poles and zeros of the 2 DOF model given in (4.33) for straight-running vehicle speeds between 20 m/s and 80 m/s. The poles are the roots of the characteristic polynomial given in (4.34). The zeros of the lateral acceleration, side-slip angle and yaw velocity transfer functions are computed using (4.36), (4.37) and (4.38), respectively. In the case of the lateral acceleration, the zeros form a complex conjugate pair. The right-hand figure shows the responses of the lateral acceleration, side-slip angle and yaw velocity transfer functions to step changes in the steering input. These responses are computed at longitudinal speeds of $u_0 = \{30 \text{ m/s}, 50 \text{ m/s}, 70 \text{ m/s}\}$ corresponding respectively to the magenta, green and cyan curves.	66
4.4	The left-hand figure shows the real parts of the frozen-time eigenvalues of a vehicle braking in straight running at -10 m/s^2 for vehicle speeds from 80 m/s to 20 m/s. The right-hand figure shows the yaw response to a non-zero initial yaw velocity at 80 m/s. . . .	70
4.5	The left-hand figure shows the real part of the frozen-time eigenvalues of a vehicle accelerating in straight running at 10 m/s^2 from 20 m/s to 80 m/s. The right-hand figure shows the yaw response to a non-zero initial yaw velocity at 20 m/s.	70

4.6	The left-hand plot contains the numerical evaluation of the bounds in (4.59) and (4.60) for vehicle speeds from 80 to 20 m/s, for a vehicle braking at -10 m/s^2 . The right-hand plot shows the vehicle critical speed, and the limit speed as functions of the vehicle acceleration.	74
4.7	Numerical evaluation of conditions (4.63) and (4.64) for vehicle speeds between 20 and 80 m/s, with a vehicle accelerating at 10 m/s^2 . Under accelerating conditions there is no critical speed.	75
4.8	The left-hand figure shows the real-part of the eigenvalues of the 3 DOF vehicle model plotted for different vehicle speeds at vehicle accelerations of $a_x = 5 \text{ m/s}^2, 0 \text{ m/s}^2, -5 \text{ m/s}^2$,. The right-hand figure shows the zeros of each of the transfer functions across different speeds and at the different acceleration levels.	78
4.9	The left-hand figure shows the root-locus plot for two of the three eigenvalues of the 3 DOF vehicle model under cornering; the varied parameter is the vehicle's speed. The eigenvalues corresponding to the longitudinal speeds $V = \{48 \text{ m/s}, 49 \text{ m/s}, 50 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right-hand figure shows the step responses of the 3 DOF model at the listed longitudinal speeds. The top plot shows the lateral acceleration responses to a step in steering angle, the centre plot shows the body side-slip angle response, while the bottom plot illustrates the yaw rate response. The cyan and magenta curves are associated with the scale on the left axes, while the green curves with those on the right.	79
4.10	The left-hand figure shows the root-locus plot of the eigenvalues of the 3 DOF vehicle model under 5 m/s^2 acceleration. At low speeds there are three negative real eigenvalues. As the speed increases, the pole near the origin remains there, whilst the other two move together, coalesce, and form a complex-conjugate pair. While the complex conjugate pair move towards the imaginary axis as the vehicle speed increases, the car loses traction before they cross into the right-half plane. The eigenvalues corresponding to speeds of $V = \{40 \text{ m/s}, 45 \text{ m/s}, 50 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right-hand figure shows the step response plots for the 3 DOF model at the cited longitudinal speeds. From top to bottom, the plots show the lateral acceleration, the side-slip angle and the yaw velocity responses to a step change in the steering angle. The green curves are associated with the right-hand axes, while the cyan and magenta curves are associated with the left-hand axes.	81

4.11	The left figure contains the root-locus plot of two of the eigenvalues of the 3 DOF vehicle model under -5 m/s^2 braking. As the vehicle speed increases, one pole moves away from the origin as the other moves towards it. These then coalesce and form a complex conjugate pair that migrates into the right-half plane. At higher speeds these become two positive real poles, with one moving towards the origin and the other away from it. The eigenvalues corresponding to the longitudinal speeds $V = \{35 \text{ m/s}, 36 \text{ m/s}, 37 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right figure contains the step response plots for a 3 DOF model, computed at the listed longitudinal speeds. From top to bottom, the sub-plots contain the responses of the lateral acceleration, body side-slip angle and yaw velocity transfer functions to a step in steering angle. The cyan and magenta curves are associated with the scale on the left axes, while the green curves with those on the right.	83
4.12	Plan view of a Formula One car with some of its basic geometric parameters.	84
4.13	The left figure contains the eigenvalues for the Formula 1 car under steady-state conditions, 7.5 m/s^2 of acceleration and -15 m/s^2 of braking. The right figure contains the root-locus of the eigenvalues for increasing vehicle speed.	86
5.1	The topology of the standard optimal cueing algorithm, modified to consider only the longitudinal motion.	92
5.2	The basic optimal, translational cueing algorithm recast in the generalised regulator framework.	93
5.3	The basic optimal, translational cueing algorithm recast in the generalised regulator framework with frequency weights applied to the elements of the cost function.	95
5.4	The translational optimal cueing algorithm with a miscue penalty introduced to the cost function.	95
5.5	The left figure contains the error introduced by a typical cueing filter for a race-car manoeuvre, and the error sensed by the vestibular system. The right figure, similarly, compares the yaw velocity error in the simulator with the sensed error.	96
5.6	Optimal filter cueing structure with an additional anti-backlash module.	98
5.7	Structure of the cueing filter with a miscue reshaping module.	99
5.8	The kinematics of an oversteering vehicle navigating a turn are illustrated in the left-hand figure, while those of an understeering vehicle are illustrated in the right-hand diagram.	101
5.9	The optimal lateral cueing filter design problem cast in the generalised regulator framework [30, 60].	103

LIST OF FIGURES

5.10	The yaw-cueing washout filter optimisation problem in generalised regulator form.	104
5.11	The plant dynamics of the MPC cueing problem.	105
6.1	The racing line on the Barcelona Circuit with corner numbers. . .	109
6.2	The left figure contains the speed profile for a lap of the track and the right figure contains the lateral and longitudinal accelerations and the yaw velocity around the circuit.	109
6.3	The Bode magnitude plots of the longitudinal cueing filters tested with the race drivers.	111
6.4	The left figure contains the longitudinal acceleration cueing signals produced by the different filter options for the braking into and accelerating out of Turn 1 of the track. The right figure shows the onset cues (the first 2s) more clearly.	111
6.5	The two plots contain the platform longitudinal velocity and position demands for each of the cueing options, for the braking/accelerating manoeuvre in Figure 6.4.	112
6.6	The platform displacement during an entire lap of the Barcelona track is given for each of the longitudinal cueing options. The platform limits are also shown to illustrate the fraction of workspace that is utilised.	112
6.7	The left plot contains the longitudinal acceleration demands produced by the anti-backlash and miscue reshaping techniques and the Option 2 linear filter for comparison. The right figure contains the acceleration cueing signals for the first 2s to show the difference in the onsets.	119
6.8	The two plots contain the longitudinal velocity and position demands, respectively, associated with the anti-backlash and miscue reshaping techniques and the Option 2 linear filter for comparison.	120
6.9	The left plot contains the platform displacement for an entire lap, as well as the platform limits. The right figure contains a more detailed view of the motion cues under acceleration.	120
6.10	The left-hand plot contains the longitudinal vehicle acceleration for the first eight corners of the Barcelona Circuit. The vehicle's lateral velocity (blue) and the yaw velocity (red) are displayed in the right-hand plot.	123
6.11	The Bode magnitude plot of the three lateral cueing filter options.	124

6.12	Reference lateral acceleration for the section of the Barcelona circuit highlighted in Figure 6.1. Figure (a) shows (in black) the vehicle's lateral acceleration. The three coloured plots are the simulator lateral acceleration demands for the three $\mathcal{H}_2$ optimal filters. Figures (b) and (c) show, respectively, the associated platform velocity and displacement demands for each of the three filters. Figure (d) shows the vehicle responses from Figure 4.9, filtered using $F_1(s)$, to determine the platform steering-angle step response at three different vehicle speeds. Figures (e) and (f) contain the same analysis for $F_2(s)$ and $F_3(s)$	126
6.13	The yaw cueing filter Bode magnitude response.	128
6.14	Yaw velocity and side-slip angle cues. The left-hand figure shows the vehicle's yaw velocity (associated with the right axis), the filtered yaw velocity and the vehicle side-slip angle velocity (both associated with the left axis). The right-hand plot shows the platform yaw angle for the filtered yaw velocity and the vehicle side-slip angle.	129
6.15	The left plot contains the longitudinal acceleration demands produced by the the two MPC options and the miscue reshaping technique for comparison. The right figure contains the acceleration cueing signals for the first 2s to show the difference in the onsets.	134
6.16	The left plot contains the longitudinal velocity and the right plot the position demands associated with the two MPC options and, for comparison, the miscue reshaping technique.	134
6.17	The three plots contain a comparison of the MPC lateral cue and the best linear lateral cue option (Option 2). The figures contain from left to right the acceleration, velocity and position demands for the simulator.	135
6.18	The left figure compares the MPC yaw velocity demand and the yaw velocity produced by the linear filter. The right plot compares the corresponding platform yaw angles with the vehicle side-slip angle.	135
7.1	Formulation of numerical optimal control problem.	141
7.2	This collection of figures contains longitudinal cueing signals determined by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands produced for a braking manoeuvre using different cost functions. The bottom two figures contain the platform velocity and displacement demands, respectively.	144
7.3	The plot shows the nonlinear weighting function given in (7.13). The output of the function is 1 when the platform and vehicle accelerations are in opposite directions and 0 when they are in-phase.	147

LIST OF FIGURES

7.4	This set of figures contains longitudinal cueing signals determined for an accelerating manoeuvre by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands, and the bottom two figures contain the platform velocity and displacement demands, respectively.	150
7.5	This collection of figures contains the lateral cueing signals determined by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands produced for a cornering manoeuvre. The bottom two figures contain the associated platform velocity and displacement demands, respectively.	153
7.6	This figure contains the vehicle yaw velocity, and the platform yaw velocity cue produced by the numerical optimal control computation using a cost function that minimises the angular velocity error.	154
7.7	Track and racing line for the first four corners of the Barcelona circuit used in numerical optimal control simulations [38, 51]. The red markers are spaced 5 seconds apart, with the first being at 0 seconds and the last at 20; the car travels in a clockwise direction.	156
7.8	The left figure shows the longitudinal acceleration cueing signal determined by the numerical optimal control computation for the given longitudinal acceleration reference. The right figure shows the corresponding simulator velocity and position demand.	157
7.9	The left figure shows the lateral acceleration cueing signal determined by the numerical optimal control computation for the given lateral acceleration reference. The right figure shows the corresponding simulator velocity and position demand.	158
7.10	The left figure shows the yaw velocity reference and cueing signal computed by the numerical optimal control solver. The right figure shows the associated platform rotational position demand.	159
7.11	The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis. The lateral motion is given priority.	160
7.12	The simulator velocity and position demands for the longitudinal (left figure) and lateral (right figure) directions. These are associated with the acceleration demands given in Figure 7.11.	161
7.13	The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis. The longitudinal motion is given priority.	162

LIST OF FIGURES

7.14	The simulator velocity and position demands for the longitudinal (left figure) and lateral (right figure) directions. These are associated with the acceleration demands given in Figure 7.13.	163
7.15	The platform boundary and the use of the workspace over the four corners for the cueing option associated with the acceleration demands in Figure 7.11 (lateral cue priority) and with those in Figure 7.13 (longitudinal cue priority).	163
7.16	The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The problem set-up includes all 3 DOF, and the yaw motion is given a low priority. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis.	165
7.17	The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The problem formulation includes all 3 DOF, and the yaw motion is given a higher precedence than the translational cues. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis.	165
7.18	The vehicle yaw velocity and the simulator yaw velocity cues produced when the yaw motion is given a low (blue) and a high (red) priority in the full 3 DOF motion optimal control computation.	166
7.19	This set of figures contains the simulator translational velocity and position, and rotational position demands for the full 3 DOF motion problem. The platform motion signals for both the low (Opt 1) and high (Opt 2) yaw priority are given. The top right figure contains the longitudinal velocity and position demands. The top left figure shows the lateral velocity and position demands. The bottom figure shows the two yaw platform position demands.	168
7.20	These two figures describe the simulator workspace utilisation over a lap of the Barcelona circuit, as determined by the optimal control. The left figure shows the workspace use computed using the standard hexapod actuator limits, and the right figure the platform motion using actuators with double the velocity limits.	169
7.21	These figures show the improvement in the lateral cueing signal for an increase in simulator workspace. The left figure contains the lateral acceleration demand computed by the optimal control solver for the standard workspace (Opt 1) and with a 50% increase in workspace size (Opt 2). The right figure displays the associated platform velocity and position demands.	169

LIST OF FIGURES

7.22	These figures compare the longitudinal cueing demands produced by the optimal control computation, and the linear and nonlinear filters. The left figure shows the acceleration cues, and the right top and bottom plots the platform velocity and displacement demands, respectively.	171
7.23	These figures compare the lateral cueing demands produced by the optimal control computation, and the linear filtering method. The left figure shows the acceleration cues, and the right figure contains the platform velocity and displacement demands for both cueing options.	173
7.24	These figures compare the yaw cueing demands produced by the optimal control computation, and the linear filtering approach. The left figure shows the yaw velocity cues, and the right figure the associated platform yaw angle.	174
A.1	The right figure contains the longitudinal tyre force for different normal loads and zero side-slip angle. The left figure contains the lateral tyre force for different normal loads and zero longitudinal tyre slip.	187
B.1	Plan view of a Formula One car with some of its basic geometric parameters.	189
B.2	Suspension system of a Formula 1 car. L_1 through L_4 represent the double-wishbones; L_5 is the steering push rod, and L_6 is the suspension push rod to which is connected the spring-damper pair.	190

List of Tables

1.1	Table of typical ranges in acceleration, velocity and displacement in the vehicle 6 DOF; all quantities are in SI units with the exception of angles that are given in degrees.	5
3.1	A Comparison of Rotational Velocity Threshold Values	36
3.2	Otolith Model Parameters	38
3.3	Linear Acceleration Threshold Values	38
A.1	Formula 1 car vehicle parameters.	186
A.2	Tyre model parameters.	187
C.1	Table of the values of the weighting parameters for the longitudinal optimal filter designs.	193
C.2	Table of the values of the weighting parameters for the lateral optimal filter designs.	194
C.3	Table of the values of the weighting parameters for the yaw optimal filter designs.	195

Abbreviations

COM	C entre of M ass
DOF	D egree of F reedom
EOM	E quation of M otion
LQR	L inear Q uadratic R egulator
MPC	M odel P redictive C ontrol
SAE	S ociety of A utomotive E ngineers

Publications

I G Salisbury and D J N Limebeer, “Motion Cueing in High-Performance Vehicle Simulators”, *Vehicle Systems Dynamics*, under review.

I G Salisbury and D J N Limebeer, “Optimal Motion Cueing for Race Cars”, *IEEE Trans Control System Technology*, Vol 24. No 1, January, 2016

I G Salisbury, D J N Limebeer, A Tremlett and M Massaro, “The Unification of Acceleration Envelope and Driveability Concepts”, 24th International Symposium on Dynamics of Vehicles on Road and Tracks (IAVSD), Graz, Austria, 2015.

I G Salisbury and D J N Limebeer, “An Algorithm for False Cue Reduction and Prepositioning”, *Driving Simulation Conference*, Paris, France, 2014.

I G Salisbury and D J N Limebeer, “Optimal Race Car Motion Cueing”, The 19th World Congress of the International Federation of Automatic Control, Cape Town, South Africa, 2014.

Chapter 1

Introduction

The rising costs of vehicle development have resulted in an increases requirement for simulation and virtual testing. Driving simulators are used to conduct driver-in-the-loop experiments. In this manner, vehicle concepts and new designs can undergo a first stage of testing without the need to make any physical part of the vehicle. Furthermore, driving simulators can simulate critical situations to test vehicle safety features, which cannot be attempted in a real vehicle.

In recent years, driving simulators have become more important in the Formula 1 industry, where new regulations limit the amount of track testing time. Additionally, with the ever increasing complexity of the vehicle software systems more troubleshooting and debugging is required, and it is more efficient to do this in a simulator. In racing, each track requires a different vehicle set-up and much of the preliminary preparation can be done in the simulator. In these tests the driver feedback about the vehicle handling is critical, because the vehicle behaviour heavily influences the ability of the driver to operate the vehicle on the limit. The need for and advantages of using driving simulators are clear, however, the value of the simulator results are dependent on the fidelity of the simulator environment, and hence the extent to which the drivers respond realistically.

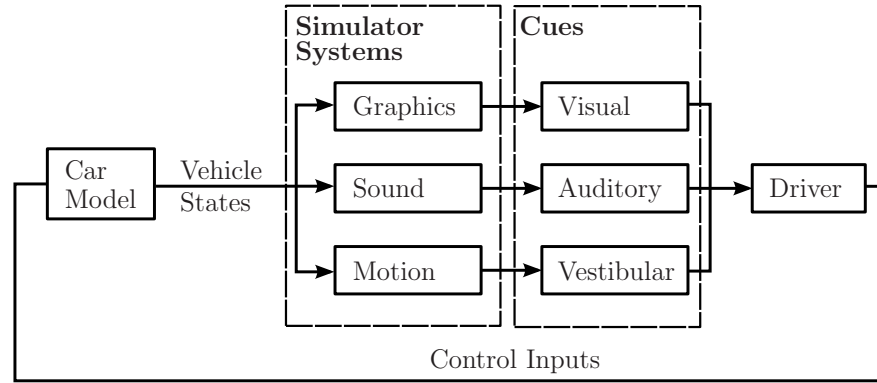


FIGURE 1.1: Components of a generic vehicle simulator and their relationship to the driver.

1.1 Simulator Components

A realistic driving environment is created in a high-fidelity simulator by mounting a vehicle cab on an actuated platform surrounded by audio-visual equipment. The driver cockpit is a replica of the simulated vehicle and is equipped with controls that mimic those in the car. The driver also receives realistic sensory stimuli (referred to as cues) and thus, ideally, behaves and feels as he/she would in the actual vehicle.

There are numerous systems that supply the driver with the required sensory information. Visual display graphics are of primary importance, providing images of the location and driving conditions, while also conveying the vehicle's speed. Auditory cues and haptic feedback, which is provided through the steering wheel and pedals, communicate a range of vehicle-related dynamic information to the driver including the engine speed and tyre slip. Acceleration and orientation is detected in humans by the vestibular system in the inner ear [17]. A dynamic simulator is able to stimulate this organ thereby providing an enhanced sense of the motion of the vehicle.

Figure 1.1 shows the integration of various subsystems into a complete simulator and illustrates how the driver generates control commands in response to sensory

cues. These inputs are fed into a vehicle model that generates the vehicle-related state information including its acceleration, velocity and position. This data is then used to update the visual, audio and motion systems.

The fidelity of the simulator driving experience is necessarily limited by the quality of the cues. Advanced 3D image projection with fast refresh rates and low latency together with high-quality computer generated images (CGI) produce an excellent visual environment. In the case of motion stimuli, however, the platform workspace and bandwidth restrict the fidelity of the motion cues. This is a physical limitation rather than a technological one, since reproducing the vehicle motion exactly, requires a workspace as large as the track itself. Since this is practically infeasible, motion cueing algorithms have been developed that seek to mimic driving-related motions within the confines of the physical system. To observe these limits, motion cueing algorithms often contain washout filters that cause the acceleration to decay as the platform nears the limits, thereby "washing out" the motion of the simulator.

Motion cueing algorithms have to balance the trade-off between cueing fidelity and platform motion. A successful cueing strategy is one that sacrifices the unnecessary vehicle motion, while retaining the acceleration stimuli that are required to produce realistic driver responses. The nub of the cueing problem is identifying these important acceleration stimuli. The motion cues needed in the simulator depend on a number of factors, including the experience of the driver, the type of vehicle being driven and the driving scenario.

1.2 Race Car Applications

While there are common criteria for all driving simulators, there are also situation-specific differences that must be addressed. The simulator discussed here is used

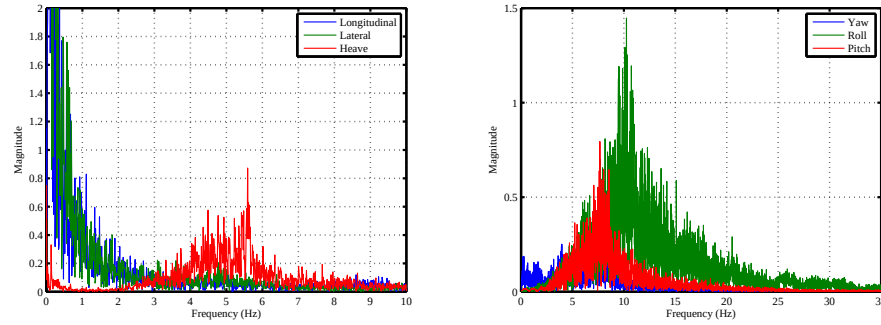


FIGURE 1.2: The left- and right-hand figure contain the frequency responses of the translational and rotational acceleration signals respectively.

in race car applications, where the primary functions are driver training and car set-up ahead of races.

Race car driving is characterised by high levels of longitudinal and lateral acceleration (in excess of 4Gs), which cannot be sustained in a confined simulator workspace. This makes the motion cueing problem challenging and it is important to identify the information the drivers wish to receive from the motion platform. The drivers make it clear that they appreciate that a simulator cannot mimic a real race car. However, they require accurate feedback about the vehicle behaviour during braking to detect lock-up, and during cornering to detect under and oversteer. Understanding the cues that the drivers require assists in both the design and assessment of cueing algorithms.

A further characteristic of this application is that the car is typically driving around a known closed track. In racing the aim is to minimise lap-time, and to achieve this there is an optimal speed profile and racing line [38, 51]. Given that experienced drivers generally drive the car close to this optimum, there is little variation in the acceleration profiles between laps. As a result the braking and cornering situations are highly predictable.

A detailed analysis of the acceleration and angular velocity trajectories that need to be simulated is provided by examining track telemetry data. Table 1.1 contains

TABLE 1.1: Table of typical ranges in acceleration, velocity and displacement in the vehicle 6 DOF; all quantities are in SI units with the exception of angles that are given in degrees.

Direction	Acceleration	Velocity	Displacement	Frequency
Longitudinal	-40:20	-70:70	-650:500	0-5
Lateral	-50:50	-65:72	-800:200	0-5
Heave	-15:15	-0.5:0.5	-0.3:0.25	2-7
Yaw	-20:20	-1.2:1.1	0:360	0-12
Pitch	-30:30	-0.65:0.65	-2:2	2-8
Roll	-80:80	-1.2:1.2	-5:5	2-25

a representative range of the accelerations, velocities and displacements a race car will experience during a typical lap of a track. The two graphs in Figure 1.2 display the frequency content of the acceleration signals for the lap. There are some important problem features that are highlighted in this data. Firstly, in the translational freedoms, both longitudinal and lateral motions contain more low-frequency information when compared with the heave direction. The vertical displacement is small and it should be possible to reproduce this exactly in a simulator setting. This result is intuitive when one thinks of the motion of a vehicle on a hard suspension system. Similarly, in the rotational freedoms, the roll and pitch angles are only a few degrees. These angular freedoms should also require little, or no filtering before being reproduced by the simulator. The outcome of this analysis is a reduction of the motion cueing problem from 6 DOF to only 3 DOF (longitudinal, lateral and yaw). Determining the cueing requirements for these large motion directions, and implementing filters to address these, is the aim of this thesis.

1.3 Thesis Outline

The development of the race-car cueing strategies requires a model of the hardware which contains the kinematic and dynamic limitations. Chapter 2 compares the different motion platforms in use and explores the characteristics of each and

their applicability to racing. A mathematical model for the platform used in this study is developed and the shape of the workspace is determined.

The existing motion cueing methods that have been developed for the passenger car context are described in Chapter 3. How these can be applied in the racing context is discussed, and the gaps in the cueing methodology are identified.

Chapter 4 contains an analysis of the stability of a race-car. A method is developed to examine the behaviour across all operating conditions, not limited to steady-state. An understanding of the changing vehicle response is used to develop cueing requirements for the vehicle heading control directions, that is, the lateral and yaw directions.

The implementation details of the motion cueing algorithms are provided in Chapter 5, and the tuning and experimental results are then presented in Chapter 6.

The cueing algorithms focus on providing the necessary motion cues, but do not explicitly consider making optimal use of the workspace. This problem is considered in Chapter 7, where numerical optimal control is used to develop a baseline cueing signal that simultaneously satisfies the cueing requirements and fully exploits the platform workspace.

The conclusions and suggested future work are presented in Chapters 8 and 9, respectively.

1.4 Contributions of Thesis

The work presented in this thesis is based on a number of papers, some published and one still under review. A detailed list is provided previously under Publications.

The focus of this thesis is on motion cueing in driving simulators for race-car applications. This extends from understanding the particular hardware requirements, to testing cueing strategies with drivers.

The first contribution of this work is to determine the motion cues race-car drivers require in the simulator. Vehicle behaviour is studied across the performance envelope to identify the critical information for effective race-car motion simulation. The second contribution of this work is the adaptation of existing passenger car cueing strategies to race-car simulators using the cueing criteria developed. New cueing methods specifically for the race-car problem are also designed. These algorithms are implemented in the Scuderia Ferrari Formula 1 car simulator and tested with their professional drivers.

The development of an offline optimal control framework for the analysis and optimisation of the simulator workspace usage is the final contribution. This section uses the previously identified cueing requirements to develop performance indices for optimal control calculations, which are used to determine the best platform usage, and the effect of simulator architecture and sizing on the cueing fidelity.

Chapter 2

Simulator Motion Platforms

The first car simulators, and many still in existence today, do not have motion capabilities. Indeed, a realistic driving environment can be created using the cab of a vehicle with actuated steering and pedals, and advanced graphics systems. These simulators have the advantage of being cheaper to build, and simulation fidelity is sufficient for many applications, such as, hazard avoidance training of novice drivers [1]. However, there is still sensory information missing, which leads to unrealistic driving behaviour, such as excessive braking [63].

While the visual system is used to determine speed and distance [34], acceleration is detected by the vestibular system (biological accelerometers) [63]. Hence, in non-moving simulators sudden changes in acceleration are unobserved and the level of acceleration is miscalculated. Moving platforms enhance the sensory perception, and lead to improvements in driver behaviour. These include lower jerk at the onset of braking, lower lateral acceleration in cornering and lower deceleration in braking [15]. Additionally, vehicle characteristics, such as the road and ride feel, which contain high frequency vibrations, cannot be communicated visually. A motion platform, with appropriate dynamic capabilities, is required if these are to be reproduced [63].

The first part of this chapter, Section 2.1, examines the different available simulator hardware. The capabilities of the simulator platform affect the motion cueing algorithm design. The limitations of the particular platform should be taken into account, while the advantages must be fully exploited. With this in mind, the kinematics of the simulator platform used in this research are described in detail in Section 2.2.

2.1 Motion Platform Designs

There are a number of factors that affect the quality of a vehicle simulator experience, and it is costly to build a system that can excel in all of these. For example, a simulator that is highly realistic and contains a test vehicle, will necessarily be heavy, hence it will be difficult to actuate at high frequencies and so there will be a loss of road feel. Consequently, in any motion platform design, there will have to be compromises on the performance in certain areas. The choice of compromise depends on the application.

In broad terms, there are two classes of simulator: high-immersion and vehicle dynamics. The first group are simulators that aim to convince the driver that he/she is in a real vehicle. These simulators are typically used for human factors experiments. These are studies that examine the interaction of the driver with the vehicle and environment. This can be used to test human behaviour in critical situations in order to design driver assistance systems. The efficacy of which can then also be tested in the simulator.

The second class of simulators are used to study the vehicle behaviour. Typically, this will involve a professional test driver, operating the vehicle on the limit to determine the ease with which it can be controlled. These vehicle handling and response studies are particularly important in the design and set-up of high-performance road and race cars.



FIGURE 2.1: An example of a typical Stewart platform. This large, high-fidelity hexapod is installed at the Technical University of Delft in the Netherlands [49].

For the two classes of simulators, there are very different simulator motion requirements. The high-immersion simulators are usually for passenger car applications, in which the motions are typically smoother and lower in magnitude. In contrast, race car simulators are characterised by quickly changing, large magnitude accelerations. The kinematic and dynamic characteristics of the motion platform, therefore, depend heavily on the application. In the first case, the platform needs to deliver low frequency accelerations for long time periods, whereas in the second case the platform needs to be highly responsive.

With these two general types of simulator in mind, the different motion platform designs are examined. In each case, the simulator's suitability for its application is discussed, together with the constraints that make it inapplicable to other situations.

2.1.1 Stewart Platform

The Stewart platform remains one of the oldest and most commonly used motion bases in vehicle simulators. This 6 DOF hexapod parallel manipulator originated in the aircraft industry. Based on a mechanism designed by Gough in 1948 [66], Stewart presented the platform design for a flight simulator in 1965 [70]. However,

it was only 20 years later that the hexapod was first used by Daimler-Benz as a base for a car simulator [18].

Examples of typical Stewart platforms are shown in Figures 2.1 and 2.2. The simulator platform is mounted on 6 actuated legs, and by changing the lengths of these the position and orientation of the platform in space is controlled. The motion is kinematically constrained by the actuator stroke length, and the range and types of joints connecting the legs to the platform and the ground. The power of the actuators limits the achievable platform accelerations and the system bandwidth.

Being a parallel manipulator introduces a number of disadvantages. Firstly, the motion directions are coupled, so movement in one direction reduces the available workspace of the other freedoms. This makes it difficult to utilise the full platform workspace under all operating conditions. A detailed analysis of this problem is given in Section 2.2.4. Additionally, from a dynamic perspective, the frequency response and acceleration capabilities in all freedoms are the same, which is not consistent with the properties of a real vehicle. As shown in Chapter 1, the roll, pitch and heave motions associated with the vehicle suspension are of a higher frequency than the driving motions of the vehicle.

A further feature of the Stewart platform is the isometric workspace. In the context of flight simulators, where there are large accelerations in all directions, this is necessary [29]. However, as shown in Chapter 1, the nature of ground vehicles is such that the vehicle motions in the heave, pitch and roll directions are orders of magnitude smaller than those in the remaining freedoms. However, in many simulator applications sustained longitudinal and lateral accelerations are simulated by tilting the driver, so the large pitch and roll workspaces are utilised [10].

Despite the aforementioned limitations, hexapods have an enduring popularity.



FIGURE 2.2: A smaller hexapod system with simple controls and graphics, installed at the University of Loughborough [25].

They vary greatly in size and motion capabilities, and hence can be used for many different applications.

A large simulator, such as that in Figure 2.1, will usually have the cab of a car with controls, and the graphics system (projectors and screens) contained in the dome which is mounted on the actuated platform. The mechanism can support the high pay-load, in excess of 2 tonnes, since the mass is spread across the 6 legs. This type of system is highly realistic. The driver sits in a real vehicle with functioning mirrors, the correct vehicle sounds, and a visual environment that extends to a wide field of view. Such simulators are used in human behaviour studies, since it is easy for the drivers to become immersed in the simulation and believe they are in a real vehicle. The large mass of the simulator necessarily leads to a low platform bandwidth. As a result, much of the road and suspension feel is lost, and the simulator is not suitable for the study of the ride of a vehicle.

A smaller hexapod system, such as that in Figure 2.2, creates, overall, a far less realistic experience. Minimalist driving seat and controls are mounted on the platform with a simple visual system provided by three monitors. These lighter simulators can achieve a higher bandwidth, but typically have a smaller workspace. Consequently, these are more useful in applications that require the vehicle ride information, or that assess the vehicle dynamics which relies on a low



FIGURE 2.3: An inverted hexapod motion platform, at the DLR in Germany [71].

latency platform response.

A third inverted hexapod simulator design, from the Deutsches Zentrum für Luft- und Raumfahrt (DLR) in Germany, is shown in Figure 2.3. In this simulator the connection points of the legs to the dome are roughly at the level of the driver's head. This means the centre of rotation for cueing can be done at a point nearer the driver's head without sacrificing any motion performance [33].

The first two simulator examples illustrate the ends of the hexapod spectrum, with the third being an example of one of the many variations in between. The choice of hexapod workspace size and dynamic capability depends on the research application, budget, and often the available room space.

2.1.2 Advanced Hexapod Motion Platforms

The translational motion of a Stewart platform is typically limited to less than a metre. Within these constraints, only the transient vehicle dynamics (0.4Hz-3.0Hz) can be rendered, and alternative strategies, such as tilt co-ordination, need to be used to simulate the sustained accelerations [57]. To combat these limitations many hexapod simulators have been mounted on moving bases to create a system with more than 6 DOF, which has additional capacity in any or all of the lateral, longitudinal and yaw directions.

To determine the additional workspace needed in the translational directions, the motion requirements for two simulator applications were studied by Daimler during their facility upgrade [87]. In the first application, the vehicle drive dynamics are assessed by a professional driver. This includes an evaluation of the vehicle behaviour in slalom and lane change manoeuvres. Ideally, to achieve a realistic experience the lateral accelerations would be reproduced exactly, which requires a workspace of 8m [87]. The second application of the simulator is the testing of driver assistant systems, which is performed with ordinary drivers [87]. The main requirement in these scenarios is to create an overall realistic environment, which is achieved predominately through good visual cues, and the simulator motion produced by the hexapod was found to be sufficient [87]. The feasibility of generating sustained longitudinal accelerations was examined, but it was determined that a workspace much larger than 10m was required for any significant improvement. The cost-benefit ratio of such a large simulator was not favourable, and hence, in 2010 the Daimler simulator consisting of a hexapod on 12m lateral rails was built. A quasi-static turntable was included, allowing the simulator to be rotated through 90 degrees. This flexibility allows better longitudinal cueing during start-stop traffic scenarios when a large lateral workspace is not required [87].

To extend both the lateral and longitudinal capabilities, the hexapod simulators of both the Renault [13] and Citroen [10] car manufacturers were placed on XY tables [66]. The Citroen simulator is shown in Figure 2.4. These additional axes provide a large workspace. However, since the entire hexapod is being moved (hence the mass is much greater), the achievable accelerations will necessarily be lower than those of the hexapod (the Renault platform can accelerate at up to 0.5g in the horizontal directions [13]). In these duplicated DOF the high frequency and larger magnitude accelerations are reproduced by the hexapod, and the low



FIGURE 2.4: A graphic reproduction of the Citroen simulator platform with 8 DOF [10]. This is an example of a large workspace, advanced hexapod centred system.

frequency, long duration motions by the XY-moving base. In the passenger car applications for the which the simulator is used, the driving manoeuvres have relatively low accelerations, for long time periods, hence the simulator caters to these requirements. The drivers' sensations during steady-state cornering and braking and accelerating are improved with respect to the basic hexapod.

The National Advanced Driving Simulator (NADS) in the USA has 12 DOF, with all motion freedoms duplicated. The hexapod is placed on a yaw turntable which is placed on an XY plotter. This allows a large workspace in all the long duration motion freedoms. Additionally, to provide high frequency cueing information the vehicle cab within the dome is vibrated [69]. While the mechanism has 12 DOF, and a car only 6 DOF, the design uses different sub-systems to cue the different freedoms and frequency ranges. In the passenger car context this simulator satisfies many of the cueing requirements. It can maintain steady-state cueing, as well as provide ride feel in the vehicle. However, it has a limited capability for high magnitude accelerations, which feature in high performance vehicles.

Similar to the NADS, is the Toyota simulator, which is currently the largest in the world with a hexapod moving on a 25mx35m two axis sled [33], [66].

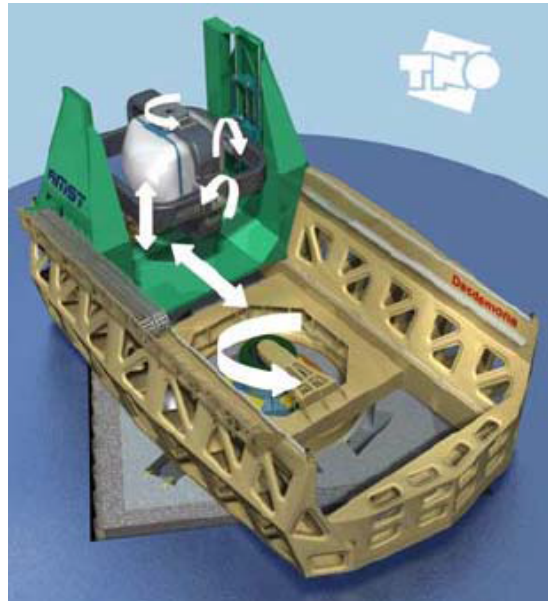


FIGURE 2.5: The Desdemona simulator platform [19], which is characterised by a cylindrical workspace with large yaw capabilities.

2.1.3 Desdemona

A novel simulator, known as Desdemona, shown in Figure 2.5, was built at the Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek (TNO) Safety and Security in the Netherlands. It is capable of producing sustained lateral accelerations useful for curve driving experiments [19].

The simulator cab is placed in a gimballed system which allows 3 unlimited rotational DOF. This system can be moved vertically by 1m and horizontally along an 8m rail. The unlimited rotational motion allows the cab to be oriented laterally or longitudinally on the rail. The rail itself is able to spin on a central vertical axis. This design allows the simulator to move along a circular path, and the centripetal acceleration of the simulator produces a sensation of sustained lateral acceleration [59].

A further advantage of this system is that the vertical, pitch, roll and yaw freedoms are not coupled so can be independently controlled - a result of being a

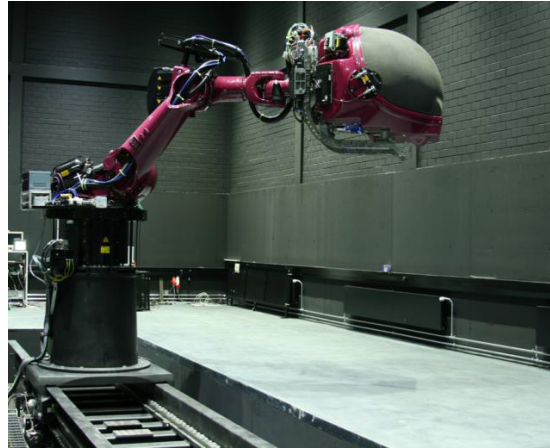


FIGURE 2.6: The CyberMotion platform [53], which is a novel simulator design, equipped with unlimited yaw, large longitudinal workspace and capable of sustaining lateral accelerations.

serial as opposed to parallel manipulator. This also means that each DOF is easily introduced sequentially, simplifying the development of the cueing algorithm. The disadvantage of the circular workspace is that, in the generation of sustained lateral and longitudinal accelerations, unwanted Coriolis and Centrifugal forces also result. Traditional cueing strategies designed for the hexapod systems do not take advantage of the simulator workspace [81]. Both of these issues are addressed by an alternative cueing algorithm which is described in Section 3.6.

2.1.4 CyberMotion Simulator

The simulator installed at the Max Plank Institute for Biological Cybernetics at the University in Tübingen has a novel architecture which can be used for studies that are not possible in a standard hexapod.

A Keller und Knappich Augsburg (KUKA) Robocoaster has been modified to create a platform with 8 decoupled DOF. The mechanism is shown in Figure 2.6, and consists of a robotic arm, with a motorised cabin attached [28]. The base of the arm is mounted on a sledge which can move on a 9.88m rail [53]. Being a serial mechanism allows independent control of the joints and results in a simulator

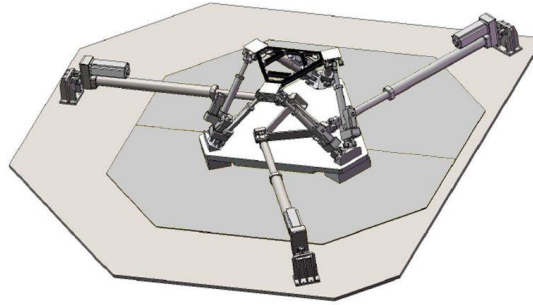


FIGURE 2.7: The Driver in motion platform produced by VI-Grade [7].

with a much larger workspace and capable of more dexterous motions [75]. The rotational joints in the robotic arm provide unlimited yaw, which can be used to produce sustained lateral acceleration, or yaw motion, depending whether the cabin is located on or perpendicular to the vertical axis. The linear rail provides additional longitudinal motion.

This simulator is used for human factors and perception experiments, and is suitable for use with a wide variety of vehicles including cars, and helicopters [46]. The cabin interior can be modified to accommodate different vehicle controls. The platform architecture is particularly useful for conducting perception threshold experiments, since yaw, roll and lateral motion can be maintained without requiring the motion washout which characterises a Stewart platform [53].

2.1.5 Driver in Motion Simulator

The Driver in Motion (DiM) simulator is produced by VI-Grade and is designed for vehicle dynamics applications. The simulator consists of a hexapod, mounted on a plate that is actuated by a planar tripod, which provides additional yaw, lateral and longitudinal motion. The DiM mechanism aims to improve the yaw capabilities of the simulator, and provide a motion envelope better suited to road car applications.

The DiM simulator is smaller than the other advanced hexapod systems described in Section 2.1.2, and only the cab of the vehicle, not the graphics are moved. The lower mass, and stiff hexapod design allow a higher bandwidth, which is required in vehicle dynamics studies [7]. Furthermore, the main purpose of the tripod is to improve the yaw motion, and not to provide sustained accelerations. Indeed, the tripod can achieve translational accelerations of approximately 1G, which, with a workspace of less than 2m, cannot be sustained. However, by offloading the yaw, the hexapod is able to render more motion in the remaining freedoms[7].

Unlike the other systems with more than 6 DOF, it is not immediately obvious which motions should be produced by each platform. The high frequency components that cannot be rendered by the tripod must be generated by the hexapod. However, the lower frequency motions that can be produced by both systems need to be split between the two mechanisms. A non-linear model predictive control (MPC) algorithm is presented in [7] that determines the division of motion, such that optimal usage is made of each platform. In certain configurations, the tripod workspace maybe be exhausted in a given direction, in which case the hexapod will be used instead.

2.1.6 Vehicle Dynamics Simulator

The design of a specialised vehicle dynamics simulator is described in a UK Patent [29]. This mechanism is intended to address the limitations of the Stewart platform in the high performance, and race-car context.

The simulator platform described in [29], and shown in Figure 2.8 consists of a front and back rail system; each comprises two sledges mounted on parallel rails. The platform is connected to the 4 sledges, which can move independently. The sides of the platform perpendicular to the base rails are tapered, not parallel. Thus when the sledges are moved apart or together the platform moves in surge.

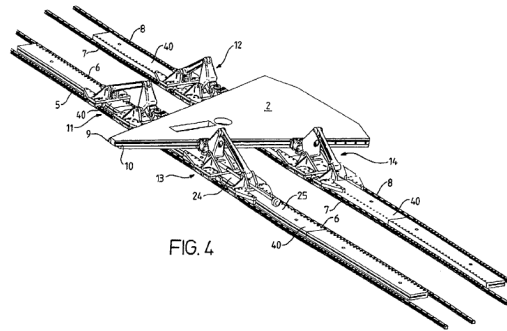


FIGURE 2.8: Simulator platform designed specifically for vehicle dynamics and race-car applications [29].

By moving all 4 sledges left or right simultaneously, sway motion is produced. Yaw is created by moving the two back sledges in the opposite direction to the two front sledges. The platform is connected to the sledges with universal joints that can be moved up and down. By moving the 4 joints independently roll, pitch and heave are created. The system is over-constrained, having 8 actuators to produce motion in 6 DOF, hence the actuators must be controlled in a coherent fashion.

The advantage of this design is that large horizontal forces and displacements can be generated by relatively small mechanisms. The system is non-isotropic and yaw, surge and sway have greater workspaces, which is consistent with the vehicle motion characteristics. These lower frequency freedoms are also decoupled from the other 3 which are characterised by higher frequency motions. It is argued in [8] that it is difficult to re-create the braking and acceleration forces in a restricted workspace so there is little value in trying to do this. It is more important that the driver receives the correct information about the vehicle in cornering, which comes from lateral and yaw cues [82]. As a consequence more workspace is given to these. Finally the smaller, light system has a higher frequency response than a standard hexapod, hence the system latency is reduced which improves the

drivers' perception of the vehicle behaviour, particularly when the vehicle exhibits fast responses, such as at the limit of stability.

2.2 Simulator Kinematics

Cueing studies have shown that, since the human mechanism for sensing translational and rotational accelerations (vestibular system) is located in the inner ear, the aim in motion cueing should be to subject the simulator driver's head to the same accelerations as those experienced by the head of the actual vehicle driver [72].

The car models used to generate the vehicle telemetry data typically produce body-fixed, inertial accelerations of the vehicle's COM. By knowing the location of the driver's head in the vehicle it is possible to compute its acceleration. The details of this calculation are provided in Section 2.2.1. In addition, if the rotational motion of the vehicle is parametrised using Euler angles, then the Euler angle rates and not the vehicle angular velocity are the appropriate stimuli to the vestibular system, since the rotational sensors are aligned with each of the axes of the body-fixed frame. The transformation of angular velocity and acceleration to Euler angle velocities and accelerations are provided in Section 2.2.2.

The motion cueing filtering is performed on the accelerations of the driver's head, and the Euler acceleration signals. The resulting cueing demands are transformed back into angular accelerations, and the acceleration of the simulator COM. Finally, the simulator actuator motion demands are computed from the simulator COM acceleration signals. The transformations from platform to actuator motion are referred to as the inverse kinematics and are detailed in Section 2.2.3.

The motion platform used in this study is a conventional Stewart Platform. The cab, which consists of the race car cock-pit, screens and projectors, is mounted on six actuated legs, which are connected to the platform with spherical joints

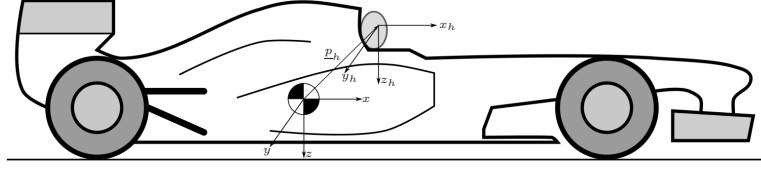


FIGURE 2.9: A schematic of a Formula 1 car, showing the body-fixed and head-fixed reference frames.

and to the ground-mounted base plate with pin joints. By controlling the six actuated leg lengths the cab can be moved with 6 DOF; see Figure 2.10.

During cueing development, the inverse kinematics of the platform are used to ensure that the platform boundaries are respected and to determine the platform workspace. In driving simulator applications, large displacements are only required in the longitudinal, lateral and yaw directions. Hence, it is convenient to visualise the changing in the translational workspace as a function of the platform yaw angle. This analysis is given in Section 2.2.4.

2.2.1 Reference Frames

There are two important reference frames in the vehicle, which are shown in Figure 2.9. The first is the body-fixed frame located at the vehicle COM, in which the vehicle accelerations are expressed as inertial quantities. The second frame is the body-fixed frame located at the driver's head. The origin of the head frame in the car body frame is given as $\underline{p}_h$.

The output of the vehicle model is the acceleration of the vehicle's COM ($\underline{a}_{COM}$), and, for cueing purposes the acceleration of the driver's head ($\underline{a}_h$), must be computed from:

$$\underline{a}_h = \underline{a}_{COM} + \dot{\underline{\omega}} \times \underline{p}_h + \underline{\omega} \times (\underline{\omega} \times \underline{p}_h), \quad (2.1)$$

where, $\dot{\underline{\omega}}$ and $\underline{\omega}$ are the vehicle angular velocity and acceleration, respectively.

2.2.2 Euler Angles and Angular Velocity

The angular velocity and acceleration of the vehicle are converted to Euler angle velocities and accelerations required for motion cueing. For consistency with the vehicle modelling, the 3-2-1 Euler angle sequence is adopted. This corresponds to a first rotation about the z -axis (yaw angle (ψ)), then about the y -axis (pitch angle (θ)) and finally about x -axis (roll angle (ϕ)). With each rotation is associated a rotation axis $e_\psi = [0 \ 0 \ 1]^T$, $e_\theta = [0 \ 1 \ 0]^T$ and $e_\phi = [1 \ 0 \ 0]^T$, and a rotation matrix, R_ψ , R_θ and R_ϕ . Lastly, three reference frames are defined which correspond with each Euler rotation, F_ψ , F_θ and F_ϕ respectively.

The angular velocity of the vehicle is the rotation of the final frame F_ϕ , and is expressed in body-fixed co-ordinates. It can be computed by summing the contributions of each of the Euler rotations, as follows:

$$\underline{\omega} = \underline{\omega}_{F_\phi} + \underline{\omega}_{F_\theta} + \underline{\omega}_{F_\psi}. \quad (2.2)$$

This can be written as a function of the Euler angle velocities:

$$\underline{\omega} = e_\phi \dot{\phi} + R_\phi^T e_\theta \dot{\theta} + R_\phi^T R_\theta^T e_\psi \dot{\psi}, \quad (2.3)$$

where, the axes of rotation have been transformed to body-fixed co-ordinates using the appropriate rotation matrices. This can be expressed more compactly as:

$$\underline{\omega} = B \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}, \quad (2.4)$$

where,

$$B = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & s_\phi c_\theta \\ 0 & -s_\phi & c_\phi c_\theta \end{bmatrix}. \quad (2.5)$$

The angular acceleration can be found by differentiating equation (2.2) and noting that the derivative of a vector in a moving reference frame is given by:

$$\frac{d}{dt} \left(\underline{\omega}_{F_i/F_j} \right)_{F_k} = \frac{d}{dt} \left(\underline{\omega}_{F_i/F_j} \right)_{F_j} + \underline{\omega}_{F_j/F_k} \times \underline{\omega}_{F_i/F_j}. \quad (2.6)$$

The expression for the angular acceleration in terms of the Euler angle velocities and accelerations is then found to be:

$$\underline{\dot{\omega}} = \underline{\alpha} = B \begin{bmatrix} \ddot{\psi} \\ \ddot{\theta} \\ \ddot{\phi} \end{bmatrix} + (e_\phi \dot{\phi}) \times (R_\phi^T e_\theta \dot{\theta}) + (R_\phi^T e_\theta \dot{\theta} + e_\phi \dot{\phi}) \times (R_\theta^T R_\phi^T e_\psi \dot{\psi}). \quad (2.7)$$

2.2.3 Hexapod Inverse Kinematics

The development of the inverse kinematic equations for a hexapod system is a standard procedure and can be found in [14].

2.2.3.1 Actuator Length

Referring to Figure 2.10, an inertial reference frame xyz with origin O is defined at the geometric centre of the hexapod base. A second body-fixed frame $x'y'z'$ is similarly defined with its origin C at the geometric centre of the platform. The translation of the platform from the origin of the inertial reference frame is described by the vector $\underline{S}$ and the matrix R describes the platform orientation in terms of the 3 Euler angles. The rotation matrix for the 3-2-1 Euler angle

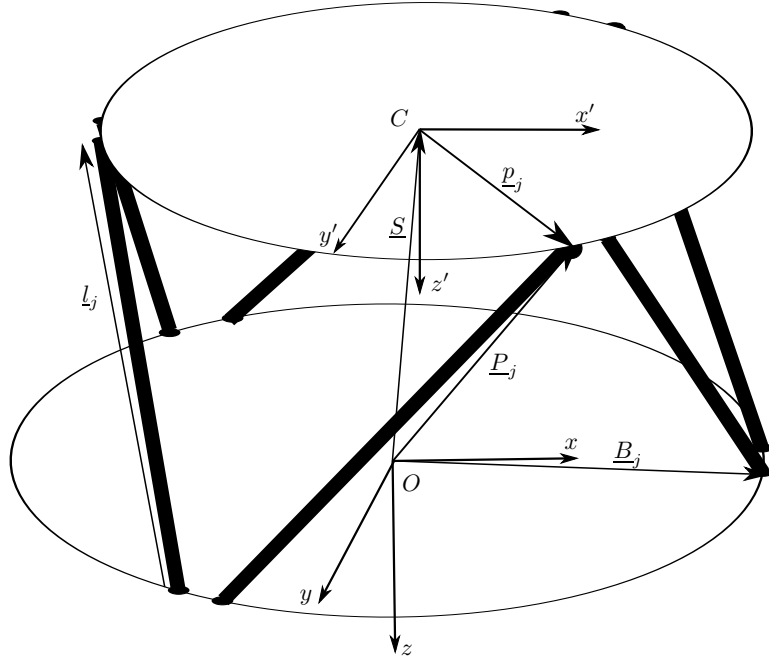


FIGURE 2.10: Hexapod motion platform with the body-fixed and inertial reference frames, and leg, base-joint and platform-joint vectors.

sequence is given by

$$R = R_z(\psi)R_y(\theta)R_x(\phi) \quad (2.8)$$

$$R = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}, \quad (2.9)$$

where $\cos(\theta)$ is abbreviated as c_θ and $\sin(\theta)$ as s_θ , and so on.

The base joints of the hexapod legs are described by six vectors $\underline{B}_j$ in the inertial frame, and the platform joints are described by vectors $\underline{p}_j$ in the body-fixed frame.

The platform joints can then be described by vectors $\underline{P}_j$ in the inertial frame, which are related to $\underline{p}_j$ by:

$$\underline{P}_j = \underline{S} + R \cdot \underline{p}_j \quad (j = 1, \dots, 6) \quad (2.10)$$

The platform legs can be described by vectors $\underline{l}_j$, which point from the base joints to the corresponding platform joints and can be found as the difference between the platform and base vectors. The leg lengths (l_j) are simply the magnitude of the leg vectors and $\hat{\underline{l}}_j$ are unit vectors associated with the directions of the legs.

$$\underline{l}_j = l_j \hat{\underline{l}}_j = \underline{P}_j - \underline{B}_j \quad (j = 1, \dots, 6) \quad (2.11)$$

2.2.3.2 Actuator Velocity

Suppose that the leg angular velocities are $\underline{\omega}_{Lj}$ and that their vector-valued rates of change of length $\dot{\underline{l}}_j$ have magnitudes $\dot{l}_j$ in directions $\hat{\underline{l}}_j$. Thus

$$\dot{\underline{P}}_j = \dot{l}_j \hat{\underline{l}}_j + \underline{\omega}_{Lj} \times \underline{l}_j. \quad (2.12)$$

It follows from (2.10) that

$$\dot{\underline{P}}_j = \dot{\underline{S}} + \underline{\omega} \times \underline{p}_j. \quad (2.13)$$

Equating (2.12) and (2.13) and dotting the result with $\hat{\underline{l}}_j$ gives

$$\dot{l}_j = \left[\begin{array}{cc} \hat{\underline{l}}_j & \underline{p}_j \times \hat{\underline{l}}_j \end{array} \right] \cdot \left[\begin{array}{c} \dot{\underline{S}} \\ \underline{\omega} \end{array} \right] \quad (2.14)$$

since $\underline{l}_j$ is parallel to $\hat{\underline{l}}_j$.

The leg angular velocities can be found from (2.12) as follows

$$\hat{\underline{l}}_j \times \dot{\underline{P}}_j = \hat{\underline{l}}_j \times \left(\dot{l}_j \hat{\underline{l}}_j + \underline{\omega}_{Lj} \times \underline{l}_j \right) \quad (2.15)$$

$$= \hat{\underline{l}}_j \times \underline{\omega}_{Lj} \times \underline{l}_j \quad (2.16)$$

$$= l_j \underline{\omega}_{Lj}, \quad (2.17)$$

since $\underline{\omega}_{Lj}$ is perpendicular to $\hat{l}_j$. Whence $\underline{\omega}_{Lj} = (\hat{l}_j \times \dot{\underline{p}}_j)/l_j$.

2.2.3.3 Actuator Acceleration

The accelerations of the platform joints can be found by differentiating (2.14) as follows

$$\ddot{l}_j = (\ddot{\underline{S}} + \underline{\dot{\omega}} \times \underline{p}_j) \cdot \hat{l}_j + (\dot{\underline{S}} + \underline{\omega} \times \underline{p}_j) \cdot \dot{\hat{l}}_j \quad (2.18)$$

since $\dot{\underline{p}}_j = 0$. Whence

$$\ddot{l}_j = \begin{bmatrix} \hat{l}_j & \underline{p}_j \times \hat{l}_j \end{bmatrix} \cdot \begin{bmatrix} \ddot{\underline{S}} \\ \underline{\dot{\omega}} \end{bmatrix} + \frac{d}{dt} \left(\begin{bmatrix} \hat{l}_j & \underline{p}_j \times \hat{l}_j \end{bmatrix} \right) \cdot \begin{bmatrix} \dot{\underline{S}} \\ \underline{\omega} \end{bmatrix} \quad (2.19)$$

in which we can use (2.11) to show that

$$\dot{\hat{l}}_j = (\dot{\underline{p}}_j - \dot{l}_j \hat{l}_j)/l_j. \quad (2.20)$$

2.2.4 Hexapod Workspace

An important feature of the hexapod motion platform is the shape of the workspace. Since movement in 6 DOF is created by the same six actuators, motion in one direction affects the capabilities of the platform to move in another. In this application large displacements are required in the longitudinal, lateral and yaw directions, and so the workspace usage relationships between these 3 freedoms are examined.

Figure 2.11 shows the workspace limits for the lateral and longitudinal motion for various values of platform yaw angle¹. By focussing temporarily on the heavy

¹For confidentiality reasons these are representative, and not the actual dimensions of the hexapod used for testing.

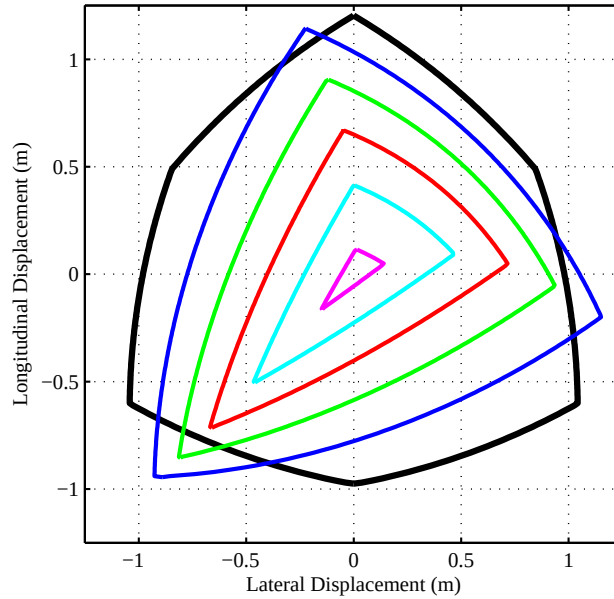


FIGURE 2.11: Body-fixed lateral and longitudinal hexapod workspace boundaries for increasing yaw angles from 0° (black) to 25° (yellow) in steps of 5° .

(black) plot, which corresponds to a yaw angle of zero, one observes that movement in either the longitudinal or lateral direction reduces the workspace available in the other. The leg configuration employed in this study makes the zero-yaw-angle workspace symmetrical about the x-axis. The workspace perimeter is reached when one or more of the leg actuators reaches its maximum or minimum length limit. The six vertices in this figure correspond to points in configuration space where multiple legs saturate and there is a change over between saturating legs.

Figure 2.11 also shows the lateral and longitudinal workspace bounds for increased values of yaw angle. As the platform rotates (yaws), the workspace is no longer symmetrical and it becomes smaller and more ‘triangular’ in shape. This irregularly shaped workspace has consequences for the motion cueing algorithm design. In most cueing algorithms each freedom is tuned individually for a predetermined workspace. If the actual available workspace is less than expected,

because of workspace coupling, the platform may not fully render the cueing signal. Consequently, it may be better to design for worst-case displacement limits, even though this results in the workspace not being fully utilised.

2.3 Summary

The different simulator motion platforms have been presented, and it seems that the most popular is the Stewart platform. It is characterised by 6 DOF and by changing the size can be used in many applications. It is also common to improve the hexapod workspace by placing it on rails or an xy-plotter.

Other simulators have been designed to meet particular criteria. In the cases of the CyberMotion and Desdemona simulators, this is to produce unlimited yaw, and sustained lateral accelerations.

In any application it is important to consider the motion requirements during the simulator design. It is also necessary to make compromises, and accept that one simulator may not be able to achieve all the desired outcomes. Indeed, Daimler, have three simulators, a fixed based car simulator, a large hexapod on rails, and finally a simplistic, actuated seat simulator which is used for ride experiments. This approach can be more cost effective, and is certainly simpler to build.

For the specific race-car application a simulator capable of producing large magnitude accelerations, with a high-bandwidth and fast response is required. Of the simulators discussed, either an appropriately specified hexapod, or the two simulators discussed in Sections 2.1.5 and 2.1.6 which are aimed specifically at vehicle dynamics applications, could be used. In this thesis a hexapod platform was available for testing, and is used in all the studies.

The kinematic characteristics of the motion platform were determined as they are necessary for cueing algorithm design. Not only are they required to ensure the platform limits are adhered to, but are also useful to determine the shape of

the platform workspace. The latter calculation helps determine the applicability of the platform in the current application, as well as the interaction between the different motions. This helps to determine a priority or precedence in the cueing directions.

Chapter 3

Motion Cueing Algorithms

In broad terms dynamic driving simulators aim to reproduce vehicular accelerations [5], thereby supplementing the visual cues and improving the driver's perception of the vehicle behaviour. For example, under braking, motion cues assist the driver to brake at more realistic rates [63]. However, in order to satisfy workspace and bandwidth constraints, vehicular acceleration signals must be filtered and/or attenuated and consequently only part of the vehicle's movement is rendered by the motion platform. Motion cueing algorithms have evolved to address the design of these filters [60].

In a real vehicle, the movement stimulates the drivers' vestibular and somatosensory systems. The vestibular system is a biological mechanism for sensing rotational and linear acceleration, and accurate mathematical models have been developed for it. The somatosensory system perceives touch and pressure. This indirectly senses acceleration, for example, the increase in pressure on the back during large accelerations. Mathematical models do not exist for this complex system, and it is, therefore, not considered in the design of motion cueing algorithms. Instead, most of the cueing approaches described in the literature aim to minimise the difference between the accelerations perceived in the simulator and

those perceived in the real vehicle, where the perception model is a linear transfer function representing only the vestibular dynamics [73].

Before the different motion cueing algorithms are described, a brief description of the vestibular models used in many of these is given in Section 3.1. The numerous motion cueing strategies that are currently applied in a variety of simulators are then assessed so that their limitations can be identified and discussed. Several criteria can be used to critique a cueing strategy, including

1. Workspace usage: how well the displacement, velocity and acceleration capabilities of the platform match those of the vehicle, and how well they are being exploited;
2. Tuneability: the ease with which the algorithm can be adjusted in response to driver comments;
3. False and missed cues: false cues refer to simulator accelerations that are opposite in sign to those produced by the car model. Missing cues refer to accelerations that are present in the car model output, but are not in the simulator motion [68];
4. Implementation: a method may in principle produce good results, but if it cannot be successfully implemented, it is of little practical value.

The classical technique (described in Section 3.2) involves the design of acceleration filters using an iterative process to improve the performance and minimise the sensation error. The nonlinear coordinated adaptive algorithm presented in Section 3.3 changes the cueing filter gains such that the sensation error is less when the platform is close to the centre of the workspace, and performance only degrades as the platform nears the physical limits. Cueing filters that minimise vestibular sensation error and platform motion simultaneously are designed using

linear quadratic regulator (LQR) theory and are described in Section 3.4. In Section 3.5 MPC theory is used to create a cueing algorithm that minimises sensation error while explicitly respecting the workspace bounds. A cueing algorithm that makes best use of a circular workspace is described in Section 3.6. Finally, in Section 3.7 a technique is presented that improves workspace usage by predicting the upcoming manoeuvres and positioning the platform appropriately.

3.1 Vestibular Models

Inertial motion cues are sensed by two biological mechanisms, the semicircular canals, and the otoliths, both located in the inner ear, and, collectively known as the vestibular system. Simple linear models are used to represent these sensors in the development of motion cueing algorithms. However, both have non-linearities, in the form of thresholds below which no motion is detected.

3.1.1 Semi-Circular Canals

Angular motion is sensed by the two (one in each inner ear) sets of semi-circular canals. Each organ consists of three ducts containing fluid which flows as the head turns. This movement bends sensory hair cells, which causes a change in the firing rate of the vestibular afferent nerves. This is interpreted by the brain, and the sensation of turning is experienced.

3.1.1.1 Mathematical Model

The movement of the head is related to the motion detected by the brain by the following model [32]:

$$\frac{\hat{\omega}}{\omega} = \frac{\tau_a s}{\tau_a s + 1} \frac{\tau_1 s}{(\tau_1 s + 1)(\tau_2 s + 1)} (\tau_L s + 1). \quad (3.1)$$

The input and output of the model are the head angular velocity and the angular velocity sensed by a person, respectively. The model contains three elements, each of which has been contributed by different researchers. The middle term, developed by [78], and, known as the torsion-pendulum model, models the decay in turning sensation for a step change in angular velocity. This function alone did not capture all the observed effects. The torsion-pendulum model predicts, that for a constant angular acceleration there is a steady-state turning sensation, however, in experiments, this feeling decays to zero. Additionally, in response to a step-change in velocity, the model predicts an exponential decay, however an overshoot is observed in the sensed velocity. To account for these deviations, an adaptation operator, the first term in (3.1), was added [85].

The final term in the function is a lead term that was added by [44] to model the minimum response time of the system to large impulsive inputs.

The time constant τ_1 is not easily measured, but can be estimated from the canal fluid dynamics as 5ms for man [86]. Values of τ_L have been found to range from 17ms [44] to 60ms [47] which agrees with the 50ms found by [21] which is used here. The parameter τ_a was found by [21] to be 80s and this value has been used in most research since [72, 77].

There is a far greater variety in the accepted value of the long time constant, τ_2 , and, in some studies, the value is different for each of the motion axes. In motion cueing either the single value of 5.73s found by [21] is used (as it is in this work), or parameters of Melville-Jones with values of 10.2s, 5.3s and 6.1s for yaw, pitch and roll axes respectively [56].

The Bode plot of the semi-circular canal model used in this work is shown in Figure 3.1, with the highlighted region corresponding to the frequency range of the rotational motion of a race-car.

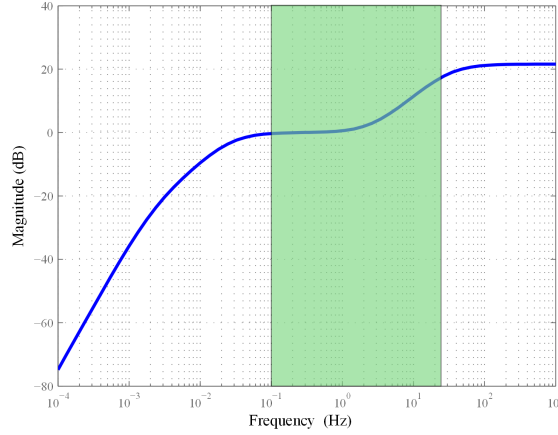


FIGURE 3.1: The Bode plot of the semi-circular canal model used in this work. The highlighted region corresponds to the frequency range of the rotational motions experienced in a race-car.

3.1.1.2 Rotational Velocity Threshold Values

The semi-circular model is approximated by a linear transfer function, however, the actual behaviour is nonlinear, with low magnitude rotational velocities below a threshold not being perceived. As it was noted in [86], there are numerous studies reporting different threshold values, but the two sets of values used by the different researchers in motion cueing development are shown in Table 3.1.

TABLE 3.1: A Comparison of Rotational Velocity Threshold Values

Reference	Yaw(deg/s)	Pitch(deg/s)	Roll(deg/s)
Nahon and Reid [56]	1.6	2	2
Telban and Cardullo [72]	4.2	3.6	2.5

3.1.2 Otoliths

Much less is known about the otolith structures, as compared with the semicircular canals [86], and, as a result, the models of the sensory function are less developed. From physiological experiments, it has been determined that the otoliths

sense both tilt and acceleration [22], referred to collectively in the literature as specific force, although the units are of acceleration.

When the head tilts or undergoes linear acceleration the otoliths move. They are attached to a gelatinous layer, and their motion bends hair cells embedded in this layer. This affects the afferent nerve firing rate in the sensory cells. This sensory mechanism cannot distinguish between a tilt and an acceleration. This sensation ambiguity is exploited by the tilt-coordination cueing technique.

3.1.2.1 Mathematical Model

The first otolith model related the sensed velocity to input velocity [41], and did not capture the system response to a steady-state tilt. Subsequently, a model was proposed that relates the sensed specific force to that experienced by the subject [84]. In this framework, linear acceleration and tilt have equivalent sensations. The transfer function of the model is given by:

$$\frac{f(s)}{f(s)} = \frac{K(\tau_L s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad (3.2)$$

Table 3.2 summarises the two common sets of parameter values for the otolith model used in motion cueing. The first set were developed by a pair of researchers [84], while the second set are a combination of a number of studies [32, 47, 84], and are used in this work. The Bode plot of the otolith model is given in Figure 3.2, and the highlighted region corresponds to the frequency range of interest in race-car motion cueing.

TABLE 3.2: Otolith Model Parameters

Parameter	Young and Meiry [84]	Telban and Cardullo [72]
τ_L	13.2	10
τ_1	5.33	5
τ_2	0.66	0.016
K	0.4	0.4

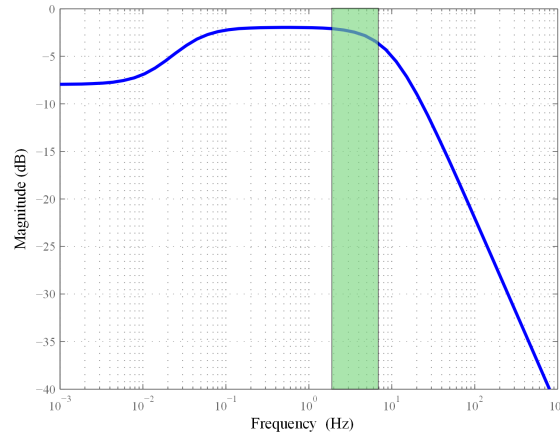


FIGURE 3.2: The Bode plot of the otolith model used in this work, corresponding to the parameters given in right-most column of Table 3.2. The highlighted range corresponds to the frequency of motions experienced in a race-car.

3.1.2.2 Linear Acceleration Threshold Values

Similarly to the semi-circular canal behaviour, the otoliths do not detect motion below a threshold. There are very few studies that determine the linear acceleration thresholds for humans, and the values found in these can differ in orders of magnitude [86]. However, the values given in Table 3.3, are used in a number of motion cueing algorithms [56, 72].

TABLE 3.3: Linear Acceleration Threshold Values

Direction	Longitudinal(x)	Lateral(y)	Vertical(z)
Value(m/s^2)	0.17	0.17	0.17

3.2 Classical Filtering Technique

The oldest cueing strategy, and still one of the most common, is the classical algorithm. This filtering technique was first formalised by Schmidt and Conrad in 1970 [61], in which they combined ideas from single axis systems to form the first washout circuitry for a full six degree-of-freedom (6 DOF) motion platform. This algorithm has subsequently been implemented in numerous flight and car simulators, which include both 3 DOF [9, 36] and 6 DOF platforms [11, 37].

In this system, simulator motion demands are generated by high-pass filtering the translational acceleration and angular velocity signals produced by the car model. The low frequency (longer duration) components of acceleration signals result in large displacements, which cannot be replicated in a simulator. Through an iterative, trial-and-error tuning process the filter gains and cut-off frequencies are adjusted so that the platform position demand remains within the confines of the workspace.

The lack of low-frequency acceleration information in the longitudinal and lateral directions can be partially mitigated by the addition of tilt-coordination. This technique exploits the fact that humans cannot distinguish between a steady-state tilt and a translational acceleration; in order to simulate a longitudinal or lateral acceleration the driver is pitched or rolled. Providing the visuals are moved appropriately, and the rotational motion is sufficiently slow, the driver will perceive a steady-state forward or sideways acceleration and not a tilt. A rate-limit on the platform rotation produces a delayed peak in the sensed acceleration. Therefore, if the acceleration changes direction rapidly, the platform moves too slowly and this technique is not usable. As illustrated in Figure 3.3, the pitch and longitudinal (and lateral and roll) freedoms are considered in combination. The translational acceleration is high-pass filtered in order to generate a position reference for the

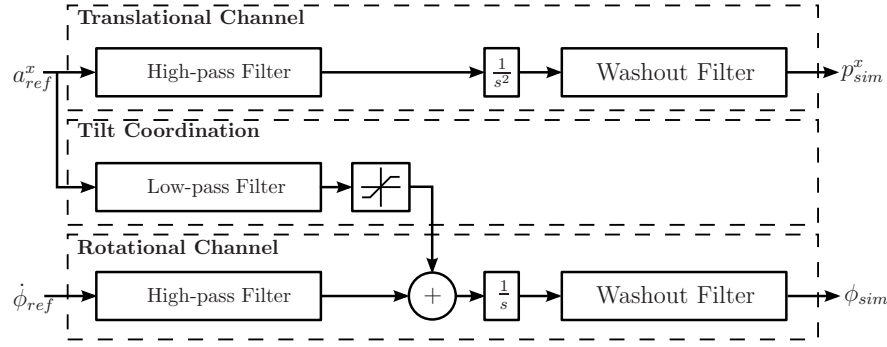


FIGURE 3.3: Structure of the classical cueing system with tilt-coordination. The input signals a_{ref}^x and $\dot{\phi}_{ref}$ are the vehicle longitudinal acceleration and pitch velocity, respectively. The output signals, p_{sim}^x and ϕ_{sim} are the platform longitudinal and pitch displacement demands.

platform. It is also low-pass filtered and added to the rotation channel in order to produce tilt coordination in the rotation demand. The translational acceleration may be scaled and limited so that it is not interpreted as a rotational motion. An additional platform-centering washout filter must be included in each channel. There is no cross-coupling between the yaw and heave freedoms.

A well-documented disadvantage of the classical approach is the generation of false cues that occur after an onset acceleration; this is a feature of any linear time-invariant filter. As explained in [57], this backlash effect is a characteristic of linear high-pass filters, and is an intrinsic limitation of any linear technique. A method is introduced in [57] that reduces the miscues through the addition of a non-linear post-filtering module. An adaptive gain is applied to the output of the filter, which reduces the amplitude of the signal during miscuing.

Another limitation of this approach is that it does not explicitly account for the platform workspace limits [11]. As a result, the filters are either tuned for the anticipated worst-case acceleration, or for a particular driving session. As noted in [74] the filters are usually tuned to accommodate the maximum amplitude acceleration and so smaller onset cues are not sustained for as long as they should be.

A final limitation of this technique, is that tuning the filter parameters is, to an inexperienced user, not clear [25]. A method is proposed in [45] to determine the range of acceptable parameter values to simplify tuning. The parameter limits are found using vestibular detection thresholds (miscues are aimed to be kept below human detection thresholds) and platform excursion constraints.

Despite the limitations of this technique it is easy to implement and, while tuning may require experience, it is simple to adjust in response to driver comments [43], hence its enduring popularity.

3.3 Nonlinear Coordinated Adaptive Algorithm

The classical algorithm is tuned to accommodate worst case accelerations and so poor performance relative to low-amplitude cues results. Adaptive techniques aim to overcome this limitation by varying the cueing filter parameters in real-time. These (adaptive) algorithms can be designed to adjust the translational high-pass, cross-term and rotational filters, or any combination of these. The filter gains and cut-off frequencies are calculated at each time step to minimise a cost function of the form [11]

$$J = (a_{ref} - a_{sim})^2 + w_1 v_{sim}^2 + w_2 p_{sim}^2 \quad (3.3)$$

in which a_{ref} is the acceleration of the vehicle, and a_{sim} , v_{sim} and p_{sim} are, respectively, the acceleration, velocity and displacement of the platform, while w_1 and w_2 are weights. The cost is a weighted sum of the platform position, velocity and the acceleration error; this error is the difference between the input and output of a dynamic system. A steepest descent method is used to find the cost function minimum [25]. By adjusting the weights in the cost function the

algorithm can be tuned so that the resulting acceleration demand lies within the platform workspace.

This method shows an improvement in the fidelity of the motion of the simulator platform when it is close to the neutral position, and accelerations are only corrupted as the platform nears its limits. The cost function can be defined in different ways making the method more flexible and intelligent [43]. The tuning is also more intuitive as the cost function weights cause a clear trade-off between the fidelity of the acceleration and the extent of the platform motion [25]. The experimental results in [25] and [45] show a reduction in the false cues produced by this algorithm as compared with the classical technique.

The disadvantages of this method are that it is computationally demanding, and so has a higher execution time, with the algorithm being difficult to adjust [43]. It has also been found that for certain motion trajectories the adaptive filter distorts the motion perception [2]. These authors suggest a more complex cost function that considers lateral and roll freedoms in combination. A model of the vestibular system is included and the adaptive cost function is adjusted to minimise either the rotational sensory error or the roll angle. This showed an improved performance in the angular sensations, but the translational perception was compromised. The cost function can also be adjusted to penalise acceleration errors when the derivative of the acceleration is high [2], thereby including a commonly employed principle of motion cueing - cue for sudden changes in motion.

3.4 Optimal Cueing

This cueing approach uses LQR theory to design optimal washout filters; a detailed description of the algorithm can be found in [43], [73] and [64] and later in this thesis.

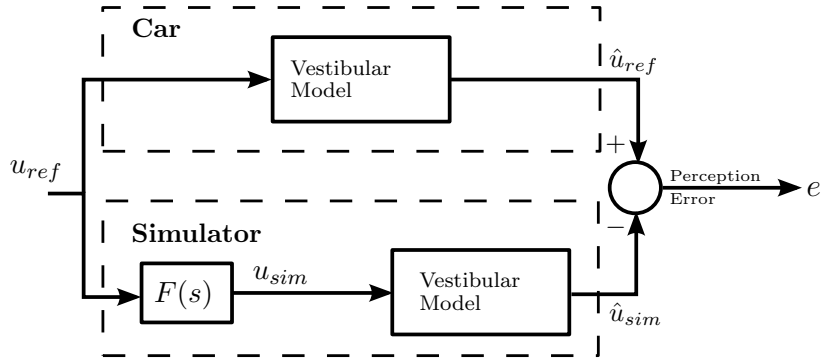


FIGURE 3.4: Formulation of the optimal motion cueing problem.

The aim of this approach is to design a filter $F(s)$ that relates the simulator acceleration demands to the vehicle reference accelerations as shown [45]:

$$u_{sim}(s) = F(s)u_{ref}(s), \quad (3.4)$$

where $u_{sim}(s)$ represents the simulator translational acceleration or rotational velocity, and $u_{ref}(s)$ represents the same quantity in the vehicle. This filter is designed using LQR theory to minimise a cost function of the form:

$$J = \int_0^\infty (e^T Q e + x_{sim}^T R_d x_{sim} + u_{sim}^T R_s u_{sim}) dt. \quad (3.5)$$

The cueing problem formulation is shown in Figure 3.4. The motion of the vehicle is given by u_{ref} , and is filtered by the vestibular system to produce perceived vehicle motion $\hat{u}_{ref}$. In the simulator the vehicle motion is filtered by the washout filter $F(s)$ to produce simulator motion demands u_{sim} which are also filtered by the vestibular model to produce perceived simulator motion $\hat{u}_{sim}$. The perceptual error e is the difference between the motion perceived in the simulator and the real vehicle.

The second term in the cost function contains x_{sim} which is the simulator velocity and position. This and the final term in the cost function require that simulator

motions as well as perceptual error are minimised.

LQR theory is used to design the optimal cueing filter $F(s)$ that balances the trade-off between perceptual fidelity and simulator motion. The weights Q , R_d and R_s are tuned iteratively so that the platform stays within its limits for the worst-case acceleration input.

The filter structure that results from this technique is similar to the classical algorithm. By using a model of the vestibular system (where the specific force in the translational direction is a function of the platform's rotational motion) the longitudinal, pitch and cross-term filters are designed simultaneously, and (3.4) becomes:

$$\begin{pmatrix} a_{sim}^x \\ \dot{\phi}_{sim} \end{pmatrix} = \begin{pmatrix} F_L(s) & F_{PL}(s) \\ F_{LP}(s) & F_P(s) \end{pmatrix} \begin{pmatrix} a_{ref}^x \\ \dot{\phi}_{ref} \end{pmatrix} \quad (3.6)$$

for the longitudinal and pitch directions. The filter $F_{PL}(s)$ is the cross-term from the pitch to longitudinal direction and is usually 0. The filter $F_{LP}(s)$ is the cross-term from longitudinal to pitch motion and is typically low-pass. The remain filters $F_L(s)$ and $F_P(s)$ are the (normally high-pass) longitudinal and pitch filters. The lateral and roll filter design problem is similarly structured, while the heave and yaw are independent. These filters are usually 3rd to 5th order [43].

The tuning of this algorithm is transparent to the user as it only involves adjusting the weights that trade-off platform workspace usage and vestibular error [25]. This approach incorporates directly a model of the vestibular system and so takes advantages of the natural filtering of humans. The disadvantages of this method are that the freedoms are assumed to be decoupled and the vehicle motion a stochastic signal [43], which is not realistic in a practical simulator. It also shares the limitations of classical filters in that the system is tuned for worst case accelerations, and so does not use the full workspace in normal operation, and it suffers from false cueing.

3.5 MPC Cueing

Recent motion cueing algorithms have explored the use of MPC [5, 13, 25]. These methods also minimise the perception error, but the platform limits and the driver motion perception thresholds are included as hard constraints.

In MPC a constrained finite-horizon optimisation problem is solved at every time step. Since this is a receding horizon technique, only the control action at the first-step is applied to the system. The numerical solution is found by reformulating the optimisation problem as a quadratic programming problem, which is solved using standard algorithms [55].

The cost function of the motion cueing problem is:

$$J = \sum_{i=0}^{H_P} y^T(t+i)Qy(t+i) + u^T(t+i)Ru(t+i), \quad (3.7)$$

where R and Q are weighting matrices, $u(t+i)$ is the simulator acceleration (control action) and $y(t+i)$ is defined as:

$$y(t+i) = x(t+i) - x_{ref}(t) \quad i = 0, \dots, H_P. \quad (3.8)$$

In (3.8) $x_{ref}(t)$ is a reference, and $x(t+i)$ are the system states that evolve according to the linear plant dynamics:

$$x(t+i+1) = Ax(t+i) + Bu(t+i) \quad i = 0, \dots, H_P, \quad (3.9)$$

The problem constraints are:

$$u_{min} \leq u(t+i) \leq u_{max} \quad i = 0, \dots, H_U - 1 \quad (3.10)$$

$$x_{min} \leq x(t+i) \leq x_{max} \quad i = 0, \dots, H_P \quad (3.11)$$

$$u(t+i) = 0 \quad i = H_U, \dots, H_P \quad (3.12)$$

In equations (3.7) to (3.12) H_P is the prediction horizon (the number of time-steps over which the cost function is minimised) and H_U is the control horizon (the number of time steps for which the control action is solved). Since $H_U \leq H_P$ the control is assumed zero from H_U to H_P , equation (3.12). The state vector $x(t+1)$ contains vestibular states, and states relating to the platform dynamics. Once again, by using the vestibular model where the translational specific force depends on the rotational motion, tilt-coordination will result. When the longitudinal constraints are encountered, pitch motion can be substituted, and similarly with lateral and roll.

The MPC problem is converted into a quadratic programming (QP) problem given by:

$$\min_x x^T H x + f^T x \quad (3.13)$$

subject to

$$Ax \leq B \quad (3.14)$$

$$A_{eq}x = B_{eq} \quad (3.15)$$

where, x is the optimisation variable, H and f are weighting matrices and (3.14) and (3.15) are the linear inequality and equality constraints, respectively. The transformation of the MPC into a QP problem follows the method developed by [79] and applied to motion cueing by [25]. The QP problem is solved using standard methods, such as *quadprog* in Matlab.

A model of the platform is required, so that the workspace limits can be observed. In the simplest form, constraints on the acceleration, velocity and position of the

platform centre can be imposed, without using a platform kinematic model. If the DOF of the platform are largely uncoupled this approach will still make good use of the available workspace [5]. Alternatively, a linearised platform model can be employed, such as in [27] which uses a Stewart platform model, simplified with small-angle approximations. A 9 DOF platform model, consisting of a hexapod mounted on a flat-bed tripod, is used in [7]. Since the platform has 3 redundant DOF, the MPC algorithm determines the optimal division of the motion between the different mechanisms, thereby making best use of the unusually shaped workspace.

In MPC theory, the aim is to determine the plant input that will minimise the difference between the system states and the reference trajectory, as shown in equations (3.7) and (3.8). A reference signal (available at least N steps in the future) is therefore required. In this problem, the reference signal is the vestibular accelerations of a driver in the actual vehicle, since sensation error should be minimised. There are two main approaches to generating the reference vestibular output over the prediction horizon. The first, and simple, approach is to assume the reference is constant, that is, the vehicle and hence vestibular accelerations are the same at the future time steps as they are at the initial one [27]. Other approaches generate a reference trajectory by using a virtual driver simulation [4] or using a simple car model where the driver inputs remain constant over the prediction horizon [3].

The error given in (3.8) and the platform acceleration $u(t+i)$ both appear in the cost function (3.7), and so both are penalised. As one would expect, the strength and smoothness of the onset and miscue in the acceleration signal are affected by the length of control and prediction horizons.

The result of the MPC algorithm is that, typically, the simulator follows the reference acceleration signal until it nears the platform limits, at which point it slows

the platform to a stop. With the hardware used in [13], and the incorporation of vestibular threshold values it is possible to decelerate the platform at a level imperceptible to the driver. This yields smooth decelerations near the limits. Washout to the neutral position is not included, but is implemented as a parallel process that is activated by the accelerations and driver actions [13].

Testing with this method has demonstrated better workspace usage than the classical technique [13]. It is also the simplest of the cueing algorithms to tune [25]. However, in the period where the platform slows towards the limits, none of the high-frequency content of the motion is present [25]. Finally, the computational burden of the MPC problem is high, and to implement in real-time the control and prediction horizons must necessarily be kept short [5]. Furthermore, the problem can become too large if the inputs to all DOF are determined simultaneously. This can be overcome by only considering the tilt-coordination pairs together [27].

3.6 Spherical Algorithms

The classical and optimal motion cueing algorithms are not applicable in the Desdemona simulator (Section 2.1.3) as they are not designed for a cylindrical workspace and will not take advantage of the kinematic characteristics of the system, such as the ability of the platform to match yaw one-to-one. As a result a so-called spherical algorithm has been designed to exploit the platform's capabilities and provide both onset and sustained cues [81].

The spherical algorithm is described in detail in [81] and functions by transforming the x and y specific forces into radial and tangential accelerations. These are high-pass filtered and integrated to generate radius, cabin and central yaw axis values. The algorithm washes out the motion to a neutral radius position and a tangentially facing cabin, but it does not wash out the central yaw angle.

For a longitudinal acceleration a curved trajectory is generated so the centripetal acceleration can be used to generate sustained motion cues. This circular motion however, causes a lateral false cue, which as mentioned in Section 2.1.3 is the primary disadvantage of the Desdemona simulator. However, by tilting the platform a component of the gravity vector can be used to compensate in part for this acceleration [20]. False cues are also generated in yaw and lateral directions during the washing out of the platform to its neutral radius position and cabin yaw position.

In the experiments conducted by [82] the spherical algorithm was proved to be more successful in the Desdemona than the classical algorithm, and the sustained lateral and one-to-one yaw cues were liked by the drivers operating the simulator.

3.7 Prepositioning

Prepositioning is not a motion cueing algorithm in itself, but it is a useful technique that can be employed in conjunction with the classical and optimal approaches in order to improve workspace usage. Developed by Weiß [80], the idea behind this technique is to anticipate the next movement of the vehicle and move the platform to a point that will increase the workspace in the required direction. If the driver is going to brake (accelerate), the platform should move to the foremost (rearmost) longitudinal limit of the workspace. In this way the workspace for braking, accelerating and cornering is increased and optimal or classical cueing algorithms can be tuned to give stronger or longer duration cues. The crux of this technique is moving the platform sufficiently slowly that the pre-manoeuving is not detected by the driver and does not interfere with the other motion cues or introduce new ones. To address this, nonlinear rate limiters are used in [10] to keep prepositioning movements below the driver perception thresholds.

Weiß presents two techniques to implement prepositioning [80]. The first is through filtering a neutral-position-offset demand to ensure a smooth transition of the platform to the prepositioning location. The filtering technique allows a slow onset and final motion, however, a limitation lies in the maximum velocity that the platform needs to achieve in order to move within the prediction time. The second technique is a logic method. The desired position and prediction time are used to calculate the velocity at which the platform must move to get there. At first this was aimed to be at sub-detectable velocities, but this does not ensure that the platform reaches the position in time and was found to be inadequate. The filtering technique is smoother, but the logic technique requires a lower peak velocity.

The limitation of the prepositioning technique is that it requires the ability to predict the future vehicle motion. In the longitudinal direction it is possible to estimate the upcoming accelerations by observing the current vehicle speed or gear ratio (braking is likely to follow high speed driving, while accelerating follows a period of low speed) [10]. The lateral direction can be prepositioned from the track data. The curve radius, vehicle speed and an estimated time-to-next-corner can be used to move the platform into a good position [10].

3.8 Summary

The cueing algorithms presented in this chapter were designed originally for either aircraft or passenger car simulators, and hence, have limited applicability in race or high-performance car simulators. The tilt coordination technique that is used to generate sustained accelerations cannot be applied in racing. The longitudinal and lateral accelerations change direction very quickly and the platform cannot be tilted fast enough. However, the sustained accelerations are of the order of 3Gs, and tilt coordination can only simulate a fraction of 1G. Hence, it would

not provide significant benefit anyway. Finally, the test drivers are more sensitive to vehicle behaviour, and are more attuned to whether the vehicle is tilting or accelerating and hence are harder to ‘trick’.

Without the use of tilt coordination, and noting that the heavy, pitch and roll freedoms only have small motions that do not require filtering, the cueing problem reduces to designing filters for only longitudinal, lateral and yaw motions. As such, any of the above algorithms which design filters for multiple directions at once, would need to be modified.

While it seems that there are more challenges in designing a racing simulator cueing algorithm, there is, however, one advantage. This is that the driver behaviour is very predictable. The race track and speed profile is known, and hence, the vehicle acceleration is consistent between laps. The ability to anticipate the driver’s actions makes the prepositioning technique applicable in this context. This assists with the exploitation of the full workspace to provide the highest fidelity cues possible.

The aim in both the optimal and model predictive control algorithms is to minimise the acceleration sensation error of the simulator driver. This approach is successful in passenger car simulators that only respond to low-magnitude accelerations and slow dynamics. Using the cueing techniques described (particularly with successful tilt coordination) it is possible to reproduce many features of the sensed accelerations. However, in race-car and high-performance vehicle applications, the vehicle accelerations are an order of magnitude larger than those in typical passenger cars, and so only a small fraction of the motion is rendered. Since the perceptual error is large relative to the simulator motion, the incorporation of a vestibular model does not offer much assistance in the cueing algorithm design. Indeed, there is little value in attempting to replicate vehicular accelerations, since it is both unachievable and unnecessary. Experienced test drivers do

not need to ‘believe’ the simulator is a car. They merely require sufficient information to be able to control and drive the car realistically. The cueing focus should therefore not be on minimising sensory error, but rather correctly communicating the vehicle behavioural information. Determining the motion cues required by the driver to control the vehicle and understand its handling properties is one of the main themes of this thesis.

Chapter 4

Vehicle Stability Analysis

4.1 Introduction

The aim of motion cueing in a race car simulator differs from that of the passenger car application. In the latter, an immersive, realistic driving experience is created by providing cues that minimise the vestibular sensory error. In the former, however, the simulators are used to evaluate the car set-up and its effect on the vehicle handling. To perform this task the drivers need motion cues from which they can infer the vehicle behaviour at the performance limit. This behaviour is influenced by the vehicle's handling and stability properties. By studying the changes in the car's stability and response characteristics as it approaches the limit, the necessary motion cues for the vehicle simulator can be analysed. This insight can be used to evaluate existing and to develop new cueing methodologies. Since the stability of the vehicle changes with speed and acceleration (both longitudinal and lateral), it is convenient to use a G-G diagram (Figure 4.1) to characterise the operating domain of the car. This well-known diagram plots the lateral vehicle acceleration against the longitudinal acceleration with a boundary line representing the physical limits of the car. The engine power is the primary

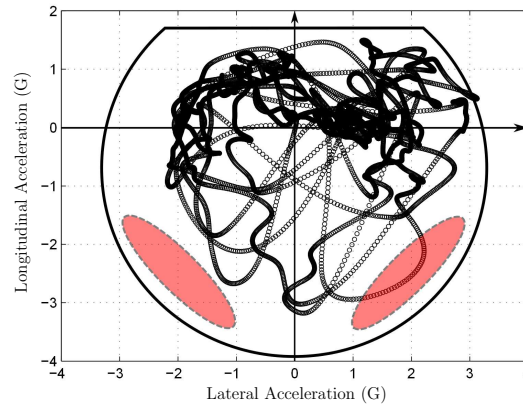


FIGURE 4.1: Sample G-G diagram of a Formula One car that shows the vehicle's lateral and longitudinal acceleration on a qualifying lap at Silverstone. The measured vehicle data has been filtered to remove noise. The track contains 18 corners, of which 10 are right-handed, that accounts for the vehicle's positive-dominated lateral acceleration. In addition, the high-speed corners, out of which the driver accelerates, are also right-handed, which is consistent with vehicle's prolonged operation in the top-right quadrant of the diagram. The highlighted regions correspond with simultaneous braking and cornering, a condition in which the vehicle is typically unstable.

influence on the positive longitudinal acceleration limit, while the tyre grip is the key factor in determining the lateral and braking limits. A graphical overlay of recorded telemetry data can be used to analyse the driver's exploitation of the performance envelope. In racing, the more the driver is able to utilise the capability of the vehicle, the lower the lap time.

Handling and stability characteristics influence the drivers' ability to operate the vehicle on the limit of performance. An unstable vehicle may undermine the drivers' confidence, or in extreme cases make the vehicle undriveable. The highlighted regions in Figure 4.1 correspond to areas where the vehicle is typically unstable and more difficult to control, thus requiring highly skilled drivers. Characterising the vehicle stability across the G-G diagram is not only useful for understanding motion cueing requirements, but can also assist in predicting the extent to which the driver will be able to exploit the vehicle's performance.

The treatment of stability over the full operating range of the vehicle has produced a number of analysis techniques, but all have their limitations. Milliken [42] describes the use of stability and control derivatives to assist in the assessment of vehicle handling such as oversteer and understeer [54]. These derivatives are generated from linearised vehicle models at particular steady-state operating points. Used originally in the aircraft industry, this approach is practically useful, and characterises the vehicle response to physical stimuli, such as an applied yaw moment. Despite providing valuable insight into vehicle behaviour, stability derivatives are only applicable to the portion of the G-G envelope corresponding to steady-state cornering conditions. More sophisticated methods use phase plane and bifurcation analysis to determine vehicle stability [83]. These techniques are only usable with relatively simple vehicle models.

This work uses Lyapunov’s indirect method to study the stability of a vehicle under steady-state cornering conditions [62]. The analysis is extended to accelerating and braking scenarios following the approach developed for motorcycle stability analysis [12, 39] and more recently applied to the case of a racing vehicle G-G envelope in [76]. In essence, the car’s stability properties are inferred from the eigenvalue of linearised vehicle models. While high-fidelity models can be used, due care must be taken with this type of analysis since it is only valid when the nonlinear system is ‘varying slowly’ [16, 58, 65].

The time-varying eigenvalue analysis is first developed for accelerating and braking under straight running conditions. A simple two degree-of-freedom (2 DOF) car model facilitates the derivation of simple analytic expressions for the vehicle’s stability and response. The stability analysis is then extended to the remainder of the operating chart by considering first steady-state and then accelerating and braking under cornering. A 3 DOF model is used that includes nonlinear, combined slip ‘magic formula’ type tyres. This is then extended to a high-fidelity,

4-wheel car model with suspension and aerodynamic forces.

The changes in the vehicle stability and response under different operating conditions are analysed to determine the motion cues that are required by race-car simulator drivers.

The remainder of this chapter is arranged as follows: the method to determine the vehicle stability is explained in Section 4.2, the analysis of stability under straight-running conditions is presented in Section 4.3. The extension to cornering scenarios is given in Section 4.4, and the analysis of a high fidelity model is given in Section 4.5.

4.2 Stability Theory

Lyapunov's Indirect method is applied to the study of the non-linear vehicle [67]. When the vehicle is travelling at a constant velocity, the system is in equilibrium and can be linearised about the steady-state operating conditions. However, for accelerating and braking vehicle scenarios the system is changing, and the stability along a nominal motion trajectory is studied.

The vehicle dynamics are represented by the following non-linear equation:

$$\dot{x} = f(x, u). \quad (4.1)$$

If the braking or accelerating vehicle motion is represented by a nominal state trajectory, $x_n(t)$, which has an associated input trajectory $u_n(t)$, the stability of the system to a perturbation away from $x_n(t)$ can be studied. The perturbation of the state and input are given by:

$$x(t) = x_n(t) + \Delta x(t), \quad (4.2)$$

and

$$u(t) = u_n(t) + \Delta u(t), \quad (4.3)$$

respectively. Since $(x(t), u(t))$ is also a solution to (4.1):

$$\dot{x}_n(t) + \Delta \dot{x}(t) = f(x_n(t) + \Delta x(t), u_n(t) + \Delta u(t)). \quad (4.4)$$

This can be rewritten using a Taylor expansion:

$$\dot{x}_n(t) + \Delta \dot{x}(t) = f(x_n(t), u_n(t)) + \left. \frac{df}{dx} \right|_{x_n(t), u_n(t)} \Delta x(t) + \left. \frac{df}{du} \right|_{x_n(t), u_n(t)} \Delta u(t) + h.o.t \quad (4.5)$$

and, since Δx is small, this can be simplified by neglecting the higher order terms in Δx and Δu :

$$\Delta \dot{x}(t) = A(t)\Delta x(t) + B(t)\Delta u(t), \quad (4.6)$$

where,

$$A(t) = \left. \frac{df}{dx} \right|_{x_n(t), u_n(t)}, \quad (4.7)$$

and

$$B(t) = \left. \frac{df}{du} \right|_{x_n(t), u_n(t)}. \quad (4.8)$$

The stability of the accelerating or braking vehicle is inferred from the stability of the associated time-varying linear system. The use of frozen-time eigenvalues to infer the stability of this system is discussed in Section 4.2.1 below.

4.2.1 Stability of Linear, Time-varying systems

It is shown in [58] and [16] that the time-varying linear system represented by:

$$\dot{x} = A(t)x, \quad (4.9)$$

is stable when the associated frozen-time eigenvalues are negative, and the rate of change of $A(t)$ is sufficiently slow. A detailed derivation of the upper bound on the rate of change of $A(t)$ is given in [58], with the results quoted here.

The time-varying matrix $A(t)$ is restricted to the form that corresponds to the differential equation:

$$x^n + a_n(t)x^{n-1} + \dots + a_2(t)\dot{x} + a_1(t)x = 0, \quad (4.10)$$

where, x^n is the n^{th} derivative of x . The eigenvalues of $A(t)$ are uniquely related to the coefficients of the differential equation, and therefore bounds can be placed on the rates of change of the eigenvalues.

It is shown in [58] that if for all $t \geq t_0$, every $a_i(t)$ satisfies $|a_i| \leq a$ and is differentiable, and the eigenvalues λ of A are distinct and $Re(\lambda_i) \leq -\epsilon < 0$ for all i , then the point $x = 0$ is asymptotically stable if the matrix $L(t)$ is negative definite for every $t \geq t_0$ and some $\eta > 0$, where

$$L = AS + SA^T - \dot{S} + \eta I, \quad (4.11)$$

and the matrix S is given by:

$$S_{ij} = \sum_{k=1}^n \lambda_k^{i-1} \bar{\lambda}_k^{j-1}, \quad (4.12)$$

where, λ_k are the eigenvalues of A and $\bar{\lambda}_k$ the complex conjugates of the eigenvalues.

This result can be applied to the vehicle model under accelerating and braking conditions to determine the vehicle speeds and accelerations under which the system is stable.

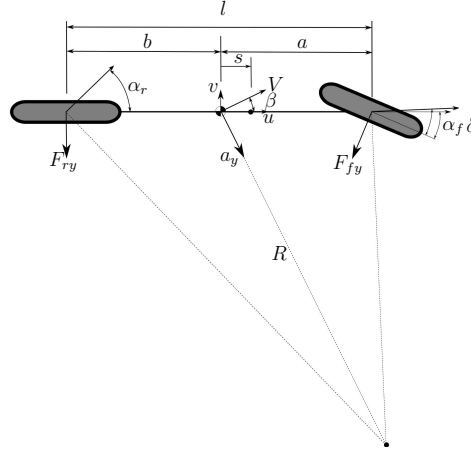


FIGURE 4.2: Kinematics of a single-track vehicle model showing its basic geometric parameters.

4.3 Straight Line Motion

The stability of the straight-running car is studied first, with analytic results derived using a 2 DOF vehicle model.

4.3.1 Vehicle Model

A single-track bicycle model is used to describe the vehicle's dynamics and stability. The kinematics of the car model are shown in Figure 4.2, with all quantities described in an SAE coordinate system. Balancing forces in the longitudinal and lateral directions, and balancing yaw moments around the car's mass centre gives the following well known equations of motion (EOM):

$$m\dot{u} = m\omega v - F_{fy} \sin \delta + F_{rx} \quad (4.13)$$

$$m\dot{v} = -m\omega u + F_{fy} \cos \delta + F_{ry} \quad (4.14)$$

$$J\dot{\omega} = a \cos \delta F_{fy} - b F_{ry}. \quad (4.15)$$

The state vector is $x = [u, v, \omega]^T$, where the states are, respectively, the body-fixed longitudinal, lateral and yaw velocities. The parameters a , b , m and J

represent respectively the distance of the front and rear axle from the centre-of-mass, the mass of the vehicle and the vehicle's yaw moment of inertia. The front wheel has steering angle δ , which is assumed 'small'. The front and rear lateral tyre forces are given by F_{fy} and F_{ry} respectively. The driven rear wheel has longitudinal tyre force F_{rx} . The side-slip angle β lies between the vehicle mass centre velocity V and the vehicle's longitudinal axis. The front and rear tyre velocities make slip angles α_f and α_r with the wheel plane directions. These three angles are computed using:

$$\beta = \arctan\left(\frac{v}{u}\right), \quad (4.16)$$

$$\alpha_f = \arctan\left(\frac{u \sin \delta - \cos \delta (\omega a + v)}{u \cos \delta + \sin \delta (\omega a + v)}\right), \quad (4.17)$$

$$\alpha_r = \arctan\left(\frac{\omega b - v}{u}\right). \quad (4.18)$$

When the vehicle is accelerated, or braked, there is a normal tyre force transfer between the front and rear axles. By balancing moments around the rear-wheel ground contact point, and noting that the sum of the tyre down-forces is equal to the weight of the vehicle, the front and rear down-force, F_{fz} and F_{rz} respectively, can be computed as:

$$F_{rz} = mg \frac{a}{a+b} + ma_x \frac{h}{a+b} \quad (4.19)$$

$$F_{fz} = mg - F_{rz}, \quad (4.20)$$

where h is the height of the centre-of-mass and a_x is the longitudinal acceleration.

The lateral forces generated by the tyres are assumed to depend linearly on the tyre normal loads and slip angles, and are computed by:

$$F_{fy} = C_{f0}F_{fz} \left(\delta - \frac{\omega a + v}{u} \right), \quad (4.21)$$

$$F_{ry} = C_{r0}F_{rz} \left(\frac{\omega b - v}{u} \right), \quad (4.22)$$

where, C_{f0} and C_{r0} are the normalised cornering stiffnesses (at the front and rear axle); (4.17) and (4.18) are simplified using small angle approximations.

Under straight-running conditions the influence of the mwv term on the longitudinal acceleration is assumed negligible, which decouples the longitudinal speed from the other two EOM. We will assume that

$$u(t) = u_0 + a_x t. \quad (4.23)$$

The tyre model is combined with the remaining EOM, and the resulting linear time-varying EOM are given by:

$$\dot{v} = -\frac{C}{mu(t)}v + \left(\frac{-Cs}{mu(t)} - u(t) \right) \omega + \frac{C_f}{m}\delta, \quad (4.24)$$

$$\dot{\omega} = \frac{-Cs}{mk^2u(t)}v - \frac{Cq^2}{mk^2u(t)}\omega + \frac{C_fa}{mk^2}\delta, \quad (4.25)$$

in which

$$C_r = C_{r0}F_{rz} \quad (4.26)$$

$$C_f = C_{f0}F_{fz} \quad (4.27)$$

$$C_s = C_f a - C_r b \quad (4.28)$$

$$C = C_f + C_r \quad (4.29)$$

$$Cq^2 = C_f a^2 + C_r b^2 \quad (4.30)$$

$$mk^2 = J. \quad (4.31)$$

Following [48], the parameter ‘ s ’ is the distance between the centre-of-mass and the neutral steer point, q is the average moment arm and k the radius of gyration. The neutral steer point, which is shown in Figure 4.2, has kinematic significance; a force applied through this point will cause the vehicle to move laterally without yawing.

4.3.2 Constant Velocity Stability

Under steady-state conditions (4.24) and (4.25) are time-invariant. When examining the behaviour of the car, the response in the lateral acceleration, yaw velocity and side-slip angle to perturbations in steering angle are determined. These quantities are given by: $y = [a_y, \omega, \beta]^T$, where, the lateral acceleration is given by:

$$a_y = \dot{v} + u\omega, \quad (4.32)$$

ω is a state, and β is given in (4.16), but linearised assuming small angles.

The state-space model of the car from the steering angle to the output vector is given by:

$$\begin{aligned} A &= \begin{pmatrix} -\frac{C}{mu_0} & \frac{-Cs - mu_0^2}{mu_0} \\ \frac{-Cs}{mk^2u_0} & -\frac{Cq^2}{mk^2u_0} \end{pmatrix} & B &= \begin{pmatrix} \frac{C_f}{m} \\ \frac{aC_f}{mk^2} \end{pmatrix} \\ C &= \begin{pmatrix} -\frac{C}{mu_0} & -\frac{Cs}{mu_0} \\ 0 & 1 \\ \frac{1}{u_0} & 0 \end{pmatrix} & D &= \begin{pmatrix} \frac{C_f}{m} \\ 0 \\ 0 \end{pmatrix} \end{aligned} \quad (4.33)$$

where u_0 is the steady-state velocity of the vehicle.

The stability of this constant-velocity model has been studied extensively, e.g. [48]. The poles of the system, which are used to determine the stability are the roots of the characteristic polynomial:

$$\lambda^2 + \frac{(a^2 + k^2)C_f + (b^2 + k^2)C_r}{mk^2u_0}\lambda + \left(\frac{l^2C_fC_r}{m^2k^2u_0^2} - \frac{Cs}{mk^2} \right) = 0. \quad (4.34)$$

Following the Routh-Hurwitz criterion, the system will be stable if all the coefficients in (4.34) are positive. With a linear tyre model all cornering stiffnesses are positive, hence only the last term in (4.34) can become negative, which occurs when the speed exceeds some critical value given by:

$$u_0 > V_{crit} = \sqrt{\frac{l^2C_fC_r}{mCs}}. \quad (4.35)$$

The existence of a positive, real critical speed is dictated by the sign of the Cs term. Recalling equation (4.28), this depends on the location of the centre-of-mass and the cornering stiffnesses of the tyres. This is linked to the concepts of over and understeer, which have a number of equivalent definitions [42, 48]. The definition given here is not restricted to a linearised tyre model, and is chosen because it links the ideas to the driver input. If a vehicle is negotiating a turn

of constant radius, and the speed is increased, the steering angle required to maintain the turn will change. In an under-steering vehicle it will increase, in an oversteering vehicle it will decrease and in a neutrally steering vehicle it will remain constant. It can be shown that if C_s is positive, the vehicle will oversteer and hence have a critical speed above which it is unstable [48]. If, on the other hand, it is negative, the vehicle will under-steer and will remain stable at all speeds.

The stability and steering characteristics of a race car will now be studied, with the vehicle parameters given in Table A.1 in the Appendix. The tyre parameters are chosen in order to give an oversteering vehicle with a critical speed of $V_{crit} = 65$ m/s.

From the state-space model in (4.33) the transfer functions from the steering angle to each of the outputs are determined. While the system poles are determined by the characteristic equation, the zeros of each of the three transfer functions are given by

$$\{z_{a_y1}, z_{a_y2}\} = -\frac{bC_rl}{2mk^2u_0} \pm \sqrt{\left(\frac{bC_rl}{2mk^2u_0}\right)^2 - \frac{C_rl}{mk^2}} \quad (4.36)$$

$$z_\beta = \frac{amu_0^2 - bC_rl}{mk^2u_0} \quad (4.37)$$

$$z_\omega = -\frac{C_rl}{mu_0a}. \quad (4.38)$$

The zeros of the lateral acceleration transfer function will either be negative real, or will form a complex conjugate pair with a negative real part. The zero of the yaw velocity response transfer function will always be negative, however, at higher speeds, the zero of the side-slip angle transfer function may become positive, resulting in a non-minimum phase response.

The system poles and the zeros of each transfer function are given in the left-half of Figure 4.3 for vehicle speeds from 20 m/s to 80 m/s. As the speed increases,

the poles of the system migrate towards the right-half plane, with one becoming positive when the critical speed of 65 m/s is reached.

The lateral acceleration, side-slip angle and yaw velocity responses to a step change in the steering angle are determined at speeds of $u_0 = \{30 \text{ m/s}, 50 \text{ m/s}, 70 \text{ m/s}\}$. The results are shown in the right half of Figure 4.3. At $u_0 = 30 \text{ m/s}$ the system responses exhibit overdamped, stable behaviour, approaching finite final values. At $u_0 = 50 \text{ m/s}$ the responses are quicker, with the outputs achieving significantly larger values in the same time. At $u_0 = 70 \text{ m/s}$ the system is unstable and the responses grow indefinitely. In the simulator the driver needs to identify the car's steering (over or under) characteristics, as well as its proximity to the stability limit. As shown in Figure 4.3, the speed of the vehicle response to a steering input increases as the vehicle becomes progressively less stable. To assist the driver, the simulator motion cues should recreate these larger magnitude motions, reduced response times and reduced damping ratios as the car approaches the stability limit.

4.3.3 Accelerating and Braking Stability

As a first step, the stability of the vehicle under braking and accelerating is studied by examining the eigenvalues of the frozen-time system (conventional approach). The vehicle is accelerated (or braked) and the frozen-time eigenvalues along the manoeuvre are computed. To verify the results, the poles are correlated with the time response of the yaw velocity to a perturbation. The frozen-time approach is then validated by computing explicit bounds on the rate of change of the system. These are used to determine the conditions (vehicle acceleration and velocity) where the eigenvalues represent the system stability.

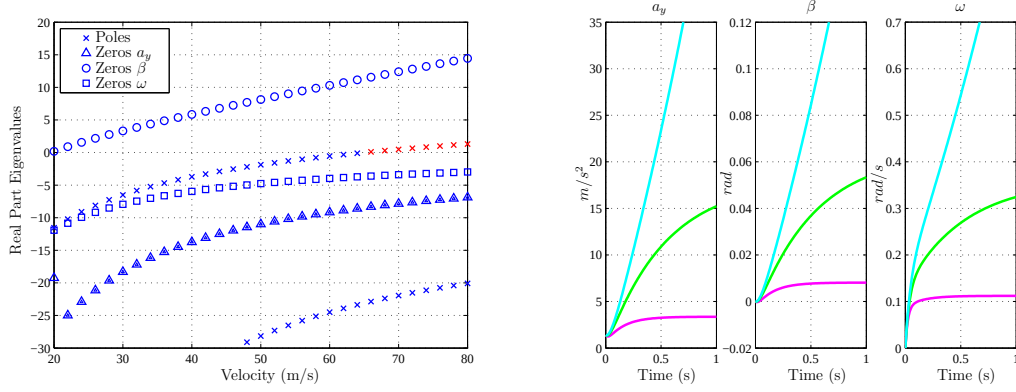


FIGURE 4.3: The left-hand figure shows the real part of the poles and zeros of the 2 DOF model given in (4.33) for straight-running vehicle speeds between 20 m/s and 80 m/s. The poles are the roots of the characteristic polynomial given in (4.34). The zeros of the lateral acceleration, side-slip angle and yaw velocity transfer functions are computed using (4.36), (4.37) and (4.38), respectively. In the case of the lateral acceleration, the zeros form a complex conjugate pair. The right-hand figure shows the responses of the lateral acceleration, side-slip angle and yaw velocity transfer functions to step changes in the steering input. These responses are computed at longitudinal speeds of $u_0 = \{30 \text{ m/s}, 50 \text{ m/s}, 70 \text{ m/s}\}$ corresponding respectively to the magenta, green and cyan curves.

4.3.3.1 Closed-Form Solution

The first-order differential equations (4.24) and (4.25) can be combined by eliminating the lateral velocity to produce a second-order differential equation in the yaw rate ω .

$$u^2(t)\ddot{\omega} + \frac{k^2ma_x + (a^2 + k^2)C_f + (b^2 + k^2)C_r}{mk^2}u(t)\dot{\omega} + \left(-\frac{Cs}{mk^2}u^2(t) + \frac{l^2C_fC_r}{m^2k^2}\right)\omega = \frac{u(t)C_f(C_rl + aa_xm)}{m^2k^2}\delta + \frac{u^2(t)aC_f}{mk^2}\dot{\delta}, \quad (4.39)$$

This equation can be simplified by defining the parameters P , Q and R as:

$$P = \frac{k^2ma_x + (a^2 + k^2)C_f + (b^2 + k^2)C_r}{mk^2}, \quad Q = -\frac{Cs}{mk^2}, \quad R = \frac{l^2C_fC_r}{m^2k^2}, \quad (4.40)$$

and, with the additional assumption that the steering angle is held at zero, (4.39) can be rewritten as:

$$u^2(t)\ddot{\omega} + Pu(t)\dot{\omega} + (Qu^2(t) + R)\omega = 0. \quad (4.41)$$

The time-varying differential equation (4.41) has a structure reminiscent of a Bessel's differential equation and so a closed-form solution can be found in terms of Bessel functions. It is assumed that the parameters P , Q and R do not vary with speed. However, these parameters do change with longitudinal acceleration. For a given constant acceleration a_x the tyre cornering stiffnesses C_f and C_r can be calculated, based on the required longitudinal driving force, and the load transfer between the front and rear axels. Aerodynamic forces are neglected in this model, however, were they to be included then P , Q and R would vary with speed, and hence the Bessel's function approach could not be applied.

As demonstrated in [35] an equation with the form:

$$u^2 \frac{d^2 \omega}{du^2} + Au \frac{d\omega}{du} + (Bu^2 + C)\omega = 0 \quad (4.42)$$

has a solution

$$\omega = c_1 u^\rho(t) J_\nu(\alpha u(t)) + c_2 u^\rho(t) Y_\nu(\alpha u(t)), \quad (4.43)$$

in which J_ν and Y_ν are Bessel functions of the first and second kind, and c_1 and c_2 are constants of integration. The parameters ρ and α , and the order of the Bessel functions ν , are related to the coefficients of the differential equation by:

$$\rho = \frac{1}{2}(1 - A), \quad \alpha = \sqrt{B}, \quad \nu = \sqrt{\frac{1}{4}(1 - A)^2 - C}. \quad (4.44)$$

Comparing equations (4.41) and (4.42) show that a change of variable from t to

u is needed. The first- and second-order derivatives of ω with respect to t are related to the derivatives with respect to u by:

$$\frac{d\omega}{dt} = \frac{d\omega}{du} \frac{du}{dt} = a_x \frac{d\omega}{du} \quad (4.45)$$

and

$$\frac{d^2\omega}{dt^2} = \frac{d^2\omega}{du^2} \left(\frac{du}{dt} \right)^2 + \frac{d\omega}{du} \frac{d^2u}{dt^2} = a_x^2 \frac{d^2\omega}{du^2}, \quad (4.46)$$

in which a_x is assumed constant.

The equation in (4.41) can now be rewritten as:

$$u^2(t) \frac{d^2\omega}{du^2} + \frac{P}{a_x} u(t) \frac{d\omega}{du} + \left(\frac{Q}{a_x^2} u^2(t) + \frac{R}{a_x^2} \right) \omega = 0, \quad (4.47)$$

The parameters ρ and α , and the order of the Bessel functions ν , can now be related to the coefficients of the differential equation and a_x by:

$$\rho = \frac{1}{2} \left(1 - \frac{P}{a_x} \right), \quad \alpha = \sqrt{\frac{Q}{a_x^2}}, \quad \nu = \sqrt{\frac{1}{4} \left(1 - \frac{P}{a_x} \right)^2 - \frac{R}{a_x^2}}. \quad (4.48)$$

If Q is negative the argument of the Bessel functions is complex. In this case, in order to obtain real solutions, modified Bessel functions I_ν and K_ν of the first and second kind must be used to obtain

$$\omega = c_1 u^\rho(t) I_\nu(\alpha u(t)) + c_2 u^\rho(t) K_\nu(\alpha u(t)). \quad (4.49)$$

The sign of Q depends on the sign of C_s . However, unlike the constant velocity case, this quantity now changes with both the car kinematic parameters and the vehicle's acceleration. For example, a car might be oversteering at a constant velocity ($C_s < 0$), however when the vehicle accelerates, normal load is transferred to the rear and C_r increases, while C_f simultaneously decreases thereby producing

possibly a negative value of Cs . Under braking the opposite occurs and Cs will remain negative (or may become negative if the vehicle is initially under-steering).

As the sign of Cs changes the appropriate Bessel functions must be used.

The constants of integration, c_1 and c_2 are found by applying the initial conditions $\omega(0) = \omega_0$ and $\dot{\omega}(0) = \dot{\omega}_0$. The first condition is applied to (4.43) to get

$$\omega_0 = u_0^\rho (c_1 J_\nu(\alpha u_0) + c_2 Y_\nu(\alpha u_0)), \quad (4.50)$$

where, $u_0 = u(0)$. The derivative of the general solution (4.43) is required to apply the second initial condition:

$$\begin{aligned} a_x \dot{\omega}_0 = c_1 & \left[\rho u_0^{\rho-1} J_\nu(\alpha u_0) + u_0^\rho \left(J_{\nu-1}(\alpha u_0) - \frac{\nu}{\alpha u_0} J_\nu(\alpha u_0) \right) \alpha \right] \\ & + c_2 \left[\rho u_0^{\rho-1} Y_\nu(\alpha u_0) + u_0^\rho \left(Y_{\nu-1}(\alpha u_0) - \frac{\nu}{\alpha u_0} Y_\nu(\alpha u_0) \right) \alpha \right]. \end{aligned} \quad (4.51)$$

If the velocity is assumed constant the yaw differential equation reduces to a linear second-order time-invariant equation with solution

$$\omega = c_1 e^{\lambda_1 t} + c_2 e^{\lambda_2 t}, \quad (4.52)$$

where λ_1 and λ_2 are the model eigenvalues. The constants c_1 and c_2 can be found, once again, through the application of the initial conditions. The time-invariant yaw response will differ from the time-varying solution over long time periods, and a piece-wise time-invariant yaw response can therefore be used. Suppose that time is divided into intervals, and that the time-invariant solution is computed on each interval with the terminal conditions from the previous interval used as the initial conditions for the current interval. As these sub-intervals decrease in duration the solution accuracy, as expected, improves.

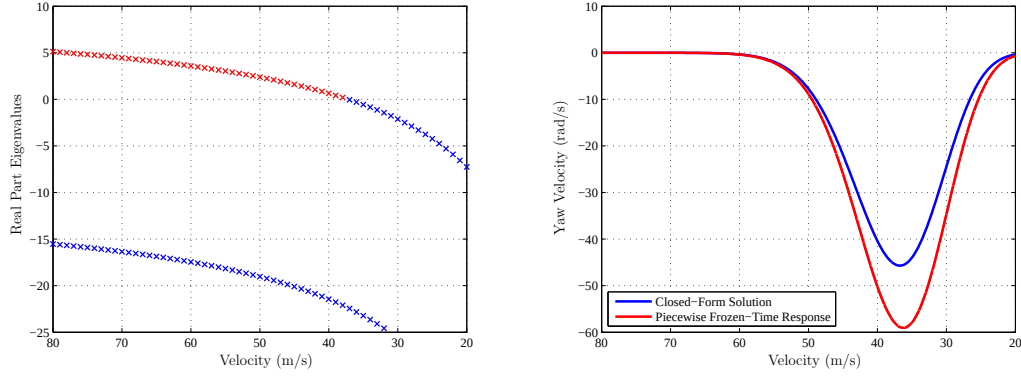


FIGURE 4.4: The left-hand figure shows the real parts of the frozen-time eigenvalues of a vehicle braking in straight running at -10 m/s^2 for vehicle speeds from 80 m/s to 20 m/s . The right-hand figure shows the yaw response to a non-zero initial yaw velocity at 80 m/s .

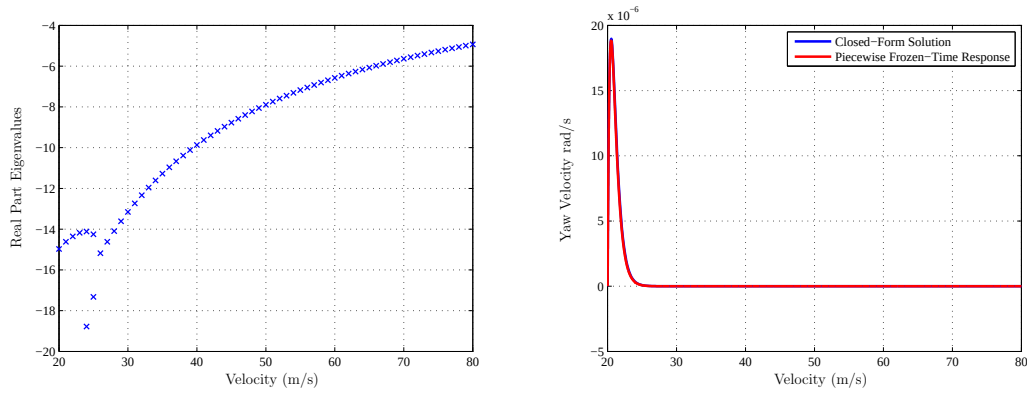


FIGURE 4.5: The left-hand figure shows the real part of the frozen-time eigenvalues of a vehicle accelerating in straight running at 10 m/s^2 from 20 m/s to 80 m/s . The right-hand figure shows the yaw response to a non-zero initial yaw velocity at 20 m/s .

In the case where Cs is negative, the eigenvalues λ_1 and λ_2 are complex hence ω in (4.52) is also complex. A real solution is found using the general solution of the form:

$$\omega = c_1 e^{\lambda_r t} \cos(\lambda_i t) + c_2 e^{\lambda_r t} \sin(\lambda_i t), \quad (4.53)$$

where, λ_r and λ_i are the real and imaginary parts of the eigenvalues.

4.3.3.2 Results

The eigenvalues and system response for a constant -10 m/s^2 of braking are shown in Figure 4.4, for speeds varying from 80 m/s to 20 m/s. One of the eigenvalues changes sign at $V_{crit} = 38 \text{ m/s}$, which is much lower than the constant velocity case (65 m/s). The load transfer due to braking reduces the rear tyre cornering stiffness. This, in turn, causes C_s to increase with an accompanying reduction in the critical speed; see (4.35). The response of the time-varying system to a non-zero initial yaw velocity at 80 m/s is determined using the closed-form solution in (4.43), as well as a piecewise time-invariant solution using (4.52). Initially, the yaw angle grows exponentially, typical of an unstable system. However, as the vehicle speed decreases, and the eigenvalues become negative, the yaw angle response slows and then decays. This behaviour suggests that a vehicle braking from 80 m/s may exhibit an undesirable ‘burst’ response.

Under acceleration the eigenvalues of the system remain negative, and indeed, the system response to a non-zero initial condition exhibits a stable characteristic. The piecewise frozen-time solution predicts almost exactly the real solution. At higher speeds the eigenvalues are complex, which manifests in an underdamped yaw response to a perturbation.

In the simulator, cues that communicate these behavioural changes must be present so that the driver can keep track of the effects of acceleration and braking.

4.3.3.3 Validation of the Eigenvalues

The results of [58], which have been described in Section 4.2.1, are applied to the vehicle model for straight-line accelerating and braking conditions. From this, it can be determined if the system is slowly-varying and, therefore, whether the eigenvalues can be used to infer stability. The computation of the matrix S , given

by (4.12), is simplified by considering the case of real and complex eigenvalues separately. Under braking, the eigenvalues for the particular vehicle studied are real. As a result of the load transfer to the rear under acceleration, the eigenvalues are complex. It is therefore convenient to study these two cases independently.

4.3.3.4 Real Eigenvalues

When the eigenvalues of $A(t)$ are real and the system second order, the matrix S is given simply by:

$$S(t) = \begin{bmatrix} 2 & -a_2 \\ -a_2 & a_2^2 - 2a_1 \end{bmatrix} \quad (4.54)$$

where, a_1 and a_2 are the coefficients of the differential equation (4.10).

The time-varying differential yaw equation in (4.41) can be rewritten in the form of (4.10), and, so S is found to be:

$$S(t) = \begin{bmatrix} 2 & -\frac{P}{u(t)} \\ -\frac{P}{u(t)} & \frac{P^2}{u^2(t)} - 2\left(Q + \frac{R}{u^2(t)}\right) \end{bmatrix}, \quad (4.55)$$

while $A(t)$ and $\dot{S}$ are given by:

$$A(t) = \begin{bmatrix} 0 & 1 \\ -\left(Q + \frac{R}{u^2(t)}\right) & -\frac{P}{u(t)} \end{bmatrix} \quad (4.56)$$

and,

$$\dot{S}(t) = \begin{bmatrix} 0 & \frac{Pa_x}{u^2(t)} \\ \frac{Pa_x}{u^2(t)} & \frac{4Ra_x - 2P^2a_x}{u^3(t)} \end{bmatrix}, \quad (4.57)$$

respectively.

The matrix L can be determined analytically:

$$L(t) = \begin{bmatrix} -2\frac{P}{u(t)} & -\frac{-2P^2+Pa_x+4Qu^2(t)+4R}{u^2(t)} \\ -\frac{-2P^2+Pa_x+4Qu^2(t)+4R}{u^2(t)} & 2\frac{-P^3+P^2a_x+3PQu^2(t)+3PR-2Ra_x}{u^3(t)} \end{bmatrix} + \begin{bmatrix} \eta & 0 \\ 0 & \eta \end{bmatrix} \quad (4.58)$$

The requirement for stability is that L must be negative definite, which, since it is symmetric, implies its eigenvalues must be negative. From the Routh criterion, this requires that the coefficients of the characteristic polynomial be positive. These are determined analytically from L and with some rearranging are given by:

$$B_{R1}(t) = \frac{u^2(t)(P - 3PQ) + P^3 - P^2a_x + 2Ra_x - 3PR}{u^3(t)} > \eta > 0, \quad (4.59)$$

and,

$$B_{R2}(t) = \frac{-8Q^2u^4(t) + u^2(t)(2P^2Q - 16QR - 4PQa_x) + 2P^2R - 8R^2 - \frac{1}{2}P^2a_x^2}{u^4(t)} > -\eta^2 + \eta \frac{u^2(t)(P - 3PQ) + P^3 - P^2a_x + 2Ra_x - 3RP}{u^3(t)} > 0, \quad (4.60)$$

where, the second condition has been simplified by assuming that the first condition holds. The time-varying vehicle system is stable when these two bounds are satisfied and the frozen-time eigenvalues are negative.

The conditions $B_{R1}(t)$ and $B_{R2}(t)$ in (4.59) and (4.60), respectively, are evaluated numerically for a vehicle braking at -10m/s^2 at speeds ranging from 80 to 20 m/s, with the results shown in the left plot of Figure 4.6. The first condition $B_{R1}(t) > 0$ always holds, but the second, $B_{R2}(t) > 0$, only holds below the limit speed $V_{\text{lim}} = 37\text{ m/s}$.

In the frozen-time system the eigenvalues are negative below the vehicle critical speed V_{crit} . This speed is compared with V_{lim} for accelerations of -20m/s^2 to

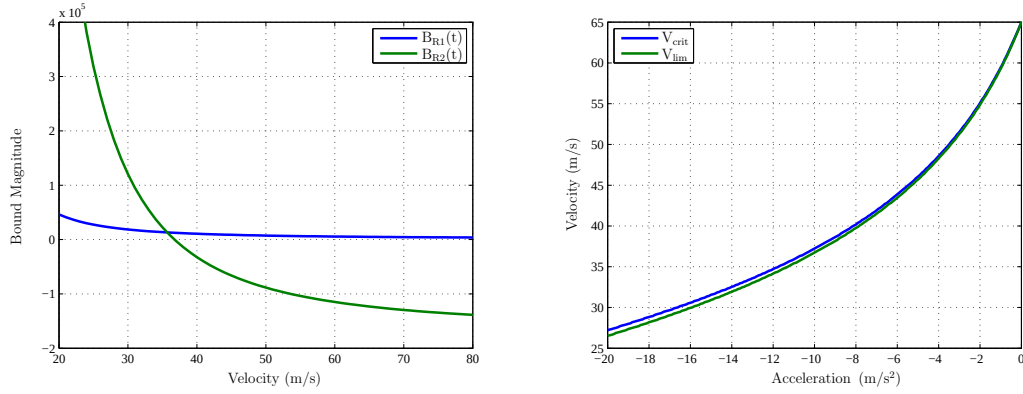


FIGURE 4.6: The left-hand plot contains the numerical evaluation of the bounds in (4.59) and (4.60) for vehicle speeds from 80 to 20 m/s, for a vehicle braking at -10 m/s^2 . The right-hand plot shows the vehicle critical speed, and the limit speed as functions of the vehicle acceleration.

0 m/s^2 in right plot in Figure 4.6. At slower accelerations the critical and limit speeds are almost equal (being exactly equal at zero acceleration), but as the braking becomes stronger the limit speed becomes increasingly lower than the critical speed. This implies that the asymptotic stability of the time-varying system is assured for $u_0 < V_{lim}$, while the negative eigenvalues between V_{lim} and V_{crit} cannot be used to infer stability.

4.3.3.5 Complex Eigenvalues

When the system eigenvalues are complex, S is given by:

$$S = \begin{bmatrix} 2 & -a_2 \\ -a_2 & 2a_1 \end{bmatrix} = \begin{bmatrix} 2 & -\frac{P}{u(t)} \\ -\frac{P}{u(t)} & 2\left(Q + \frac{R}{u^2(t)}\right) \end{bmatrix}, \quad (4.61)$$

and, thus L is found to be:

$$L(t) = \begin{bmatrix} -2\frac{P}{u(t)} & \frac{P^2 - Pa_x}{u^2(t)} \\ \frac{P^2 - Pa_x}{u^2(t)} & -2\frac{PQu^2(t) + PR - Ra_x}{u^3(t)} \end{bmatrix} + \begin{bmatrix} \eta & 0 \\ 0 & \eta \end{bmatrix}. \quad (4.62)$$

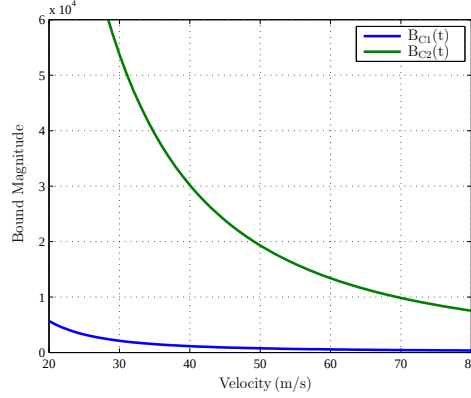


FIGURE 4.7: Numerical evaluation of conditions (4.63) and (4.64) for vehicle speeds between 20 and 80 m/s, with a vehicle accelerating at 10 m/s². Under accelerating conditions there is no critical speed.

The sufficient conditions for stability are now given by:

$$B_{C1}(t) = \frac{u^2(t)(P + PQ) - Ra_x + PR}{u^3(t)} > \eta > 0, \quad (4.63)$$

and,

$$B_{C2}(t) = \frac{2P^2Qu^2(t) - 2PRa_x - \frac{1}{2}P^4 + 2P^2R + P^3a_x - \frac{1}{2}P^2a_x^2}{u^4(t)} > -\eta^2 + \eta \frac{u^2(t)(P + PQ) - Ra_x + PR}{u^3(t)} > 0, \quad (4.64)$$

where, once again, the second condition is simplified assuming the first is true.

The bounds $B_{C1}(t)$ and $B_{C2}(t)$ are evaluated for a vehicle accelerating at 10 m/s² and given in Figure 4.7. Across all vehicle speeds the bounds are satisfied, which implies that the system is asymptotically stable since the eigenvalues of the frozen-time system, as shown in Figure 4.5, are negative under acceleration.

4.4 Cornering Stability

The straight-running analysis is now extended to the case of a cornering vehicle for both the constant and variable speed cases.

4.4.1 Constant Velocity

Under cornering conditions the longitudinal and lateral dynamics given in (4.13) and (4.14) interact. Furthermore, in steady-state cornering, the tyre side-slip angles are no longer zero and the linear tyre model given in (4.22) is no longer usable. Instead, combined-slip magic formula type tyres must be used. Empirical formulae, described in Appendix A determine the magnitude of the lateral and longitudinal tyre forces as functions of the tyre side-slip angle, longitudinal slip and normal load, and can be represented as:

$$F_x = g(\alpha, \kappa, F_z), \quad F_y = f(\alpha, \kappa, F_z). \quad (4.65)$$

If the EOM in (4.13), (4.14), and (4.15) are linearised about the steady-state (constant velocity and radius-of-turn) operating point, the state-space model is given in (4.66), where the state vector is $[u, v, \omega]^T$.

$$A = \begin{pmatrix} \frac{-C_f \delta_0 (v_0 + a \omega_0)}{m u_0^2} & \frac{C_f \delta_0 + m u_0 \omega_0}{m u_0} & \frac{C_f a \delta_0 + m u_0 v_0}{m u_0} \\ \frac{C v_0 + (C_s - m u_0^2) \omega_0}{m u_0^2} & -\frac{C}{m u_0} & \frac{-C_s - m u_0^2}{m u_0} \\ \frac{C s v_0 + C q^2 \omega_0}{m k^2 u_0^2} & \frac{-C_s}{m k^2 u_0} & -\frac{C q^2}{m k^2 u_0} \end{pmatrix} \quad B = \begin{pmatrix} -\frac{C_f \delta_0 + F_{fy0}}{m} \\ \frac{C_f - F_{fy0} \delta_0}{m} \\ \frac{C_f a - F_{fy0} a \delta_0}{m k^2} \end{pmatrix}$$

$$C = \begin{pmatrix} \frac{C s \omega_0 + v_0 C}{m u_0^2} & -\frac{C}{m u_0} & -\frac{C_s}{m u_0} \\ 0 & 0 & 1 \\ -\frac{v_0}{u_0^2} & \frac{1}{u_0} & 0 \end{pmatrix} \quad D = \begin{pmatrix} \frac{C_f - F_{fy0} \delta_0}{m} \\ 0 \\ 0 \end{pmatrix}. \quad (4.66)$$

The outputs are again $[a_y, \beta, \omega]$, where the simplifications in (4.28) to (4.31) are used and the linearisation of the front tyre forces is given by:

$$\begin{bmatrix} \frac{\partial F_{fy}}{\partial u} \\ \frac{\partial F_{fy}}{\partial v} \\ \frac{\partial F_{fy}}{\partial \omega} \\ \frac{\partial F_{fy}}{\partial \delta} \end{bmatrix} = \frac{\partial F_{fy}}{\partial \alpha_f} \bigg|_{\alpha_{f0}, \kappa_{f0}, F_{fz0}} \begin{bmatrix} \frac{\partial \alpha_f}{\partial u} \\ \frac{\partial \alpha_f}{\partial v} \\ \frac{\partial \alpha_f}{\partial \omega} \\ \frac{\partial \alpha_f}{\partial \delta} \end{bmatrix} \bigg|_{u_0, v_0, \omega_0, \delta_0} = C_f \begin{bmatrix} \frac{\omega_0 a + v}{u^2} \\ -\frac{1}{u_0} \\ -\frac{a}{u_0} \\ 1 \end{bmatrix}. \quad (4.67)$$

The rear-tyre force is treated similarly.

The operating conditions required to evaluate the linearised model are determined for a steady-state velocity (V) and radius-of-turn (R). Once V and R have been chosen, ω is determined by the kinematic relation:

$$\omega = \frac{V}{R}. \quad (4.68)$$

The resultant vehicle velocity is related to the lateral and longitudinal velocities by:

$$V^2 = v^2 + u^2. \quad (4.69)$$

Under steady-state conditions, the EOM reduce to the following algebraic relations:

$$0 = m\omega_0 v_0 - F_{fy0}\delta_0 + F_{rx0} \quad (4.70)$$

$$0 = -m\omega_0 u_0 + F_{fy0}\cos\delta_0 + F_{ry0} \quad (4.71)$$

$$0 = aF_{fy0}\cos\delta_0 - bF_{ry0}. \quad (4.72)$$

These five equations and the non-linear tyre model are solved simultaneously to determine the steady-state values of u_0 , v_0 , ω_0 , δ_0 , F_{fy0} , F_{ry0} , as well as F_{rx0} , which is a small driving force required to maintain the vehicle at a constant

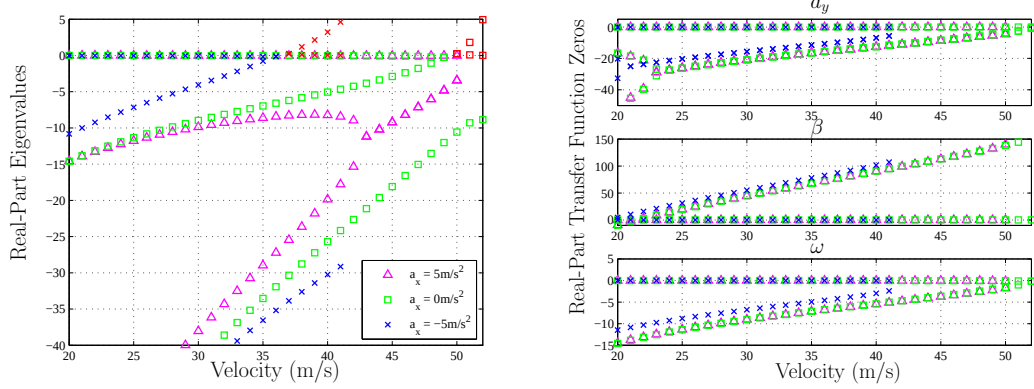


FIGURE 4.8: The left-hand figure shows the real-part of the eigenvalues of the 3 DOF vehicle model plotted for different vehicle speeds at vehicle accelerations of $a_x = 5 \text{ m/s}^2$, 0 m/s^2 , -5 m/s^2 . The right-hand figure shows the zeros of each of the transfer functions across different speeds and at the different acceleration levels.

velocity. The solution is used to evaluate the stability of the equilibrium point and to determine the response of the outputs to perturbations in the steering input.

The parameters of the vehicle studied are given in Table A.1 and the normalised tyre characteristics are given in Appendix A. The vehicle stability is determined for speeds ranging from $V = 20 \text{ m/s}$ to $V = 53 \text{ m/s}$ in a constant 170 m radius turn. At speeds above $V = 53 \text{ m/s}$ the tyres saturate and the vehicle can no longer maintain the corner. The real-parts of the system eigenvalues are plotted at different vehicle speeds in the left half of Figure 4.8. As the speed increases the eigenvalues migrate towards, and ultimately into the right-half plane. From the 2 DOF analysis, this indicates that the vehicle is over steering, which is consistent with the centre-of-mass being located towards the rear of the vehicle. In contrast to the 2 DOF model, in this case both the location and type of eigenvalue change. Initially, there are three negative real eigenvalues, which migrate towards the imaginary axis as the speed increases. Two of these eigenvalues then form a complex conjugate pair that moves across the imaginary axis. This is shown in more detail in the root-locus plot of Figure 4.9. The right half of Figure 4.8

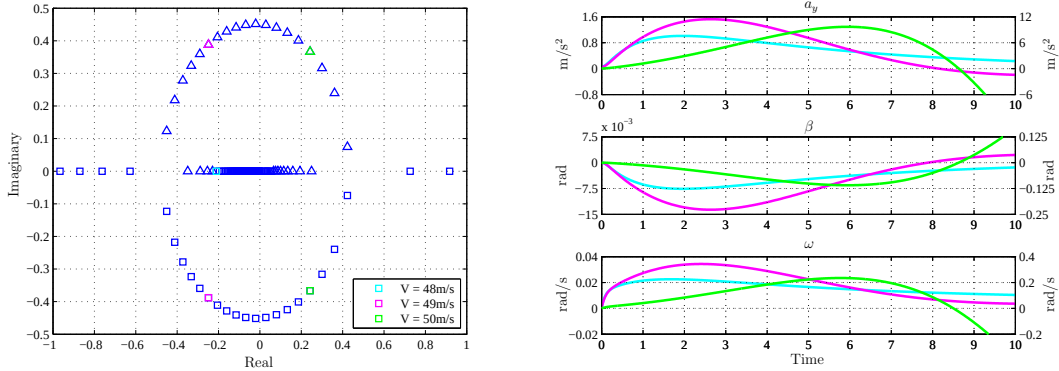


FIGURE 4.9: The left-hand figure shows the root-locus plot for two of the three eigenvalues of the 3 DOF vehicle model under cornering; the varied parameter is the vehicle's speed. The eigenvalues corresponding to the longitudinal speeds $V = \{48 \text{ m/s}, 49 \text{ m/s}, 50 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right-hand figure shows the step responses of the 3 DOF model at the listed longitudinal speeds. The top plot shows the lateral acceleration responses to a step in steering angle, the centre plot shows the body side-slip angle response, while the bottom plot illustrates the yaw rate response. The cyan and magenta curves are associated with the scale on the left axes, while the green curves with those on the right.

shows the zeros of each of the three transfer functions. Each has a zero at the origin, which implies zero steady-state gain. The side-slip angle response contains a non-minimum phase zero, similar to the 2 DOF model, this is correlated with the system responses in Figure 4.9.

In Figure 4.9 the system response is computed at $V = \{48 \text{ m/s}, 49 \text{ m/s}, 50 \text{ m/s}\}$; the eigenvalue type is different at each speed. At $V = 48 \text{ m/s}$ all three responses exhibit overdamped behaviour, consistent with real, negative eigenvalues. When the eigenvalues become complex at $V = 49 \text{ m/s}$ and $V = 50 \text{ m/s}$ the corresponding responses become underdamped and oscillatory. When the system is still stable at $V = 49 \text{ m/s}$ the oscillations decay, however, once the system is unstable at $V = 50 \text{ m/s}$ the responses grow. Once again, the side-slip response is differentiated from the yaw velocity by the non-minimum phase behaviour resulting from the right-half plane zero.

4.4.2 Accelerating and Braking Under Cornering

Straight-line accelerating and braking, and steady-state cornering only represent the axes of the G-G diagram. To determine the stability over the rest of the operating regime it is necessary to develop a method of examining acceleration and braking under cornering. In this scenario, the vehicle is no longer in equilibrium, since the states are changing with time. It is possible to study the stability of an accelerating or braking vehicle by examining perturbations from some nominal motion [67]. To conduct this type of analysis the vehicle states along a nominal prescribed trajectory are required. These evolving states can be found by simulating the manoeuvre using the non-linear vehicle model with feedback controllers that are used to generate the plant inputs that produce the required motion. This approach has a number of limitations: simulating the manoeuvre is a time-consuming process and determining vehicle stability is intended to be a way predicting the vehicle's response *without having to numerically integrate the equations of motion*. Furthermore, as the vehicle becomes unstable, and the braking and acceleration becomes more aggressive, the feedback controllers are less able to maintain the vehicle on the reference trajectory.

An alternative approach to determining the frozen-time vehicle states is to use a D'Alembert force system that acts as a surrogate for the real forces of inertia experienced under acceleration. In this approach, an accelerating or braking vehicle is approximated by one travelling at a constant velocity, but subjected to forces that capture the inertial effects of accelerating and braking. The advantage of this method is clear. The vehicle remains in equilibrium and the vehicle states can be determined from algebraic equations of motion, which now contain additional terms representing the apparent forces. The frozen-time eigenvalues of the system are computed at each point along a motion trajectory. The choice of

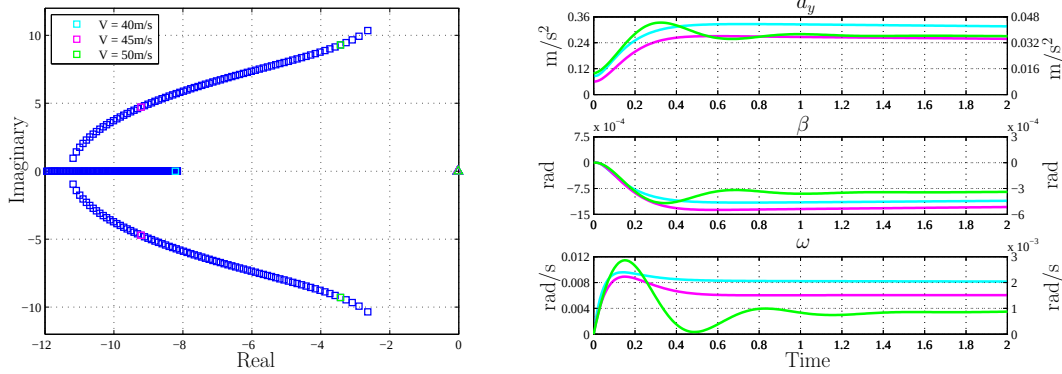


FIGURE 4.10: The left-hand figure shows the root-locus plot of the eigenvalues of the 3 DOF vehicle model under 5 m/s^2 acceleration. At low speeds there are three negative real eigenvalues. As the speed increases, the pole near the origin remains there, whilst the other two move together, coalesce, and form a complex-conjugate pair. While the complex conjugate pair move towards the imaginary axis as the vehicle speed increases, the car loses traction before they cross into the right-half plane. The eigenvalues corresponding to speeds of $V = \{40 \text{ m/s}, 45 \text{ m/s}, 50 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right-hand figure shows the step response plots for the 3 DOF model at the cited longitudinal speeds. From top to bottom, the plots show the lateral acceleration, the side-slip angle and the yaw velocity responses to a step change in the steering angle. The green curves are associated with the right-hand axes, while the cyan and magenta curves are associated with the left-hand axes.

manoeuvres used to ‘fill out’ the G-G diagram is not unique.

For example, the vehicle could be braked while the radius of turn decreases, to maintain a constant lateral acceleration. This would correspond to a line on the G-G diagram that is parallel to the longitudinal acceleration axis. A more common physical scenario, and the one studied here, is the vehicle braking or accelerating in a constant (170 m) radius-of-turn bend.

Once again, the limitation of this eigenvalue-based approach is that stability can only be inferred from the frozen-time system eigenvalues if the system is slowly time-varying.

In Figure 4.8 the eigenvalues and zeros of the vehicle model are computed for acceleration and braking of $\pm 5 \text{ m/s}^2$. In both cases the achievable top speed of

the vehicle is reduced. When the vehicle is driven or braked, the torque applied at the rear wheels increases the tyre longitudinal slip angle, which reduces the ability of the tyre to generate lateral force. Under accelerating there is a normal load transfer to the rear tyres. This changes the vehicle from oversteering to understeering. Consequently, the vehicle remains stable to the point where the tyres lose traction. Under braking the load transfer to the front axle reduces the critical speed and the vehicle becomes unstable at 37 m/s.

The responses of the vehicle outputs to a step steering angle input is once again examined at different vehicle speeds. In Figure 4.10 the eigenvalue root-locus diagram for the accelerating vehicle is given and the step-responses at the highlighted eigenvalues computed; these correspond to speeds of $V = \{40 \text{ m/s}, 45 \text{ m/s}, 50 \text{ m/s}\}$. The vehicle responses under acceleration differ significantly from the constant velocity case given in Figure 4.9. Firstly, the responses reach steady-state more quickly in the accelerating case. Secondly, the responses are more oscillatory in the accelerating case, with the damping factor decreasing as the speed increases. Thirdly, the steady-state responses are much smaller in the accelerating case, with the step-response magnitudes becoming smaller at higher speeds. Thus under acceleration at high speeds, the responsiveness of the vehicle to steering inputs reduces. In [42] it is noted that as an understeering vehicle approaches the limit of adhesion control moments which change the orientation of the vehicle can no longer be generated with front steer.

In the simulator, the change from an over to understeering vehicle due to acceleration, should be cued. Furthermore, the reduction in gains from steering input to the vehicle's outputs (lateral acceleration, yaw velocity and side-slip), must be also communicated to the simulator driver so that he/she is forewarned of their approach to the handling limit of the car.

The vehicle's braking responses and stability characteristics shown in Figure 4.11

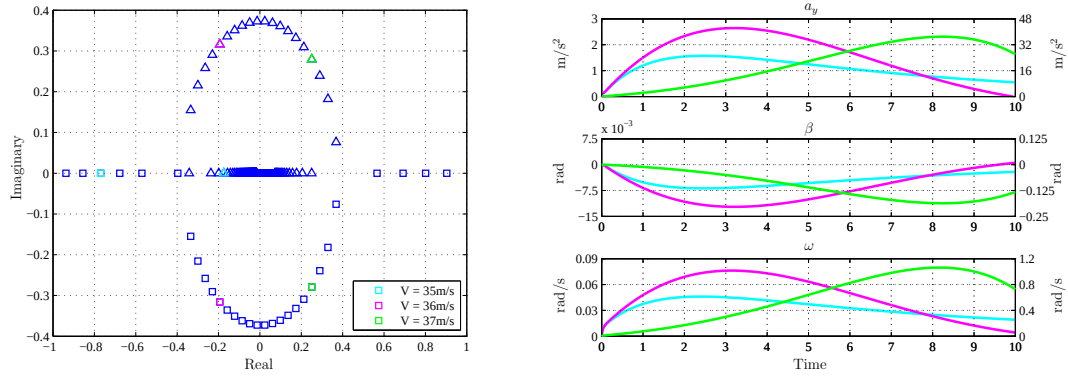


FIGURE 4.11: The left figure contains the root-locus plot of two of the eigenvalues of the 3 DOF vehicle model under -5 m/s^2 braking. As the vehicle speed increases, one pole moves away from the origin as the other moves towards it. These then coalesce and form a complex conjugate pair that migrates into the right-half plane. At higher speeds these become two positive real poles, with one moving towards the origin and the other away from it. The eigenvalues corresponding to the longitudinal speeds $V = \{35 \text{ m/s}, 36 \text{ m/s}, 37 \text{ m/s}\}$ are highlighted in cyan, magenta and green, respectively. The right figure contains the step response plots for a 3 DOF model, computed at the listed longitudinal speeds. From top to bottom, the sub-plots contain the responses of the lateral acceleration, body side-slip angle and yaw velocity transfer functions to a step in steering angle. The cyan and magenta curves are associated with the scale on the left axes, while the green curves with those on the right.

are in several ways similar to the steady-state behaviours given in Figure 4.9. The root loci and the vehicle responses bear a strong resemblance to the constant velocity case, although the limit responses occur at lower vehicle speeds. While the braking and constant-speed responses have similar general shapes, the vehicle is more responsive in lateral acceleration and yaw than it is in the constant velocity case. This increase in gain must be detectable in the simulator. The higher damping factors and the reduced overshoot in the responses should also be cued.

This study illustrates clearly the changes in vehicle behaviour under accelerating and braking (as opposed to constant speed), and highlights the need for stability and vehicle responsiveness to be analysed across the full G-G diagram when contemplating the design of simulator cueing filters.

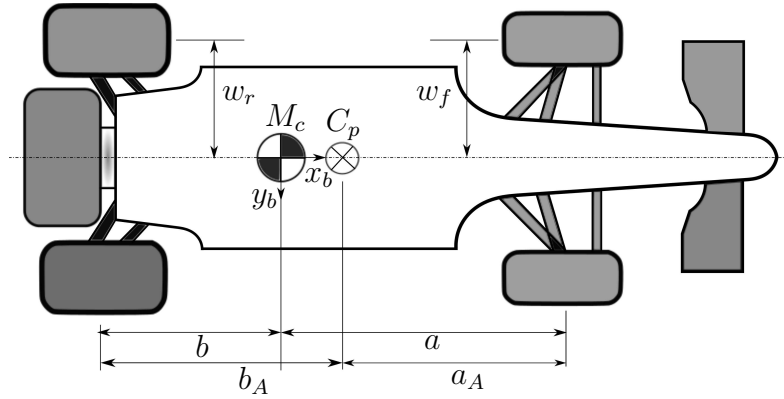


FIGURE 4.12: Plan view of a Formula One car with some of its basic geometric parameters.

4.5 Model Extensions

The 3 DOF model neglects the effects of aerodynamic forces, brake distribution and lateral load transfer. The first is important in race cars, where the increase in down-force enables higher speeds and larger braking and accelerating. In high speed corners the lateral load transfer reduces the grip on the inside tyres, and since the vehicle is limited by the lowest grip tyre, this effect influences the top-speed in a corner. To address these issues, the single-track vehicle is replaced with a higher-fidelity vehicle model, the kinematics of which are given in Fig. 4.12. While similar to the 3 DOF model, new parameters for the front and rear wheel bases (w_f and w_r) are introduced. The four wheels are connected via a massless, double wishbone suspension to the main chassis body. This captures the lateral and longitudinal load transfer effects. The centre-of-pressure (C_P), which is a distance b_A from the rear of the vehicle describes the distribution of the aerodynamic down-force between the front and rear axles. Both the aerodynamic down and drag forces vary quadratically with speed. The full, non-linear combined slip tyre model is also used. The model details are given in Appendix B.

The system is built using the multibody modelling package AUTOSIM [40], which

computes the symbolic equations of motion for the vehicle and generates FORTRAN code that solves these. The resulting program is used for non-linear simulations. By including proportional-differential (PD) feedback controllers from the vehicle speed to tyre longitudinal slip angle and yaw velocity to steering angle the vehicle can be driven at a target velocity in a specific radius-of-turn. The vehicle trim states are extracted from the output of the non-linear simulation and used to populate an equivalent linearised model. Using PD-controllers to drive the car into a steady-state condition, is analogous to determining the operating conditions by solving the algebraic equations (4.68) to (4.72) in the simple model. The state-space model of the complex system is generated by AUTOSIM and can be used to determine the stability, and the transfer functions from steering input to the required outputs.

The stability of the vehicle model is computed across the vehicle operating speeds for braking, constant velocity and accelerating conditions and given in Figure 4.13. The vehicle behaviour differs from the 3 DOF model, noticeably, under acceleration, where the vehicle now becomes unstable above speeds of 60 m/s. In this high-fidelity model, under acceleration the rear tyres are driven, and the increased longitudinal slip reduces the lateral cornering stiffness at the rear. This effect dominates the benefit from the increase in normal load due to mass transfer. Furthermore, the critical speed under acceleration is lower than that under braking. This is attributed to the aerodynamic drag force. Under accelerating the longitudinal slip of the tyres needs be large to overcome this force. However, under braking the drag assists the vehicle to decelerate and smaller longitudinal slip is required. Indeed, when aerodynamic drag is neglected, then the critical speed under braking is 10 m/s lower than accelerating. Furthermore, it is worth considering a practical situation. A Formula 1 car is power limited and as the

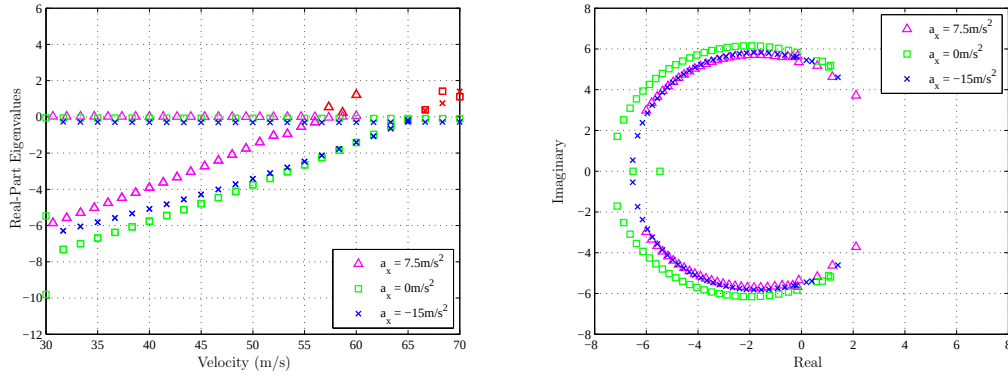


FIGURE 4.13: The left figure contains the eigenvalues for the Formula 1 car under steady-state conditions, 7.5 m/s^2 of acceleration and -15 m/s^2 of braking. The right figure contains the root-locus of the eigenvalues for increasing vehicle speed.

G-G diagram in Fig. 4.1 shows, an acceleration above approximately 1.75G cannot be achieved. At high lateral accelerations 5 m/s^2 acceleration is operating near to the boundary. However, under braking 15 m/s^2 is considered moderate, and a vehicle should be designed to be stable in this condition. Instability under braking is only expected at closer to $2\text{-}3\text{Gs}$.

Under braking the critical speed of the vehicle is reduced, which is consistent with the 3 DOF results, however the reduction in critical speed is not as dramatic, which is a result of including aerodynamic forces. At high speeds the aerodynamic loading exceeds that due to the mass of the vehicle. In the 3 DOF case braking will produce a change in the axle normal loads of approximately 18% . However, in the complex model, at high speeds the change in load is in the region of 5% . Consequently, the change in cornering stiffness of the front and rear wheels is less significant.

This high-fidelity study shows the importance of using a vehicle model that captures all of the pertinent features of the real vehicle. In Formula 1 cars the aerodynamic down and drag forces play a significant role in the behaviour of the vehicle, and a study of stability and vehicle handling is incomplete without

considering them.

4.6 Conclusion

High-performance race car simulators cannot reproduce the accelerations experienced in the actual vehicle and so cueing algorithms are used to filter the vehicle's accelerations retaining only those motions that the driver requires in order to perform specific tasks. Identifying the motion cues that the driver needs is a central problem in cueing algorithm design. The majority of the literature aims to minimise the error in the acceleration sensed by the simulator driver's vestibular system. In race car simulators the platform's inability to reproduce the large accelerations makes this approach infeasible. Small changes in tuning parameters produce significantly different cueing filters, and without more specific cueing objectives, a lengthy and subjective hit-and-miss testing process is needed.

In racing, the driver attempts to operate the vehicle on its performance limit, and to do so requires timely, detailed knowledge of the vehicle handling characteristics. The simulator must cue correctly the vehicle's changing state as it approaches, and possibly over steps, its performance limit.

In Section 4.3 the vehicle stability and response to steering perturbations are examined for straight-line constant velocity, accelerating and braking situations. While a frozen-time eigenvalue analysis can be used to study the stability of linear slowly time-varying systems, this technique must be used with caution. In most cases an alternative checking procedure, such as time simulations, or theoretically derived stability checks, should be used to verify the results. In the race-car context stability must be studied across the full performance envelope, and at large accelerations it is unclear whether an eigenvalue-based stability analysis is valid. By applying the conditions derived in [58], it has been shown that the bounds given in (4.59), (4.60), (4.63) and (4.64) can be used to check the

bicycle model stability up to some stability limit speed, which is generally lower than the critical speed given in (4.35). In the braking case, the classical critical speed is only slightly higher than the limit speed for accelerations up to 2G. In the accelerating case there is no classical critical speed, while the bounds (4.59), (4.60), (4.63) and (4.64) predict the vehicle model to be stable at high acceleration and high speed. These results support the notion that the system described by the simple bicycle model can be treated as slowly time-varying, and that the use of frozen-time eigenvalues to infer the stability in the straight-line accelerating and braking context is valid.

Based on the success of the frozen-time eigenvalue analysis in the straight-line case, the method is then applied to a cornering accelerating and braking vehicle. Finally, the stability of a high fidelity model is analysed to include the effects of lateral load transfer and aerodynamic forces.

The analysis shows that the lateral and yaw responses of the vehicle, and its over and understeer characteristics, vary significantly with the operating condition. If the simulator driver is to operate the vehicle on its performance limit under cornering, and assess continuously the vehicle's handling behaviour, these changes must be cued to the simulator driver by appropriate platform motions.

An oversteering vehicle becomes progressively unstable as its speed increases. The vehicle also becomes more responsive and changes in its disposition occur quickly. To control the vehicle in this region, the driver requires low-latency motion cues that correctly indicate changes in the vehicle's motion. This contrasts with an understeering vehicle that does not become unstable, and has smaller responses to perturbations that change more slowly.

As the vehicle speed and acceleration (braking) varies the gain of the from the steering input to vehicle heading direction also changes. The simulator driver should be able to detect these differences through the motion cues. As the vehicle

becomes more unstable the magnitude of the response grows, and this should be associated with stronger cueing. An overdamped or oscillatory response in the vehicle should also be perceptible in the simulator. It is important that all these changing behaviours are not filtered out and “lost” in the simulator. Different vehicle characteristics should be associated with different motion cues and a different feeling.

The side-slip angle behaviour is key in cueing vehicle stability. Following a disturbance input to a stable car the side-slip angle returns to zero thereby re-aligning the vehicle velocity with its longitudinal direction [42]. The side-slip angle is also an indicator of the proximity of the tyres to lateral force saturation. This information derived from the side-slip angle is useful to the driver in his/her efforts to push the vehicle to its performance limit.

The results of the vehicle stability and behavioural analysis from this chapter are applied to the design and tuning of lateral and yaw filters in Chapter 5.

Chapter 5

Motion Cueing Implementation

A number of cueing algorithms that were developed for flight simulator and passenger car applications, were discussed in Chapter 3. The optimal and MPC approaches are adapted here and applied to the race-car motion cueing problem. These two algorithms are selected, because, in the case of the optimal approach it has intuitive tuning and is used successfully in many applications, and in the case of MPC, it is the most recently developed algorithm that is able to consider the kinematic platform constraints directly.

These two algorithms cannot be applied directly to the race-car problem, since, as discussed in Chapter 3, the tilt co-ordination technique which is a significant element of these methods is not applicable. The modifications to the algorithms to allow the independent design of the filters for the longitudinal, lateral and yaw freedoms are detailed in this chapter.

The longitudinal motion is considered first in Section 5.1, with a number of variations of the optimal algorithm being presented. Thereafter, the details of two non-linear modifications to the standard optimal algorithm are discussed.

The lateral and yaw cues are discussed together, as the driver perceives a combined sensation and cannot always distinguish one from the other. This link

between the two cues is described in more detail in Section 5.2.

Finally, in Section 5.3, the MPC method is used to generate longitudinal, lateral and yaw cueing signals simultaneously.

5.1 Longitudinal Cueing Filter Design

The longitudinal motion of a race-car is characterised by high levels of acceleration, particularly under braking, where the high down-force makes decelerations of 4G possible. The magnitude of these accelerations cannot be replicated in the simulator, and furthermore, there can be no sustained motion. The general cueing wisdom is to produce some strong onset motion at the beginning of braking to communicate the sensation of having “hit the brakes”. This acceleration cannot be maintained for any length of time, and will quickly decay, and thereafter the visual aids become the dominant source of information. In the latter stages of braking the driver requires cues that indicate wheel lock-up.

Typically, in motion cueing, after a braking or accelerating manoeuvre the platform will be moved from the extreme of the workspace back to the centre in preparation for the following, unpredicted motion that could be in any direction. This ‘washout’ effect is a commonly used strategy to prevent the platform reaching the workspace limits. The advantage of the race driving scenario is that every braking is consistently followed by an acceleration, since the drivers brake into a corner and accelerate out of it. So, after braking, the platform will be at the rearmost extreme of the longitudinal workspace, and since it is known that an acceleration will follow, there is no need to return the platform to the centre.

The remainder of this section considers the application of the optimal algorithm to longitudinal cueing, as well as the implementation of two nonlinear modules, that aim to reduce the platform miscues.

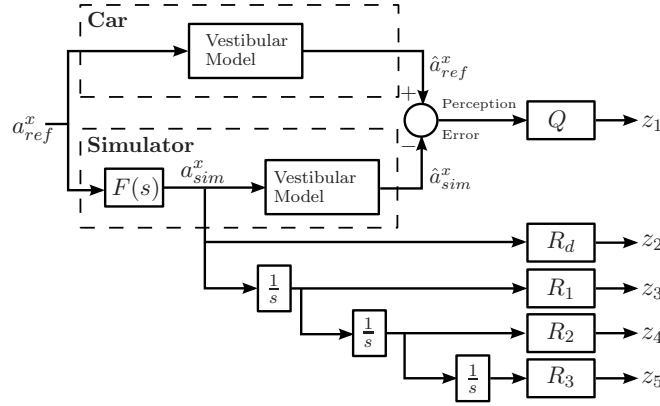


FIGURE 5.1: The topology of the standard optimal cueing algorithm, modified to consider only the longitudinal motion.

5.1.1 Optimal Filter Design

In the literature, the optimal cueing filter is determined by solving the algebraic Ricatti equation associated with a given cost function and plant model. This first principles approach is inflexible, and any modifications to the algorithm require the plant model to be recomputed, which is time-consuming. In this work, the algorithm is recast in the generalised regulator format, and the optimal filter designed using standard algorithms (e.g. the Matlab function *H2syn* which determines the H_2 optimal controller). Within this framework it is easy to modify and extend the system.

5.1.1.1 Basic Optimal Algorithm

The longitudinal cueing design structure is given in Figure 5.1. The acceleration produced by the car model, a_{ref}^x , is the system input. The top signal path generates the vestibular response of a driver in the “real” car. The vehicular acceleration is filtered by the otolith system to generate a perceived motion sensation, $\hat{a}_{ref}^x$. This differs from the method presented in the literature [72], where both the pitch angular velocity and longitudinal acceleration are inputs to a vestibular model that includes both semi-circular canals and otoliths.

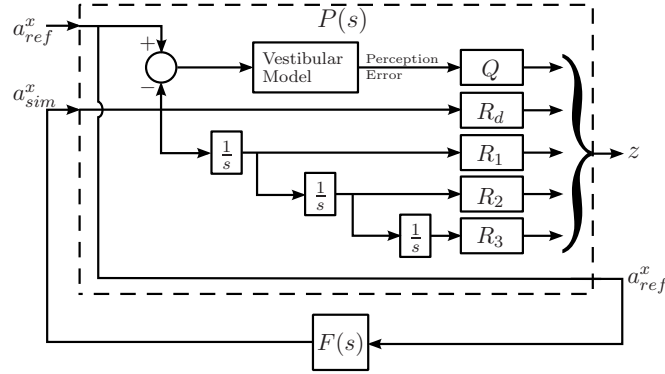


FIGURE 5.2: The basic optimal, translational cueing algorithm recast in the generalised regulator framework.

The lower signal path represents the simulator, where the vehicle accelerations are first filtered by $F(s)$ to produce acceleration demands, a_{sim}^x . The driver's otolith system filters the platform motion to produce a perceived motion, $\hat{a}_{sim}^x$. The perception error is defined as the difference between the motion sensed by the driver in the car and by the driver in the simulator; $\hat{a}_{ref}^x - \hat{a}_{sim}^x$. This error is weighted by Q , and the result z_1 forms the first output. The remaining four outputs z_2, z_3, z_4, z_5 are weighted simulator acceleration, velocity, displacement and absement signals respectively. These are included in the cost function to produce the washout effect, and prevent workspace saturation. The absement signal, which is the time integral of displacement, is included to penalise time spent away from the platform neutral position. The filter design problem can be solved by embedding it in the standard generalised regulator shown in Figure 5.2 [30]. The filter $F(s)$ is synthesized so as to minimise the $\mathcal{H}_2$ -norm of the transfer function $a_{ref}^x \mapsto z$ in which $z = [z_1 \ z_2 \ z_3 \ z_4 \ z_5]'$. This cost function clearly results in a trade-off between the perception error and the simulator motion. If the platform motion is heavily penalised, a larger perception error will result, conversely if the perception error is to be small, increased platform motion is required. With these simple principles in mind, the weights can be tuned until the platform remains within its workspace constraints for the desired test scenario.

5.1.1.2 Frequency Weighted Cost Function

In the first modification of the basic algorithm the perceptual model is still used, and frequency weighting is introduced. The constant weights Q , R_d , R_1 and R_2 in 5.2 are replaced by filters $w_1(s)$, $w_2(s)$, $w_3(s)$ and $w_4(s)$, respectively. This allows the perception error and platform states to be both frequency-shaped and weighted before being included in the optimisation cost function. The platform absement is neglected in this implementation. It was previously included to allow a penalty to be placed on the platform having a large large displacement for a long time. This behaviour is associated with low frequency motion, which in the current implementation can be penalised more heavily than high frequency. This added flexibility makes the absement redundant.

In the simulator low frequency perception errors are unavoidable, because the low frequency accelerations correspond to motions with large displacements and cannot be rendered. Consequently, the filter $w_1(s)$ is a high-pass filter so only high-frequency perception errors are penalised. Since the workspace is consumed by low frequency motions, the platform states are all low-pass filtered.

The weighting filters are implemented as second-order Butterworth filters with a cut-off frequency of ω_c and a gain of K . The low-pass transfer function is given by:

$$w_{LP}(s) = \frac{K\omega_c}{s^2 + \omega_c\sqrt{2}s + \omega_c^2}, \quad (5.1)$$

and, the high-pass by:

$$w_{HP}(s) = \frac{Ks^2}{s^2 + \omega_c\sqrt{2}s + \omega_c^2}. \quad (5.2)$$

The weighting filters provide an alternative tuning technique from the standard approach. By changing the filter cut-off frequencies and gains a more direct

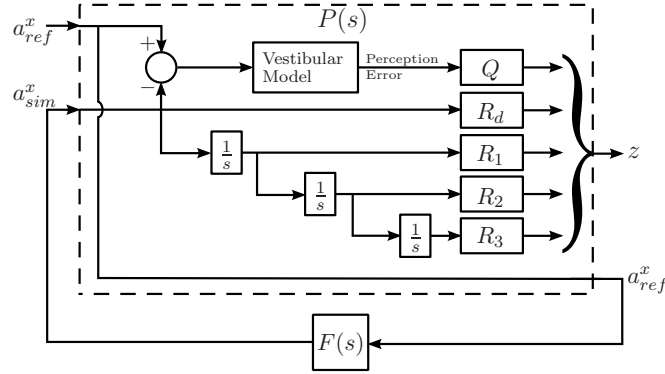


FIGURE 5.3: The basic optimal, translational cueing algorithm recast in the generalised regulator framework with frequency weights applied to the elements of the cost function.

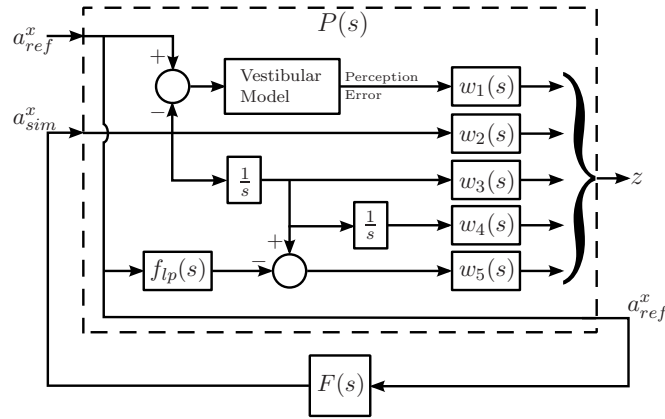


FIGURE 5.4: The translational optimal cueing algorithm with a miscue penalty introduced to the cost function.

control of the errors and platform motions is achieved.

5.1.1.3 False Cue Penalty

A well-documented and observed problem with the optimal filter approach is the miscues (or false cues). These occur after the onset braking cue, and are defined as the platform moving in the opposite direction to the vehicle acceleration vector. As explained in [57], this backlash effect is a characteristic of linear high-pass filters, and is an intrinsic limitation of this technique. They are also a physical necessity: false cues slow down the platform and prevent actuator saturation. They also occur during the washout phase of the cueing.

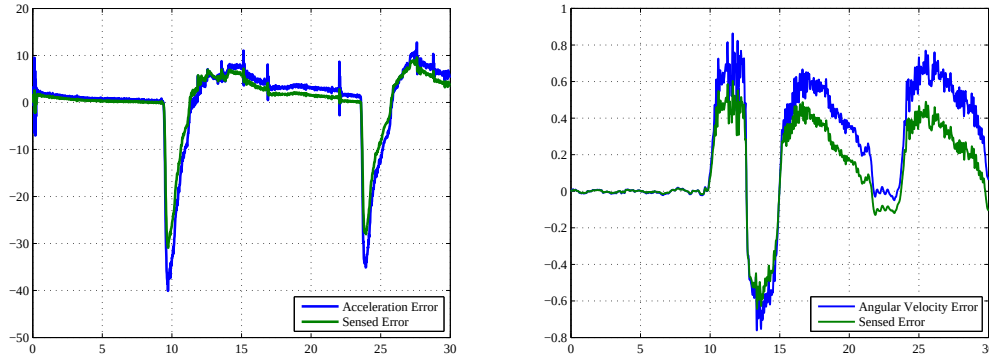


FIGURE 5.5: The left figure contains the error introduced by a typical cueing filter for a race-car manoeuvre, and the error sensed by the vestibular system. The right figure, similarly, compares the yaw velocity error in the simulator with the sensed error.

Ideally, the platform velocity should always be in-phase with the vehicle acceleration. Accordingly, a false cue penalty is introduced into the cost function by calculating the error between the low frequency vehicle acceleration and the platform velocity. This additional term is shown in the cueing topology in Figure 5.4, which retains the perceptual model and frequency weighting described in the previous sections. The false cue penalty term will reduce the miscues, however, in doing so it will also reduce the re-centring behaviour of the platform. The balance between these are found by tuning the weighting filter parameters.

5.1.1.4 Removal of Vestibular Model

A model of the human vestibular system was introduced to flight motion cueing algorithms by [64], with the assumption that it played an important role in the sensing of the cues in the simulator. This hypothesis is reconsidered here in the context of race-cars.

From the vestibular models provided in Chapter 2, the frequency responses of both the semi-circular canal and otolith organs are mostly flat in the region of race-car operation, with a constant gain. This suggests that there is little difference

between the sensed motion, and the motion physically experienced by the body. The left plot in Figure 5.5 contains the longitudinal acceleration error introduced by a typical cueing filter for a normal race-car manoeuvre. The error sensed by the otolith model is also given. Similarly, in the right plot the yaw velocity error and sensed error are given. These signals differ by a constant gain factor, as anticipated. Furthermore, the difference between the vehicle and simulator acceleration error, and sensed error is insignificant in comparison with the magnitude of the error. This evidence argues that the inclusion of the vestibular model has no benefit, while adding to the complexity of the algorithm.

To examine the effect of the vestibular model, optimal filters are also designed without including the perception filters in the plant model. The system structure is similar to that in Figure 5.1, however the vestibular model is removed and the frequency weighted translational acceleration error replaces the perception error in the cost.

5.1.2 Non-Linear Modifications

The linear filtering approach in the longitudinal direction is now extended to include two non-linear post-filtering modules. The first is the anti-backlash filter described in [57]. The performance of this algorithm is compared with an alternative miscue-resaping procedure that is described in section 5.1.4. These two modules reduce false cues, and their design does not involve a perceptual model.

5.1.3 Anti-backlash Filter

The system designed in [57] is illustrated in Figure 5.6.

This additional module applies an adaptive gain to the output of the cueing filter and reduces the magnitude of the acceleration cue during false cueing. The acceleration reference signal, a_{ref}^x , is filtered by the high-pass LQR-designed cueing

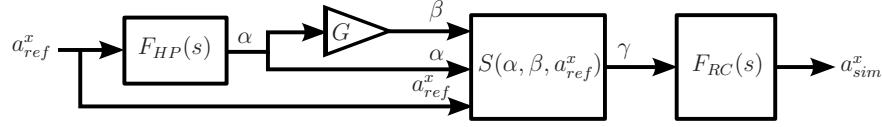


FIGURE 5.6: Optimal filter cueing structure with an additional anti-backlash module.

filter, $F_{HP}(s)$. The adaptive gain G , given by

$$G = \exp(-k|\alpha - a_{ref}^x|), \quad (5.3)$$

is applied to the filtered output α ; k is a tuning parameter. The output of the adaptive gain β , the unfiltered signal α , together with the reference acceleration are inputs to a switching function $S(\alpha, \beta, a_{ref}^x)$. The switching function is given by:

$$\gamma = (1 - K)\alpha + K\beta, \quad (5.4)$$

in which K is a smooth switching function given by:

$$K = 0.5(1 + \tanh(|\alpha - a_{ref}^x| - |\beta - a_{ref}^x|)). \quad (5.5)$$

The switch toggles between 0 and 1 as the argument of the hyperbolic-tangent function changes from negative to positive. Therefore, the output γ switches from α to β so as to achieve least error.

The cueing filter is required to slow the platform as well as washout the acceleration demand. The miscue reduction produced by the switch $S(\alpha, \beta, a_{ref}^x)$ may produce a non-zero platform terminal velocity. To correct for defect, the output γ is passed through a high-pass re-centring filter $F_{RC}(s)$ that assists in maintaining the platform within its workspace.

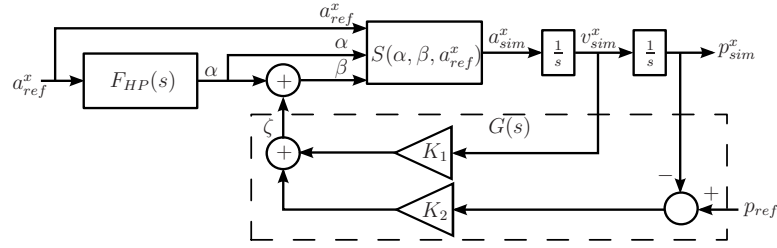


FIGURE 5.7: Structure of the cueing filter with a miscue reshaping module.

5.1.4 Miscue Reshaping Module

The anti-backlash approach described in the previous section does not include prepositioning. Additionally, an extra re-centring filter is required, which affects the shape of the onset cues. To address these issues another non-linear module has been designed with the following requirements:

1. The additional module must not change the acceleration onset cues;
2. The false cues must have lower magnitude, with necessarily longer duration than the standard LQR filter;
3. The platform must be prepositioned before a manoeuvre;
4. The shape of the onset cues and false cues should be easily, and independently, adjustable.

A system shown in Figure 5.7 fulfils these requirements. The reference acceleration a_{ref}^x is filtered by $F_{HP}(s)$ to produce an output α . The filter is second-order, high-pass and designed to produce large-magnitude onset cues that do not exceed workspace limits.

The output of the feedback controller $G(s)$ is:

$$\zeta = K_1 v_{sim}^x + K_2 (p_{ref} - p_{sim}^x). \quad (5.6)$$

where ζ is an acceleration signal that drives the platform velocity to zero and the position to p_{ref} . The reference position is $+1$ when the vehicle is accelerating, -1 during braking and so the platform is prepositioned for the next manoeuvre. The selection of the feedback gains K_1 and K_2 in (5.6) influences the shape of the acceleration trajectory. The switching function $S(\alpha, \beta, a_{ref}^x)$ produces the acceleration cue as before.

In the onset phase of a manoeuvre $a_{sim}^x = \alpha$ and so the simulator acceleration demand is the high-pass filtered reference signal. After this the onset cue $a_{sim}^x = \beta$ and the platform is slowed to a halt at the reference position. The switching function detects the beginning of a manoeuvre through the sign change in the vehicle's acceleration. At this time the output is set to α and after Δt seconds the output switches to β . The delay Δt is chosen to maximise the duration of the miscue phase without interfering with the onset cue.

5.2 Lateral and Yaw Cueing Filter Design

From the vehicle stability analysis in Chapter 4, it is shown that the lateral and yaw responses of the vehicle, and its over and understeer characteristics, vary significantly with the operating condition. The changes in the vehicle behaviour as it nears the performance limit should be cued to the simulator driver by appropriate motion cues.

In cornering, the lateral and yaw motion cues must be examined simultaneously, since in concert, they create the sensation of rotating about a centre of turning. The location of the centre of turning varies continuously and is strongly influenced by the car's over and understeer characteristics; these changes should be considered in the design and tuning of cueing algorithms, and are discussed in more detail below.

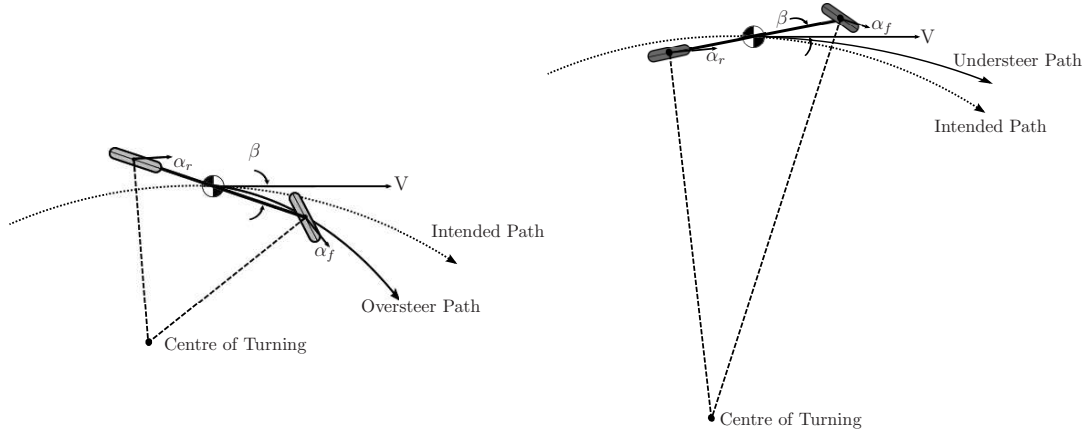


FIGURE 5.8: The kinematics of an oversteering vehicle navigating a turn are illustrated in the left-hand figure, while those of an understeering vehicle are illustrated in the right-hand diagram.

5.2.1 Centre of Turning

The centre of turning is a point in the reference frame of the vehicle that is absolutely stationary. One way to find the centre of turning is to determine the intersection point of perpendiculars to the front- and rear-tyre ground contact point velocity vectors. This point moves as the front- and rear-tyre slip angles change. The centre of turning can also be found in terms of the vehicle's longitudinal, lateral and yaw velocities.

The absolute velocity of a general point, $\mathbf{P}$, in the vehicle's reference frame is given by

$$\mathbf{v}_P = \boldsymbol{\omega} \times \mathbf{P} + \mathbf{v}_{car}, \quad (5.7)$$

where $\mathbf{v}_{car}$ is the car's absolute velocity. The centre of turning is a point $\mathbf{P}_t$, which is absolutely stationary, and therefore given by

$$0 = \boldsymbol{\omega} \times \mathbf{P}_t + \mathbf{v}_{car}, \quad (5.8)$$

or

$$\begin{pmatrix} P_{tx} \\ P_{ty} \end{pmatrix} = \frac{1}{\omega} \begin{pmatrix} v \\ -u \end{pmatrix}. \quad (5.9)$$

The components of $\mathbf{P}_t$ are P_{tx} and P_{ty} , which are the distances of the rotation centre from origin of the car-fixed coordinate system. It is clear that

$$P_{tx}^2 + P_{ty}^2 = \frac{u^2 + v^2}{\omega^2} = \frac{V^2}{\omega^2} = R^2,$$

where R is the radius of turn.

The locations of the centres of turning for oversteering and understeering vehicles are compared in Figure 5.8.

When navigating a turn, a vehicle is said to exhibit *oversteer* if $R_{int} > R$, where R_{int} is the intended radius of turn. The vehicle tends to steer inside the intended path (oversteer) if there is less grip available at the rear of the car than at the front.

Understeering is associated with reduced front-tyre grip. In this case the vehicle will tend to run outside the intended path, with the front slip angle increasing more rapidly than that at the rear.

Changes in the perceived location of the centre of turning is important in the simulator. Although it is not possible to create a sustained sensation of cornering, miscues that create a false perception of the centre of turning location can lead to misinterpretations of the vehicle's behaviour.

5.2.2 Lateral Filter Design

The lateral filter design framework, shown in Figure 5.9, is identical to the longitudinal freedom, and includes frequency weighted states as well as a vestibular model. The difference between the lateral and longitudinal approaches lies in

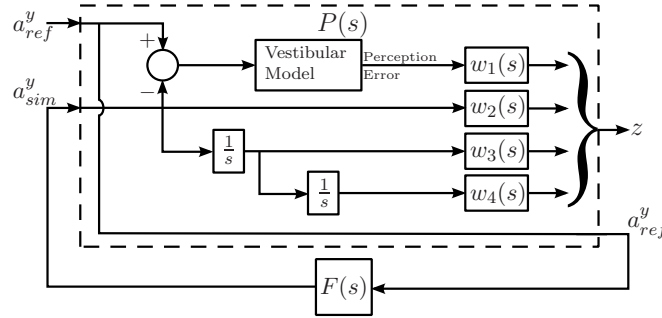


FIGURE 5.9: The optimal lateral cueing filter design problem cast in the generalised regulator framework [30, 60].

the tuning of the cueing filters, since the lateral direction is not characterised by onset cues as strong as those of the longitudinal direction. Furthermore there can be two lateral manoeuvres that are in the same direction. This implies more washout would be necessary.

5.2.3 Yaw Filter Design

The design of the yawing filter follows a similar process to that used in the design of the lateral acceleration filter. Within the capabilities of the motion platform, the filter is designed to minimise the yaw velocity (not acceleration) error, since humans can sense the vehicle's angular velocity.

The yaw cue filters presented in the literature are designed to take the vehicle yaw velocity as input [73]. The simulator hardware requires coherent yaw acceleration, velocity and displacement signals. Since numerical differentiation is noisy, it is necessary to design a yaw acceleration cueing filter. The filter design problem that achieves this is shown in Figure 5.10. This resembles the translational freedoms, although the input to the vestibular model is yaw velocity error.

An alternative yaw cueing strategy is presented in [26], where the vehicle side-slip angle is the platform yaw demand rather than the filtered yaw velocity. There are two advantages associated with using the side-slip angle as a cue: firstly, no filtering (which may remove critical information) is required, and secondly, the

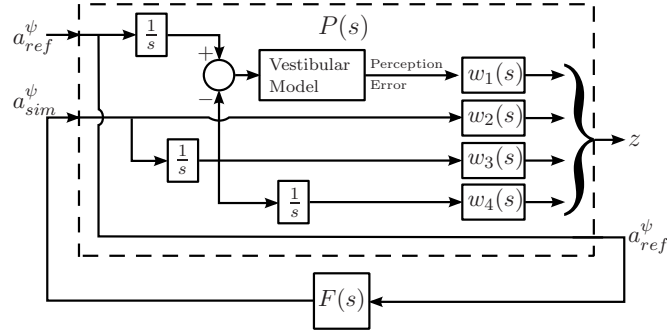


FIGURE 5.10: The yaw-cueing washout filter optimisation problem in generalised regulator form.

driver receives correct vehicle side-slip information. The tests provided in [26] show that in ‘normal’ driving scenarios side-slip cueing gives better results than traditional yaw-rate cueing. Side-slip angle cueing and filtered yaw-rate cueing will be compared for limit driving in this paper.

5.3 MPC

The MPC cueing algorithm described in Chapter 3 is applied to the race-car problem. The plant dynamics are shown in Figure 5.11, with the state vector given. The control inputs are the platform longitudinal, lateral and yaw accelerations. These are filtered by the vestibular models to generate sensation error. The accelerations are also integrated twice and transformed to platform actuator demands. These signals are constrained to observe physical limits.

Recalling that the cost function is given by:

$$J = \sum_{i=0}^{H_P} y^T(t+i)Qy(t+i) + u^T(t+i)Ru(t+i), \quad (5.10)$$

where,

$$y(t+i) = x(t+i) - x_{ref}(t) \quad i = 0, \dots, H_P. \quad (5.11)$$

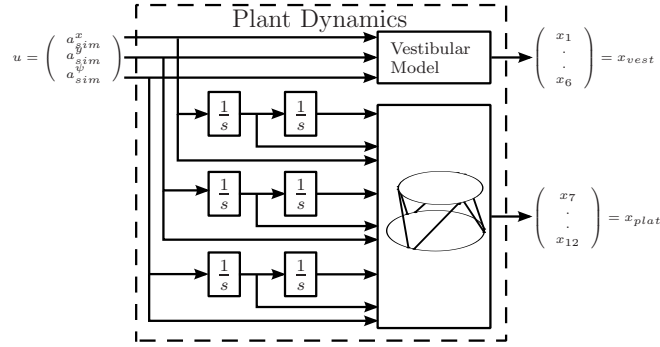


FIGURE 5.11: The plant dynamics of the MPC cueing problem.

A reference state vector is required for every time step in the prediction horizon. The reference vestibular states are generated by assuming the vehicle accelerations are constant in this time. The reference platform states are assumed to be zero, which since these are included in the performance index, introduces a penalty on platform motion.

The MPC algorithm can be tuned by adjusting the control and prediction horizons, and the cost function weights Q and R . The weights associated with the platform acceleration, velocity and position can be set to 0. This implies that while the workspace constraints are not violated the platform will follow the reference acceleration. Once the limits are encountered the platform motion will be brought to a stop. By including the platform states in the cost function, it is possible to change this characteristic, and the solver will need, similarly to the optimal cueing algorithm, to balance platform motion and perception error. This provides more flexibility for adjusting the algorithm in response to driver feedback and it also allows more than just an onset cue to be provided.

5.4 Summary

The application of the optimal cueing algorithm to race-car simulators has been explained in this chapter. The filters produced using this approach are heavily dependent on the tuning weights. Adjustments to these change the cueing

characteristics, and the amount of platform workspace required.

The details of longitudinal filter tuning and the experimental results are presented in the next chapter. These are compared with the outcome of the nonlinear filter cueing systems.

While longitudinal motion can be considered independently, lateral and yaw interact in the generation of over and understeer feeling. In the analysis of the cueing results, the lateral and yaw signals should be considered in conjunction. The details of the filter tuning and the driver comments from testing are given in the following chapter.

Finally, the MPC algorithm has been adapted to cue the 3 large motion freedoms. In the next chapter, the tuning of this method and the cueing signals produced are compared with the results of the other algorithms.

Chapter 6

Experimental Results

The cueing algorithms described in the previous chapter are implemented and tested with experienced test drivers in the Scuderia Ferrari (Formula 1) race-car simulator. Each motion freedom was tested individually, with the newly designed cueing filters replacing the existing filter in the simulator system. The motion in the other directions was unchanged so that the drivers were able to comment on the particular filter.

The simulator drivers are accustomed to their existing cueing system and have developed an internal model that maps simulator cues to vehicle behaviour. When the cueing is changed, the simulator motion to vehicle behaviour associations need to be adjusted. Being skilled professionals, they are able to do this within a couple of laps of a track and are then able to give useful feedback about the cueing.

It is well-known that people will judge simulator motion differently. In passenger car applications, with a sufficient number of subjects and by performing a variety of tests and asking for ratings on the cueing systems, it is possible to perform statistical analysis to judge which cueing algorithm is better. In racing simulators however, there are only a small number of professional test drivers in any team and quite often each driver will have his/her own and different opinion. Some drivers

are also resistant to change and will be consistently negative about anything that feels different. With these factors in mind, when testing the cueing filters, the drivers were specifically asked to comment on which elements of the vehicle behaviour were present or missing and the manner in which this affected their ability to drive the vehicle on the limit.

As mentioned earlier in this thesis, the race drivers use the simulator to evaluate vehicle set-ups. The drivers regard a good set-up as one in which the car has good grip and balance, there is little understeer, and no snap oversteer. All these elements affect the limit of the vehicle, i.e. how late the vehicle can brake, and how quickly it can navigate a turn. The purpose of the simulator cueing is to assist the driver to find this limit and a good cueing system is one in which the driver is easily able to detect the effect of set-up changes.

The different cueing strategies were tested on the Barcelona closed-circuit track shown in Figure 6.1. This track is commonly used for cueing testing because it features large magnitude braking, and both low and high speed corners. The speed profile, longitudinal and lateral accelerations and yaw velocity for a typical lap are shown in Figure 6.2. At the beginning of a lap the driver brakes into Turn 1 and then accelerates out of the corner. In Turns 2 and 3, the driver lifts off the throttle on entry into the corner and then accelerates at the exit of the bend. Similar decelerate-on-entry and accelerate-on-exit speed profiles are exhibited for the other corners in the lap.

The remainder of this chapter is arranged as follows: the results of the longitudinal linear and non-linear cueing strategies are given in Section 6.1, the lateral and yaw cueing test results are described in Section 6.2, and finally in Section 6.3 the results of the MPC algorithm are discussed.

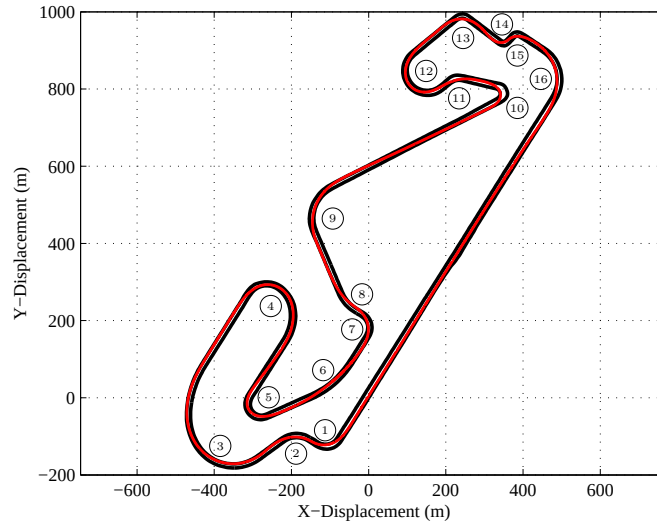


FIGURE 6.1: The racing line on the Barcelona Circuit with corner numbers.

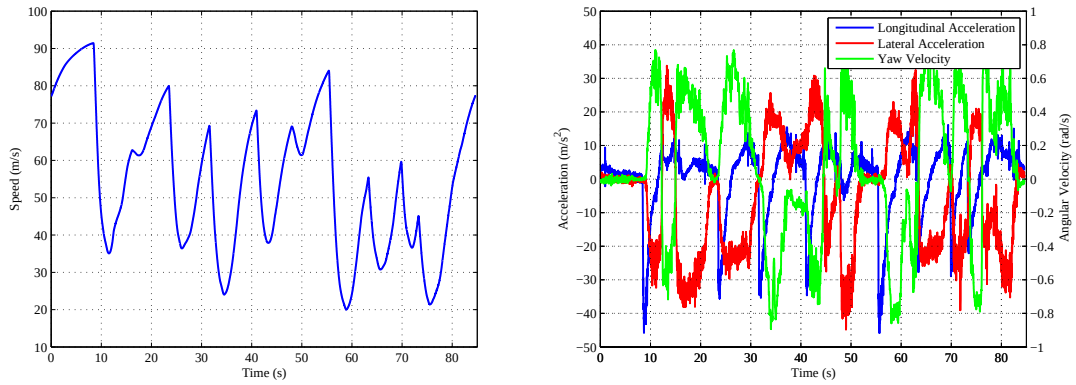


FIGURE 6.2: The left figure contains the speed profile for a lap of the track and the right figure contains the lateral and longitudinal accelerations and the yaw velocity around the circuit.

6.1 Longitudinal Testing

Different linear filters designed using the methods in the previous chapter are tested and compared here. The driver response to the non-linear longitudinal filter methods described in the previous chapter are also discussed.

6.1.1 Optimal Filter Designs

The cueing filters produced by the optimal methods depend not only on the problem set-up but also on the choice of cost function weights. While changing the plant model affects the filter order, it is by changing the weights that the shape of the filter response is influenced. The cost function weights are selected through trial-and-error. The weights are adjusted, the cueing filter designed, a sample lap of track data is filtered, and platform motion demands are generated. This process is repeated until the cueing demand does not exceed the workspace. This aspect of the tuning addresses only half the cueing problem, which is to generate a cueing signal that respects the platform limitations. The second part of the problem is to produce motion cues that communicate the required vehicle information to the driver. This is the more challenging task and requires some prior knowledge and extensive testing with drivers. From discussions with the test drivers, it is known that they prefer strong cues at the beginning of braking and for the remaining time use the longitudinal cue to assist with the detection of wheel spin and lock-up. With this in mind, the cueing filters are tuned to produce strong onset cues and are then tested with a race-driver for feedback. The driver comments on each cueing filter are discussed below and the different options are compared. The numerical values of the tuning weights and the filter transfer functions are given in Appendix C.1 and the Bode response of the different filters are given in Figure 6.3. The acceleration cues and associated platform velocity and displacement demands are shown in Figures 6.4 and 6.5 for Turn 1 of Barcelona since this corner features a large deceleration at the end of the straight. The platform position for each of the cueing options for the entire lap of the circuit is given in Figure 6.6

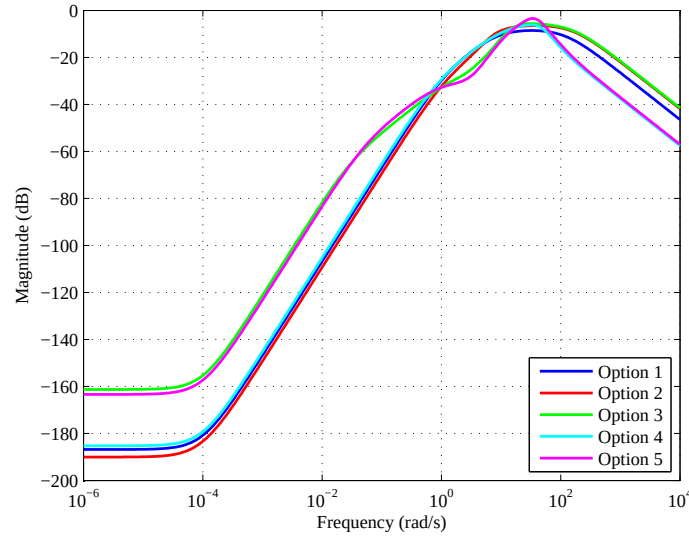


FIGURE 6.3: The Bode magnitude plots of the longitudinal cueing filters tested with the race drivers.

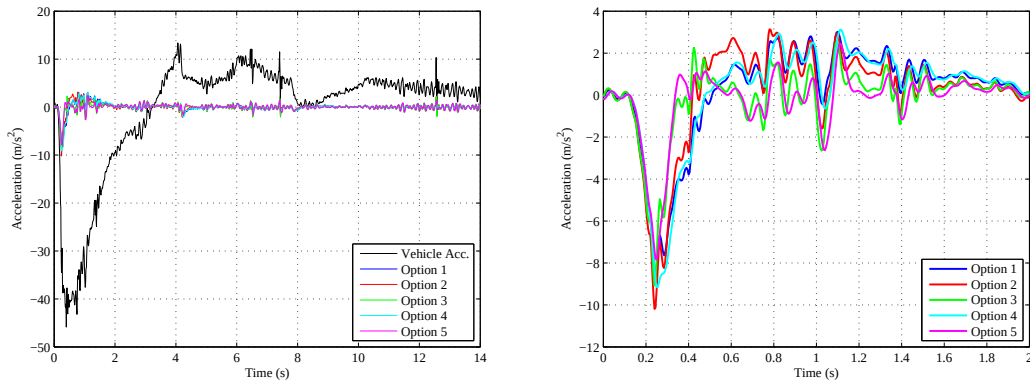


FIGURE 6.4: The left figure contains the longitudinal acceleration cueing signals produced by the different filter options for the braking into and accelerating out of Turn 1 of the track. The right figure shows the onset cues (the first 2s) more clearly.

6.1.1.1 Basic Optimal Filter

The basic filter design problem contains 5 tuning parameters, which are associated with the perception error and platform acceleration, velocity, position and absence. The platform movement is confined to the available workspace by placing sufficiently large weights on the motion states. The shape of the acceleration cue,

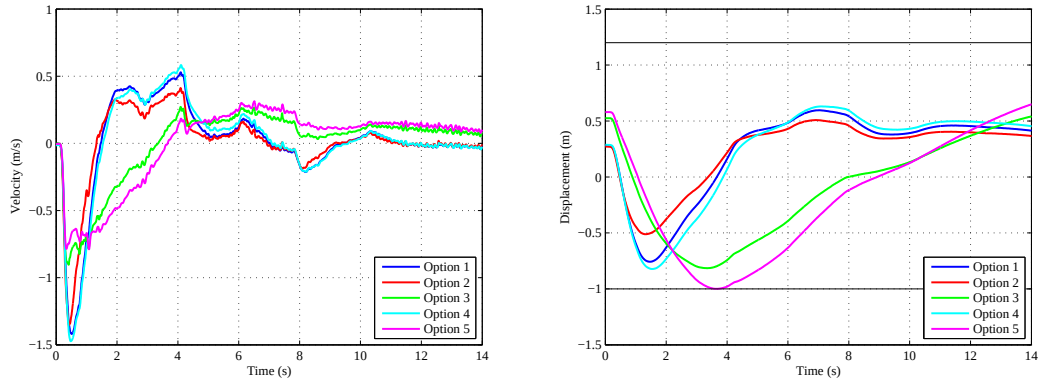


FIGURE 6.5: The two plots contain the platform longitudinal velocity and position demands for each of the cueing options, for the braking/accelerating manoeuvre in Figure 6.4.

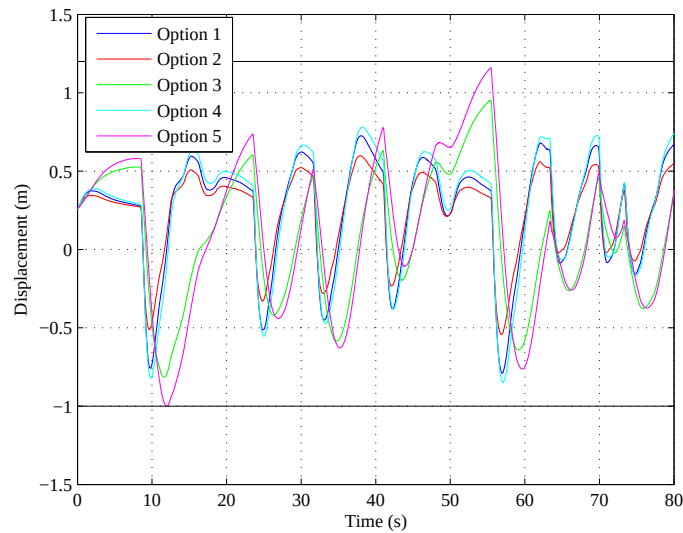


FIGURE 6.6: The platform displacement during an entire lap of the Barcelona track is given for each of the longitudinal cueing options. The platform limits are also shown to illustrate the fraction of workspace that is utilised.

however, depends on the ratio between the weights. For example, a large acceleration penalty will reduce the peak value of the onset cue, but will allow a long duration, low magnitude acceleration. In contrast, a lower acceleration penalty but a relatively higher displacement weight will produce a shorter but stronger onset acceleration and more of the high frequency information will be retained. Since it is known from experience that this cueing behaviour is preferred, the

filter tested was tuned accordingly.

The final filter was 4th order and the braking cue generated by it is characterised by a strong onset cue followed by a large miscue, which is shown as Option 1 in Figure 6.4. This false cue from 0.4s to 2s is correlated with the platform velocity that is out-of-phase with the vehicle acceleration and the re-centring of the platform during the vehicle braking phase, which can be seen from 1.5s to 3s in the plots of Figure 6.5.

After completing two laps of the Barcelona circuit the test driver made the following comments about the cueing filter.

1. The large magnitude onset cue provides a strong sensation of stopping and is a good indicator that the vehicle is braking.
2. The motion sensed throughout braking is not constant. There is a feeling of braking, then a loss of braking motion, followed by a sensation of braking again.
3. Under acceleration, when the driver lifts his foot off the throttle, or reaches the maximum power, he receives a braking feeling.
4. The driver feels that the platform is too sensitive in the longitudinal direction. Overall the simulator moves too much, which makes the driver feel slightly sick.

The driver comments can be correlated with the platform motion demands. The first comment is a positive response to the strong onset braking cues, however, the second comment is associated with the false cue that immediately follows. The positive acceleration of the platform during the miscue is interpreted by the driver not as a forward acceleration but as a loss of braking feeling. In the simulator, the visual cues help to compensate for the washout of the onset cue. However, the

large miscues cannot be disguised so the platform motion still interferes with the driver's perceptions and makes it more difficult to operate the vehicle. The driver responds to the loss in braking feeling by attempting to brake harder. Towards the end of the braking manoeuvre the false cue also decays and the sensation of braking returns.

The third comment made by the driver relates to the high-pass nature of the cueing filter. In the vehicle, the driver lifting his foot off the throttle causes a decrease in the positive longitudinal acceleration. Once this signal is high-pass filtered this variation in the positive acceleration becomes a negative acceleration cue. This is seen in the left plot of Figure 6.4 at 8s. This is inevitable in a high-pass filter, but this effect is compounded by the velocity washout. Since the platform velocity has returned to zero, the negative acceleration will cause it to move backwards, not just slow down (while continuing to move forwards). A backwards motion in the simulator is associated with a braking manoeuvre, hence the driver's comments.

The final comment made by the driver indicates that, in general, there is more platform motion than the driver desires. Furthermore, sickness is associated with the platform and visuals providing inconsistent cues. This can be associated with the washout of the platform to the centre of the workspace. Ideally, under braking the platform will only move backwards and under acceleration only forwards. However, the inclusion of the platform displacement and absement in the optimal cost function causes the inherent washout characteristic in the final cueing filter. In racing, the predictable acceleration pattern makes washout unnecessary and undesirable. The platform workspace is not fully utilised, since the platform is continually being re-centred, which makes only half the workspace available for a manoeuvre. Finally, the washout effect causes the large false cues and all those associated problems.

This basic cueing filter has two main limitations, the large false cues and sub-optimal use of the platform workspace. While the effects of these to some extent can be adjusted through different parameter tuning, the issues are features of the problem set-up and are addressed using the modifications suggested in the previous chapter.

6.1.1.2 Frequency Weighting Functions

The inclusion of frequency weighting functions on the elements of the cost function allows greater control over the characteristics of the cueing filter. In the first implementation of this method only the perception error was given a frequency weight, in this case a 2nd order high-pass filter. The tuning parameters for this are the filter cut-off frequency and gain. These, together with the motion state weights, are adjusted to produce a filter with a strong onset cue.

The driver's comments following testing were the following:

1. The braking consistency is better as compared with the basic algorithm, however, it is still not constant throughout.
2. The onset braking cue is good and the feeling at corner exit is good, since he can feel the traction of the vehicle.
3. In some corners, when braking lightly, the cues were too strong.

These comments indicate an improvement over the basic algorithm and can also be correlated with the cueing signals given by Option 2 in Figures 6.4 and 6.5.

The first comment, once again, deals with the miscue that follows the onset cue. The shape of the false cue produced by this filter differs from the previous option. The peak miscue occurs immediately after the onset and then decreases constantly in magnitude, resulting in a more consistent miscue, which, based

on the comments, is more acceptable to the driver. This can be understood by considering a typical braking manoeuvre, which can be divided into three phases: the initial application of the brake during which the brake pressure builds up, the middle phase where the pressure is maintained at a constant level, and the release of the brake towards the end of the manoeuvre. The cueing filter performs well in the first and last phases, delivering a realistic feeling during brake pressure build up and reduction. During the second phase, although the onset cue cannot be maintained, the platform motion should be constant to correlate with a constant brake pressure. In the first cueing filter the miscue changes during this phase, so the perception of the vehicle motion changes which is inconsistent with constant applied brake. With the current cueing option, the miscue is more constant, which improves the consistency of the cueing. Furthermore, the platform velocity during the miscue phase (approximately 1-3s in Figure 6.5) is lower than the first cueing option, which is an improvement.

By examining the filter frequency responses in Figure 6.3, it is seen that this second cueing option has a higher gain at high frequencies as compared with the first cueing filter. This accounts for the stronger onset cue as seen in Figure 6.4, as well as, the third comment that in some corners these cues were too strong.

The modification to the basic filter problem resulted in cueing filter that the driver deemed better. The problems of false cues, however, still persist and are addressed with the false cue penalty option.

6.1.1.3 False Cue Penalty Filter

The aim of this cueing strategy is to reduce the false cues and keep the platform velocity in-phase with the vehicle acceleration. By placing a large penalty on the acceleration-velocity relationship, the false cues are greatly reduced. This,

necessarily, reduces the washout of the platform motion so that it no longer re-centres. This re-centring of the platform is normally required so that the platform at any time can accommodate a braking or accelerating manoeuvre. With reduced washout, after the driver brakes the platform is near the back of the workspace, accommodating only an accelerating manoeuvre. This is acceptable in the race-driving context, since drivers follow a pattern of braking into a corner and then accelerating out of it.

This filter option has a large number of tuning parameters, which provides more flexibility and control. The acceleration error is high-pass filtered and the platform acceleration is low pass filtered. Both filters have gain and cut-off frequency tuning parameters. The platform position and velocity have constant weights. The vehicle acceleration is low-pass filtered (this has a cut-off frequency and gain to be tuned) and this is compared with the platform velocity. The error between these is the miscue penalty which is also weighted by a constant. All of these parameters are adjusted until the cueing filter produces an onset cue and the platform velocity maintains the same sign as the vehicle acceleration.

After being tested with the driver, the following comments were made:

1. There is insufficient cue during the onset of braking.
2. The driver did not feel a mismatch between the visual and motion cues, or notice any false cues.
3. There is a good feeling on corner exit and the driver is able to feel the wheel spin.

This cueing filter addressed the problems discussed in the previous two options. The miscues and inconsistencies were removed, but the strength of the onset cue was sacrificed. It is seen in Figure 6.4 that the onset cue for this Option 3 is slightly lower in magnitude and significantly shorter in duration than the previous

two. This lack of motion is a problem, the driver does not feel like the vehicle is stopping so applies more brake and causes the wheels to lock-up.

The platform velocity remains in-phase with the vehicle acceleration, seen in Figure 6.5. The platform displacement is not washed out to the centre of the workspace. Indeed, at the end of braking the platform is located almost at the rear of the workspace. Over the whole lap, as seen in Figure 6.6, this cueing option uses more of the platform workspace than the other two. A limitation of using a single linear filter is that the workspace is not explicitly respected which makes it difficult to utilise fully.

While this cueing option shows improved workspace usage and false cue reduction, the lack of onset is a significant drawback. The filter can be retuned to provide more onset, however, this will produce a larger platform velocity, which will require washout and then the problems encountered in the first two cueing tests will reappear. This balance of onset cue versus miscues is the challenge in longitudinal cueing. A lot of tuning, tweaking and iteration is required to achieve a filter that will satisfy the drivers. This cueing framework, however, shows the most promise to achieve this.

6.1.1.4 Cueing Filters without Vestibular Model

The final two longitudinal tests were conducted to determine whether the vestibular model is of any assistance in the design of race-car cueing filters. In the first test the vestibular model was removed from the second cueing option, and in second test, the model was removed from the miscue penalty framework. Some retuning of parameters was required, but the cueing demands remained similar to the original options. In both tests the drivers found that there was little difference between the options with and without a vestibular model. The purpose of including the vestibular model is to produce a cueing filter that reproduces with

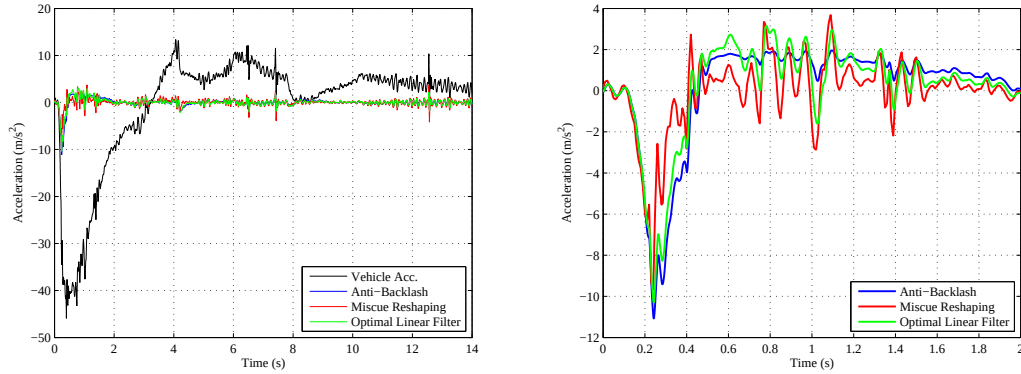


FIGURE 6.7: The left plot contains the longitudinal acceleration demands produced by the anti-backlash and miscue reshaping techniques and the Option 2 linear filter for comparison. The right figure contains the acceleration cueing signals for the first 2s to show the difference in the onsets.

a higher accuracy the accelerations in the frequency ranges with a higher sensory gain. In the race-car scenario this filtering of the vestibular system is insignificant in comparison with the errors introduced by the cueing, and the inclusion of the model does not better inform the cueing filter design.

6.1.2 Non-Linear Filter Testing

The two miscue reduction techniques are compared with the second linear cueing option. The platform acceleration, velocity and position signals are given in Figures 6.7, 6.8 and 6.9.

The adaptive gain in the anti-backlash filter is tuned to reduce the false cues, whilst the re-centring filter was adjusted to ensure compliance with the actuator limits. In the acceleration responses shown in Figure 6.7, the anti-backlash system response is attenuating the miscue present in the linear response. However, all the high-frequency information in the acceleration signal is being lost. This is undesirable, because the associated motions are small (hence do not consume workspace); it is also inconsistent with the rest of the acceleration signal. Referring to the right plot in Figure 6.8 and the left plot of Figure 6.9, one observes that

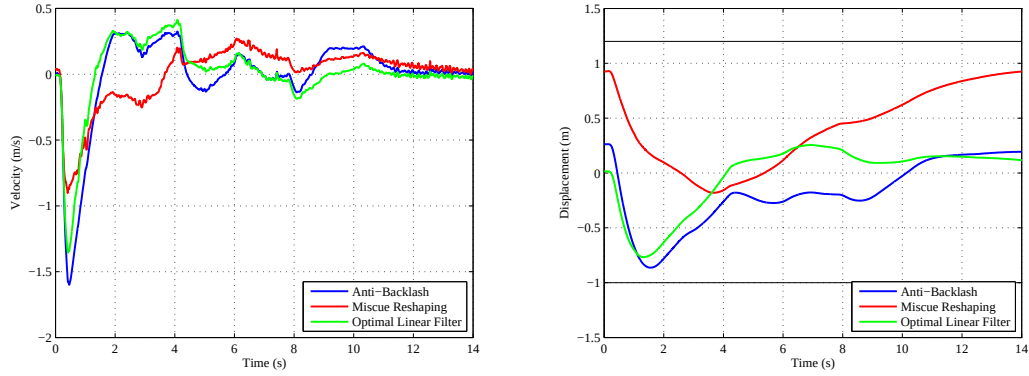


FIGURE 6.8: The two plots contain the longitudinal velocity and position demands, respectively, associated with the anti-backlash and miscue reshaping techniques and the Option 2 linear filter for comparison.

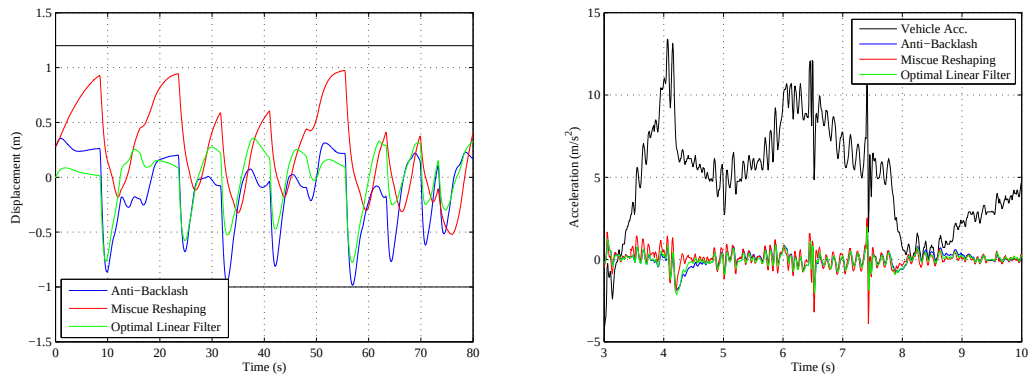


FIGURE 6.9: The left plot contains the platform displacement for an entire lap, as well as the platform limits. The right figure contains a more detailed view of the motion cues under acceleration.

the anti-backlash system does not utilise the full platform workspace, although it is an improvement over the linear filter.

The high-pass cueing filter in the miscue reshaping system was tuned to produce a strong onset cue (Figure 6.7), with the feedback controller adjusted to ensure the platform is correctly positioned for the subsequent acceleration manoeuvre. In contrast to the anti-backlash system, the miscue-reshaping module preserves the high-frequency information in the acceleration reference. The miscue associated with the miscue-reshaping module is weaker than both the linear filter

and anti-backlash methods. This miscue reshaping module makes full use of the displacement workspace through the inclusion of prepositioning, shown in the position plots in Figures 6.8 and 6.9.

Both nonlinear methods were tested with a driver. The following comments were made about the anti-backlash approach.

1. The onset cues are too aggressive.
2. The cueing is inconsistent and confusing.
3. He loses feeling during braking and the system is too sensitive on corner exit.

The onset cues produced by this approach are larger in magnitude and longer in duration than the linear filter option, and the driver found these to be too strong. The main problem, however, with the cueing strategy is the nonlinear reduction in the miscue. As noted, this reduces the high frequency information, and feels inconsistent. Overall this approach has too many different elements that affect the final cueing signal. It contains a linear filter, a non-linear gain to reduce miscues and a re-centring filter. The result is an acceleration cue that is difficult to interpret, and does not communicate the vehicle behaviour.

The driver made the next comments in reference to the miscue reshaping method.

1. The simulator cues are consistent through-out braking, with a strong onset and none of the loss in feeling experienced in the linear filters.
2. The cues during acceleration are too strong which does not correlate with the applied throttle.
3. The cues received when braking lightly are too strong. There needs to be consistency in between the cue strength and the applied brake.

The first comment is a positive response to the acceleration cue that, while slightly lower in magnitude as compared with the linear filter, is as long in duration (left plot in Figure 6.7). The miscue is greatly reduced as compared with the linear filter option, and this is preferred by the driver as it is more consistent with the braking phases in the vehicle.

The limitation of this approach is the strength of the prepositioning. During the accelerating manoeuvres the platform is moved towards the front of the workspace by both the prepositioning effect and the acceleration cues. The result is an acceleration cue that is too strong. It can be seen in the left plot of Figure 6.9 that the acceleration cues of this method are larger than those of the linear filter. This prepositioning problem manifests again during light braking. The braking cue is stronger than expected because of the backwards motion contributed by the prepositioning element.

These two nonlinear methods have illustrated a number of points. Firstly, it is possible to achieve much lower miscues during braking without exceeding the workspace limits. Secondly, as shown in the right plot of Figure 6.9 the nonlinear methods make better use of the available platform workspace than the linear approach. This is particularly true of the miscue reshaping method that prepositions the platform and provides more space to travel during a manoeuvre. Thirdly, these two tests brought to light another issue, and that is consistency between corners. The second method produces strong cues in each braking, attempting to use full workspace in every manoeuvre. However, the drivers need a linear relationship between the magnitude of the braking vehicle and the motion cue to be able to interpret the vehicle behaviour correctly.

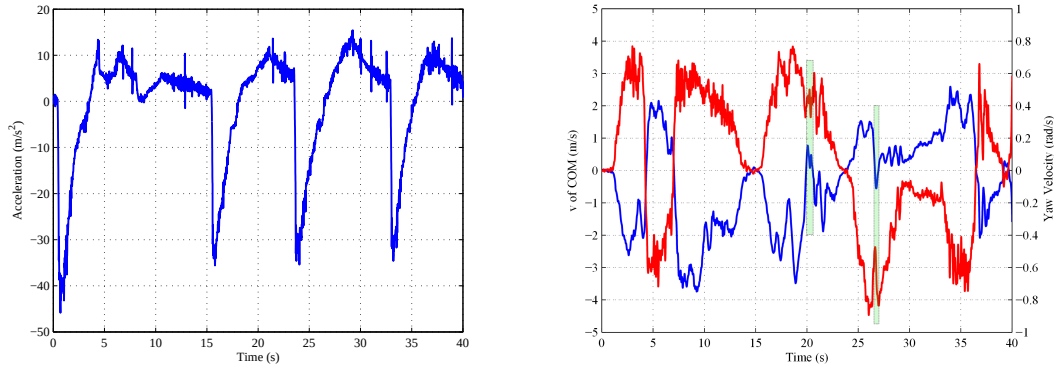


FIGURE 6.10: The left-hand plot contains the longitudinal vehicle acceleration for the first eight corners of the Barcelona Circuit. The vehicle’s lateral velocity (blue) and the yaw velocity (red) are displayed in the right-hand plot.

6.2 Lateral and Yaw Testing

The results of the lateral and yaw cueing filters are discussed for the the first 8 turns in Barcelona. The vehicle’s lateral and yaw velocities for this section of track are displayed in the right half of Figure 6.10, and the longitudinal acceleration in the left half. For the most part, the lateral acceleration and yaw velocity quantities have opposite signs, implying that the car’s centre of turning is behind the car’s centre of mass; see equation (5.9). At approximately 20 s and 27 s there is a change in direction of the lateral velocity and the centre of turning moves in front of the car’s mass centre. In both cases the driver is accelerating out of a corner, and the reduced front-tyre loading causes the vehicle to understeer and run wide in the corner, and so additional corrective steering is required.

6.2.1 Lateral Filter Tuning

Several lateral cueing filters were designed by changing the weighting functions $w_i(s)$ for the different components of the output z in Figure 5.9. The different tuning weights for each filter option are given in Appendix C.2.

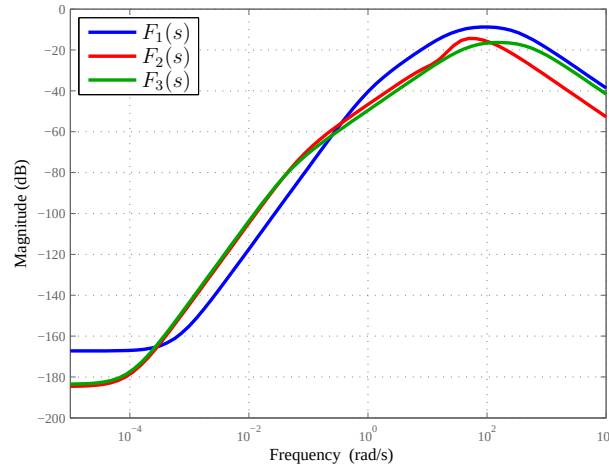


FIGURE 6.11: The Bode magnitude plot of the three lateral cueing filter options.

The first filter option is based on the frequency-weighted optimal approach of Section 5.1.1.2. The perception error is high-pass filtered, while the platform states have constant weights.

The second option is tuned using the false cue penalty approach of Section 5.1.1.3. The perception error is still high-pass filtered and an additional element is included in the cost to penalise miscues.

The final cueing filter is designed similarly to the second option. However, the vestibular model is removed from the system, and a lower order filter results.

The transfer functions for the three filters are given in Appendix C.2, while the Bode magnitude plots for these filters are illustrated in Figure 6.11, where their high-pass properties can be seen.

Figure 6.12 illustrates the performance of the three filters. The platform acceleration, velocity and displacement for the highlighted section of the Barcelona track shown in Figure 6.1 are given in plots (a), (b) and (c), respectively. The response of the vehicle lateral acceleration to a perturbation in steering angle, computed in Section 4.4.1, and shown in Figure 4.9, are filtered by each of the cueing filter options to determine the platform response to a step steering angle input. These

are shown in Figure 6.12 as plots (d), (e), and (f).

The cueing filter $F_1(s)$ is characterised by a strong onset cue as is evident in the platform velocity response, which must be washed out in order to prevent workspace saturation. This cueing option uses the largest ‘per-corner’ fraction of the workspace (of the three filters) during each corner and returns the platform to a near-central position following each cornering manoeuvre. The discussion in Section 5.2 stressed the importance of cueing behaviours that are qualitatively similar to the vehicle’s actual response. In this instance this qualitatively similarity is not retained, because as the vehicle speed increases the platform response to steering perturbations decreases in magnitude (shown in (e) in Figure 6.12) as opposed to the opposite trend shown in Figure 4.9, and no warning of impending instability is provided. When the vehicle is unstable, the cueing filter produces an onset cue, which, while longer in duration, is not in the first half second significantly different from the stable-car responses. While this filter was tuned to produce a cueing demand that aims to make full use of the workspace in each corner, and position the platform in the centre in a manner consistent with general cueing wisdom, it loses some critical vehicle behavioural information. Additionally, there is a significant difference between the platform’s lateral velocity and the lateral velocity of the car. During the 10-12s washout period the platform’s lateral velocity is in the opposite direction to that of the vehicle. This implies that in the simulator the centre of cornering will move behind the vehicle mass centre in a manner not consistent with the vehicle’s actual turning behaviour. The behaviour predicted from this analysis is correlated with the comments of the test driver who complained of confusing cues that were the opposite of the real vehicle behaviour. He was unable to control the vehicle on the performance limit, which was reflected in increased simulator lap-times. The driver also complained that the platform ‘moved too much’, suggesting that such strong cues,

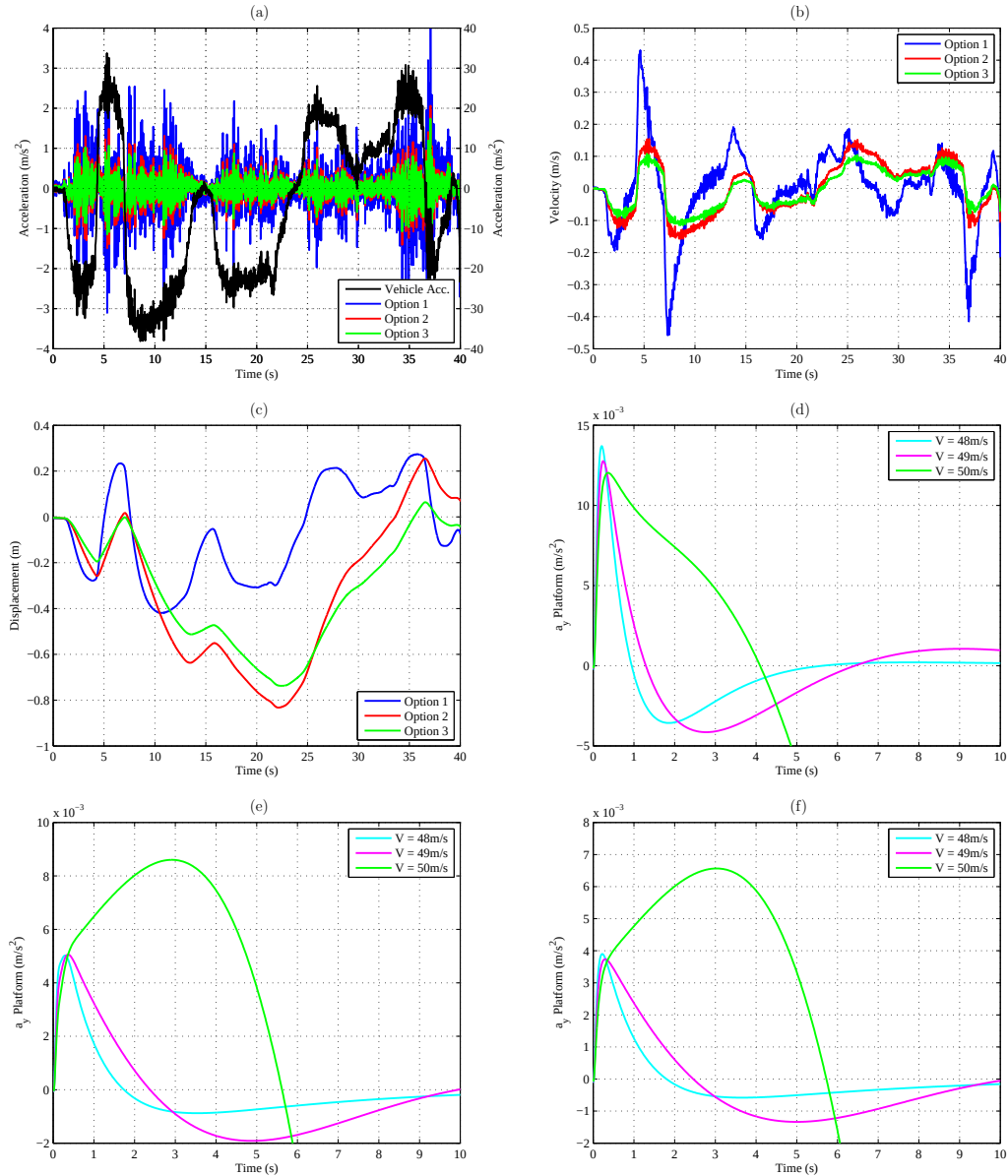


FIGURE 6.12: Reference lateral acceleration for the section of the Barcelona circuit highlighted in Figure 6.1. Figure (a) shows (in black) the vehicle's lateral acceleration. The three coloured plots are the simulator lateral acceleration demands for the three $\mathcal{H}_2$ optimal filters. Figures (b) and (c) show, respectively, the associated platform velocity and displacement demands for each of the three filters. Figure (d) shows the vehicle responses from Figure 4.9, filtered using $F_1(s)$, to determine the platform steering-angle step response at three different vehicle speeds. Figures (e) and (f) contain the same analysis for $F_2(s)$ and $F_3(s)$.

when accompanied by washout, are undesirable.

The plots (e) and (f) in Figure 6.12 show that the cueing filters $F_2(s)$ and $F_3(s)$ produce a platform response that is similar to that of the car. The onset cues increase as the vehicle approaches the stability boundary and once unstable, the acceleration cues are much larger. In these designs there is no platform washout and consequently little or no false cueing. However, the removal of washout effects enforces the need for low-amplitude cueing. The absence of strong onset cues means that only the high-frequency content of the lateral acceleration is retained in the platform demand signal. At first sight these filters appear flawed - without washout there is no means of preventing workspace saturation. However, in a racing context, where the car is operating on a fixed closed-circuit track, only a limited number of left-hand corners, for example, can follow in sequence, which makes the simulator workspace easier to manage. A right-hand corner that follows a left-hand corner will ‘unwind’ the platform and move it back towards the centre of the workspace. For the Barcelona track there are never more than three consecutive left-hand or right-hand corners, and so it is possible to tune the filters to avoid significant platform washout. When the vehicle is unstable the platform cue is large and would normally exceed the workspace. However, with professional drivers, provided the initial cueing is correct, the driver will bring the car promptly under control thereby creating a immediate reduction in the lateral acceleration of the simulator platform. If the driver does lose control, the platform cues no longer matter since the car is likely to have left the track. In the case of the $F_2(s)$ and $F_3(s)$ filters the platform lateral velocity is in many ways similar to that of the vehicle. Large rapid variations in the platform velocity are no longer present, and the false cues at 20s and 27s have been removed. Any loss of stability-related information can be partially compensated by yaw cueing strategies that are discussed in more detail in the next section. When the $F_2(s)$

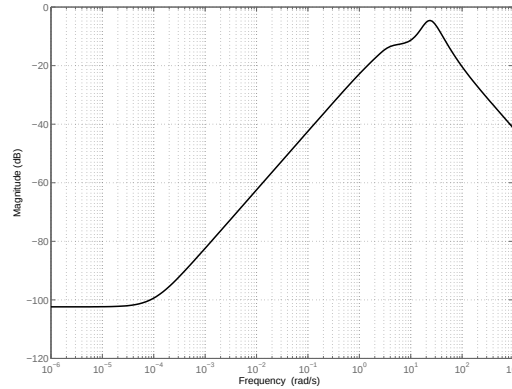


FIGURE 6.13: The yaw cueing filter Bode magnitude response.

and $F_3(s)$ filter options were tested, the driver found that the cueing provide realistic vehicular information, which helped him to control the oversteer on the entry to corners. The lower-order filter $F_3(s)$ was deemed superior, because it has a fast response which delivers timely cues, although they are of lower magnitude. The test driver found that the sooner the cues arrived the easier it was to respond to them and control the vehicle. He also found that $F_3(s)$ provided realistic lateral-grip and car ‘balance’ feedback.

6.2.2 Yaw Filter Testing

The efficacy of using the side-slip angle as a yaw cue is compared with that of using a more conventional filtered yaw velocity cue. The yaw cueing filter was designed as described in Section 5.2.3, with a second order high-pass weighting filter applied to the angular velocity sensation error, and second order low-pass filters applied to the platform position and velocity.

The tuning parameters and cueing filter transfer function are given in Appendix C.3, and the high-pass Bode magnitude plot illustrated in Figure 6.13.

As shown in Chapter 2, as yaw motion is produced by the hexapod the longitudinal and lateral workspaces decay. The platform yaw displacement is therefore

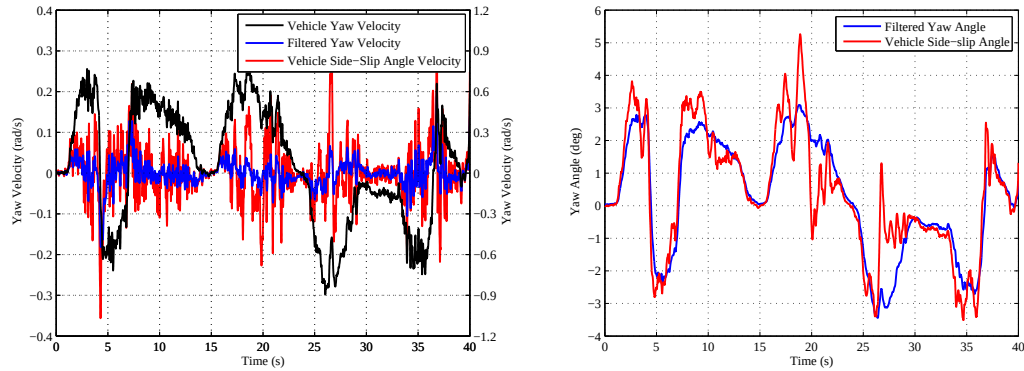


FIGURE 6.14: Yaw velocity and side-slip angle cues. The left-hand figure shows the vehicle’s yaw velocity (associated with the right axis), the filtered yaw velocity and the vehicle side-slip angle velocity (both associated with the left axis). The right-hand plot shows the platform yaw angle for the filtered yaw velocity and the vehicle side-slip angle.

restricted to an artificial, smaller workspace to prevent interference with the other motions. For this study the yaw motion is limited to $\pm 4^\circ$ and so the cueing signal necessarily contains only high-frequency yaw information. The yaw cue demands for the side-slip angle and filtered yaw angle algorithms are shown in Figure 6.14. The left-hand plot contains the yaw velocity of the vehicle, the filtered yaw velocity cue and the vehicle side-slip angle velocity. The filtered yaw angle is compared with the vehicle side-slip angle in the right-hand plot.

The two yaw algorithms were tested with the professional driver. During testing the lateral cues were generated using the final filter (Option 3) and as shown in Figure 6.12. With both yaw algorithms the driver was able to control the oversteer on entry to the corners, interpret the vehicle behaviour and operate the car on its performance limit. The driver reported that the two options felt ‘similar’, however, when compared graphically, the cueing signals show similarities as well as differences. As shown in Figure 6.14 both cueing options demand similar platform yaw angle swings during corner entry. This is consistent with the driver being able to control the oversteer in corner entry with both algorithms. During corner exit, when the yaw angle is returning to 0° , the cueing demands are

almost identical. The differences between the two options occur in mid-corner. In Corner 4 for example, which starts at approximately 15 s, the filtered yaw angle demand increases to 3° and then decreases to 0° , with only small fractional-degree fluctuations. In contrast, the side-slip angle reached a peak of over 5° at approximately 18.5 s when the vehicle understeers and ‘runs wide’ in the turn. The differences in the platform behaviour during this understeer was not noticed by the driver.

As mentioned in Section 6.2, and shown in Figure 6.4, at 20 s the lateral velocity of the vehicle rapidly changes sign, which would cause the rotation centre to move in front of the vehicle COM, and would indicate to the driver that the vehicle is ‘running wide’; this behaviour is not indicated by the lateral cue. As shown in Figure 6.12 (b), the lateral velocity of the platform remains negative at 20 s; see the $F_3(s)$ response. Figure 6.14 indicates that the yaw velocity of the vehicle remains positive at this time. The yaw velocity cue and the side-slip angle velocity are, however, both negative at 20 s. The platform’s centre of turning will therefore move forwards, and a vehicle understeer is cued. This motion is stronger in the side-slip angle case. Referring to Figures 4.10 and 4.11, the vehicle yaw velocity response to a steering perturbation is an order of magnitude smaller when the vehicle is accelerating and understeering, as compared with a braking and oversteering vehicle. The lower magnitude angular velocity response will produce a slower change in vehicle position. The driver was thus able to detect understeer from visual cues. The driver also receives feedback through the steering wheel about the vehicle understeer behaviour. There are thus at least three cues contributing to the driver’s detecting the vehicle understeer and changes in the motion cues are therefore less noticeable.

6.3 MPC Cueing Strategy

Since the challenges of real-time and system integration made driver testing of the MPC cueing algorithm practically infeasible, the results are instead compared with cueing strategies liked by the drivers in testing. This provides a first level of insight into the potential success of this method. The MPC algorithm is implemented for the three degrees of freedom, and at first each motion is considered individually.

The MPC algorithm is implemented for the longitudinal direction. In the first iteration only the acceleration perception error is included in the cost function. The cueing demands are given in Figures 6.15 and 6.16 and are referred to as MPC Option 1. The acceleration cue produced by the algorithm closely follows the reference acceleration until the velocity saturates, at which point the acceleration is zero. As the platform approaches the workspace limit it is decelerated at close to the maximum acceleration. This behaviour is consistent with how the problem is posed. When the state constraints are not active the platform will follow the reference trajectory. However, once the velocity constraint is active, the perception error is minimised by maintaining the platform at a constant velocity. The platform will only be slowed when the position constraint becomes active, hence the abruptness with which it is stopped. However, the longer the prediction horizon the smoother the deceleration of the platform. From experience this cueing characteristic will be disliked for a number of reasons. Firstly, while the velocity constraint is active there are no high frequency cues, and a completely still platform with no vibration or ride information will feel irregular. Secondly, the large miscue that slows the platform will disorient the driver and most likely make him/her feel sick.

The main problem with the first MPC implementation is that the velocity constraint becomes active. To overcome this issue, the platform velocity is weighted and included in the cost function. The cueing signals produced by this approach are denoted by MPC Option 2 and shown in Figures 6.15 and 6.16. It is immediately obvious that this is an improvement on the previous method. The large miscues are replaced by lower magnitude false cues, and much of the high frequency information is retained, particularly when the vehicle is accelerating. Overall, this option is much closer to the nonlinear miscue reshaping option that proved quite successful in testing. The main difference occurs after the onset cue, during the acceleration decay phase there is little high frequency content in the cueing signal. The inclusion of a penalty on the platform velocity prevents the velocity constraint from ever becoming active which ensures a more consistent and smooth cueing. While the full platform workspace is not used under braking, it is better exploited than the miscue reshaping option.

Having produced a feasible cueing signal in the longitudinal direction the MPC method is applied to the lateral cue. From the linear cueing experiments, it is known that miscues in the lateral motion should be avoided where possible. From the first longitudinal MPC option, it is apparent that encountering the platform limits introduces false cues, and to prevent this, the motion should be included in the cost function. The lateral motion produced by the MPC algorithm is compared with the best lateral linear filter cueing signal in the plots in Figure 6.17. The two cueing signals are very similar which is to be expected. If the constraints in the MPC problem are not active, the solution is exactly the solution to the unconstrained LQR problem. The differences between the two cueing options depends on the choice of tuning parameters and the problem set-up, since the linear filter framework included frequency weighting.

The MPC algorithm is implemented for the yaw motion with the results shown

in Figure 6.18. The MPC cue is larger in magnitude although similar in shape to the linear filter based approach. This is once again due to differences in the cost function. The linear filter was tuned to have a similar magnitude to the side-slip angle, although this did not utilise the full workspace.

Finally, the MPC algorithm is implemented to compute the cueing signals of all three motions simultaneously. This prevents infeasible actuator demands when large displacements are required in two freedoms at once. In this event one motion freedom must be prioritised, which is achieved by the relative weights of the perception error of each motion. However, interference between cueing freedoms produces unexpected, irregular motion cues, and therefore should be avoided. Once again, it is preferable to penalise the platform motion such that this scenario is not encountered. As shown in Chapter 2, with the hexapod platform, if the yaw angle is kept small, and the longitudinal and lateral motions are somewhat conservative in the workspace usage, then interference between the different motion freedoms can be mostly avoided. When the algorithm is tuned to avoid interaction between the cueing directions there is little change in the cues, and hence the plots are not shown.

In summary, the most commonly stated advantage of using MPC cueing techniques is that the algorithm explicitly observes the platform constraints. However, as it has been shown, allowing the platform to encounter the constraints introduces abrupt false cues that would be highly disruptive to the driver. By including the penalties on the platform motion in the MPC cost function this problem can be avoided. However, in this case the solution of the MPC problem is that of an unconstrained LQR problem, in which case it is preferable to use the LQR-based filtering, which has none of the real-time implementation challenges of MPC. An advantage of the MPC approach that still remains, is that it allows less conservative weighting of the platform motion. The parameters can be selected so

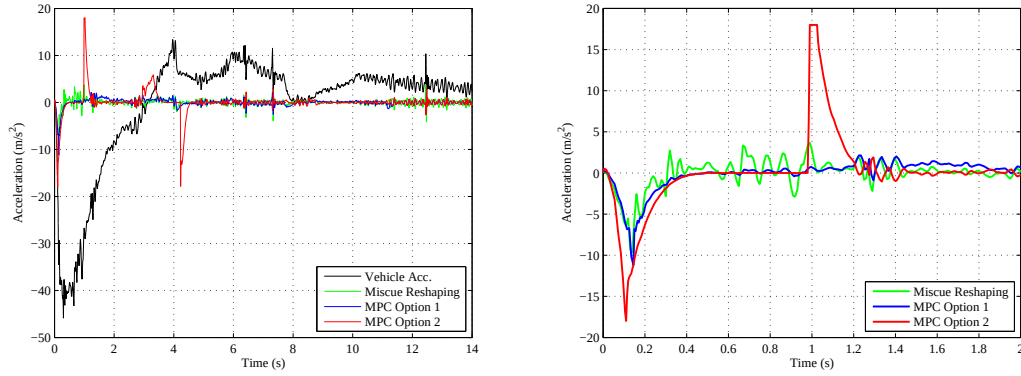


FIGURE 6.15: The left plot contains the longitudinal acceleration demands produced by the the two MPC options and the miscue reshaping technique for comparison. The right figure contains the acceleration cueing signals for the first 2s to show the difference in the onsets.

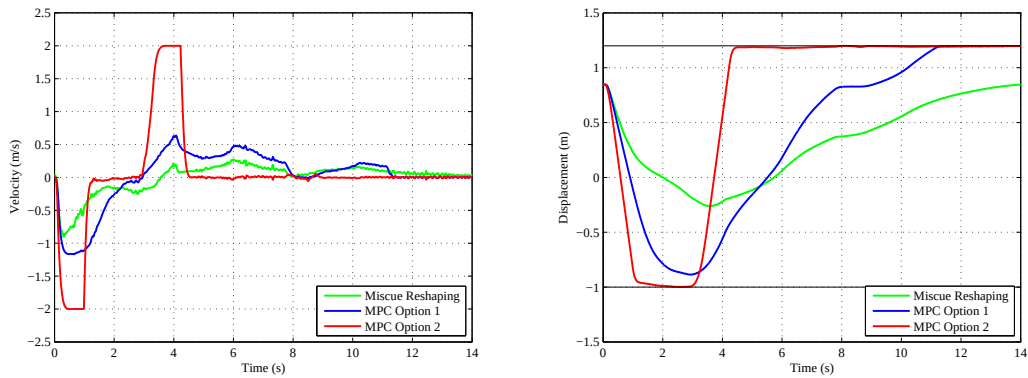


FIGURE 6.16: The left plot contains the longitudinal velocity and the right plot the position demands associated with the two MPC options and, for comparison, the miscue reshaping technique.

that the platform utilises close to the full workspace, while the constraints ensure the platform is safely within the limits in an unexpected event.

6.4 Conclusion

The different cueing methods for longitudinal, lateral and yaw were implemented and tested with professional drivers. From the tests with the linear and non-linear longitudinal cueing options a number of themes emerged from the driver

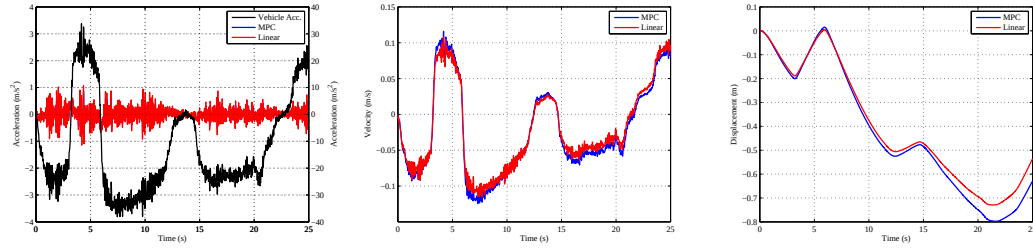


FIGURE 6.17: The three plots contain a comparison of the MPC lateral cue and the best linear lateral cue option (Option 2). The figures contain from left to right the acceleration, velocity and position demands for the simulator.

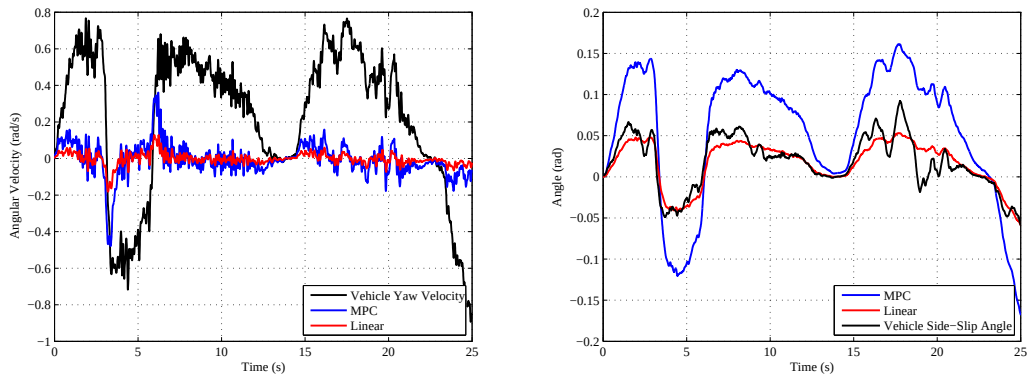


FIGURE 6.18: The left figure compares the MPC yaw velocity demand and the yaw velocity produced by the linear filter. The right plot compares the corresponding platform yaw angles with the vehicle side-slip angle.

comments. Firstly, the drivers insist on strong onset cues and are prepared to tolerate a shorter cue as a consequence. Secondly, continuity is required throughout the braking manoeuvre. While the drivers tolerate the decay in the onset cue, they do not like false cues, and it is important to keep the platform velocity in phase with the vehicle acceleration. This implies that washout to the platform centre is unachievable. However, in racing, where driving behaviour is very consistent and predictable, an acceleration will follow a braking manoeuvre, and the re-centring will be unnecessary. The third recurring comment from the drivers is a need for consistency. The same cues should occur from braking manoeuvres of similar intensities and lighter braking should have correspondingly smaller cues.

This suggests that methods, such as the nonlinear adaptive and some implementations of MPC, that adjust the strength of the cues given the platform's location are not applicable. These methods make full use of the workspace for each manoeuvre, and the aim of the cueing method should rather be to produce consistent cues while using the whole workspace during a lap.

The lateral and yaw cueing filters were tuned to cue the changing vehicle behaviour. Importantly, yaw and lateral velocity motions combine to produce a stationary centre of turning. This, as shown in Section 5.2.1, moves behind the vehicle's mass centre for an oversteering vehicle and in front of it for an understeering one. Any loss of information in the lateral cue can be compensated for by the yaw cue so that together they produce a turning centre in the simulator that agrees qualitatively with that of the car itself. This approach appears successful as the driver cannot usually identify which cue provides this information. Indeed, during testing, the yaw cue was changed and the driver was convinced that the lateral cue had been adjusted.

The driver testing of the lateral and yaw cueing filters is discussed in Section 6.2. The lateral filters were tuned to capture the vehicle's changing behaviour and provide realistic cues. These allowed the driver to control the vehicle at the performance limit; particularly the oversteer at the entry to corners. Both the yaw filter and side-slip angle cueing approaches performed well during testing. The difference between the two options was most noticeable during understeering, but since this is also cued by steering feedback and visual cues this difference was not important to the driver.

While the MPC algorithm was not tested with drivers, the cueing signals were compared with those from the other tests. The MPC implementation yields similar results when it is tuned to avoid hitting the constraints. In this case, the added complexity of implementing the algorithm in real-time is not immediately

justifiable. However, with simulator hardware that has more than 6 DOF and the motion needs to be split between two mechanisms the MPC algorithm is clearly the better solution.

The cueing experiments have shown that much can be achieved in simulator cueing with well-tuned linear filters. The main, well-documented drawback with this approach is that the workspace limits are not explicitly respected. The filters need to be tuned to operate within the workspace for a variety of race tracks. Additionally, it is unsafe to produce platform demands that are close to the platform boundary when it is not explicitly recognised. Consequently, a filter with a conservative workspace usage will normally be implemented. This problem can be overcome by using the MPC approach.

The lateral and yaw experiments have also shown that by studying the vehicle behaviour, and identifying the important motion cues, it is possible to design cueing filters that more nearly meet the race-drivers' requirements. In addition, good cueing results can be found more quickly thereby reducing the test time needed for cue development.

Chapter 7

Numerical Optimal Control

The cueing algorithms described thus far have been concerned with the filtering/attenuating of the vehicle acceleration to produce a signal that can be rendered by a simulator platform without exceeding physical limitations. None of these strategies directly address the problem of seeking best usage of the available workspace. The LQR method, for example, constrains the output to be a linearly filtered version of the input, and does not attempt to make optimal use of the workspace. The MPC method recognises constraints on the workspace, but it only considers the platform motion over a short horizon, and so does not optimise the workspace usage over a full driving session.

In a racing situation the acceleration profile for a lap of a given track is essentially known, although there may be some minor variations between drivers. The aim of this study is to determine the optimal way to use the platform workspace in order to cue the driver over an entire lap. Numerical optimal control is applied to this problem, and in this framework non-linear models can be used, and constraints imposed.

As has been discussed repeatedly, the problem in cueing is determining which

vehicle motions need to be reproduced. This issue is relevant again in this context because the two-fold aim is to provide good motion cues, while using the workspace optimally. In the optimal control framework the cueing characteristic is influenced by the choice of cost function. For example, the acceleration sensation error may be minimised. The first part of this study examines the cues produced by different cost function definitions and develops ones that capture the desirable cueing characteristics.

Having designed cost functions to produce appropriate cueing signals, these can be used to analyse the optimal workspace usage. The interactions between the different motion directions is also studied to determine the effects of the platform kinematic design. The effect of increasing the platform dimensions and dynamic capabilities is also explored.

Finally, the solution of the optimal control problem is compared with the standard cueing techniques. This provides insight into the areas in which the algorithms can be improved.

The vehicle response data for this study is generated by the minimum-lap-time optimal control simulation described in [38, 51]. This 3 DOF car model includes tyre and aerodynamic forces, and captures the vehicle's low-frequency drive dynamics but not the higher frequency car ride (suspension) dynamics. Since workspace analysis is concerned with the filtering of low-frequency data, which results in large displacements, the use of a simplified vehicle model in this context is legitimate. The higher frequency ride dynamics that come from the vehicle suspension can be rendered by the simulator without filtering as a separate exercise and are not a necessary component of this analysis.

The remainder of this chapter is laid out as follows: the optimal control theory is described in Section 7.1, and the cueing problem is formulated in Section 7.2. The results of the optimisation over a lap are discussed in Section 7.3, the enlarged

workspace analysis is described in Section 7.4, and the comparison with the other cueing methods is presented in 7.5.

7.1 Optimal Control Theory

The purpose of an optimal control calculation is to determine the state and control associated with a system in order to minimise a performance index [6]. When expressed in Bolza form, the performance index is given by

$$J = \Phi(t_0, x(t_0), t_f, x(t_f), p) + \int_{t_0}^{t_f} l(t, x(t), u(t), p) dt, \quad (7.1)$$

while the system and operating constraints are given by

$$\left\{ \begin{array}{l} \frac{dx}{dt} - f(t, x(t), u(t), p) = 0 \\ g(t, x(t), u(t), p) = 0 \\ h(t, x(t), u(t), p) \leq 0 \\ g_b(x(t_0), x(t_f), u(t_0), u(t_f), p) = 0 \end{array} \right. \quad (7.2)$$

where $t_0 \leq t \leq t_f$ is the optimisation interval with t_f either fixed, or free to be optimised. The vector $p \in \mathbb{R}^{n_p}$ contains any fixed parameters to be optimised¹; $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ are the state and control vectors respectively. The vector-valued function $f(\cdot) \in \mathbb{R}^n$ describes the system dynamics. The vector functions $g(\cdot) \in \mathbb{R}^{n_g}$ and $h(\cdot) \in \mathbb{R}^{n_h}$ define the equality and inequality constraints for the system. The subscript b refers to the boundary constraints with $g_b(\cdot) \in \mathbb{R}^{n_{g_b}}$. The scalar function $l(\cdot)$ is the stage cost that is a function of the state, the controls and the parameters.

¹ $\mathbb{R}^n$ denotes the set of n -dimensional real vectors.

Direct methods of the type employed here transcribe, or convert, infinite-dimensional optimal control problems into a finite-dimensional optimisation problem with algebraic constraints; a nonlinear programming problem (NLP) [6].

The work in this paper makes use of a Gauss-Legendre-Radua psuedospectral optimal control solution method that has been implemented in the software package GPOPS-II [24, 50].

7.2 Optimal Control Problem Formulation

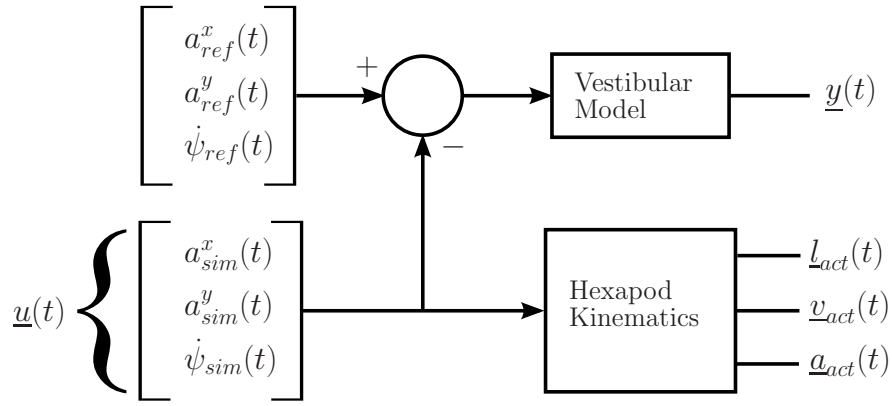


FIGURE 7.1: Formulation of numerical optimal control problem.

The optimal control problem is first formulated similarly to the majority of cueing algorithms, where the aim is to reproduce the accelerations a driver would experience in a real car. If the car's translational accelerations are of the form $a_{ref}^x(t)$ (where the superscript defines the direction of motion), then the acceleration sensed by the driver is given by $\hat{a}_{ref}^x(s) = V(s)a_{ref}^x(s)$, in which $V(s)$ is a model of the human otoliths that are part of the human vestibular system [41]; s is the Laplace variable. The aim of a cueing algorithm is to make the difference $\hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t)$ as small as possible. In the yaw freedom the sensed yaw velocity is given by $\dot{\hat{\psi}}_{ref}(s) = W(s)\dot{\psi}_{ref}(s)$, where $W(s)$ is a model of the semi-circular

canals. The difference $\dot{\psi}_{ref}(t) - \dot{\psi}_{sim}(t)$ must be minimised. The hexapod displacement, velocity and acceleration constraints are included in the problem to prevent the simulator from matching exactly the motions of the vehicle.

The aims and constraints described above can be embedded in an optimal control problem with a quadratic form type of performance index

$$J = \int_{t_0}^{t_f} \underline{y}(t)^T Q \underline{y}(t) dt. \quad (7.3)$$

The optimal control problem set up is shown in Figure 7.1.

The problem constraints are of the form

$$\dot{\underline{x}}(t) = f(\underline{x}(t), \underline{u}(t)) \quad (7.4)$$

$$\underline{y}(t) = g(\underline{x}(t), \underline{u}(t)) \quad (7.5)$$

$$\underline{l}_{min} \leq \underline{l}_{act}(t) \leq \underline{l}_{max} \quad (7.6)$$

$$\underline{v}_{min} \leq \underline{v}_{act}(t) \leq \underline{v}_{max} \quad (7.7)$$

$$\underline{a}_{min} \leq \underline{a}_{act}(t) \leq \underline{a}_{max} \quad (7.8)$$

in which $f(\underline{x}(t), \underline{u}(t))$ describes the plant and vestibular model dynamics. The remaining constraints are associated with the motion platform. The optimal control problem is to determine the input $\underline{u}(t)$ such that the performance index is minimised with the constraints respected.

As shown in Figure 7.1, the control input is a vector of simulator longitudinal and lateral accelerations and yaw velocity $\underline{u}(t) = [a_{sim}^x(t), a_{sim}^y(t), \dot{\psi}_{sim}(t)]^T$. This input and the motion demand $[a_{ref}^x(t), a_{ref}^y(t), \dot{\psi}_{ref}(t)]^T$ are used to compute the system states $\underline{x}(t)$ and the outputs $\underline{y}(t)$ from the system dynamics described in (7.4) and (7.5). The system dynamics relate the reference accelerations and the

control input, through a nonlinear hexapod model and linear vestibular system model, to the actuator accelerations, velocities and displacements as well as the system output $\underline{y}(t)$. The actuator accelerations, velocities and displacements are components of the state vector and are limited in (7.6), (7.7) and (7.8) respectively. The output $\underline{y}(t)$, which contains the error (the difference between the motion sensed in the simulator and the car) is a function of the system state.

$$\underline{y}(t) = \begin{bmatrix} \hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t) \\ \hat{a}_{ref}^y(t) - \hat{a}_{sim}^y(t) \\ \hat{\psi}_{ref}(t) - \hat{\psi}_{sim}(t) \end{bmatrix} \quad (7.9)$$

In the cost function given in (7.3), the diagonal matrix Q allows the relative weighting of the error sensed in each motion direction. The optimal control solver will determine the platform input that minimises the weighted sensed translational acceleration error and the weighted sensed rotational velocity error.

7.2.1 Longitudinal Cost Development

At first, each motion direction is considered separately, and the error terms in the cost function are modified to capture desirable cueing characteristics. In the longitudinal directions the braking and accelerating scenarios need to be cued, with the numerical optimal control used to determine the optimal acceleration profile for each situation. The characteristics of this cueing acceleration are influenced by the choice of error cost function. At first, for consistency with standard cueing approaches, the sensed acceleration is minimised. Thereafter, modifications are made to this to improve the cueing behaviour in accordance with comments from drivers and the experience gained during the testing of cueing filters. The accelerating and braking scenarios are treated separately, with braking being examined first since it is the more critical manoeuvre.

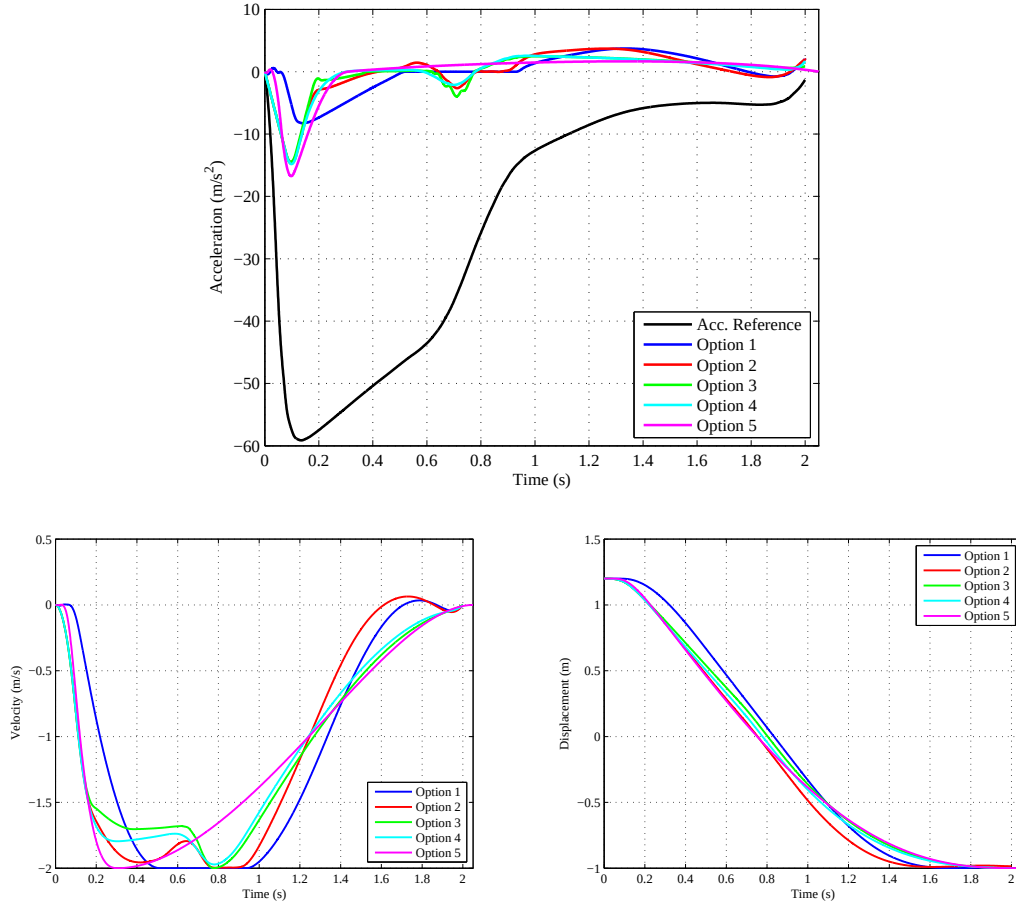


FIGURE 7.2: This collection of figures contains longitudinal cueing signals determined by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands produced for a braking manoeuvre using different cost functions. The bottom two figures contain the platform velocity and displacement demands, respectively.

The plots in Figure 7.2 contain the acceleration profile and the platform velocity and displacement demands for the vehicle braking into Turn 1 of the Barcelona track. The vehicle acceleration reference trajectory is given by the black curve. The blue curve, denoted by Option 1, is the simulator trajectory computed to minimise the square of the acceleration error; a standard time-invariant quadratic cost of the form (7.3) is used. The optimised response is constrained to begin the manoeuvre with the platform at the front of the workspace under the assumption that it has been correctly prepositioned. There is no constraint on

where the platform finishes the manoeuvre although, since an acceleration will follow, it is preferable if the platform is further backwards. Finally, the platform is constrained to begin and finish with 0m/s velocity.

There are a few features of the Option 1 cueing signal that deserve comment. Firstly, the onset cue response is delayed, such that the peak deceleration of the platform coincides with that of the car. Secondly, during the initial 0.1s of braking the platform acceleration fluctuates and moves with the opposite acceleration to the vehicle. This allows the platform to build a positive velocity, which decays when the onset cue begins. The third feature of the cueing signal is the mid-braking loss of motion, which occurs at 0.6s when the platform reaches its maximum velocity. Finally, the miscue that prevents the platform exceeding the workspace limitations occurs in the last phase of braking. This is poor workspace management, as the longer the duration of the miscue the lower the peak magnitude. Overall this response is not satisfactory because it does not provide the strong initial onset cue required by the race car drivers who were consulted. Furthermore, the drivers prefer cueing systems with consistent, lower magnitude miscues. The cueing acceleration resulting from (7.3) does not exhibit these characteristics and thus required modification.

Recalling that the cost function for the optimal control problem, when considering only the longitudinal motion, is given by:

$$J_x = \int_{t_0}^{t_f} \underline{y}_x^T(t) Q \underline{y}_x(t) dt, \quad (7.10)$$

where the subscript x refers to the x-direction, and when considering only sensation error in the cost,

$$y_x(t) = \hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t). \quad (7.11)$$

Additional terms can be included in $y_x(t)$ to improve the cueing characteristic.

The first adjustment is made as follows:

$$\underline{y}_x(t) = \begin{bmatrix} \hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t) \\ (\hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t))\dot{a}_{ref}^x(t) \end{bmatrix}. \quad (7.12)$$

The second element in the vector is the sensation error multiplied by the vehicle jerk signal. This implies that the minimisation of the sensation error is more critical when the vehicle jerk is large. The jerk of the vehicle is largest at the onset of braking, and hence this term encourages a higher fidelity onset cue. The effect of this is seen in Figure 7.2, where the platform acceleration cue is given in the top figure, and denoted by Option 2. There is a stark difference between this and the first option. The onset is earlier, higher in magnitude and shorter in duration. The platform runs into the velocity constraint almost 0.4s earlier than in the previous case. As a result, the mid-braking no motion phase begins earlier. However, the miscueing phase is longer, as the change in acceleration from 0.7-1s also exhibits large jerk and must be cued.

While this cueing signal is an improvement, there are still some issues, one of which is the large miscue in the latter stage of braking from 1-1.6s. This problem is addressed with the introduction into the cost function of a miscue penalty, which penalises simulator accelerations which are in the opposite direction to the vehicle acceleration. This is implemented by including the simulator acceleration in the cost (the third term in (7.14)) and multiplying it by a nonlinear weighting function given by:

$$w(t) = \frac{1}{2} (1 - \tanh(\beta \hat{a}_{ref}^x(t) \hat{a}_{sim}^x(t))). \quad (7.13)$$

The behaviour of $w(t)$ is shown in Figure 7.3. It is a smooth switch that has a value of 0 when the vehicle and simulator accelerations are in the same direction

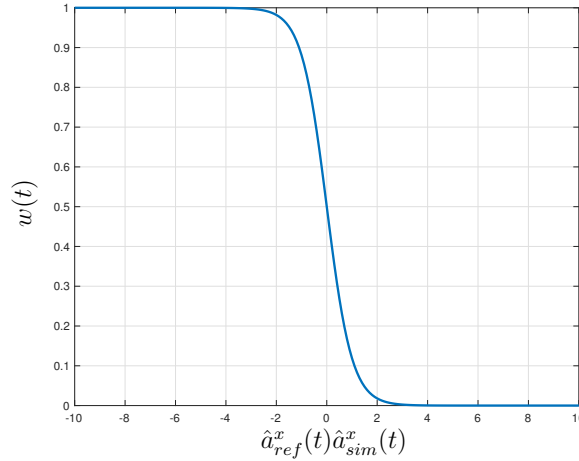


FIGURE 7.3: The plot shows the nonlinear weighting function given in (7.13). The output of the function is 1 when the platform and vehicle accelerations are in opposite directions and 0 when they are in-phase.

and a value of 1 when they are in opposite directions. This introduces a penalty only on false acceleration cues. The cueing signal produced with this additional term is given by Option 3 in Figure 7.2, and exhibits lower magnitude miscues. The final problem with the cueing signal is the lack of smoothness in the acceleration signal. The study in [31] showed that large jerk causes accelerations to be perceived as stronger. During the miscue phase this is particularly undesirable. However, by placing a penalty on the simulator jerk (given by the fourth term in (7.14)), the acceleration demand is made smoother, as shown by Option 4 in the plots in Figure 7.2. This cueing signal meets the requirements of a strong onset cue, and smooth low magnitude false cue while tracking some of the acceleration changes during the latter parts of braking.

The final cost function used to cue the longitudinal braking manoeuvre is given by:

$$\underline{y}_x(t) = \begin{bmatrix} \hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t) \\ (\hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t)) \dot{a}_{ref}^x(t) \\ \hat{a}_{sim}^x(t) \\ \dot{a}_{ref}^x(t) - \dot{a}_{sim}^x(t) \end{bmatrix}, \quad (7.14)$$

and, the weighting matrix is given by

$$Q_x = \begin{bmatrix} Q_1 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 \\ 0 & 0 & w(t)Q_3 & 0 \\ 0 & 0 & 0 & Q_4 \end{bmatrix}, \quad (7.15)$$

where the diagonal elements are used tuned to balance the different components of the cost function.

The cueing characteristic is compared to an alternative strategy which has a time dependent cost function. This approach is designed to produce a high fidelity onset cue and a minimal false cue, accepting that in the latter phase of braking it is not possible to reproduce the vehicle acceleration. The terms in the cost function in (7.10) are defined as follows to induce the desired response characteristics.

$$Q = \begin{bmatrix} Q_1 w(t) & 0 \\ 0 & Q_2(1 - w(t)) \end{bmatrix}, \quad (7.16)$$

and,

$$\underline{y}(t) = \begin{bmatrix} \hat{a}_{error}^x(t) \\ \hat{a}_{sim}^x(t) \end{bmatrix}. \quad (7.17)$$

where

$$\hat{a}_{error}^x(t) = \hat{a}_{ref}^x(t) - \hat{a}_{sim}^x(t) \quad (7.18)$$

and

$$w(t) = \frac{1}{2}(1 - \tanh(\beta(t - t_s))). \quad (7.19)$$

The first term in the cost function penalises the acceleration error during the first part of the braking phase. The second term then penalises the platform acceleration during the remainder of the braking manoeuvre. The weights Q_1 and Q_2 are used to balance the relative strengths of the onset cue and the miscue; not surprisingly, the stronger the onset cue, the worse the miscue. The function $w(t)$ is a ‘smooth’ switch that is 1 from $t = 0$ to $t = t_s$ and 0 thereafter; the constant β controls the sharpness of the switch. The switching time and weights are selected to produce the strongest, and longest onset cue that can be achieved within the actuator constraints. Since braking manoeuvres have similar strengths, the tuning parameters are consistent for a given lap.

The result of this adjusted cost function is shown as Option 5 in Figure 7.2. The acceleration cue contains a strong timely onset cue, while the false cue has been extended, but with a much reduced amplitude. This cueing signal is a baseline, describing the strongest miscue, and the lowest magnitude false cue. However, no acceleration changes are cued after the onset. It can be compared with the Option 4 cueing demand. The two onset cues are similar in magnitude and duration, and the miscue of the Option 4 is only larger for 0.5s. This suggests that the cost function developed to produce the Option 4 is sufficient for further use in optimal control studies, as it captures the desirable cueing characteristics while producing close to optimal onset and false cues.

The cost function for longitudinal cueing has been developed for a braking manoeuvre. The application of this function to the accelerating scenario is examined

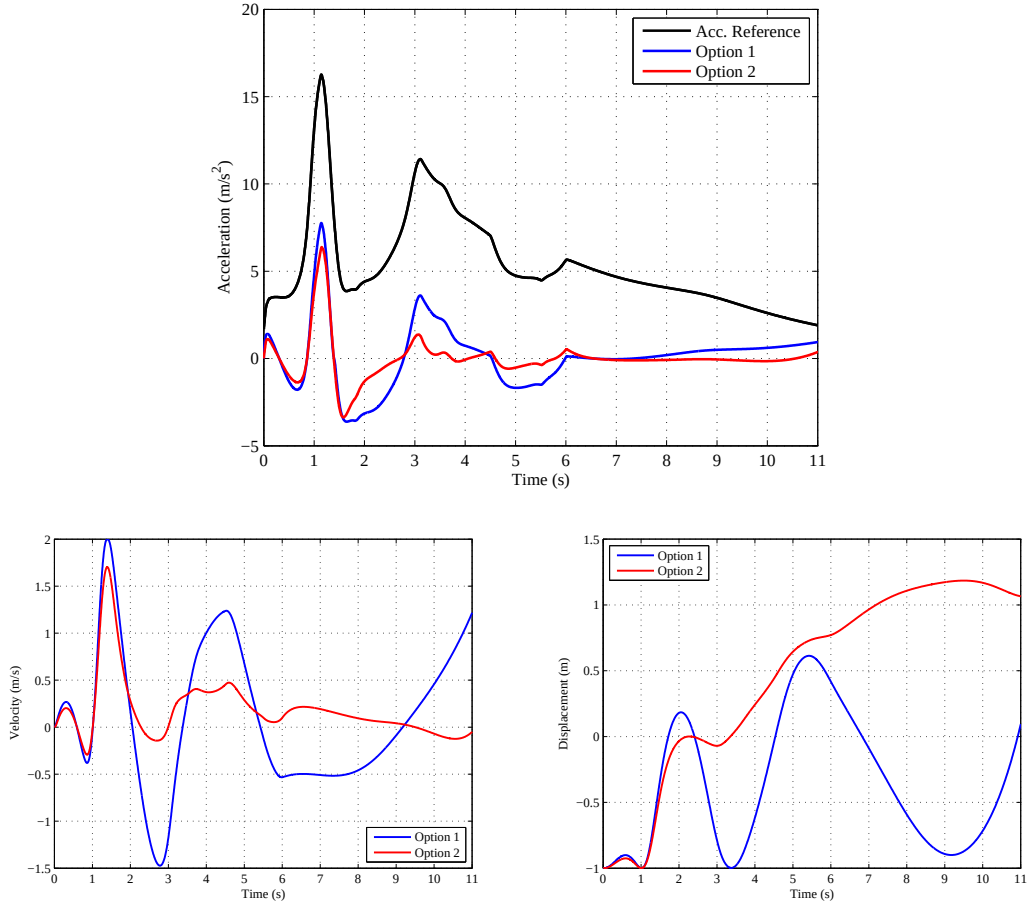


FIGURE 7.4: This set of figures contains longitudinal cueing signals determined for an accelerating manoeuvre by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands, and the bottom two figures contain the platform velocity and displacement demands, respectively.

next by comparing it to the cueing signal produced by the naive perception error cost function. A comparison of the platform demands is shown in Figure 7.4. In this situation the platform is constrained to start at the back of the workspace, and once again, start and finish with 0m/s velocity. The cueing acceleration produced by the naive cost function (denoted by Option 1) retains the general shape of the acceleration reference, but the low-frequency information has been removed. Unlike the braking case, there are no discernible onset cues and so the general approach is to cue acceleration changes. The problem with using a simple

quadratic cost function, is that the velocity of the platform varies between positive and negative values as the acceleration changes are tracked. This causes a lot of forwards and backwards platform motion, particularly at 1.2s and 3s when the throttle is being lifted. These sudden decreases in the platform acceleration are being cued similarly to a braking manoeuvre, which, as it was found in testing, the drivers dislike. The motion cue produced by the cost function in (7.14), denoted by Option 2, has lower false cues and while it does not preserve the shape of the acceleration cue as well, it does not, however, produce large backwards motions. This cost function is therefore suitable for using in both accelerating and braking situations, and can be applied to further optimal control studies.

7.2.2 Lateral Cost Development

The development of the lateral acceleration follows a similar procedure to that of the longitudinal although the requirements are different. The lateral acceleration experienced in a vehicle has a different shape (as compared with the longitudinal) and may be of high amplitude and long duration in a protracted high-speed corner. Unlike the longitudinal motion, the drivers do not have a clear preference for onset cues. Of more importance in this case is the need for no false cues. The lateral velocity of the platform should be in the appropriate direction to maintain the correct perception of the vehicle turning centre.

The terms in the cost function used in the lateral direction are given by:

$$\underline{y}_y(t) = \begin{bmatrix} \hat{a}_{ref}^y(t) - \hat{a}_{sim}^y(t) \\ (\hat{a}_{ref}^y(t) - \hat{a}_{sim}^y(t)) \dot{a}_{ref}^y(t) \\ \dot{a}_{ref}^y(t) - \dot{a}_{sim}^y(t) \\ v_{sim}^y(t) \\ \hat{a}_{sim}^y(t) \end{bmatrix}, \quad (7.20)$$

and, the weighting matrix is given by

$$Q_y = \begin{bmatrix} Q_1 & 0 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 & 0 \\ 0 & 0 & Q_3 & 0 & 0 \\ 0 & 0 & 0 & w_1(t)Q_4 & 0 \\ 0 & 0 & 0 & 0 & w_2(t)Q_5 \end{bmatrix}. \quad (7.21)$$

where,

$$w_1(t) = \frac{1}{2} \left(1 - \tanh(-\beta \hat{a}_{ref}^y(t) v_{sim}^y(t)) \right), \quad (7.22)$$

and

$$w_2(t) = \frac{1}{2} \left(1 - \tanh(-\beta \hat{a}_{ref}^y(t) \hat{a}_{sim}^y(t)) \right). \quad (7.23)$$

The contribution of each of the elements is shown in Figure 7.5. The first cueing signal, given by Option 1, is produced by the optimal control for a cost function containing only the acceleration perception error. The acceleration cue contains a miscue during the first 0.25s. The acceleration magnitude becomes progressively larger through the corner, until, near the end, there is a large negative acceleration to slow the platform to avoid exceeding the workspace limits. The two main problems with this cue lie in the miscues in the initial and final 0.25s.

While onset cueing is not generally considered in lateral filters to the same extent as it is with longitudinal filters, it still occurs. A linear high-pass filter will provide stronger cues at the beginning of a corner, since the rapid change in the acceleration is a high-frequency motion. Consequently, the onset acceleration will be cued, and to recreate this feature in optimal control solution the second element in (7.20) is included. This, like the longitudinal case, is an additional penalty on the acceleration error when the vehicle jerk is large. The effect of this term is seen by examining Option 2 in Figure 7.5. The onset is larger, but the

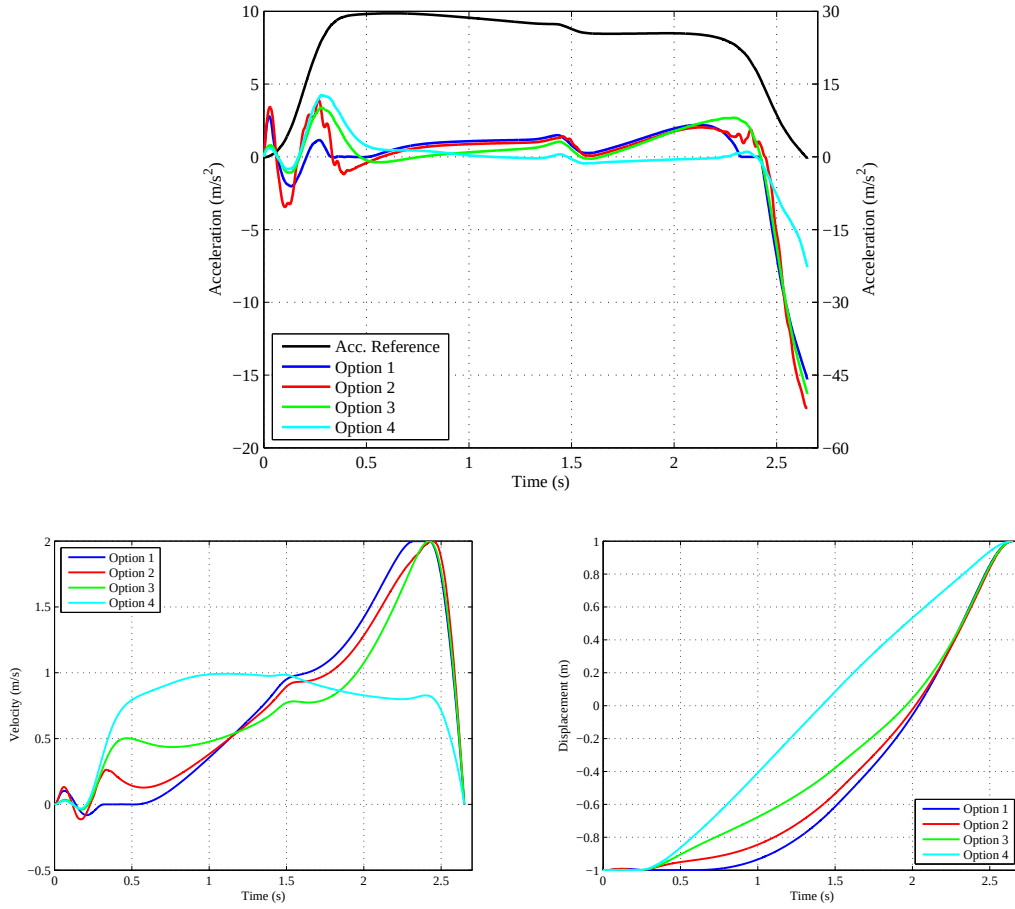


FIGURE 7.5: This collection of figures contains the lateral cueing signals determined by the numerical optimal control calculation for different cost function definitions. The top plot contains the platform acceleration demands produced for a cornering manoeuvre. The bottom two figures contain the associated platform velocity and displacement demands, respectively.

signal still features a large initial miscue, and is not smooth.

These two issues are addressed with the inclusion of the 3rd and 4th elements in the cost function. The first of these places a penalty on the jerk error, as is done in the longitudinal case. The second element is a velocity miscue penalty. The platform velocity is weighted by a nonlinear function, $w_1(t)$ that is 0 when the platform velocity is in the same direction as the vehicle acceleration and 1 when the two signals are out-of-phase. This causes the platform velocity to remain positive throughout the cornering manoeuvre. The improvements caused

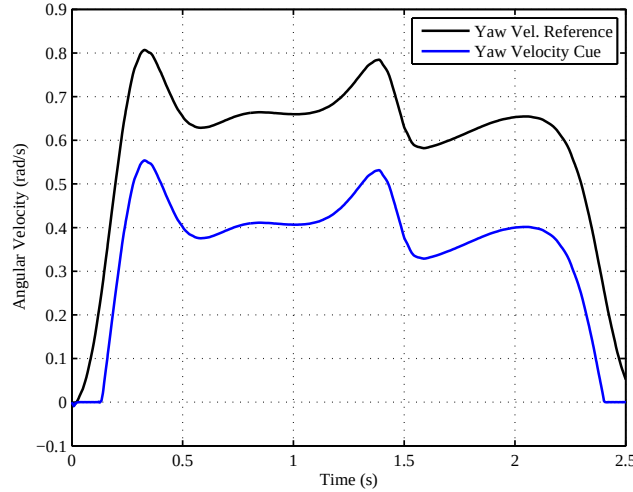


FIGURE 7.6: This figure contains the vehicle yaw velocity, and the platform yaw velocity cue produced by the numerical optimal control computation using a cost function that minimises the angular velocity error.

by these two terms are shown by Option 3 in Figure 7.5.

The final element in (7.20) is an acceleration false cue penalty (as in the longitudinal cost function), which is included to combat the large negative acceleration produced at the end of the corner. This term has a significant impact on the cueing signal, which is seen by examining the platform demands produced by Option 4 in Figure 7.5. The onset cue remains similar to the earlier options. However, the mid-corner cues are much lower in magnitude, and the final miscue also lower. The platform velocity trajectory is also altered. The velocity no longer reaches the maximum value, but tends to maintain a more constant value through the mid-corner. This behaviour is more consistent with the cues produced by the linear filters that were tested with, and approved by the drivers. The cost function in (7.20) is, therefore, suitable for further use in optimal control studies.

7.2.3 Yaw Cost Development

The yaw freedom differs from the previous two in that the platform is able to render much more of the vehicle yaw velocity signal as compared with the longitudinal or lateral acceleration signals. As a result, it is sufficient to use a cost function that simply minimises the sensed yaw velocity error. This is given by:

$$y_\psi(t) = \hat{\psi}_{ref}(t) - \hat{\psi}_{sim}(t). \quad (7.24)$$

The yaw velocity cueing demand produced by the optimal control computation is shown in Figure 7.6. The platform yaw velocity tracks the changes of the vehicle yaw velocity, although the signal magnitude is attenuated.

7.3 Optimal Cueing for a Racing Lap

The cost functions previously developed are now used to determine an optimal simulator cueing strategy for a lap of the Barcelona Circuit de Catalunya. The results focus on corners 1-4, shown in Figure 7.7. This section of the track is chosen because it features high-amplitude braking cues, high-speed corners (3 and 4) and two consecutive right-hand corners (the relevance of which is discussed in Section 7.3.2 on lateral acceleration cueing). In Sections 7.3.1, 7.3.2 and 7.3.3, the longitudinal, lateral and yaw cueing demands are computed separately. In Section 7.3.4 the longitudinal and lateral cues are considered in combination, with three-dimensional cueing assessed in Section 7.3.5. Multi-dimensional cueing is important, because the various degrees of freedom are coupled, and motions in one dimension might compromise those in another.

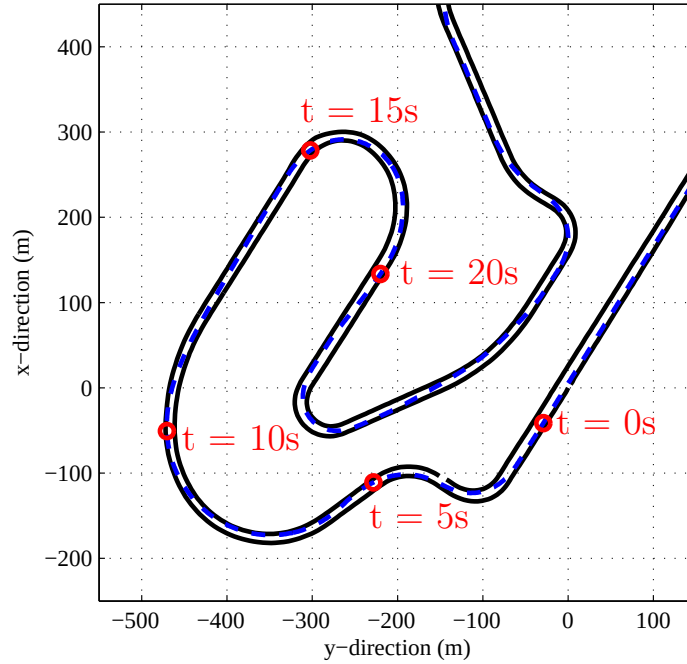


FIGURE 7.7: Track and racing line for the first four corners of the Barcelona circuit used in numerical optimal control simulations [38, 51]. The red markers are spaced 5 seconds apart, with the first being at 0 seconds and the last at 20; the car travels in a clockwise direction.

7.3.1 Longitudinal Cueing

The longitudinal acceleration produced by the numerical optimal control computation for the first 4 corners of Barcelona is given in the left plot of Figure 7.8. The corresponding platform velocity and displacement demands are shown in the right plot. The purpose of the open-loop numerical optimal control solution is to provide insight into the best use of the platform workspace. As can be seen, the braking manoeuvres are characterised by strong onset cues, while during acceleration changes are tracked, although somewhat attenuated. Under braking, the motion cueing is being restricted by the velocity capability of the platform. By examining the velocity and displacement signals it is seen that the velocity achieves its peak value before the platform reaches the displacement limit. It is

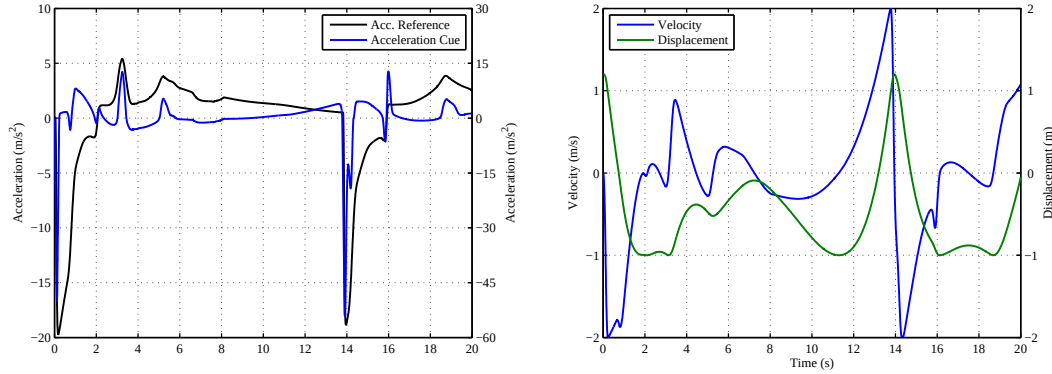


FIGURE 7.8: The left figure shows the longitudinal acceleration cueing signal determined by the numerical optimal control computation for the given longitudinal acceleration reference. The right figure shows the corresponding simulator velocity and position demand.

thus the velocity limit that prevents the onset cue from being sustained.

After a braking or accelerating manoeuvre the platform is prepositioned forwards, or rearwards, of the neutral position in anticipation of the next manoeuvre. Before the second braking manoeuvre from 12-14s the platform accelerates to the maximum velocity. During the onset braking cue at 14s the platform is first braked, to zero velocity at the front of the workspace, and then accelerated backwards. This velocity-type “prepositioning” allows a stronger braking cue to be generated since the platform is being decelerated from maximum forward to maximum rearwards velocity, instead of from 0m/s. This approach makes optimal use of the available platform workspace. In a practical situation this would be very difficult to implement as the timing is critical. If the braking manoeuvre does not occur timeously then the simulator will reach the workspace limits and a harsh false cue would be experienced. However, if the principle is employed, but with a low magnitude velocity magnitude velocity, the workspace of the platform can be better utilised.

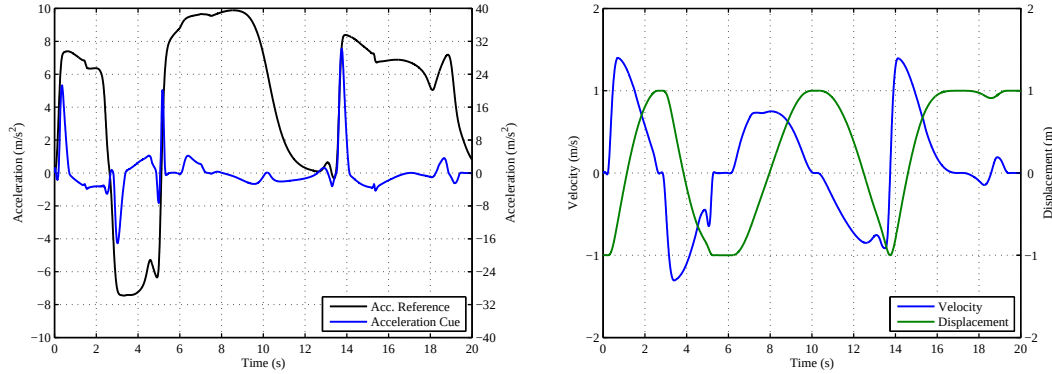


FIGURE 7.9: The left figure shows the lateral acceleration cueing signal determined by the numerical optimal control computation for the given lateral acceleration reference. The right figure shows the corresponding simulator velocity and position demand.

7.3.2 Lateral Cueing

The lateral acceleration cue is determined by the optimal control solver for turns 1-4 of the Barcelona circuit. Unlike cueing in the longitudinal direction, the lateral acceleration trajectory can have two consecutive manoeuvres in the same direction; see Figure 7.9 between 6s and 12s, and between 14s and 20s. In this case it would be best to position the platform away from the same extreme edge of the workspace at the start of each manoeuvre. However, between corners, the platform will then need to move in the opposite direction to the vehicle acceleration and a false cue will result. A balance therefore needs to be established between a miscue and the workspace needed for the following acceleration cue. In this case, the platform is returned to the left extreme from 11s to 13s when the vehicle lateral acceleration is closest to 0m/s^2 .

The lateral acceleration does not run into the velocity constraints as in the longitudinal case. This is a result of lower onset cues, and an acceleration cue that tracks the changes in acceleration.

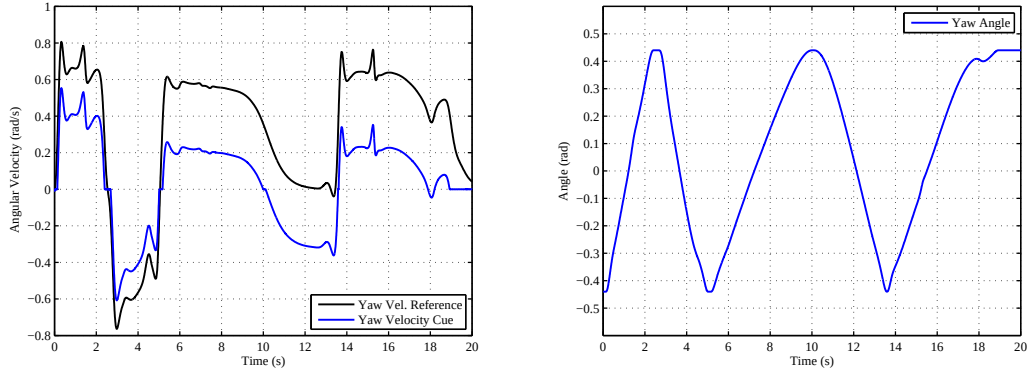


FIGURE 7.10: The left figure shows the yaw velocity reference and cueing signal computed by the numerical optimal control solver. The right figure shows the associated platform rotational position demand.

7.3.3 Yaw Cueing

The yawing motion differs from the longitudinal and lateral accelerations in that the cues can be sustained for longer and in certain circumstances, be almost fully reproduced by the platform, if its position limits are not exceeded. As with the lateral freedom, there can be multiple manoeuvres in the same direction. In this case, a bad miscue is required to preposition the platform: this can be seen from 10s - 14s in Figure 7.10. This is required to move the platform from the one extreme back to the other in preparation for the next corner.

7.3.4 Longitudinal and Lateral Combined Cueing

Having considered each DOF separately, the next step is to study the two translational freedoms in combination. The cost functions and weighting matrices developed for the two freedoms are combined as follows:

$$\underline{y}(t) = \begin{bmatrix} \underline{y}_x(t) \\ \underline{y}_y(t) \end{bmatrix} \quad (7.25)$$

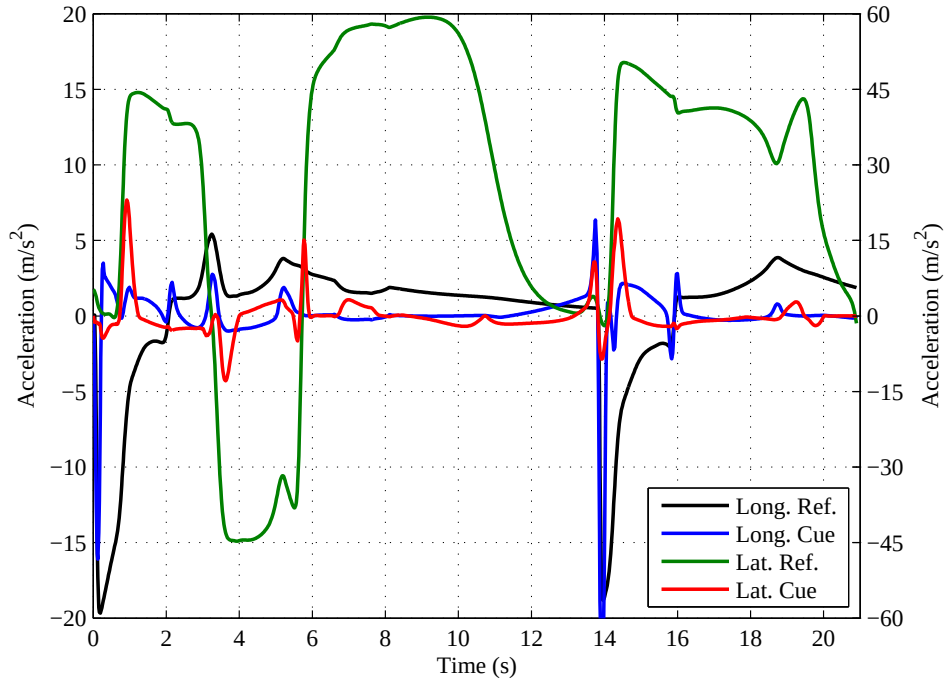


FIGURE 7.11: The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis. The lateral motion is given priority.

and,

$$Q = \begin{bmatrix} Q_x & 0 \\ 0 & R_1 Q_y \end{bmatrix}, \quad (7.26)$$

where, $\underline{y}_x(t)$, $\underline{y}_y(t)$, Q_x and Q_y are defined in (7.14), (7.20), (7.15), and (7.21), respectively. The additional parameter R_1 is used to assign a relative priority to the longitudinal and lateral freedoms. By changing the value of R_1 it is possible to study the interaction between the two motion freedoms.

In the first implementation, the lateral freedom is given precedence over the longitudinal motion by making $R_1 > 1$. The longitudinal and lateral acceleration cues are given in Figure 7.11, and the associated platform velocity and position demands in the plots in Figure 7.12. Since the lateral direction is given priority,

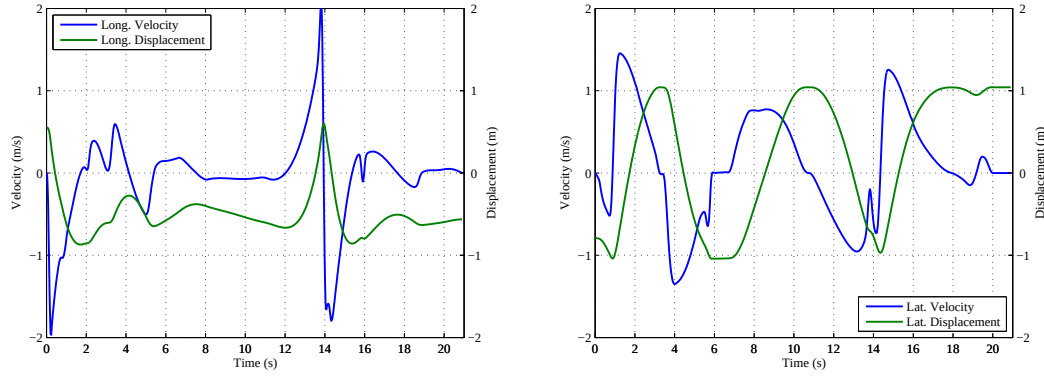


FIGURE 7.12: The simulator velocity and position demands for the longitudinal (left figure) and lateral (right figure) directions. These are associated with the acceleration demands given in Figure 7.11.

the lateral cueing demand is unchanged from the lateral only cueing presented in Section 7.3.2. The longitudinal cueing is necessarily different. Firstly, during the onset cues at 0s and 14s the platform no longer reaches the maximum velocity. Additionally, the miscues that follow the onset are larger in order to washout the platform velocity more quickly. These two changes in the longitudinal behaviour ensures that none of the actuators are moving at their maximum speed, so the platform still has the ability to move laterally.

The cueing demands in Figures 7.13 and 7.14 are produced by making $R_1 < 1$, and assigning priority to the longitudinal direction. In this case, the longitudinal cueing remains unchanged, while the lateral acceleration now features lower onset cues and larger miscues. These miscues are required to move the platform away from the edge of the workspace where the longitudinal motion is more constrained. The workspace usage of the two cueing options is compared by examining the xy-displacement of the platform in Figure 7.15. For the first cueing option the platform operates mainly in the longitudinal range from -0.8m to -0.2m, where the lateral workspace is the widest. In contrast, the second cueing option produces the majority of the platform motion in the longitudinal range from -1m to 0m.

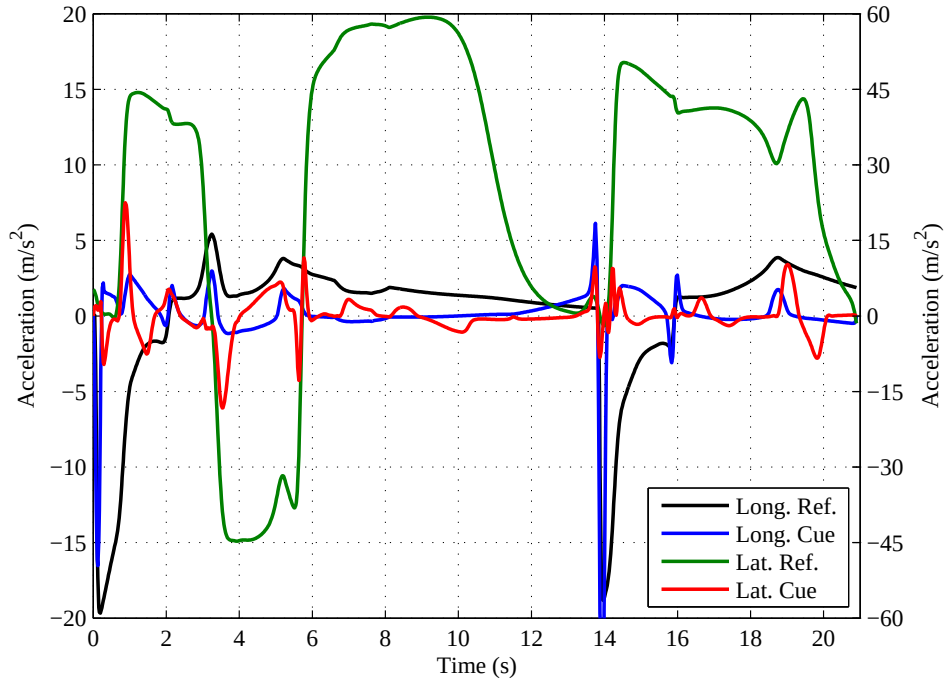


FIGURE 7.13: The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis. The longitudinal motion is given priority.

The increased travel in the longitudinal direction is consistent with the higher priority of this motion freedom.

While there is an interaction between longitudinal and lateral motions, it is possible to select R_1 to achieve a compromise such that the quality of the cue in each direction is only degraded slightly. This is possible since the longitudinal motion is mainly velocity limited whereas the lateral direction is predominantly workspace limited. Furthermore, it is only at the workspace extremes that the two motions interact strongly, so while the platform is operating towards the centre of the workspace there is no coupling.

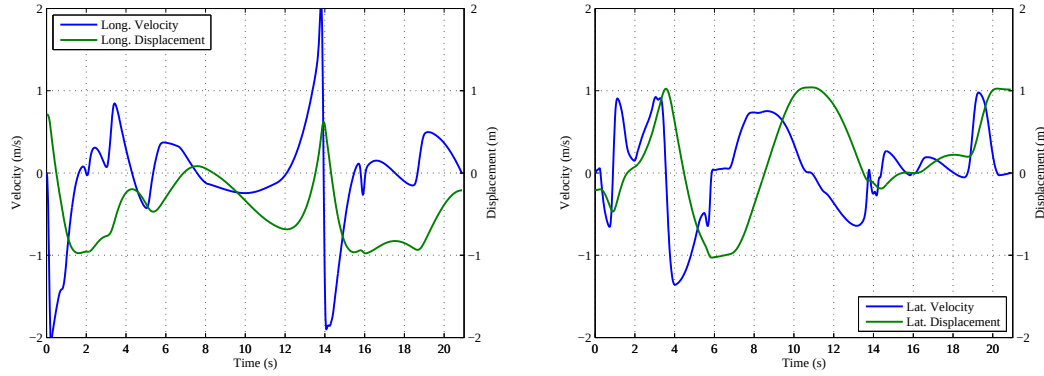


FIGURE 7.14: The simulator velocity and position demands for the longitudinal (left figure) and lateral (right figure) directions. These are associated with the acceleration demands given in Figure 7.13.

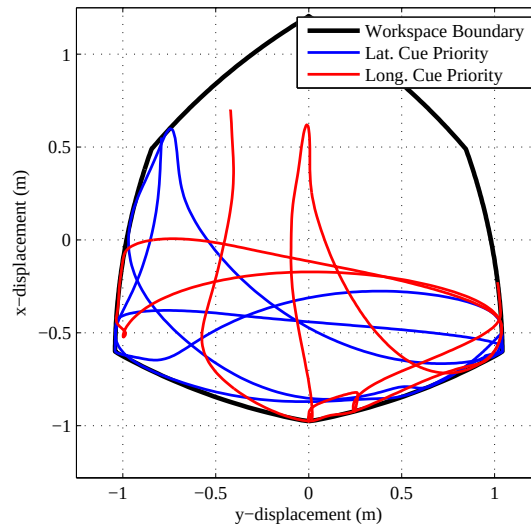


FIGURE 7.15: The platform boundary and the use of the workspace over the four corners for the cueing option associated with the acceleration demands in Figure 7.11 (lateral cue priority) and with those in Figure 7.13 (longitudinal cue priority).

7.3.5 Full 3 DOF Cueing

Adding the yawing freedom to the simulator's kinematics has important implications for the utilisation of the workspace. The hexapod workspace analysis

in Chapter 2 shows that the motion platform's translational workspace is reduced significantly by simultaneous yawing. Once again, it is necessary to assign a weighting to each of the various freedoms so that a balanced performance is achieved. The optimal control problem cost for the full 3 DOF problem is given by:

$$J = \int_{t_0}^{t_f} \underline{y}^T(t) Q \underline{y}(t) dt \quad (7.27)$$

where,

$$\underline{y}(t) = \begin{bmatrix} \underline{y}_x(t) \\ \underline{y}_y(t) \\ y_\psi(t) \end{bmatrix} \quad (7.28)$$

and,

$$Q = \begin{bmatrix} Q_x & 0 & 0 \\ 0 & R_1 Q_y & 0 \\ 0 & 0 & R_2 \end{bmatrix}, \quad (7.29)$$

where, $\underline{y}_x(t)$, $\underline{y}_y(t)$, $y_\psi(t)$, Q_x and Q_y are defined in (7.14), (7.20), (7.24), (7.15), and (7.21), respectively, and R_1 and R_2 provide a relative weight between the motion freedoms.

The optimal control computation is performed twice, in the first instance the yaw motion is given a low precedence, and in the second it is given priority over the translational motions. The lateral and longitudinal cueing demands for the two problems are shown in Figures 7.16 and 7.17. In the first case the cueing demands remain mostly unchanged from the translational motion only case. The onset cues are strong, and the changes in the two accelerations are tracked. The corresponding longitudinal and lateral velocity and position demands are shown in Figure 7.19. The velocity in the longitudinal direction still saturates under braking, while the workspace in the lateral motion also reaches the boundary.

The yaw velocity cue shown in Figure 7.18 is highly attenuated with only the

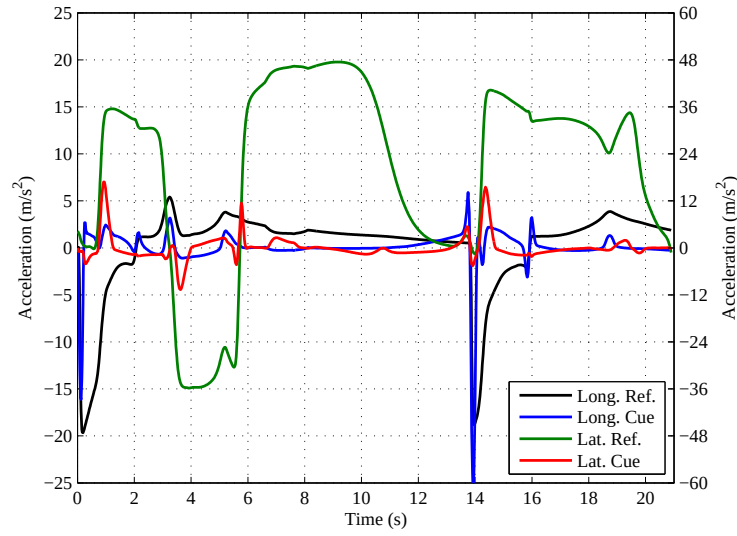


FIGURE 7.16: The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The problem set-up includes all 3 DOF, and the yaw motion is given a low priority. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis.

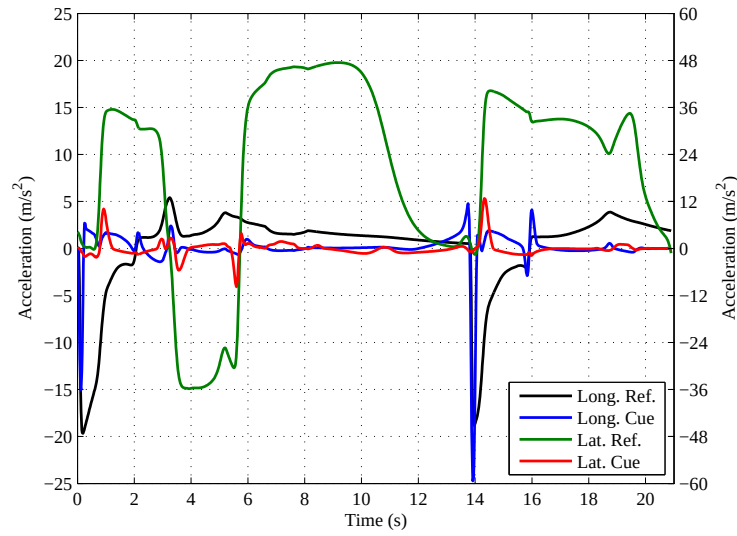


FIGURE 7.17: The longitudinal and lateral platform acceleration demands computed using the numerical optimal control solver for given reference accelerations. The problem formulation includes all 3 DOF, and the yaw motion is given a higher precedence than the translational cues. The reference accelerations are associated with the right axis and the platform acceleration demands with the left axis.

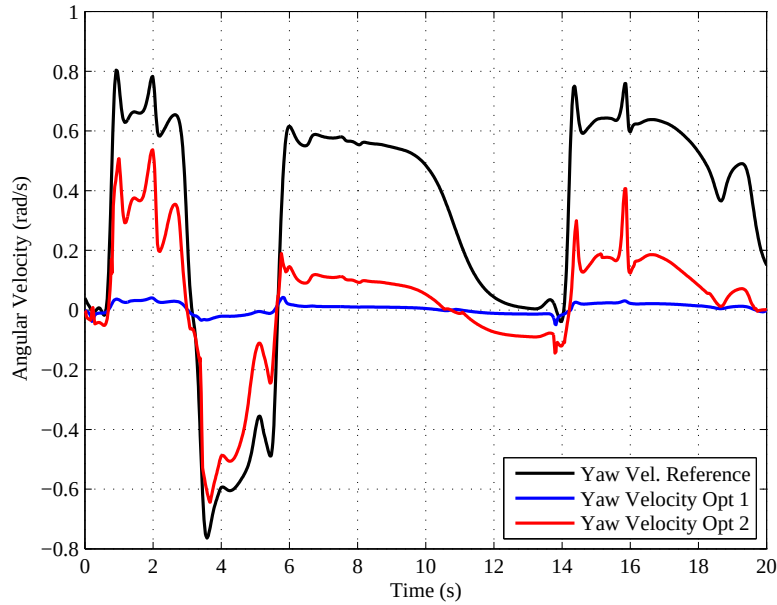


FIGURE 7.18: The vehicle yaw velocity and the simulator yaw velocity cues produced when the yaw motion is given a low (blue) and a high (red) priority in the full 3 DOF motion optimal control computation.

higher frequency changes being tracked. As a result only one-fifth of the yaw workspace is utilised, as shown in Figure 7.19.

In contrast, the second cueing option is characterised by higher magnitude and fidelity yaw cues which make full use of the platform yaw capability, as shown in Figures 7.18 and 7.19. Necessarily, the longitudinal and lateral motion is penalised. The lateral onset cues are reduced while the longitudinal miscues are larger, and fewer higher frequency changes are tracked (Figure 7.17). The velocity and position demands for the translational motions are correspondingly smaller. The longitudinal velocity decays quickly and the platform remains closer to the workspace centre. In the lateral direction the workspace usage is approximately half of that used in the first option.

This analysis highlights the biggest limitation of the hexapod design which is the rapid deterioration of the translational workspace with yaw motion. While in

specification the platform is capable of large yaw angles, in practice the motion must be limited to a few degrees. In cornering there is a combination of large lateral and yaw motion, which it is particularly difficult to render. The longitudinal motion is more easily combined with the lateral or yaw freedoms, as the strongest braking occurs before the vehicle starts turning, so it is possible to cue the longitudinal, and washout the motion before the other directions must be cued.

This analysis illustrates a limitation of the hexapod as a high-performance motor vehicle motion platform and provides a means with which to balance the trade-off between rotational and translational motion.

7.4 Workspace Extensions

Having set-up the simulator cueing problem in an optimal control framework, it is possible to analyse the effects of improving the platform actuator capabilities as well as different platform hardware.

In the development of the longitudinal cueing and the subsequent multi-DOF cueing systems, the velocity limitation of the actuators has interfered significantly with the longitudinal cueing direction. Not only is the onset cue duration limited, but to allow motion in other freedoms, the onset must be quickly washed out. The platform motion is determined for a lap of the Barcelona circuit with the actuator velocity limits doubled. The workspace usage is then compared with the workspace utilisation during a lap with the standard velocity limits. The results are presented in Figure 7.20. The additional velocity capability allows the platform to more fully utilise the workspace. The trajectories of the platform in the longitudinal direction have longer subarcs, and overall the platform travels more. The additional velocity capability, improves the platform cues, allowing

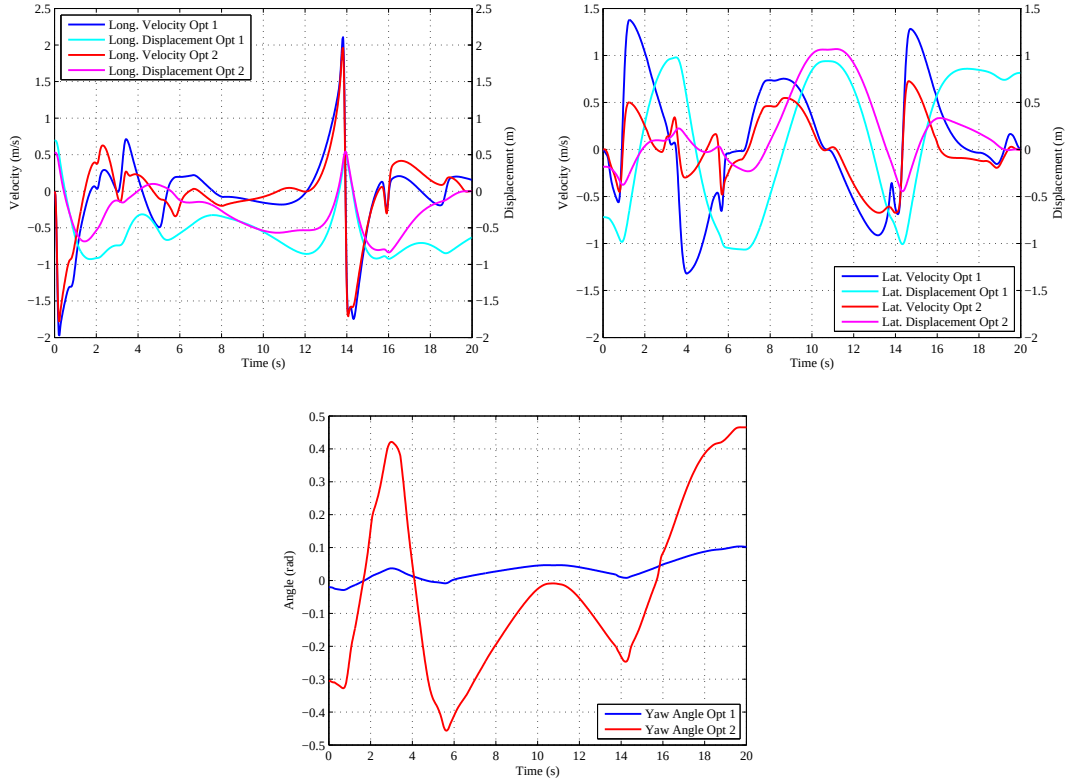


FIGURE 7.19: This set of figures contains the simulator translational velocity and position, and rotational position demands for the full 3 DOF motion problem. The platform motion signals for both the low (Opt 1) and high (Opt 2) yaw priority are given. The top right figure contains the longitudinal velocity and position demands. The top left figure shows the lateral velocity and position demands. The bottom figure shows the two yaw platform position demands.

higher speeds in the longitudinal and lateral directions simultaneously. The platform appears to exceed the workspace boundary but this is due to the change in workspace shape with yaw angle.

The constraints on the platform position are the main limitation in motion cueing, and deciding on the size of platform is an important element in hardware design. Inevitably, a balance is needed between the workspace size and the cost and other practical factors. It is difficult to determine *a priori* the cueing advantages in increasing the platform size. However, in the optimal control framework it is

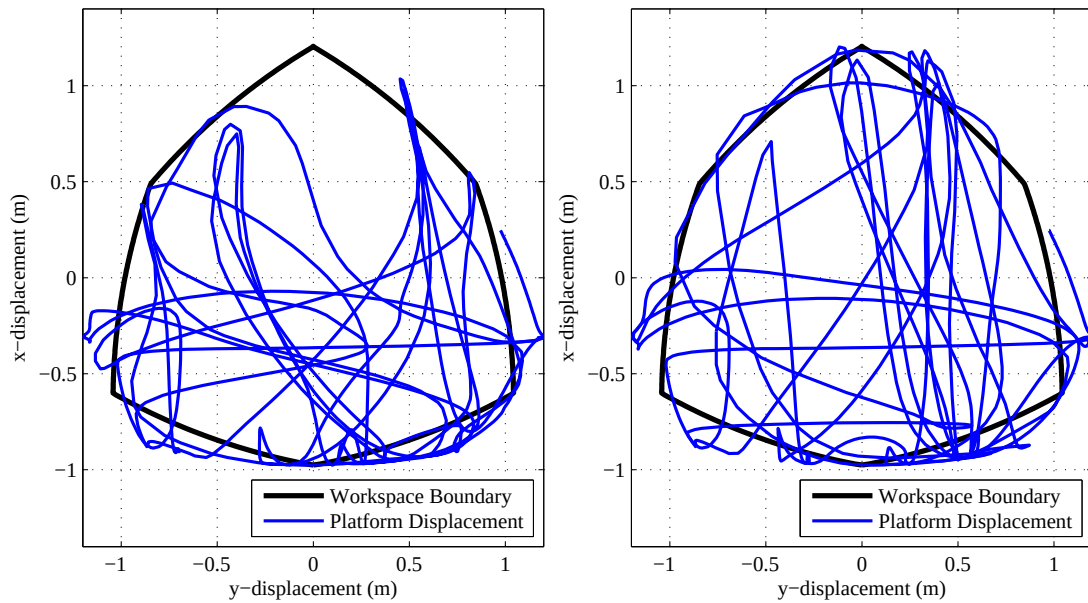


FIGURE 7.20: These two figures describe the simulator workspace utilisation over a lap of the Barcelona circuit, as determined by the optimal control. The left figure shows the workspace use computed using the standard hexapod actuator limits, and the right figure the platform motion using actuators with double the velocity limits.

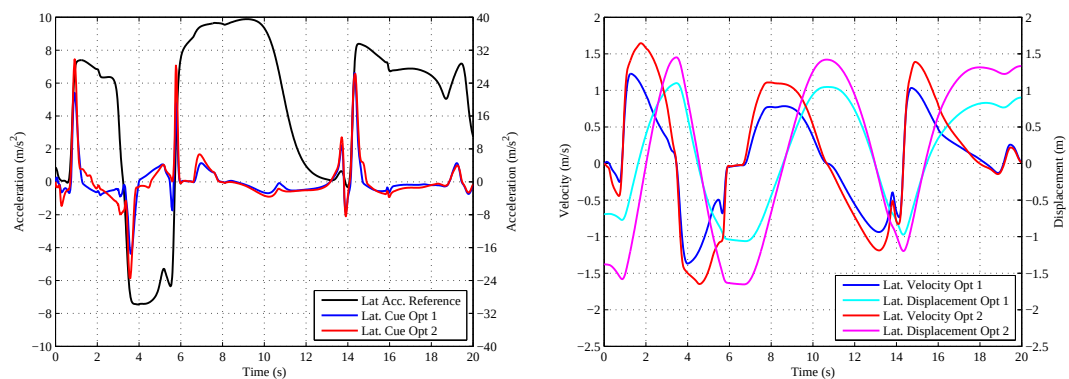


FIGURE 7.21: These figures show the improvement in the lateral cueing signal for an increase in simulator workspace. The left figure contains the lateral acceleration demand computed by the optimal control solver for the standard workspace (Opt 1) and with a 50% increase in workspace size (Opt 2). The right figure displays the associated platform velocity and position demands.

possible to examine the relationship between size and improvement in cueing. In this example, the platform actuator stroke is increased by 45% which increases the workspace by approximately 50%. Since the lateral motion is more heavily influenced by this improvement the platform cues in this direction are examined in Figure 7.21. The improved workspace allows the lateral cues to be rendered with a higher fidelity. The motion is sustained for longer, resulting in fewer miscues. More of the platform velocity capability is used, and overall the cueing motion is strengthened.

These two examples have illustrated the use of developing cost functions that capture good cueing characteristics and applying these to the problem of hardware design. This type of study also highlights the current platform limitations.

7.5 Comparison of Cueing Techniques

The solution of the optimal control computation is compared with the cues produced by the techniques described in Chapter 4, with the intention of determining how these could be improved.

7.5.1 Longitudinal Comparison

The longitudinal cueing signal produced by the optimal control computation is compared with the most successful linear filtering option that was tested with drivers, and the nonlinear miscue reduction method that also showed promise in testing. The platform demands are compared in Figure 7.22.

In the first braking the onset cue of all three options is the same. The linear and nonlinear systems produce miscues that immediately follow the onset, whereas the optimal control only produces a miscue towards the end of the manoeuvre when the acceleration decreases. This is lower in magnitude, although slightly longer than the other two options. In terms of workspace usage, the optimal control

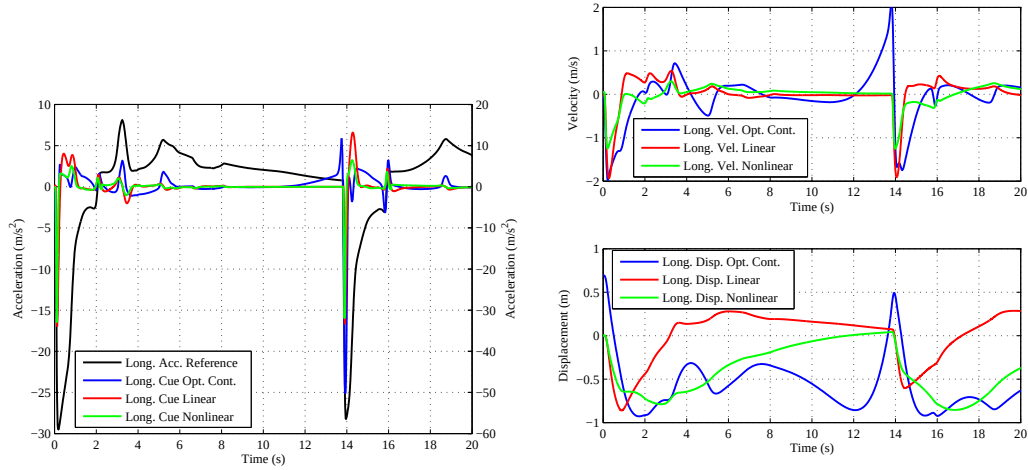


FIGURE 7.22: These figures compare the longitudinal cueing demands produced by the optimal control computation, and the linear and nonlinear filters. The left figure shows the acceleration cues, and the right top and bottom plots the platform velocity and displacement demands, respectively.

makes full use of the platform workspace, whereas the linear and nonlinear filters use approximately half. Additionally, the velocity of the platform in the optimal control case is both larger in magnitude and maintains a higher speed for longer. This comparison suggests that the two implemented cueing methods do not fully utilise the platform capability. The onset cues could be longer, and the miscues lower in magnitude.

Under acceleration all three options track the acceleration changes, although the cueing characteristics differ. In this manoeuvre none of the cueing methods makes use of the full velocity and displacement workspaces. The only significant difference is that the optimal control system prepositions the platform towards the front at the end of the manoeuvre, while the linear and nonlinear filters centralise the platform. It is evident that prepositioning is key to utilising the full workspace. The optimal control also gives the platform a large forward velocity just before braking, as mentioned previously, this strategy is in theory good, but in practise difficult to implement. The advantage of this is seen in the magnitude

of the following onset braking cue.

In summary, there are three areas where the tested cueing methods are sub-optimal. Firstly, the platform is not prepositioned for large braking. Secondly, the onset cues are not maintained, and finally, the false cues are unnecessarily large.

7.5.2 Lateral Comparison

The lateral linear filter implemented and tested successfully with the driver is compared with the optimal control solution in Figure 7.23.

It is immediately obvious that the platform motion cues produced by the optimal control solver are very different from those produced by the linear filter. The onset cues in the optimal control approach are over 6 times larger in magnitude, more acceleration changes are tracked through the corner, and the miscues are also bigger. The linear filter method is characterised by an almost constant platform velocity through the corner which provides the driver with a consistent cornering cue. Whereas, in the optimal control approach the platform velocity tends to decay through the corner.

Setting aside the differences in cueing characteristics, it is possible to compare the workspace usage. The linear filter method does not use the full platform workspace: in a short corner only 12.5% of the workspace is used, and in a longer corner 25% of the available workspace. This contrasts with the optimal control that moves from one extreme to the other in the course of a corner. When there are two consecutive same direction corners the optimal control prepositions the platform towards the end of the first to provide sufficient space for the following corner.

This comparison shows that the linear filter could be modified to provide stronger cues and make better use of the workspace.

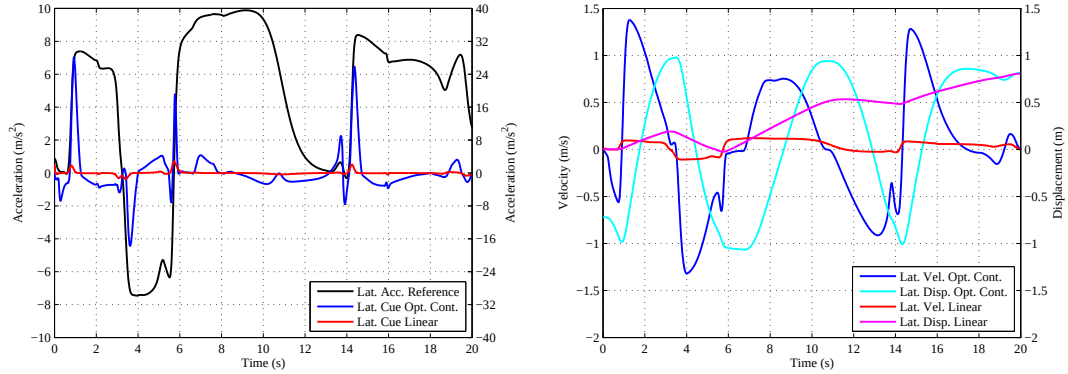


FIGURE 7.23: These figures compare the lateral cueing demands produced by the optimal control computation, and the linear filtering method. The left figure shows the acceleration cues, and the right figure contains the platform velocity and displacement demands for both cueing options.

7.5.3 Yaw Comparison

The yaw cues produced by the optimal control are also compared with those produce by the linear filtering method. From the comparison of the yaw velocity in Figure 7.24 it is seen that the characteristics of the two cueing signals are similar. Both methods cue the onset and then retain the higher frequency velocity changes. The optimal control approach however is, once again, stronger in magnitude. During the middle phase of a corner the linear filter does not move the platform, whereas the optimal control approach does. This uses more of the workspace and provides a longer cue throughout.

Similarly to the lateral case, the yaw linear filter appears to be very conservative in the workspace usage. While these filters have proved successful in the implementation with the driver, this analysis suggests both yaw and lateral cues could be larger, and the hexapod workspace be more fully exploited.

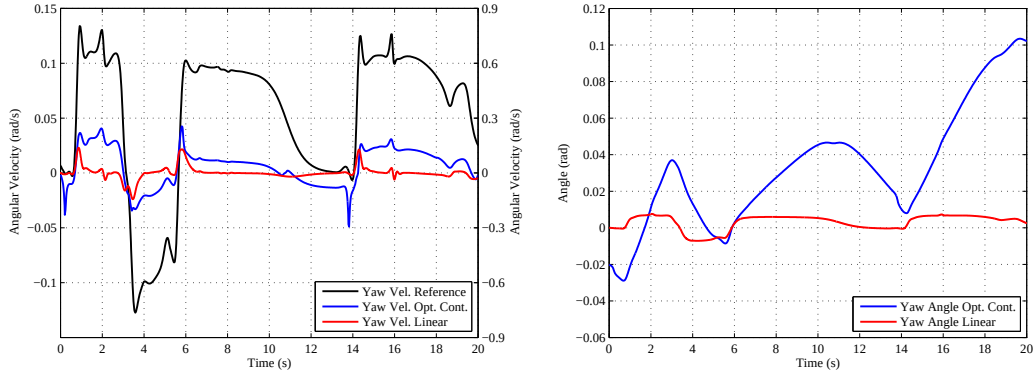


FIGURE 7.24: These figures compare the yaw cueing demands produced by the optimal control computation, and the linear filtering approach. The left figure shows the yaw velocity cues, and the right figure the associated platform yaw angle.

7.6 Summary

The use of numerical optimal control to determine optimal platform trajectories for race-car motion cueing provides insight into platform usage, and can be used to inform platform design, as well as evaluate existing cueing methods.

The first aspect of this work was to formulate the cueing problem in an optimal control framework such that the solution can be computed using the software package GPOPS-II. The second stage of the study involved developing cost functions for the longitudinal, lateral and yaw directions that capture the drivers' cueing preferences. This is necessary to ensure that the optimal control computation produces cueing demands that not only utilised the workspace optimally but also have realistic cueing behaviours. This element of the work was based on the practical experience and feedback gained during the testing with drivers.

Having developed performance indices, the optimal control problem setup could then be used to determine the platform cues for a lap of the Barcelona circuit.

At first each of the DOFs is considered separately, and thereafter in combination. This development provides insight into the interaction between the different motions when rendered using a hexapod platform. When combining the lateral and longitudinal motions, additional miscues were required to ensure that the workspace saturation (either velocity or position) did not compromise the platforms capability in the other direction. The addition of the yaw motion highlighted the limitation of a hexapod, which is the rapid deterioration of translational motion with angular displacement. With relative weighting parameters between the different motion freedoms it was possible to balance the cues and miscues in each direction.

The advantage of using the optimal control framework is the ease with which the platform kinematic model, as well as actuator constraints can be adjusted. Analysis of the cueing improvements that can be achieved by changing the platform limitations can then be conducted. The velocity constraint of the hexapod is the limiting factor in the longitudinal direction and by increasing the maximum actuator velocity the platform workspace is better exploited and the longitudinal cues more accurately rendered. In the lateral motion the workspace is the constraint and by increasing the actuator stroke, the cueing improvements can be visualised. Finally, the solution produced by the optimal control can be used as a baseline to which other cueing methods can be compared. This approach does not compete with the standard cueing algorithms and is not a real-time, online strategy. Rather, it generates a set of optimal reference cues that describe the upper limit of the platform performance. Other cueing strategies can be compared to this, and their limitations highlighted. This can also be used to determine whether these methods have room for improvement. The longitudinal linear and non-linear filters both produce larger miscues than needed and do not exploit the platform velocity and displacement capability. Both the yaw and lateral linear filters use

only a fraction of the workspace and could be retuned or designed to provide stronger cues.

Chapter 8

Conclusion

This thesis discussed the problem of motion cueing in race-car simulator applications. In this context, large magnitude accelerations and fast dynamics pose challenging hardware requirements. Furthermore, there is a high need to exploit the full potential of the simulator platform. It is critical to determine the important motion cues and algorithms to recreate these within the hardware constraints. The problem of fully utilising the hardware to render motion cues with specific characteristics is addressed in a nonlinear optimal control framework which can optimise the platform motion demands over specific race tracks.

Race-car motion features both large magnitude and high frequency motion, and reproducing both of these in a simulator is a difficult task. The simulator systems analysed were divided into two classes: passenger car and vehicle dynamic platforms. The first class contains a number of simulators with large workspaces, which is ideal for sustaining large magnitude accelerations. However, the mass of the moving platforms is often too large to allow any fast dynamics. Systems of the second class are capable of rendering higher frequencies, but typically have only small workspaces. While no simulator system meets all the requirements, a compromise is made in the form of a hexapod. The system used in this work has low

latency, can accelerate at more than 1G and allows a displacement of over 1m in lateral and longitudinal motions. There are many drawbacks of using a hexapod in driving applications. Primarily, the hexapod workspace is isotropic, whereas a vehicle has 3 large motion freedoms and 3 small motion, higher frequency freedoms. However, the limitations are out-weighed by the practical consideration that hexapods are commonly available and well-established in vehicle simulation. Existing motion cueing algorithms mostly originated in flight simulators and have migrated to passenger car simulators. The majority of these algorithms are based on two ideas. The first is that the vehicle acceleration signals are high-pass filtered to produce motion demands that adhere to the platform constraints. The second principle is that the loss in low frequency motion in the longitudinal and lateral directions is compensated for by tilting the platform in pitch and roll respectively. The tilt co-ordination technique is used in the majority of simulators with great success. Motion cueing algorithms have advanced so that the cueing filters, both the high-pass and low-pass cross-term filters, are designed to minimise both perception error and platform motion. This problem is solved using LQR theory, or more recently, MPC. As they stand, these algorithms are not applicable to race-cars. Tilt co-ordination relies on slow dynamics, and perception models do not assist in the filter design since the difference between acceleration error and perception error is insignificant. Alternative approaches are needed to determine the motion cueing requirements in race-car simulators.

By examining the changes in behaviour as the vehicle becomes progressively less stable, the important motion cues are highlighted. Since test drivers use the simulator to analyse the vehicle dynamics and handling, cueing this behaviour is necessary to communicate the information to the driver. Stability analysis across all vehicle operating conditions is conducted using eigenvalue analysis. Under acceleration the system changes with time. However, as the system changes

slowly, it is shown that stability can still be inferred from the location of the system poles. This simple method is easily applied to high fidelity vehicle models and a variety of non-steady-state braking and cornering scenarios.

The results of the stability analysis and preliminary conversations with the drivers are used to modify and extend the basic cueing algorithms for the racing context. The LQR theory filter design approach is reformulated in a generalised regulator framework, and frequency weighting on error and platform motion is introduced. Different parameter tunings yield numerous longitudinal filters, which are then tested with the drivers. Many of these are found to perform well in certain areas (onset cueing) but poorly in others (miscueing). Nonlinear filtering methods are used to improve the miscue behaviour, and while these overall yield better results, the drawback of nonlinear approaches is inconsistency throughout a circuit.

The lateral filters are tuned to preserve the response of the vehicle as it becomes unstable. The testing with the driver produces a positive response, with the oversteer being easy to recognise and control. The yaw filter is compared with a yaw cue based on the side-slip angle, both of which the driver likes and finds assist in the vehicle control. The approval with which both the lateral and yaw methods are met supports the notion that filters tuned to retain vehicle behavioural information more quickly yield good results.

The drawback of the linear and nonlinear filtering approaches is that they do not exploit the workspace and they do not recognise the workspace constraints, often leading to a conservative workspace usage. The latter problem is overcome with the use of MPC based cueing techniques. This method includes the actuator limits as hard constraints. By including the platform velocity and acceleration states in the cost function, it is possible to tune the final cueing characteristic. The real-time limitations and complexity of integrating this with the existing system means driver testing is not possible, but the comparison with the linear

filter cues suggests the method would be successful.

While MPC observes the platform limits it still does not address the problem of optimal workspace usage. Optimality is always measured with respect to a performance index, and in motion cueing it is necessary to develop these cost functions. The definition of these is critical to producing good cueing characteristics that reproduce the vehicle behaviour that the drivers desire.

Within an optimal control framework cueing costs are designed which are then applied to a lap of the Barcelona circuit. The optimal control computation produces cueing demands that simultaneously, optimise the workspace usage, and exhibit cueing behaviours for which drivers express a preference. This problem set-up enables the study of motion platform extension, such as actuator velocity or stroke length improvements. It also produces a base-line to which the implemented methods can be compared to determine in which areas these should be improved.

The contributions of this thesis lie in the development of cueing methods for race-car simulators. Race-car applications are a largely undocumented field of research, and identifying the difference in cueing requirements and developing algorithms to meet these has required a change in mind-set from the passenger car cueing approaches. The hardware limitations are the most significant constraints in the problem and the cueing methods need to exploit completely the platform capability. Finally, the contribution of the optimal control approach is that it is not only a means to optimise the use of existing platforms, but also a framework within which new simulator hardware systems can be evaluated. Studies that balance the increase in cost and size against the cueing improvement can be conducted. Furthermore, this approach is appropriate to investigate higher DOF platforms that require the cueing motion to be split between multiple mechanisms.

Chapter 9

Future Work

9.1 Further Experimental Testing

A limitation of working with race drivers and race-car simulators is their availability for experimentation. Simulator time is at a premium, and particularly when working with highly-paid test drivers it is difficult to schedule many cueing tests. The result is that there is always scope for more experiments to be conducted.

Firstly, in the tuning of motion cueing algorithms there are often a large number of parameters which affect the final filter result. In the development it would be ideal to spend a lot of time making small adjustments to these, testing these changes with the drivers and slowly improving the cueing algorithm. This iterative process is required to fine tune an algorithm and ensure that it is suitable for use on all tracks with all the drivers. While a number of the algorithms proved successful with the drivers, more time could be spent making small adjustments and improving them further.

From the comparison of the optimal control solution and the linear filter results it was clear that the platform workspace was not optimally exploited with the

implemented filters. While these filters were acceptable to the driver, with re-tuning, a less conservative workspace usage could be achieved, as well as stronger motion cues. Further testing with the drivers would need to be conducted. In addition, tuning filters to closely match the optimal control solution would assist in the validation of the cost functions used in the optimal control. Further experimental verification of the optimal control method could be made by reducing the workspace limits, tuning the cueing filters to conform to the optimal control use of this workspace, and testing these with the driver. This could be compared with the results of using the full workspace to establish a subjective driver view of the improvement in cueing. This could be correlated with the reduction in the cost function. This could help to estimate the advantages in increasing the size of the workspace.

A second type of cueing test relates to the driver perceptions and motion sensations. Many studies have been conducted with non-professional drivers to determine the thresholds of motion perception. Test drivers have more attuned senses, and as such a reduction in these thresholds is expected. However, establishing numerical values for these would help with determining acceptable false cues, and prepositioning the driver.

Additional perception tests to determine the relationship between the strength of yaw and lateral cues would be useful in understanding more of the interaction between these motions. In the simulator drivers have commented that they do not like the lateral motion to be too strong. This is believed to be because small yaw cues are used, and a mismatch in the strength of these is worse than lower magnitude cues in both directions. Tests to confirm this theory would be useful. Furthermore, testing with a strong yaw cue and reduced lateral would be helpful to improve the understanding of this link. Many newer simulators have a turntable to allow full yaw motion. However, if there are lateral constraints this

may prevent the yaw capability being used completely.

Finally, testing the MPC cueing strategy with race-drivers is key to future cueing development. MPC algorithms are more becoming more important with higher DOF platforms. This method uses one control algorithm to determine both the cueing and the optimal motion splitting between mechanisms. A challenge in MPC is computation speed, and particularly in racing, where low latency cues are required, this is a critical issue. Furthermore, as the platform models become more complicated, and multiple DOF are computed simultaneously the number of states increases, making the problem larger. Through implementation and testing with a simulator system, the size of the problem that can be solved in real-time will be determined.

Experimental validation of cueing methods is a lengthy process and often fraught with practical problems. However, it is a necessary step and tests that verify the offline optimal control methods are valuable. The optimal control methods aim to assist the cueing tuning process and thereby ultimately reduce the amount of driver testing time.

9.2 Platform Design Optimisation

Motion cueing research is usually conducted from the perspective of producing motion cues to fit within the limitations of a given platform. The optimal control framework allows this process to be reversed, and a platform designed to be capable of rendering the desired motion cues.

Much of the focus of this thesis has been on determining the important motion cues in racing and examining cueing strategies to reproduce these. These ideas have been combined with an optimal control formulation that allows the optimisation of a platform workspace usage. The natural extension of this work is to determine the benefits of different platform kinematic designs and sizing. There

are two potential courses of investigation. One option would be studying the change in the optimal control computation cost with different platforms, and examining how the cost decreases as the platform size increases. This could be used to balance the physical cost and complexity with the cueing fidelity.

A second course of investigation would be to determine the motion cueing quality required, and use the optimal control solver to determine the minimum hardware requirements to meet these. This can be used to determine the required size and power capabilities of different kinematic designs.

These different studies could be combined to provide a thorough analysis of the potential of a new platform. Combined with an understanding of the driver requirements, it would be possible to design a simulator to specifically meet the needs of the race-car problem.

9.3 Passenger Car Applications

While much of this thesis has been concerned with the motion cueing requirements in race-cars and developing cueing algorithms for this application, the methods used here could also be applied to passenger car simulators. In particular, the offline optimal control method has considerable potential in passenger cars.

The extension of this work to the passenger car problem has the advantage that there are many published studies on the factors influencing human perception in passenger vehicles. For example, from experiments much is known about the relative strength of yaw and lateral motion cues [23, 52]. A lot of research has also been conducted on the determination of motion perception thresholds while performing driving tasks. The results of these studies can be included in the optimal control to develop more appropriate cost functions, as well as to introduce new constraints.

In passenger car simulators many of the systems are capable of reproducing a high percentage of the sensed motion. This implies that false cues due to re-centring and prepositioning have the potential to be below threshold detection levels. Constraints to enforce this behaviour can be added to the problem formulation.

Much research has been conducted in the field of driver modelling. This work is relevant in passenger car simulators because the aim is to stimulate the driver so that the driving behaviour in the simulator is the same as on the road. Driver models can be used similarly to vestibular models, to determine the motion cues that will minimise the change in driving behaviour. Once again, this investigation fits within the framework of the optimal control solver. The use of driver models in race car cueing is limited, as the test drivers are sufficiently experienced that they can, after a short adaptation period, reproduce nearly the same driving behaviour with different cueing systems.

The strength of the optimal control based cueing approach is its flexibility in terms of the inclusion of different nonlinear driver and platform models as well as nonlinear cost function definitions. This framework can be used to study many different driving scenarios and determine the optimal cues that fully exploit the kinematic and dynamic capability of a platform.

Appendix A

Vehicle and Tyre Parameters

The values of the vehicle kinematic parameters used in the vehicle modelling for stability analysis are given in Table A.1.

TABLE A.1: Formula 1 car vehicle parameters.

Symbol	Description	Value
m	Vehicle mass	720 kg
J	Moment of inertia about the z-axis	450 kg m ²
l	Vehicle length	3.4 m
a	Distance of mass centre from front axle	1.8 m
b	Distance of mass centre from rear axle	1.6 m
h	Height of mass centre	0.35 m

The non-linear tyre model used in the this thesis is described in detail in [51], and the parameters used in this model are given in Table A.2. The left plot of Figure A.1 contains the longitudinal tyre forces as a function of the longitudinal slip angle for different normal loads and assuming a zero side-slip angle. Similarly, the right plot shows the lateral tyre forces as a function of side-slip angle for different normal loads and assuming no longitudinal slip.

TABLE A.2: Tyre model parameters.

Symbol	Description	Value
F_{z1}	Reference normal load 1	2kN
F_{z2}	Reference normal load 2	6kN
μ_{x1}	Peak longitudinal friction co-efficient at load 1	1.75
μ_{x2}	Peak longitudinal friction co-efficient at load 2	1.40
μ_{y1}	Peak lateral friction co-efficient at load 1	1.80
μ_{y2}	Peak lateral friction co-efficient at load 2	1.45
κ_1	Slip co-efficient at peak friction at load 1	0.11
κ_2	Slip co-efficient at peak friction at load 2	0.10
α_1	Slip angle at peak friction at load 1	9°
α_2	Slip angle at peak friction at load 2	8°
Q_x	Longitudinal shape factor	1.9
Q_y	Lateral shape factor	1.9

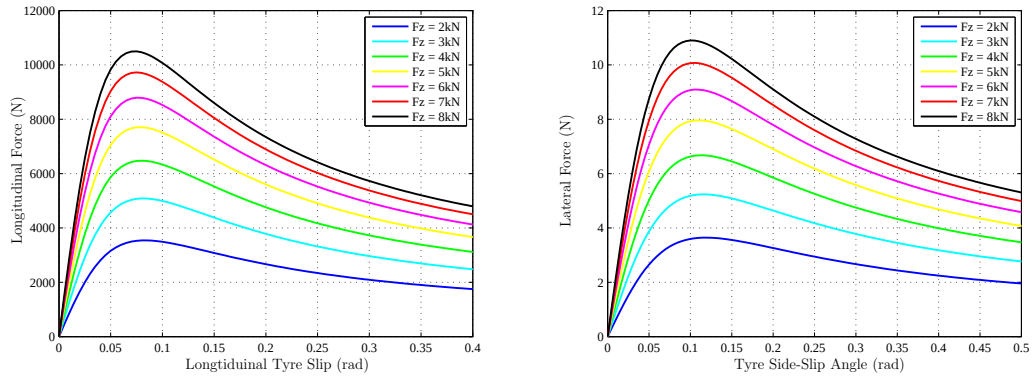


FIGURE A.1: The right figure contains the longitudinal tyre force for different normal loads and zero side-slip angle. The left figure contains the lateral tyre force for different normal loads and zero longitudinal tyre slip.

Appendix B

4-Wheeled Vehicle Model

The stability of a 4-wheeled vehicle model is studied, with the results discussed in Chapter 3. The details of the vehicle model which is built in the multibody modelling package AUTOSIM [40] are presented here. This software tool computes the symbolic equations of motion for the vehicle and generates FORTRAN code that solves these. The resulting program is used for non-linear simulations. In addition, AUTOSIM generates a linearised model code, which can be used in MATLAB to study system stability.

A vehicle model containing a rigid body chassis, double wishbone suspension and 'magic' formula tyres is used, and in addition to the tyre forces, aerodynamic, steering and driving effects are included in the model.

B.1 Vehicle body

A schematic of the vehicle with the relevant parameters is given in Figure B.1. The conventional body-fixed SAE reference frame is used, with the x-axis being aligned with the longitudinal (forward) direction of the vehicle, the y-axis in the lateral (right side-ways) direction and the z-axis vertical (downwards) direction. The vehicle body has length l , an inertia $J_{body} = [J_x, J_y, J_z]^T$ and a

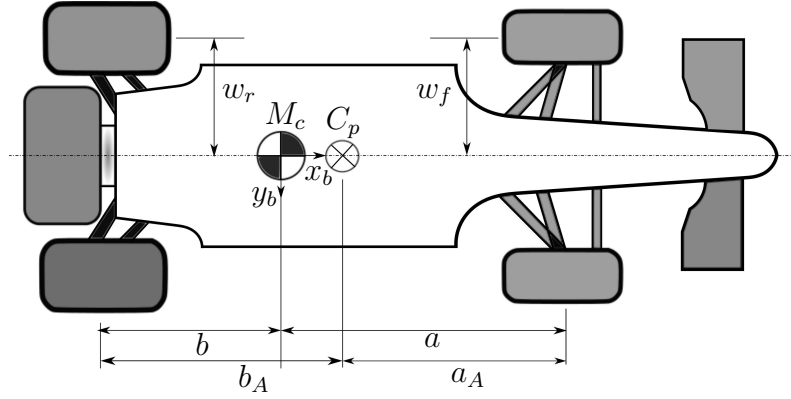


FIGURE B.1: Plan view of a Formula One car with some of its basic geometric parameters.

mass, m_{body} located at a distance b from the rear of the vehicle, and at a height h above the ground. The massless double wishbone suspension connects the wheels to the rigid-body chassis. Each wheel has mass m_{RR} , where the subscript refers to the wheel's location: right-rear (RR), left-front (LF) etc. The front wheels are steered and thus have an inertia (J_{FR} and J_{FL}) about their vertical axes.

B.2 Suspension System

The double wishbone suspension is modelled as a parallel spring-damper pair, shown in Figure B.2. This arrangement produces a torque that acts on the main chassis about the body-fixed x-axis, and reacts on the lower wishbone. Anti-roll bars are included in the model, and generate a torque that opposes vehicle roll and is proportional to the difference between the angles the left and right lower wishbones make with the vehicle body. The stiffness of the front and rear anti-roll bars can be adjusted to set the overall roll balance of the vehicle. The equation for the left-front suspension moment is given in Equation B.1; the other three are calculated equivalently.

$$\tau_{sus} = -D_{sus}\omega_{LWB} - K_{sus}q_{LWB} - K_{ARF}(q_{RWB} - q_{LWB}) \quad (B.1)$$

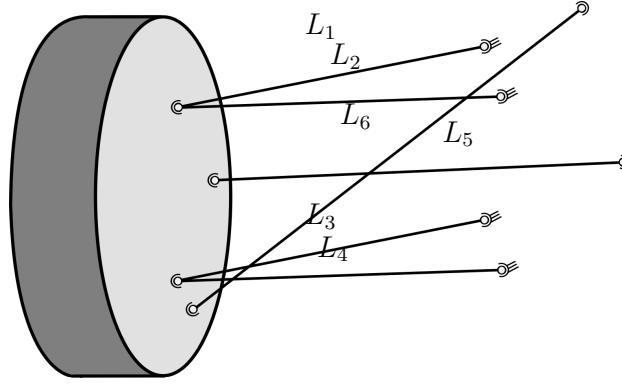


FIGURE B.2: Suspension system of a Formula 1 car. L_1 through L_4 represent the double-wishbones; L_5 is the steering push rod, and L_6 is the suspension push rod to which is connected the spring-damper pair.

where, D_{sus} and K_{sus} are the suspension damping and spring stiffness; q_{LWB} and w_{LWB} are the angle and rotational speed of the left lower-wishbone relative to the vehicle body (L signifies left); and K_{ARF} is the anti-roll bar stiffness (where F signifies front); finally, q_{RWB} is the angle between the right lower-wishbone and the vehicle body.

B.3 Aerodynamic Forces

The aerodynamic model used in this work includes speed-dependent drag and down forces, however, neglects the side-force and aerodynamic moments. Both forces depend quadratically on speed and can be calculated using the standard equation B.2, where K represents the co-efficient of either drag K_{drag} or down-force K_{down} , and F either the down F_{down} or drag F_{drag} force.

$$F = K \left(\frac{v_{long}}{\mu_{ref}} \right)^2 \quad (\text{B.2})$$

The drag force acts through the centre of mass and opposes the direction of travel. The down force acts vertically and is applied to the front and rear vehicle axles. The distribution of the down-force between the two application points depends on the location of the vehicle's centre-of-pressure, which is represented as a fraction

of the vehicle length D_{aero} . The resulting front and rear force distribution is calculated by:

$$F_{front} = F_{down} D_{Aero}, \quad (B.3)$$

and

$$F_{rear} = F_{down}(1 - D_{Aero}), \quad (B.4)$$

respectively.

B.4 Tyres

The tyres generate longitudinal, lateral and normal forces that are applied to the vehicle through the wheel contact points. Empirical formulae, given in [51] determine the magnitude of the forces as functions of the tyre slip angle, longitudinal slip and normal load, with the tyre parameters given in Appendix A.

All the tyre forces are applied through the contact point of the tyre and the road, with one force in each of the wheel vertical, longitudinal and lateral directions. The vertical force depends on the tyre stiffness and the vertical loading on each wheel. By modelling the tyres as spring-dampers, the compression is used to compute the force being exerted on the tyre. Equation B.5 calculates the force, where K_t and D_t are the tyre stiffnesses and Δl_n is the deformation of the tyre.

$$F_z = K_t(\Delta l_n) + D_t \frac{d}{dt}(\Delta l_n) \quad (B.5)$$

B.5 Speed and Steering Controllers

The car is driven by the longitudinal slip of the rear tyres. A proportional plus integral controller is used to generate the slip demand from the speed error - the difference between a reference and the vehicle longitudinal velocity.

The vehicle is steered by applying a torque about the vertical axis of the front wheels. A lateral acceleration reference is computed from the reference longitudinal speed and desired radius of curvature. The lateral acceleration error is fed back into a proportional plus differential controller to generate the steering torque.

Appendix C

Linear Cueing Filters

The weighting parameters and final transfer functions from the optimal linear filters are given here.

C.1 Longitudinal Weights and Filters

The longitudinal cost function weights are summarised in Table C.1.

TABLE C.1: Table of the values of the weighting parameters for the longitudinal optimal filter designs.

	$w_1(s)$	$w_2(s)$	$w_3(s)$	$w_4(s)$	$f_{lp}(s)$
$F_1(s)$	2.4	1.2	20	20	-
$F_2(s)$	$\frac{4.4s^2}{s^2+10.66s+56.85}$	1.5	40	40	-
$F_3(s)$	$\frac{85s^2}{s^2+44.43s+986.96}$	20	1000	$\frac{1.26 \times 10^5}{s^2+17.77s+157.91}$	-
$F_4(s)$	$\frac{0.3s^2}{s^2+39.99s+799.44}$	$\frac{3.95}{s^2+4.44s+9.87}$	1	0.03	$\frac{0.32}{s^2+7.99s+31.98}$
$F_5(s)$	$\frac{0.1s^2}{s^2+75.53s+2852.3}$	$\frac{19.61}{s^2+4.44s+9.89}$	1.5	0.09	$\frac{16.6}{s^2+25.77s+332.0}$

The transfer functions for each of the longitudinal filters are given as follows:

$$F_1(s) = \frac{47s^3 + 9.5s^2 + 0.0019s + 9.5 \times 10^{-8}}{s^4 + 128s^3 + 1189s^2 + 1274s + 208} \quad (C.1)$$

$$F_2(s) = \frac{81s^4 + 807s^3 + 3229s^2 + 0.64s + 2.9 \times 10^{-5}}{s^5 + 182s^4 + 3643s^3 + 2.9 \times 10^4s^2 + 1.2 \times 10^5s + 9.4 \times 10^4} \quad (C.2)$$

$$F_3(s) = \frac{13s^6 + 1945s^5 + 1.0 \times 10^5s^4 + 1.4 \times 10^6s^3 + 1.0 \times 10^7s^2 + 2140s + 0.10}{s^7 + 146s^6 + 9463s^5 + 3.4 \times 10^5s^4 + 6.4 \times 10^6s^3 + 6.3 \times 10^7s^2 + 2.9 \times 10^8s + 1.9 \times 10^8} \quad (C.3)$$

$$F_4(s) = \frac{84s^8 + 2762s^7 + 3.7 \times 10^4s^6 + 2.8 \times 10^5s^5 + 1.2 \times 10^6s^4 + 3.2 \times 10^6s^3 + 4.0 \times 10^6s^2 + 811.3s + 0.04}{s^9 + 208s^8 + 8427s^7 + 1.7 \times 10^5s^6 + 2.1 \times 10^6s^5 + 1.5 \times 10^7s^4 + 6.7 \times 10^7s^3 + 1.6 \times 10^8s^2 + 1.6 \times 10^8s + 4.7 \times 10^6} \quad (C.4)$$

$$F_5(s) = \frac{14s^7 + 1782s^6 + 5.1 \times 10^4s^5 + 7.9 \times 10^5s^4 + 2.9 \times 10^6s^3 + 5.7 \times 10^6s^2 + 1142s + 0.056}{s^8 + 113s^7 + 6311s^6 + 2.1 \times 10^5s^5 + 4.0 \times 10^6s^4 + 4.5 \times 10^7s^3 + 2.2 \times 10^8s^2 + 1.7 \times 10^8s + 8.3 \times 10^6} \quad (C.5)$$

C.2 Lateral Weights and Filters

The weighting functions used in the design of the lateral filters that were tested with drivers are given in Table C.2. The transfer functions of the three lateral

TABLE C.2: Table of the values of the weighting parameters for the lateral optimal filter designs.

	$w_1(s)$	$w_2(s)$	$w_3(s)$	$w_4(s)$	$f_{lp}(s)$
$F_1(s)$	$\frac{5s^2}{s^2 + 0.89s + 0.39}$	1	140	140	-
$F_2(s)$	$\frac{0.15s^2}{s^2 + 57.76s + 1668}$	0	2.3	0.2	$\frac{1.58}{s^2 + 17.77s + 157.91}$
$F_3(s)$	$\frac{0.15s^2}{s^2 + 57.76s + 1668}$	0	2.3	0.2	$\frac{1.58}{s^2 + 17.77s + 157.91}$

filters are given by:

$$F_1(s) = \frac{117s^4 + 105s^3 + 46s^2 + 0.009s - 1.5e^{-5}}{s^5 + 324s^4 + 9355s^3 + 16935s^2 + 11355s + 3454} \quad (C.6)$$

$$F_2(s) = \frac{23s^6 + 1223s^5 + 27709s^4 + 320888s^3 + 1921608s^2 + 384s + 0.019}{s^7 + 186s^6 + 12250s^5 + 434247s^4 + 8441725s^3 + 85856787s^2 + 419659437s + 32924299} \quad (C.7)$$

$$F_3(s) = \frac{83s^3 + 17s^2 + 0.0033s - 1.7e^{-7}}{s^4 + 546s^3 + 25136s^2 + 6255s + 250}. \quad (C.8)$$

C.3 Yaw Weights and Filter

The weighting functions that produce the linear filter for the yaw motion are given in Table C.3.

TABLE C.3: Table of the values of the weighting parameters for the yaw optimal filter designs.

	$w_1(s)$	$w_2(s)$	$w_3(s)$	$w_4(s)$
$F_1(s)$	$\frac{s^2}{s^2 + 44.43s + 986.96}$	$\frac{0.15}{s^2 + 57.76s + 1668}$	$\frac{315.83}{s^2 + 17.77s + 157.9}$	0

The transfer function of the resulting filter is given by:

$$F_1(s) = \frac{8.47s^6 + 635s^5 + 12012s^4 + 117627s^3 + 404881s^2 + 696129s + 70}{s^7 + 70s^6 + 2421s^5 + 49718s^4 + 590875s^3 + 2910807s^2 + 7942101s + 9214671}. \quad (C.9)$$

Bibliography

- [1] R.W. Allen, G.D. Park, M.L. Cook, and D. Fiorentino. The effect of driving simulator fidelity on training effectiveness. In *Driving Simulation Conference North America*, Iowa City, USA, 2007.
- [2] D. Ariel and R. Sivan. False cue reduction in moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-14(4):665–671, 1984.
- [3] B. Augusto. Motion cueing in the chalmers driving simulator: An optimization-based control approach. Master’s thesis, Chalmers University of Technology, Sweden., 2009.
- [4] M. Baseggio, A. Beghi, M. Bruschetta, F. Maran, and D. Minen. An MPC approach to the design of motion cueing algorithms for driving simulators. In *Proceedings of 14th International IEEE Conference on Intelligent Transportation Systems*, Washington, USA, October 2011.
- [5] A. Beghi, M. Bruschetta, and F. Maran. A real time implementation of MPC based motion cueing strategy for driving simulators. In *In Proceedings of 51st IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 2012.
- [6] J.T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. Advances in Design and Control. Society for Industrial and Applied Mathematics, 2001. ISBN 9780898714883.

- [7] M. Bruschetta, F. Maran, A. Beghi, and D. Minen. An MPC approach to the design of motion cueing algorithms for a high performance 9 DOFs driving simulator. In *Driving Simulator Conference*, Paris, France, September 2014.
- [8] K. Cammaerts and R. Stevens. Mobile platform. UK Patent 2474279, 2011.
- [9] A. Capustiac, D. Banabic, D. Schramm, and U. Ossendoth. Motion cueing: From design until implementation. In *Romanian Academy*, volume 12(2) of A, 2011.
- [10] T. Chapron and J-P. Colinot. The new PSA Peugeot-Citroën advanced driving simulator overall design and motion cue algorithm. In *Driving Simulation Conference North America*, Iowa City, USA, 2007.
- [11] F. Colombet, M. Dagdelen, G. Reymond, C. Pere, F. Merienne, and A. Kemeny. Motion cueing: What is the impact on the driver’s behaviour? In *Driving Simulation Conference*, Monaco, France, February 2008.
- [12] V. Cossalter, R. Lot, and M. Massaro. The chatter of racing motorcycles. *Vehicle Systems Dynamics*, 46(4):339–353, 2008.
- [13] M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier, and N. Maizi. Model-based predictive motion cueing strategy for vehicle driving simulators. *Control Engineering Practice*, 17:995–1003, 2009.
- [14] B. Dasgupta and T.S. Mruthyunjaya. A Newton-Euler formulation for the inverse dynamics of the Stewart platform manipulator. *Mechanical Machine Theory*, 33(8):1135–1152, 1998.
- [15] S. de Groot, M. Mulder, and P.A. Wieringa. Nonvestibular motion cueing in a fixed-base driving simulator: Effects on driver braking and cornering performance. *Presence*, 20(2):117–142, 2011.

- [16] C.A. Desoer. Slowly varying systems $\dot{x} = A(t)x$. *IEEE Transactions on Automatic Control*, 14(12):780–781, 1969.
- [17] J. Dichgans and T. Brandt. Visual-vestibular interaction: Effects on self-motion perception and postural control. In *Handbook of Sensory Physiology*, volume 8, pages 755–804. Springer Berlin Heidelberg, 1978.
- [18] J. Drosdol, W. Käding, and F. Panik. The Daimler-Benz driving simulator new technologies demand new instruments. *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, 15(1):44–57, 1986.
- [19] P. J. Feenstra, I. M. Wentink, I. Z. Roza, W. Bles, and I. A. R. V. Pais. Desdemona: a new driving-simulation platform. Technical report, AMST-Systemtechnik GmbH, 2007.
- [20] P.J. Feenstra, M. Wentink, Z.C. Roza, and W. Bles. Desdemona, an alternative moving base design for driving simulation. In *Driving Simulation Conference North America*, Iowa City, USA, September 2007.
- [21] C. Fernandez and J. Goldberg. Physiology of peripheral neurons innervating semicircular canals of the squirrel monkey. ii. response to sinusoidal stimulation and dynamics of peripheral vestibular system. *Journal of Neurophysiology*, 34(4):661–675, 1971.
- [22] C. Fernandez and J. Goldberg. Physiology of peripheral neurons innervating otolith organs of the squirrel monkey. *Journal of Neurophysiology*, 34:635–675, 1971.
- [23] N. Filliard, B. Vailleur, G. Reymond, and A. Kemeny. Combined scale factors for lateral and yaw motion rendering. In *Driving Simulation Conference*, Monaco, France, February 2009.

- [24] D. Garg, M. A. Patterson, W. W. Hager, A. V. Rao, A. V. Benson, and G. T. Huntington. A unified framework for the numerical solution of optimal control problems using pseudospectral methods. *Automatica*, 46(11):1843–1851, 2010.
- [25] N.J.I. Garrett and M.C. Best. Driving simulator motion cueing algorithms - a survey of the state of the art. In *10th International Symposium on Advanced Vehicle Control (AVEC)*, Loughborough, UK, August 2010.
- [26] N.J.I. Garrett and M.C. Best. Evaluation of a new body-sideslip-based driving simulator motion cueing algorithm. *Proceedings of the Institution of Mechanical Engineers Part D: Journal of Automobile Engineering*, 226(11):1433–1444, 2012.
- [27] N.J.I. Garrett and M.C. Best. Model predictive driving simulator motion cueing algorithm with actuator-based constraints. *Vehicle System Dynamics*, 51(8):1151–1172, 2013.
- [28] P.R. Giordano, C. Masone, J. Tesch, M. Breidt, Pollini L., and Bülthoff. A novel framework for closed-loop robotic motion simulation - part II: Motion cueing design and experimental validation. In *IEEE International Conference on Robotics and Automation*, Anchorage, USA, May 2010.
- [29] A.R. Glover. Motion arrangement for a vehicle simulator. UK Patent 2378687, February 2003.
- [30] M. Green and D. J. N. Limebeer. *Linear Robust Control*. Dover Books on Electrical Engineering. Dover Publications Inc., 2012.
- [31] B. Haycock and P.R. Grant. The influence of jerk on perceived simulator motion strength. In *Driving Simulation Conference North America*, Iowa City, USA, September 2007.

BIBLIOGRAPHY

- [32] R.J.A.W. Hosman and J.C. van der Vaart. *Vestibular Model and Thresholds of Motion Perception. Results of Tests in a Flight Simulator*. PhD thesis, Delft University of Technology, 1978.
- [33] A.H.J. Jamson. *Motion Cueing in Driving Simulators for Research Applications*. PhD thesis, The University of Leeds, 2010.
- [34] A. Kemeny and F. Panerai. Evaluating perception in driving simulation experiments. *TRENDS in Cognitive Sciences*, 7(1):31–37, 2003.
- [35] W.T. Koiter and H.B. Pacejka. Skidding of vehicle due to locked wheels. *Proceedings of the Institute of Mechanical Engineers*, 183(8):3–18, 1968.
- [36] Witaya Krissada, Chundang & Wannasuphoprasit. Washout filter for a 3 DOF simulator. In *The 5th International Conference on Automotive Engineering*, Bangkok, Thailand, 2009.
- [37] C.D. Larsen. Comparison of three degree of freedom and six degree of freedom motion bases utilizing classical washout algorithms. Master’s thesis, Iowa State University, 2011.
- [38] D. J. N. Limebeer, G. Perantoni, and A. V. Rao. Optimal control of formula one car energy recovery systems. *International Journal of Control*, 87(10):2065–2080, 2014. doi: 10.1080/00207179.2014.900705.
- [39] D.J.N. Limebeer, R.S. Sharp, and S. Evangelou. The stability of motorcycles under acceleration and braking. *Journal of Mechanical Engineering Science*, 215(9):1095–1109, 2001.
- [40] Mechanical Simulation Corporation. Autosim 2.5+ reference manual, 1998.
- [41] J.L. Meiry. *The Vestibular System and Human Dynamic Space Orientation*. PhD thesis, Massachusetts Institute of Technology, 1965.

BIBLIOGRAPHY

- [42] WF Milliken and DL Milliken. *Race car vehicle dynamics*. Warrendale (PA) SAE, 2005.
- [43] M.A. Nahon and L.D. Reid. Simulator motion-drive algorithms: A design's perspective. *Journal of Guidance, Control and Dynamics*, 13(2):356–362, 1990.
- [44] L.M. Nashner. *Sensory Feedback in Human Posture Control*. PhD thesis, Massachusettes Institute of Technology, 1970.
- [45] L. Nehaoua, H. Mohellebi, A. Amouri, H. Arioui, S. Espié, and A. Kheddar. Design and control of a small-clearance driving simulator. *IEEE Transactions on Vehicular Technology*, 56(2):736–746, 2008.
- [46] F.M. Nieuwenhuizen and H.H. Bühlhoff. The MPI cybermotion simulator: A novel research platform to investigate human control behavior. *Journal of Computing Science and Engineering*, 7(2):122–131, 2013.
- [47] C.C. Ormsby. *Model of Human Dynamic Orientation*. PhD thesis, Massachusetts Institute of Technology, 1974.
- [48] H.B. Pacejka. *Tyre and Vehicle Dynamics*. Butterworth-Heinemann Ltd, 2002.
- [49] I. A. R. V. Pais. TU Delft: Modelling human self-motion perception thresholds in vehicle simulation, 2011. URL www.lr.tudelft.nl.
- [50] Michael A. Patterson and Anil 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*. University of Florida, ACM Trans. Math. Soft. 39, 3, Article 1, July 2013.

- [51] G. Perantoni and D. J.N. Limebeer. Optimal control for a formula one car with variable parameters. *Vehicle System Dynamics*, 52(5):653–678, 2014. doi: 10.1080/00423114.2014.889315.
- [52] P. Pretto, H. Nusseck, H. Teufel, and H. H. Bühlhoff. Effect of lateral motion on drivers’ performance in the MPI motion simulator. In *Driving Simulation Conference*, Monaco, France, February 2009.
- [53] P. Pretto, A. Nesti, S. Nooij, M. Losert, and H.H. Bühlhoff. Variable roll-rate perception in driving simulation. In *Driving Simulation Conference*, Paris, France, September 2014.
- [54] H Radt and D. Van Dis. Vehicle handling responses using stability derivatives. SAE Technical Paper 960483, 1996.
- [55] J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.
- [56] L.D. Reid and M.A. Nahon. Flight simulation motion-base drive algorithm: Part 1 - developing and testing the equations. Technical Report UTIAS Report No. 296, Institute for Aerospace Studies, University of Toronto, Toronto, Canada, 1985.
- [57] G. Reymond and A. Kemeny. Motion cueing in the Renault driving simulator. *Vehicle System Dynamics*, 34(4):249–259, 2000.
- [58] H Rosenbrock. The stability of linear time-dependent control systems. *Journal of Electronic Control*, 15(1):73–80, 1963.
- [59] M. Roza and M. WenM. Wentink. Feenstra. Performance testing of the desdemona motion system. In *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Hilton Head, USA, August 2007.

BIBLIOGRAPHY

- [60] I.G. Salisbury and D.J.N. Limebeer. Optimal motion cueing for race cars. *IEEE Transactions on Control Systems Technology*, 24(1):200–215, 2016.
- [61] S.F. Schmidt and B. Conrad. Motion drive signals for piloted flight simulators. Technical Report NASA CR-1601, Washington: National Aeronautics and Space Administration, 1970.
- [62] R. Sharp. Motorsport vehicle dynamics: Dynamic analysis (2). MSc Lecture Notes, Cranfield University, 2001.
- [63] I. Siegler, G. Reymond, A. Kemeny, and A. Berthoz. Sensorimotor integration in a driving simulator: contributions of motion cueing in elementary driving tasks. In *Driving Simulation Conference*, Sophia Antipolis, France, September 2001.
- [64] R. Sivan, J. Ish-Shalom, and J. Huang. An optimal control approach to the design of moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, 12(6):818–827, 1982.
- [65] R. Skoog and C. Lau. Instability of slowly varying systems. *IEEE Transactions on Automatic Control*, AC-17(1), 1972.
- [66] J.J. Slob. State-of-the-art driving simulators, a literature survey. Technical report, Eindhoven University of Technology, 2008.
- [67] J. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, 1991.
- [68] N. Spooner, K. Seidler, A. Feldman, R. Langlois, M.J.D. Hayes, and T. Pearce. Development of the several integrated degree-of-freedom demonstrator (sidfred) simulator. In *Canadian Society of Mechanical Engineers Forum*, London, Canada, June 2004.

- [69] D.A. Stall and S. Bourne. The National Advanced Driving Simulator: Potential applications to its and AHS research. In *1996 Annual Meeting of ITS America*, 1996.
- [70] D. Stewart. A platform with six degrees of freedom. *Proceedings of the Institution of Mechanical Engineers*, 180(15):371–386, 1965-1966.
- [71] R. Suikat. The new dynamic driving simulator at DLR. In *Driving Simulation Conference North America*, Orlando, USA, November 2005.
- [72] R.J. Telban and F.M Cardullo. Motion cueing algorithm development: Human-centred linear and non-linear approaches. Technical report, State University of New York, 2005.
- [73] R.J. Telban, F.M. Cardullo, and J.A. Houck. Developments in human centered cueing algorithms for control of flight simulator motion systems. In *AIAA Modeling and Simulation Technologies Conference*, Portland, USA, 1999.
- [74] R.J. Telban, W. Wu, and F.M Cardullo. Motion cueing algorithm development: Initial investigations and redesign of the algorithms. Technical Report CR-2000-209863, State University of New York, 2000.
- [75] H. J. Teufel, H.G. Nusseck, K. A. Beykirch, J. S. Butler, M. Kerger, and H. H. Bühlhoff. MPI motion simulator: Development and analysis of a novel motion simulator. In *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Hilton Head, USA, August 2007.
- [76] A.J. Tremlett, F. Assadian, D.J. Purdy, N. Vaughan, A.P. Moore, and M. Halley. Quasi-steady-state linearisation of the racing vehicle acceleration envelope: a limited slip differential example. *Vehicle Systems Dynamics*, 52(11):1416–1442, 2014.

- [77] F.A.M. Van der Steen. *Self-Motion Perception*. PhD thesis, Delft University of Technology, 1998.
- [78] A.A.J. van Egmond, J.J. Groen, and L.B.W. Jongkees. The mechanics of the semicircular canal. *Journal of Physiology*, 110:1–17, 1949.
- [79] Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Control System Technologies*, 18(2):267–278, 2010.
- [80] C. Weiß. *Control of a Dynamic Driving Simulator: Time-Variant Motion Cueing Algorithm and Prepositioning*. PhD thesis, Deutsches Zentrum für Luft und Raumfahrt, 2006.
- [81] M. Wentink, W. Bles, and R. Hosman. Design & evaluation of spherical washout algorithm for Desdemona simulator. In *AIAA Modeling and Simulation Technologies Conference and Exhibit*, San Francisco, USA, August 2005.
- [82] M. Wentink, R.V. Pais, M. Mayrhofer, P. Feenstra, and W. Bles. First curve driving experiments in the Desdemona simulator. In *Driving Simulation Conference*, Monaco, France, January 2008.
- [83] J. Yi, J. Li, J. Lu, and Z. Liu. On the stability and agility of aggressive vehicle maneuvers: A pendulum-turn maneuver example. *IEEE Transactions on Control Systems Technology*, 20(3):663–676, 2012.
- [84] L.R. Young and J.L. Meiry. A revised dynamic otolith model. *Medical Physiology*, art. 13th, 1974.
- [85] L.R. Young and C.M. Oman. Model for vestibular adaptation to horizontal rotation. *Journal of Aerospace Medicine*, 40(10):1076–1080, 1969.

BIBLIOGRAPHY

- [86] G.L. Zacharias. Motion cue models for pilot-vehicle analysis. Technical Report AMRL-TR-78-2, Department of Defense, 1978.
- [87] E. Zeeb. Daimler's new full-scale, high-dynamic driving simulator: A technical overview. In *Driving Simulation Conference*, Paris, France, September 2010.