



AUGUST 2019

MASTER THESIS

# Analysis of Dynamic Loads induced by Spinning Gondolas on a Roller Coaster

H.E. van den Hoorn

Supervisor: ir. Frank de Ruiter CEng

**Van Velzen Extern Engineering**  
STADHOUDERSMOLENWEG 70  
7317 AW APELDOORN  
THE NETHERLANDS

**University of Twente**  
P.O. Box 217  
7500 AE ENSCHEDE  
THE NETHERLANDS



UNIVERSITY OF TWENTE.

# Analysis of Dynamic Loads induced by Spinning Gondolas on a Roller Coaster

FACULTY:

ENGINEERING TECHNOLOGY

PROGRAMME:

MECHANICAL ENGINEERING

DEPARTMENT:

MECHANICS OF SOLIDS, SURFACES & SYSTEMS

RESEARCH CHAIR:

STRUCTURAL DYNAMICS, ACOUSTICS & CONTROL

AUTHOR:

Hendrik Evert van den Hoorn

DATE:

August 30, 2019

*Thesis Committee:*

Prof. dr. ir. A. de Boer (Chairman)

Dr. ir. J.P. Schilder (University supervisor)

Ir. F. de Ruiter CEng (Company supervisor)

Dr. ir. A. Martinetti (External member)

# Abstract

The competition between rollercoaster manufacturers to build the most thrilling rollercoasters has always been fierce. In an attempt to design more spectacular rides than their competitors, rollercoaster manufacturers continuously try to push the technical limits without compromising the safety of the passengers. The design of safe rollercoasters according to the prevalent regulations has the first priority within the industry, in spite of the eagerness to build ever more spectacular rides. Partial redesigns of rollercoasters due to violations of the safety regulations or excessive maintenance costs have namely proven to be a costly endeavour. Operating on the limits while minimizing the risk of exceeding them makes the design of rollercoasters a delicate process. The technical specifications of rollercoasters can be calculated beforehand with ever more accuracy, thanks to the progress in computer-aided modelling. The development of sophisticated software enables engineers to predict accelerations, forces, stresses, and other important parameters during the design stage. Due to limited computational power, a trade-off should often be made between accuracy and the computational effort though.

The loads induced on the main chassis beam of a spinning rollercoaster vehicle are typically converted into a minimum number of loadcases, which consequently leads to a reduction in computational effort at the costs of accuracy. A spinning rollercoaster is characterized by a gondola that pivots freely under the effects of track dynamics and passenger weight distribution. The loads induced on the beam are converted into loadcases for a subsequent finite element analysis, which should reveal the resultant stresses and beam deformation. Due to the desire to minimize the computational time, the forces and moments are represented by a minimum number of relatively conservative loadcases. In other words, the forces and moments described by the loadcases are more severe than the actual loads acting on the main chassis beam. The predicted stresses and deformations are consequently larger than what can be expected based on the actual loads. This conservative approach might lead to more conservative rollercoaster designs, which conflicts with the vision of designing the most thrilling and spectacular rollercoasters. Therefore, a need exists for a general methodology that is capable of accurately predicting stress and deformation levels, while minimizing the increase in computational effort with respect to the conventional yet conservative approach.

Prior to the development of such a methodology, a kinematic model of a spinning rollercoaster should be created first. The report commences with the creation of a realistic kinematic model, from which the forces and moments acting on the main chassis beam can be extracted. Hence, the multi-body model aims to describe the dynamics of a spinning rollercoaster vehicle as realistically as possible by specification of the correct constraints and joints between the vehicle components and at the wheel-track interface. The normal forces at this interface are constantly measured and used for the real-time calculation of the bearing and rolling frictional forces. Simulations are additionally performed to determine the drag coefficient of the gondola at various velocities. Hence, the drag forces acting on the gondola and the chassis can be computed at each time-step as a function of the vehicle velocity and gondola rotation. The inclusion of these friction and drag forces ensures that the rollercoaster vehicle travels along the track lay-out with a realistic pace. The velocity of the vehicle and the rotation of the gondola are prescribed by a reference profile at certain track sections, so the vehicle can travel along the entire track lay-out. The vehicle should accelerate from the station up to the constant velocity that has been prescribed on the lifthill, while in the meantime the gondola is kept at its initial position. On the brake-section and subsequently at the re-entry of the station, the vehicle is braked to a standstill while the gondola is rotated back to the regular configuration. The dynamic behaviour of the spinning rollercoaster vehicle can be accurately described thanks to the previously-described additions to the kinematic model.

The accelerations experienced by the passengers are compared to the regulations for tolerable passenger accelerations. These passenger acceleration limits are namely of interest to ensure that the kinematic model includes a realistic track lay-out, so the forces and moments that are exerted on the main chassis beam are of

realistic magnitude as well. The passenger accelerations are evaluated for four unique passenger occupancy configurations, since it is unknown beforehand which configuration corresponds to the most severe forces and moments induced on the beam. By considering a variable number of passengers for each configuration, the loads extracted from the kinematic model represent a wide range of forces and moments. The linear acceleration magnitudes, acceleration combinations, and acceleration reversals are compared to the regulatory limits. Several design iterations of the rollercoaster track were required to meet the acceleration norms. The forces and moments at the joints should first be validated before these loads can be applied on the main chassis beam in a subsequent finite element analysis. The validation of the reaction forces is based on the masses and the acceleration values that have been acquired from the kinematic model. Special attention is paid to the validation of the forces and moments at the joints between the beam and the other vehicle components, since they are directly used in the finite element analysis of the beam.

Once the forces and moments have been validated successfully, they can be converted into loadcases. The load data is first sampled and subsequently two different load envelopes are created that enclose all force and moment data. The conventional method results in cubical and rectangular envelopes for respectively the force and torque data, whose shapes are defined by a total of twelve unique loadcases. The conventional loadcases describe forces and moments of relatively large magnitude in comparison to the actual load data though. On the other hand, the optimized method is characterized by envelopes whose boundaries coincide with the actual boundaries of the load data, so the loadcases correspond to actual loads exerted on the main chassis beam. The optimized method results in a total of almost forty unique loadcases. After the constraints have been modelled correctly in the finite element model, the beam can be subjected to the conventional and optimized loadcases. The finite element model is validated by means of a comparison with the kinematic model in terms of the reaction forces at the front and rear axle. The relative percentage difference between the reaction forces from both models is generally acceptable. Therefore, the stress and displacement results from the finite element model can be regarded as trustworthy. The maximum stress and displacement values predicted by the optimized method are nearly ten percent lower than the results obtained with the conventional method, which confirms the earlier hypothesis. However, the stress and displacement values could have been predicted too optimistically by the optimized method, and hence these static results are compared to the results of a transient analysis. A transient analysis is typically characterized by a large number of loadcases, and all loads should be evaluated in the correct sequence. A comparison between the results obtained with the optimized method and the transient analysis shows that the maximum stress and displacement values are practically similar for both analyses. The duration of the computations for the transient analysis is at the same time orders of magnitude longer than the computational time for the optimized method. Hence, if the prediction of the maximum stress and deformation levels is the primary objective of the analysis, it is advisory to use the optimized method.

Certain types of analyses require a preliminary stress analysis for each load in a load cycle, such as a fatigue analysis. Contrary to the optimized method, a transient analysis typically comprises a vast number of loadcases in chronological order, and the resulting stress profile can directly be used as input for a subsequent fatigue analysis. The duration of the computations for a full transient analysis can be substantial though, and therefore the computational time is reduced by means of the mode-superposition method at a minimum cost of accuracy. The principle of the mode-superposition technique is that the displacement of a structure can be described by a linear combination of its eigenmodes. Hence, this method requires a preliminary modal analysis to determine the number of eigenmodes that should be taken into account. Four different mode-superposition transient analyses are performed, where the analyses with ten and thirteen eigenmodes yield satisfactory results. A minor deviation with the results from a full transient analysis is obtained at the benefit of an eighty percent reduction in computational time. The stress and displacement profiles can therefore be calculated relatively accurately and very efficiently by means of a mode-superposition analysis when a sufficiently large number of eigenmodes is taken into account.

# Preface

This thesis is written as a partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering at the University of Twente. The research topic of this thesis originates from my age-long interest in rollercoasters. Until recently I have constructed scale-model rollercoasters in my spare time, including recreations of existing rollercoasters. Although I currently lack the time to pursue this hobby, I still enjoy visiting amusement parks to ride actual rollercoasters. My interest in rollercoasters has also been a major motivation for me to start studying Mechanical Engineering at the University of Twente approximately five years ago. Back then I could not even remotely foresee that my hobby would evolve into the subject of my graduation project. It has been a pleasure to work on this research topic and I am very satisfied with the final result and the insights I have gained during the process.

My graduation assignment has been conducted between January and August 2019 at Van Velzen Extern Engineering BV in Apeldoorn. This company is specialized in the structural analysis of dynamically loaded structures, such as rollercoasters and other amusement rides. Van Velzen is namely part of Intamin AG, a manufacturer of amusement rides based in Liechtenstein that operates on global level. Intamin has built record-breaking rollercoasters over recent years and they are known for their innovative ride designs. Van Velzen focuses on the mechanical and structural analysis of the amusement rides constructed by Intamin. Thanks to the delicate work done by Van Velzen and others, rollercoasters are not only exciting but also safe and comfortable to ride.

My graduation assignment was supervised by ir. Frank de Ruiter, who is the managing director at Van Velzen Extern Engineering. I would like to thank him for his guidance and support throughout the research and for providing me with the feedback that lead to this thesis. I would also like to express my gratitude to dr. ir. Jurnan Schilder for his contribution to my thesis in terms of valuable advice during our meetings. His approval of this rewarding assignment is greatly appreciated. I also wish to thank my colleagues at Van Velzen Extern Engineering for the pleasant time during my research.

Helmer van den Hoorn

August 2019

# Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b>                                | <b>1</b>  |
| <b>2</b> | <b>Methodology</b>                                 | <b>5</b>  |
| 2.1      | Rollercoaster Track . . . . .                      | 5         |
| 2.2      | Multibody Model of Rollercoaster Vehicle . . . . . | 6         |
| 2.3      | Point-on-Curve Constraints . . . . .               | 7         |
| 2.3.1    | Constraint Configuration . . . . .                 | 7         |
| 2.3.2    | Track Control Point Increment . . . . .            | 8         |
| 2.3.3    | Reaction Forces between Track and Wheels . . . . . | 9         |
| 2.4      | Modelling Rolling and Bearing Friction . . . . .   | 10        |
| 2.4.1    | Rolling Frictional Force . . . . .                 | 10        |
| 2.4.2    | Bearing Frictional Moment . . . . .                | 11        |
| 2.4.3    | Resultant Rolling and Bearing Friction . . . . .   | 11        |
| 2.5      | Modelling Air Resistance . . . . .                 | 13        |
| 2.5.1    | Frontal Surface . . . . .                          | 13        |
| 2.5.2    | Gondola Drag Coefficient . . . . .                 | 13        |
| 2.5.3    | Gondola Drag Force . . . . .                       | 15        |
| 2.5.4    | Resultant Drag Forces . . . . .                    | 15        |
| 2.6      | Motion Control . . . . .                           | 16        |
| 2.7      | Tolerable Passenger Accelerations . . . . .        | 18        |
| 2.7.1    | Passenger Occupancy Configurations . . . . .       | 19        |
| 2.7.2    | Acceleration Combinations . . . . .                | 20        |
| 2.7.3    | Acceleration Reversals . . . . .                   | 21        |
| 2.8      | Model Validation . . . . .                         | 22        |
| 2.8.1    | Forces at Pivot Joint . . . . .                    | 22        |
| 2.8.2    | Torques at Pivot Joint . . . . .                   | 22        |
| 2.8.3    | Forces at Front and Rear Axle Joints . . . . .     | 23        |
| 2.8.4    | Torques at Rear Axle Joint . . . . .               | 25        |
| <b>3</b> | <b>Results</b>                                     | <b>26</b> |
| 3.1      | Loadcases . . . . .                                | 26        |
| 3.2      | Constraints . . . . .                              | 28        |
| 3.3      | Reaction Forces . . . . .                          | 30        |
| 3.4      | Resultant Stresses and Displacements . . . . .     | 32        |
| 3.5      | Transient Analysis . . . . .                       | 32        |
| 3.6      | Mode-Superposition Method . . . . .                | 37        |
| 3.6.1    | Modal Analysis . . . . .                           | 37        |
| 3.6.2    | Mode-Superposition Transient Analysis . . . . .    | 38        |
| <b>4</b> | <b>Conclusions</b>                                 | <b>40</b> |
| <b>5</b> | <b>Recommendations</b>                             | <b>41</b> |
|          | <b>References</b>                                  | <b>44</b> |
|          | <b>Appendices</b>                                  | <b>45</b> |

|          |  |           |
|----------|--|-----------|
| <b>A</b> | <b>Simulink Model</b>                                | <b>45</b> |
| A.1      | Main Scheme . . . . .                                | 45        |
| A.2      | Front-left Wheel Carrier Schemes . . . . .           | 46        |
| A.3      | Front-right Wheel Carrier Schemes . . . . .          | 47        |
| A.4      | Rear-left Wheel Carrier Schemes . . . . .            | 48        |
| A.5      | Rear-right Wheel Carrier Schemes . . . . .           | 49        |
| A.6      | Friction Force Schemes . . . . .                     | 50        |
| A.7      | Front Axle Scheme . . . . .                          | 52        |
| A.8      | Rear Axle Scheme . . . . .                           | 52        |
| A.9      | Gondola Scheme . . . . .                             | 53        |
| A.10     | Main Chassis Beam Scheme . . . . .                   | 54        |
| A.11     | Air Drag Force Scheme . . . . .                      | 54        |
| A.12     | Beam Scheme . . . . .                                | 55        |
| A.13     | Pivot Bearing Friction Force Scheme . . . . .        | 56        |
| A.14     | Output Scheme . . . . .                              | 57        |
| A.15     | Motion Control Scheme . . . . .                      | 58        |
| A.16     | Translation Control Brake Section Scheme . . . . .   | 58        |
| A.17     | Translation Control Station Scheme . . . . .         | 59        |
| A.18     | Translation Control Lifthill Scheme . . . . .        | 60        |
| A.19     | Rotation Control Brake Section Scheme . . . . .      | 61        |
| A.20     | Rotation Control Station Scheme . . . . .            | 62        |
| A.21     | Rotation Control Lifthill Scheme . . . . .           | 63        |
| A.22     | Rotation Control Turnaround . . . . .                | 63        |
| <b>B</b> | <b>Matlab Functions in Simulink Model</b>            | <b>64</b> |
| B.1      | Copy of FuncRunning.m . . . . .                      | 64        |
| B.2      | Copy of GondolaDragForce.m . . . . .                 | 66        |
| B.3      | Copy of ChassisDragForce.m . . . . .                 | 68        |
| <b>C</b> | <b>Track Generator</b>                               | <b>69</b> |
| C.1      | NoLimits Track Data to Rail Coordinates . . . . .    | 69        |
| C.2      | Copy of TrackGenerator.m . . . . .                   | 71        |
| <b>D</b> | <b>Track Oscillations</b>                            | <b>73</b> |
| <b>E</b> | <b>Hertzian Elliptic Contact Model</b>               | <b>74</b> |
| E.1      | Modelling Rolling Contact . . . . .                  | 74        |
| E.2      | Modelling Viscoelasticity . . . . .                  | 75        |
| <b>F</b> | <b>Bearing Friction Calculation</b>                  | <b>78</b> |
| F.1      | Rolling Frictional Moment . . . . .                  | 78        |
| F.2      | Sliding Frictional Moment . . . . .                  | 80        |
| F.3      | Frictional Moment of Seals and Drag Losses . . . . . | 80        |
| <b>G</b> | <b>Modelling Air Resistance</b>                      | <b>81</b> |
| G.1      | Frontal Surface Values . . . . .                     | 81        |
| G.2      | Cubic Interpolation Spline . . . . .                 | 81        |
| G.3      | Boundary Conditions . . . . .                        | 82        |
| G.4      | Reynolds Number . . . . .                            | 84        |
| G.5      | Initialization . . . . .                             | 84        |
| G.6      | Air Flow Simulations . . . . .                       | 86        |
| G.7      | Discussion on the Residuals . . . . .                | 86        |
| G.8      | Frontal Surfaces used in Drag Analysis . . . . .     | 88        |

|          |  |            |
|----------|--|------------|
| G.9      | Residuals Plots from Drag Analysis . . . . .                   | 89         |
| G.10     | Air Velocity Contour Plots from Drag Analysis . . . . .        | 93         |
| <b>H</b> | <b>Motion Control</b>  | <b>97</b>  |
| H.1      | Translation Control Brake Section . . . . .                    | 97         |
| H.2      | Translation Control Station . . . . .                          | 98         |
| H.2.1    | Station Exit . . . . .   | 98         |
| H.2.2    | Station Entry . . . . .  | 99         |
| H.3      | Translation Control Lifthill . . . . .                         | 100        |
| H.4      | Rotation Control Brake Section . . . . .                       | 101        |
| H.5      | Rotation Control Station Entry . . . . .                       | 103        |
| H.6      | Rotation Control Lifthill . . . . .                            | 103        |
| H.7      | Rotation Control Turnaround . . . . .                          | 104        |
| <b>I</b> | <b>Acceleration Effects on Passengers</b>                      | <b>105</b> |
| I.1      | Time Duration Limits for Accelerations . . . . .               | 105        |
| I.2      | Resulting Passenger Accelerations . . . . .                    | 106        |
| I.2.1    | Configuration 1 . . . . .                                      | 106        |
| I.2.2    | Configuration 2 . . . . .                                      | 108        |
| I.2.3    | Configuration 3 . . . . .                                      | 110        |
| I.2.4    | Configuration 4 . . . . .                                      | 111        |
| I.3      | Copy of AdmissibleX.m . . . . .                                | 112        |
| I.4      | Copy of AdmissibleY.m . . . . .                                | 112        |
| I.5      | Copy of AdmissibleZ.m . . . . .                                | 113        |
| I.6      | Copy of AdmAccelerations.m . . . . .                           | 113        |
| <b>J</b> | <b>Model Validation</b>  | <b>122</b> |
| J.1      | Forces at Front and Rear Hinge Joints . . . . .                | 122        |
| J.2      | Forces between Track and Wheels . . . . .                      | 123        |
| J.3      | Model Validation Graphs . . . . .                              | 124        |
| J.3.1    | Forces at Pivot Joint . . . . .                                | 124        |
| J.3.2    | Torques at Pivot Joint . . . . .                               | 125        |
| J.3.3    | Forces at Front and Rear Axle Joints . . . . .                 | 125        |
| J.3.4    | Torques at Rear Axle Joint . . . . .                           | 126        |
| J.3.5    | Forces at Front Hinge Joints . . . . .                         | 126        |
| J.3.6    | Forces at Rear Hinge Joints . . . . .                          | 127        |
| J.3.7    | Forces between Track and Wheels at Front-Left Bogie . . . . .  | 127        |
| J.3.8    | Forces between Track and Wheels at Front-Right Bogie . . . . . | 128        |
| J.3.9    | Forces between Track and Wheels at Rear-Left Bogie . . . . .   | 128        |
| J.3.10   | Forces between Track and Wheels at Rear-Right Bogie . . . . .  | 129        |
| <b>K</b> | <b>Loadcases and Reaction Forces</b>                           | <b>130</b> |
| K.1      | Loadcases Optimized Method . . . . .                           | 130        |
| K.1.1    | Forces and Torques . . . . .                                   | 130        |
| K.1.2    | Accelerations . . . . .  | 131        |
| K.2      | Torques at Rear Revolute Joint . . . . .                       | 132        |
| K.3      | Comparison between Reaction Forces . . . . .                   | 133        |
| <b>L</b> | <b>Ansys Command Lines</b>                                     | <b>136</b> |
| <b>M</b> | <b>Resultant Stresses and Displacements</b>                    | <b>141</b> |

|          |  |            |
|----------|--|------------|
| <b>N</b> | <b>Stress and Displacement as a function of Time</b> | <b>142</b> |
| N.1      | Configuration 2 . . . . .                            | 142        |
| N.2      | Configuration 3 . . . . .                            | 142        |
| N.3      | Configuration 4 . . . . .                            | 142        |
| <b>O</b> | <b>Mode-Superposition Method</b>                     | <b>143</b> |
| O.1      | Modal Analysis . . . . .                             | 143        |
| O.2      | Comparison with Full Transient Analysis . . . . .    | 144        |
| O.2.1    | Configuration 2 . . . . .                            | 144        |
| O.2.2    | Configuration 3 . . . . .                            | 144        |
| O.2.3    | Configuration 4 . . . . .                            | 144        |
| O.3      | Tabulated Results . . . . .                          | 145        |

# 1. Introduction

Roller coasters are regarded by the majority of visitors as the most iconic and popular attractions in amusement parks. While wooden roller coasters were the highlight of every amusement park during the first half of the previous century, it has been the development of steel roller coasters over the last decades that has opened a wide array of design possibilities. Steel roller coasters are namely more capable of incorporating complex elements and inversions in the ride lay-out in comparison to their wooden equivalents. Especially the introduction of virtual modelling techniques at the end of the previous century has enabled manufacturers to design taller and faster roller coasters than ever before. The progress in terms of engineering knowledge and computational power has also led to the development of a wide variety of different roller coaster types over recent years, since amusement parks wish to compose a diverse portfolio of rides to attract visitors of all ages. Whereas larger roller coasters typically create the most thrilling experience, the development of new roller coaster types has proven that smaller family-oriented roller coasters can also be exciting.



Figure 1.1: An example of a recently opened spinning roller coaster [1]

The development of a so-called spinning roller coaster is a proper example of a ride innovation that has made relatively small and compact roller coasters more exciting. Hence, the amusement industry has witnessed a large increase in popularity of these spinning roller coasters over the last two decades. The trains or coaches on these roller coasters are characterized by a gondola that is capable of rotating freely under the effects of the track dynamics and passenger weight distribution, making every ride a unique experience. The entire ride experience is taken to a new level, since the speed of a roller coaster is combined with the experience of a carousel ride. The spinning coaster is an exciting ride for the entire family, which explains the addition of several spinning coasters to amusement parks over recent years, such as the one depicted in Figure 1.1.

Although the thrilling effect of spinning coasters will appeal to the riders mostly, one should also appreciate the high level of safety that is maintained during the operation of the spinning coasters. Aside from active safety measures such as an adequate maintenance program, the safe operation of these roller coasters is also possible thanks to the execution of detailed calculations and dynamic simulations during the design phase. These calculations are typically characterized by a two-fold focus, since the safety measures need to guarantee both rider safety and structural integrity. One of the aspects of passenger safety regulations is for instance the use of a train clearance envelope, which specifies a certain minimum clearance between the train envelope and any surrounding object to prevent any serious injuries. Another principal objective with respect to rider safety is to keep the maximum linear accelerations induced on the passengers below a specified value, so the passengers will feel comfortable throughout the entire ride.

Stringent regulations restrict the linear accelerations resulting from the track lay-out to which the passengers may be subjected during the ride. The accelerations that are imposed on the riders are measured at a specified reference point, which is referred to as the passenger measurement point. The exact location of the passenger measurement point is defined by certain standards that specify the distance between the

passenger measurement point and respectively the seat pan and seat back. These distances are typically formulated in terms of a range to take into account a wide variety of shapes and sizes of the passengers. The tolerable longitudinal, lateral, and vertical accelerations that are experienced by the passenger have been specified as a function of time according to the norms described in EN-13814. Although different standards can be applied as well, the norms in EN-13814 can be regarded as the European standard for the design and manufacturing of amusement rides. A dynamic simulation is performed to obtain the linear accelerations that the passengers are subjected to during the ride. Accelerations in different directions are generally encountered on different segments along the track-lay-out. The minimum and maximum longitudinal g-loads are typically experienced by the passenger at respectively a brake section and a drop or launch, whereas the lateral accelerations are the largest in turns and bends. The extreme values within the domain of permissible vertical accelerations are generally encountered at valleys and hills. In case of spinning roller coasters, special attention is paid to the simulation of different passenger loading configurations due to a change in dynamics of the rotating part for each configuration. The passenger loading configurations can differ in terms of seat occupancy, where the number of occupied seats can vary between zero and four with several different configurations possible.

The linear accelerations resulting from the dynamic analysis of a ride cycle can be combined into a two-dimensional or three-dimensional graph. When this procedure is executed for a variety of different passenger occupancy configurations, a two-dimensional or three-dimensional cloud of g-loads is obtained. This cloud of g-loads is bounded by lines that correspond to the minimum and maximum acceleration values that were obtained during the dynamic simulation, which results in a rectangular or cubical envelope of g-loads. This envelope can be regarded as the limits given by vehicle design load cases. The vertical and lateral linear accelerations, resulting from the assessments of a variety of configurations, are for instance combined into a two-dimensional graph of g-loads for the main beam of the chassis. This cloud of g-load combinations is then bounded by a rectangle, which represents the limits given by vehicle design load cases. Figure 1.2 shows a typical g-load distribution for the main beam of the chassis that could be observed in reality. In this cloud of g-loads, the different colours and symbols represent a variety of gondola configurations for which the linear accelerations were determined. The four red lines that bound the cloud of g-loads correspond to the extreme values found in the set of linear accelerations. The red dots that connect the red lines therefore represent the four vehicle design load cases that correspond to the corners of the envelope. Aside from the directions shown in Figure 1.2, a graph of g-load combinations can also be composed for the longitudinal accelerations in combination with the accelerations in other directions.

The pivot bearing point forms the centre of rotation on the main chassis beam for the rotating platform. Hence, the forces and moments induced by the dynamics of a rotating gondola are transferred to the main chassis beam through the pivot bearing point. The forces and moments can be computed for a variety of different gondola configurations, after which the resultant forces and moments in different directions can again be merged into a single graph. In a similar way as for the linear accelerations, all forces and moments in the graph should be within the boundaries dictated by the vehicle design load cases. It can typically be stated that a design is safe when each component can successfully withstand the vehicle design load cases corresponding to the corners of the envelope. All accelerations, forces, and moments that are encountered during the simulation are namely bounded by the rectangular or cubical envelopes, which means that the vehicle design load cases are a sufficient test case for assessing the structural integrity of the main chassis beam. Although this procedure is a respected method for ensuring the safety of spinning coaster trains, it might also be a too conservative approach. Especially on the corners of the envelope, the distance between the edges of the g-load cloud and the vehicle design load cases is substantial, as shown in Figure 1.2. The envelope is made symmetric to keep the proceedings during the structural analysis limited, which increases the distance between the observed g-load combinations and the load cases on the right-hand side of Figure 1.2 even further. As a consequence, the structure is tested by means of load cases that are more stringent than required.

If the vehicle design load cases could be optimized based on the actual boundaries of the g-load clouds instead of the envelope corners, the resultant maximum stresses in the structure would simultaneously be lower and more realistic. This could ultimately lead to a less conservative and more economical chassis design, for instance for the main beam. The pivot bearing point facilitates the transfer of forces and moments to the main chassis beam that are induced by the dynamic behaviour of the rotating gondola. A typical distribution of forces and moments that can usually be observed at the pivot bearing point is shown in Figure 1.3. When the optimization procedure leads to less severe vehicle design load cases in terms of forces and moments at the pivot bearing point, this will consequently lead to lower stress levels in the main beam. If the stress levels appear to be substantially lower in comparison to the conservative approach, one might subsequently conclude that the main beam is overdimensioned and that a more light-weight beam design would also suffice. However, it should be noted that load factors and other sources of loading should also be taken into account before any re-dimensioning of the beam can be considered. To improve the current procedure, an approach is proposed in which the dynamic behavior of a coach on a spinning roller coaster vehicle is modelled in Simscape Multibody. A physical multibody system can be constructed in the Simscape Multibody package, consisting of multiple rigid bodies that are linked to each other by means of a block scheme. The purpose of the Simscape model is to let a coach with a freely spinning gondola traverse a roller coaster track that is comparable to existing spinning roller coaster lay-out designs. A dynamic simulation of the spinning roller coaster is in principle a transient analysis in which the main chassis beam is subjected to forces and moments at the pivot bearing point and at the axles. The resulting stresses and strains can subsequently be determined using finite element software.

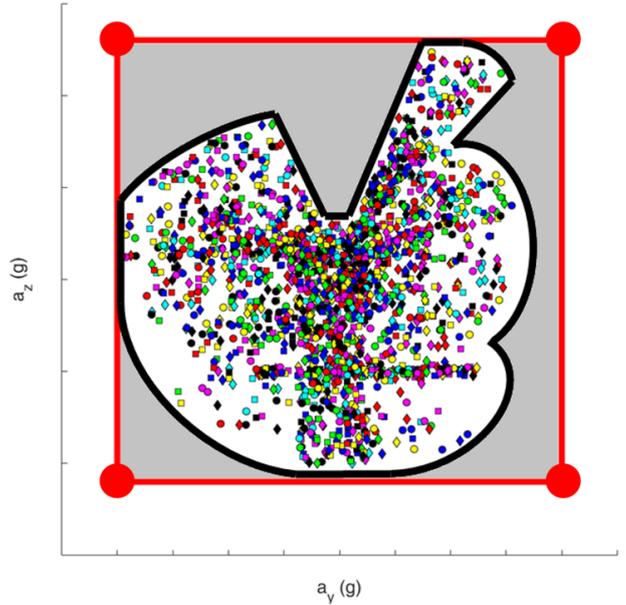


Figure 1.2: A typical yet random distribution of g-loads for a variety of configurations and gondola orientations on a spinning roller coaster, including the vehicle design load cases (red dots) and corresponding boundaries (red lines)

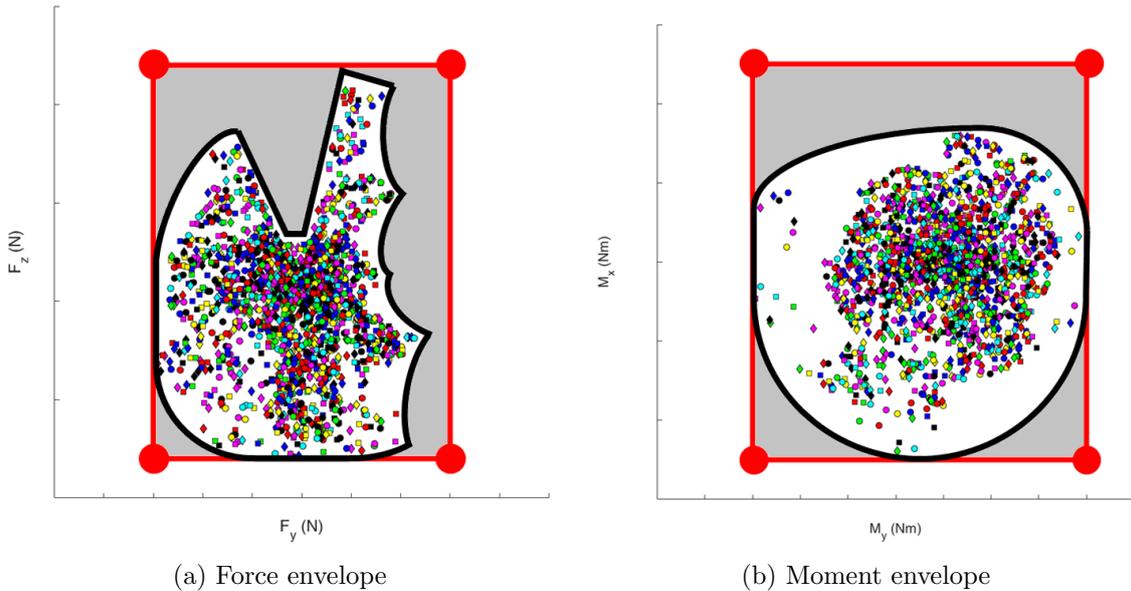


Figure 1.3: Typical yet random distribution of forces and moments induced by the spinning gondola on the pivot bearing point, where the room left for optimization of the loadcases is represented by grey surfaces

The primary objective of this thesis is to subject the roller coaster vehicle to more accurate load cases that are determined by means of dynamic simulations in Simscape Multibody. The anticipated outcome of the optimized structural finite element analysis is a decrease of the predicted strain and stress levels in comparison to the conventional methodology. The structural results of a full transient analysis are subsequently used as a reference for a comparison between the resultant stresses and strains as acquired with the conventional and optimized methods. Additionally, the mode-superposition technique is applied to reduce the simulation time of a transient analysis at a minimum cost of accuracy. Special attention is contributed towards a minimization of the computational effort during the structural analysis, while meeting the safety requirements and without any concessions with respect to the safety of passengers during a ride on a spinning roller coaster.

## 2. Methodology

### 2.1 Rollercoaster Track

To acquire the forces and moments that are exerted on the main chassis beam, a dynamic model should be developed that is capable of simulating a rollercoaster car with a freely spinning gondola that traverses a rollercoaster track lay-out. The dynamic analysis of the rollercoaster vehicle requires an accurate description of the track geometry. A methodology for the description of three-dimensional track geometries is proposed in which track data generated by the commercially available program NoLimits 2 is used. This simulator allows the user to create realistic rollercoaster lay-outs of all kinds, such as the spinning rollercoaster lay-out depicted in Figure 2.1. In this lay-out the car is first pulled to the top of the lifthill by a chain, and upon reaching the crest of the lifthill the rollercoaster vehicle is disengaged from the lift chain. From that point on, the car is allowed to freely coast along the track trajectory under the effect of gravity. When the rollercoaster vehicle traverses through the two almost flat and barely banked turns that follow directly after the lifthill, the car is subjected to relatively large lateral accelerations. This induces a rotational motion of the eccentrically mounted gondola. After the drop the car follows a curved section of track trajectory just above ground level, during which the rider experiences substantial vertical g-forces. The combination of a reduced velocity and a relatively large banking angle on the next turnaround spark a rotational acceleration of the gondola during a moment of so-called hang-time. The quick transition on the next element slightly amplifies the rotation of the gondola, after which the lay-out concludes with another turnaround and helix before the car reaches the brake section and the station.

The professional version of NoLimits allows the user to export the track geometry as a data file. The exported data file describes the position coordinates of the control points that are representative of the track heartline. Additionally, the banking of the track is described by the direction of three vectors that together form a moving frame along the track lay-out. Appendix C describes the conversion of the track data exported from NoLimits to the coordinates of the control points that describe both the left and right rail. After the rail coordinates have been imported into the Simulink model, the rigid rollercoaster track is defined using piece-wise cubic interpolation to parameterise the geometry of the rails. Hence, the geometry of the rigid rollercoaster track consists of cubic spline segments that are interconnected by transition curves that ensure the continuity of the first and second derivatives of the track in the transition points [2]. The continuity of the first derivative implies that adjacent cubic spline segments share a common control point and a tangent line at the junction of the two spline curves. Continuity of the accelerations between two adjacent spline segments is achieved by the continuity of the second derivative, since this continuity condition forces the two curves to possess equal curvature at their joint. The enforcement of these continuity conditions guarantees a smooth track trajectory without the appearance of large impulses in the transitions between segments [3].



Figure 2.1: The spinning rollercoaster lay-out that has been created in NoLimits 2

## 2.2 Multibody Model of Rollercoaster Vehicle

A rollercoaster vehicle generally consists of a collection of bodies whose motion is prescribed by the track trajectory. These bodies are considered as rigid bodies due to their high structural stiffness, while kinematic joints dictate their relative motion. The multibody model of the rollercoaster vehicle depicted in Figure 2.2 is assembled using 38 rigid bodies in total. The elliptically shaped gondola accommodates two pairs of seats that face in opposite direction, including the restraints that secure the riders in their seats. The seats and restraints are under all circumstances completely fixed to the gondola geometry according to a predefined configuration, and hence the mentioned components can be regarded as a single body. This reduces the total number of rigid bodies within the multibody model to 32, including 24 wheels. A pivot axle connects the gondola to the main chassis beam, which is modelled by means of a revolute joint that constrains all degrees of freedom except for the rotational motion of the gondola. The distance between the rotational axis and the centre of gravity of an empty gondola equals ten millimeters. The front and rear axle are attached to the main chassis beam by respectively a spherical and a rotational joint. The spherical joint imposes three kinematic constraints between the front axle and the chassis main beam to control their relative motion, allowing only three relative rotations. Thanks to the spherical joint the front axle could be regarded as a steering axle, which is often referred to as the so-called zero car [4]. The axles are on both sides attached to wheel bogies by means of a revolute joint, and each wheel bogie holds at least three different types of wheels. The running wheels carry the main load of the vehicle, whereas the side wheels guide the vehicle on the track. The upstop wheels keep the vehicle on the track during negative vertical accelerations [5]. The motion of the rollercoaster vehicle along the track is in reality facilitated by the rotational motion of the wheels. However, as will be elucidated in the next section, the point-on-curve constraints used in the Simulink model do not require the wheels to accommodate a rotational degree of freedom. Four of the eight running wheels are assigned a prismatic joint with respect to

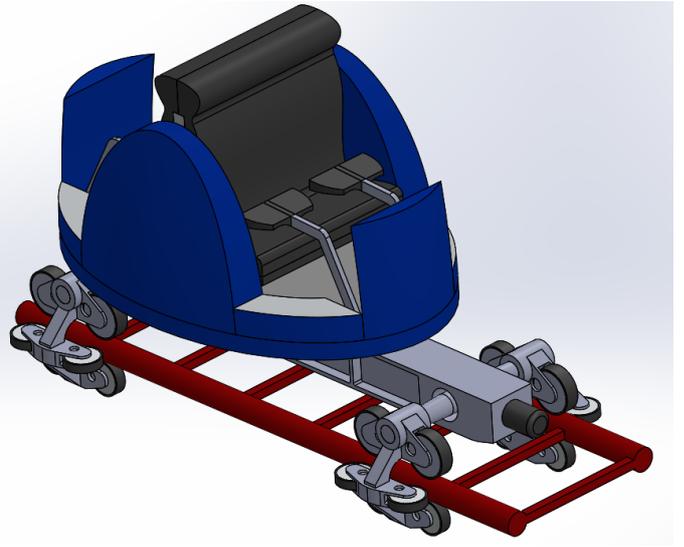


Figure 2.2: Geometry of the vehicle that will traverse the spinning rollercoaster track lay-out

|    | <b>Rigid Bodies</b>                                   | <b>Mass [kg]</b> |
|----|---|------------------|
| 1  | Gondola (including seats and restraints)              | 532.2            |
| 2  | Main chassis beam                                     | 955.3            |
| 3  | Front axle  | 88.6             |
| 4  | Rear axle   | 112.4            |
| 5  | Front-left bogie (including wheels)                   | 135.7            |
| 6  | Front-right bogie (including guide and upstop wheels) | 122.9            |
| 7  | First front-right running wheel                       | 6.4              |
| 8  | Second front-right running wheel                      | 6.4              |
| 9  | Rear-left bogie (including guide and upstop wheels)   | 122.9            |
| 10 | First rear-left running wheel                         | 6.4              |
| 11 | Second rear-left running wheel                        | 6.4              |
| 12 | Rear-right bogie (including wheels)                   | 135.7            |

Table 2.1: Mass properties of each rigid body

|    | <b>Kinematic Constraint</b> | <b>Body <math>i</math></b> | <b>Body <math>j</math></b> | <b><math>N_c</math></b> |
|----|-----------------------------|----------------------------|----------------------------|-------------------------|
| 1  | Revolute joint              | 1                          | 2                          | 5                       |
| 2  | Spherical joint             | 2                          | 3                          | 3                       |
| 3  | Revolute joint              | 2                          | 4                          | 5                       |
| 4  | Revolute joint              | 3                          | 5                          | 5                       |
| 5  | Revolute joint              | 3                          | 6                          | 5                       |
| 6  | Revolute joint              | 4                          | 9                          | 5                       |
| 7  | Revolute joint              | 4                          | 12                         | 5                       |
| 8  | Prismatic joint             | 6                          | 7                          | 5                       |
| 9  | Prismatic joint             | 6                          | 8                          | 5                       |
| 10 | Prismatic joint             | 9                          | 10                         | 5                       |
| 11 | Prismatic joint             | 9                          | 11                         | 5                       |

Table 2.2: The connected bodies and number of constraints for each type of kinematic constraint used in the multibody model of the rollercoaster vehicle

the accompanying bogie, whereas all remaining wheels are completely fixed to the wheel carrier. Hence, the wheel carrier and the accompanying wheels can be considered as a single body for two of the four bogies. The four running wheels are regarded as separate bodies for the other two wheel carriers. This reduces the number of distinctively moving bodies within the multibody model to only twelve. The mass of each rigid body is presented in Table 2.1, whereas the types of kinematic constraints used to assemble the rollercoaster vehicle model are listed in Table 2.2. The total mass of the rollercoaster vehicle, excluding passengers, is equal to 2231.5 kilograms.

## 2.3 Point-on-Curve Constraints

### 2.3.1 Constraint Configuration

No degrees of freedom have been assigned to the running wheels at the front-left and rear-right bogie. Hence, the motion of these four running wheels is completely constrained to the movement of the accompanying wheel carrier. A follower frame is added underneath each running wheel at some distance from the wheel centre. Since this distance is equal to the distance between the wheel centre and the heartline of the track, the follower frames are aligned exactly with the track heartline. The four running wheels cannot move with respect to the corresponding wheel carriers, which implies that the motion of the follower frames is also constrained to the movement of the accompanying bogie. On the other hand, the running wheels at the front-right and rear-left bogie are constrained by means of a prismatic joint that only facilitates sideways translation. A follower frame is attached to each running wheel in a similar manner as for the other four running wheels, so each follower frame is aligned with the track heartline. These four follower frames are capable of moving sideways since the motion of the running wheels is not constrained in this direction. The point-on-curve constraint is a kinematic constraint that allows the origin of a follower frame to translate only along a prescribed curve. The follower frame is free to rotate depending on other constraints in the model [6]. Hence, each point-on-curve constraint introduces two kinematic constraints to the model. The origins of the eight follower frames are only allowed to translate along the heartline of respectively the left and right rail. Two pairs of running wheels and the accompanying follower frames cannot move with respect to the front-left and rear-right wheel carrier. These follower frames cannot move with respect to each other either, and their origins are constrained to the track heartline. Therefore, the orientation of the front and rear axle is directly dictated by the orientation of respectively the front-left and the rear-right wheel carrier. The prismatic joints between the running wheels and the other wheel carriers allow sideways translation of the corresponding follower frames with respect to the track. Hence, these follower frames are capable of compensating for any track irregularities or variable track width, as depicted in Figure 2.3.

The configuration of each body is uniquely described by six independent generalized coordinates  $N_q$  in three-dimensional space. The number of bodies  $N_b$  within the multibody model of the rollercoaster vehicle equals twelve, so the total number of generalized coordinates equals:

$$N_q = 6 \cdot N_b = 6 \cdot 12 = 72 \tag{2.1}$$

The kinematic joints imply relations between the generalized coordinates, and hence the generalized coordinates are generally not all independent. For each dependent generalized coordinate, there is a kinematic constraint equation that belongs to a certain kinematic constraint. Summation of the kinematic constraints listed in Table 2.2 yields a total number of kinematic constraints equal to 53. The rollercoaster vehicle is constrained to the left and right track by means of eight point-on-curve constraints with two kinematic constraints each. Hence, the total number of kinematic constraints that can be found on the rollercoaster model equals 69. The number of independent generalized coordinates equals the number of the degrees-of-freedom  $N_{DOF}$  of a system. This parameter is related to the number of kinematic constraints and generalized coordinates, as shown in Equation 2.2 for the rollercoaster model.

$$N_{DOF} = N_q - N_c = 72 - 69 = 3 \tag{2.2}$$

The three degrees-of-freedom on the rollercoaster model are the rotation of the gondola and the translation along respectively the left and right rail.

### 2.3.2 Track Control Point Increment

The rollercoaster track is defined using piece-wise cubic interpolation to parameterise the geometry of the rails. However, a major disadvantage of the cubic spline formulation is that it leads to undesired oscillations in the track model. These track oscillations are sensed by the follower frames at the front-right and rear-left wheel carriers, since these follower frames can translate in  $z$ -direction (Figure 2.3) relative to their accompanying bogies. The  $z$ -translation of the follower frames at the front-left and rear-right bogie is constrained, which implies that the  $z$ -translation sensed at the other four follower frames could be regarded as small variations in the track width. Table 2.3 presents the  $z$ -translation measured at the four follower

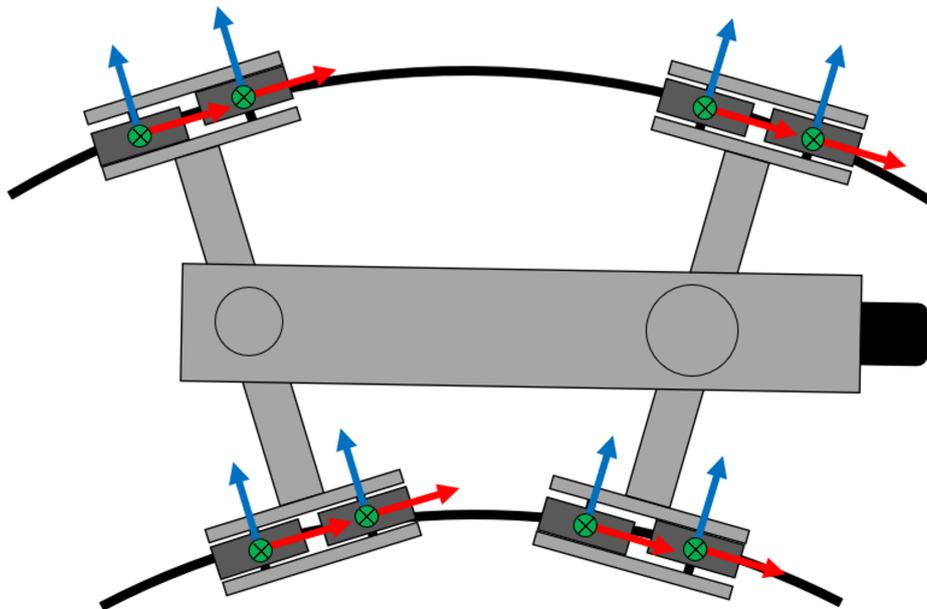


Figure 2.3: Configuration of the running wheels and follower frames with respect to a curved section of track, where the  $x$ -axis is depicted in red, the  $y$ -axis in green, and the  $z$ -axis in blue

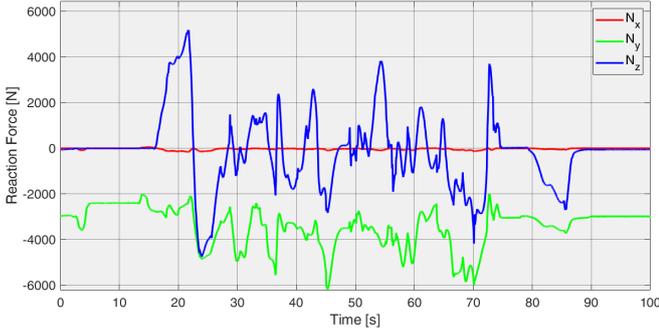
frames for four different levels of control point discretization. These undesired wiggles in the track should in principle be avoided, since the track perturbations could lead to the modelling of higher reaction forces between the vehicle wheelsets and the track. However, it can be deduced from Table 2.3 that the track perturbations are typically small, with a maximum  $z$ -translation equal to approximately 0.31 millimeters. For the considered control point increments it is shown that a smaller distance between between the control points does not necessarily lead to a noticeably smoother track. Although the difference between the resultant  $z$ -translations is negligible for most levels of control point discretization, it is shown in Table 2.3 that the smallest control point increment is not in alignment with the lowest  $z$ -translation values. Hence, the oscillations cannot be reduced by choosing a smaller control point increment. Using smaller distances between the control points on the track centre line leads to a substantial increase in the computational time though. The use of a control point increment equal to one meter is preferred, since it can be deduced from Table 2.3 that this control point increment yields the most favourable ratio between computational time and the level of track oscillation. Figure D.1 in Appendix D shows the track oscillations sensed at the front-right and rear-left wheel carrier for a control point increment  $\Delta s$  of one meter.

| $\Delta s$ [m] | $\Delta z$ F-R Bogie<br>Frame 1 [m] | $\Delta z$ F-R Bogie<br>Frame 2 [m] | $\Delta z$ R-L Bogie<br>Frame 1 [m] | $\Delta z$ R-L Bogie<br>Frame 2 [m] | CPU time [s] |
|----------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------|
| 0.25           | $1.9915 \cdot 10^{-4}$              | $1.7384 \cdot 10^{-4}$              | $1.8565 \cdot 10^{-4}$              | $1.7981 \cdot 10^{-4}$              | 726          |
| 0.50           | $1.9108 \cdot 10^{-4}$              | $1.5700 \cdot 10^{-4}$              | $1.8041 \cdot 10^{-4}$              | $1.6197 \cdot 10^{-4}$              | 448          |
| 1.00           | $1.9532 \cdot 10^{-4}$              | $1.3163 \cdot 10^{-4}$              | $1.7653 \cdot 10^{-4}$              | $1.3592 \cdot 10^{-4}$              | 265          |
| 2.00           | $1.9717 \cdot 10^{-4}$              | $3.0447 \cdot 10^{-4}$              | $1.9411 \cdot 10^{-4}$              | $3.1174 \cdot 10^{-4}$              | 183          |

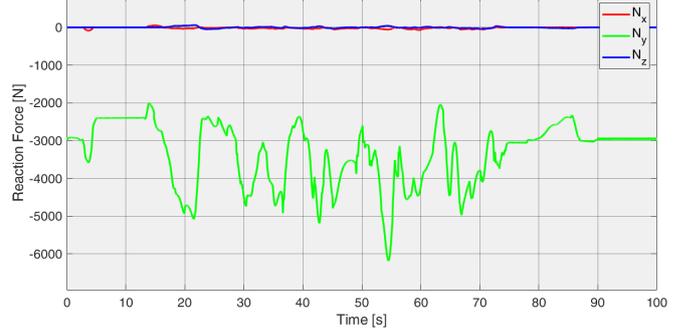
Table 2.3: Extent of track oscillations for various control point increments at the front-right (F-R) and rear-left (R-L) wheel carriers, including the CPU time

### 2.3.3 Reaction Forces between Track and Wheels

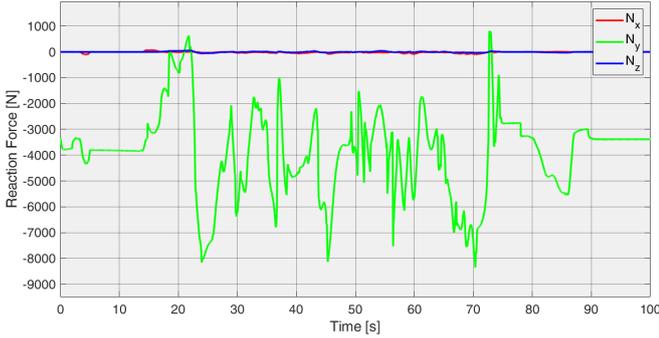
Four of the eight running wheels are connected to the accompanying bogies by means of prismatic joints, so the rollercoaster vehicle is capable of coping with track irregularities. However, a disadvantage of the prismatic joints in this context is their inability to sense reaction forces that are exerted by the track on the wheels in  $z$ -direction (Figure 2.3). The reaction forces between the track and wheels are presented in Figure 2.4, which result from a simulation of the rollercoaster vehicle traversing the entire track lay-out starting at the station. It is physically correct that the reaction force in  $x$ -direction equals approximately zero for all four bogies, since the point-on-curve constraints allow free translation of the follower frames along the left and right rail. However, the reaction forces in  $z$ -direction for the front-right and rear-left bogies that feature a prismatic joint are equal to zero as well, which is physically incorrect. It appears that the reaction forces for the front-right and rear-left bogie are sensed at the point-on-curve constraints of respectively the front-left and rear-right wheel carrier. The reaction forces are registered at the front-left and rear-right wheel carrier in the positive and negative  $z$ -direction. This would erroneously suggest that the guide wheels are always in contact with the track when the rollercoaster vehicle travels for instance through the two turns directly after the lift. The negative reaction forces in  $z$ -direction in Figure 2.4 are not in accordance with the actual physical set-up of the guide wheels with respect to the track, since a guide wheel cannot pull on a rail. Instead, a guide wheel simply comes loose from the track without any reaction forces being exerted on the track. Hence, the negative reaction forces in  $z$ -direction for the front-left and rear-right bogie actually account for the missing reaction forces at respectively the front-right and rear-left wheel carrier. The issue of the missing reaction forces is circumvented by transferring the reaction forces from the wheel carriers with fully constrained follower frames to the bogies with prismatic joints. The main Simulink scheme in Figure A.1 in the appendices shows that the reaction forces in  $z$ -direction are transferred from the front-left bogie to the front-right bogie and from the rear-right bogie to the rear-left bogie.



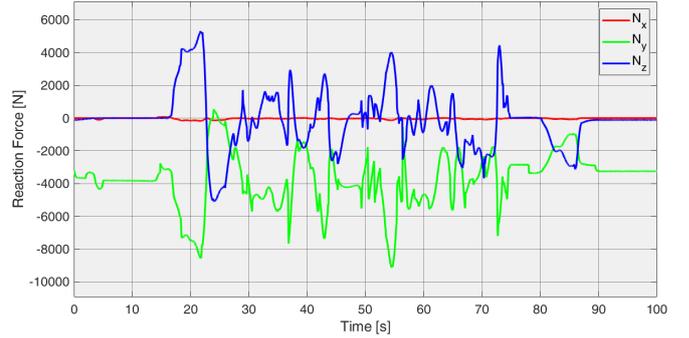
(a) Reaction forces between track and front-left bogie



(b) Reaction forces between track and front-right bogie



(c) Reaction forces between track and rear-left bogie



(d) Reaction forces between track and rear-right bogie

Figure 2.4: Reaction forces between track and wheels specified for each wheel carrier

## 2.4 Modelling Rolling and Bearing Friction

Now that the reaction forces between the track and wheels have been modelled appropriately, they can be used to compute the rolling and bearing frictional forces. The reaction forces from the previous time-step are used to determine the frictional values at the next time-step.

### 2.4.1 Rolling Frictional Force

As described elaborately in Appendix E, the Hertzian elliptic contact model is used to model rolling friction. This model is capable of describing contact between two elastic cylindrical bodies, where the contact area between the bodies is shaped elliptically. The rate of deformation is unequal to the rate of recovery during rolling contact due to elastic hysteresis. This internal friction causes the work that is provided to the contact material to be partly transformed into heat during the deformation process. Hence, part of the work is not regained, and a torque is experienced as a resistance to rolling. A friction force  $P$  can be defined by equating the hysteresis energy loss to the distance travelled in rolling motion  $s$ , as shown in Equation 2.3.

$$P \cdot s = \alpha_r \cdot W \quad (2.3)$$

where  $\alpha_r$  is the hysteresis loss factor for a rolling motion that defines the percentage of work  $W$  spent on deforming the contact bodies that is not regained after relaxation. For the Hertzian elliptic contact model, Equation 2.3 can be rewritten to the expression shown in Equation 2.4.

$$P = \alpha_r \frac{3}{16} \frac{N}{R} b = \mu N \quad \text{with} \quad \mu = \left( \alpha_r \frac{3}{16} \frac{b}{R} \right) \quad (2.4)$$

where  $R$  is the undeformed outer radius of the rolling body,  $N$  defines the normal forces on the bodies in contact,  $b$  represents the longest semi-axis of the elliptic contact area, and  $\mu$  symbolizes the friction

coefficient. Substituting the correct values into Equation 2.4 and solving this expression will yield the value of the frictional force and rolling friction coefficient.

### 2.4.2 Bearing Frictional Moment

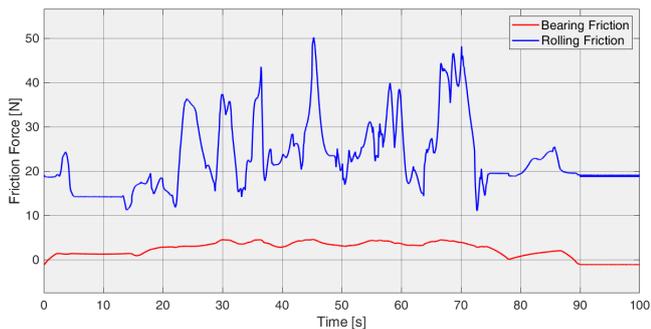
A smooth motion between separate components is facilitated by the use of bearings, which are characterized by a non-constant frictional moment that depends on certain tribological phenomena. Although the frictional moment induced by the bearings is typically small in comparison to other sources of resistance such as rolling friction or air drag, its effect is assigned a sufficient level of significance for it to be included in the model. The frictional moment induced by the bearings is modelled at two distinct contact areas, namely between the wheels and the accompanying hubs and at the revolute joint between the gondola and the main chassis beam. The frictional moment is not modelled for the spherical and revolute joint between respectively the chassis main beam and the front and rear axle, since the degree of relative motion between these components is small in comparison to the other components. The SKF model is used for calculating the frictional moment induced by the bearings, in which the total frictional moment is calculated according to Equation 2.5.

$$M = M_{rr} + M_{sl} \quad (2.5)$$

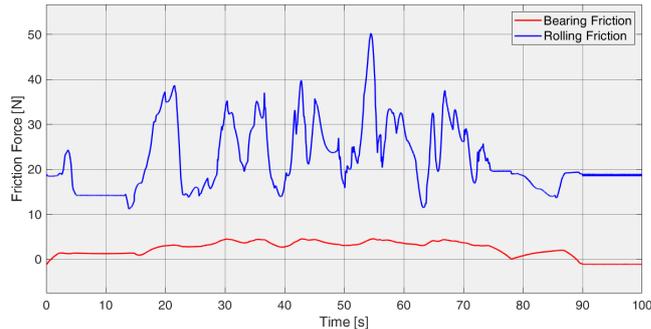
where  $M$  is the total frictional moment,  $M_{rr}$  is the rolling frictional moment, and  $M_{sl}$  is the sliding frictional moment. Please note that the the frictional moment of the seals  $M_{seal}$  equals zero for the types of bearings considered, and that the frictional moment of drag losses  $M_{drag}$  is neglected. The rolling and sliding frictional moments can be calculated according to the expressions presented in Appendix F.

### 2.4.3 Resultant Rolling and Bearing Friction

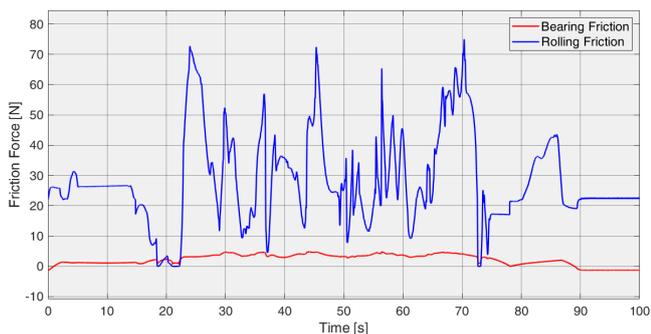
The rolling and bearing frictional forces are determined at each time-step during the simulation for each individual wheel. The resultant frictional values are depicted in Figure 2.5 and 2.6 for respectively one of the two running and guide wheels on each of the four wheel carriers. These frictional forces result from a simulation with a fixed time-step equal to 0.01 second. The duration of the simulation (100 s) equals the time that it takes for the rollercoaster vehicle to complete a single cycle when released from the station at  $t = 0$  s. An eight-order Dormand-Prince formula was used to compute the model state at the next time-step as an explicit function of the current model state. Please note that the frictional values found for the upstop wheels are not shown, since the rolling and bearing frictional forces equal zero for the upstop wheels on three of the four wheel carriers. The frictional forces are only unequal to zero at the upstop wheels on the rear-right bogie, but these values are orders of magnitude lower than the frictional forces found at the other wheels. Proof for the correct modelling of reaction forces between the track and wheels is provided by the graphs in Figure 2.5 and 2.6. The rollercoaster vehicle traverses through the first almost flat turn directly after the lifthill between  $t = 16$  s and  $t = 22$  s. As anticipated, relatively large frictional forces can be found at the running wheels for the bogies on the right-hand side of the rollercoaster car, whereas the frictional values are highest at the guide wheels on the left-hand side bogies during this time interval. An opposite pattern can be recognized in the graphs between approximately  $t = 22$  s and  $t = 27$  s when the rollercoaster vehicle traverses through the next barely banked turn. High frictional forces can namely be found at the running wheels on the left-hand side bogies and at the guide wheels on the right-hand side bogies of the rollercoaster car. Additionally, the frictional values at the guide and upstop wheels are correctly set equal to zero when the guide wheels are not in contact with the track. This also implies that values equal to zero can be omitted from the calculation of the average rolling friction coefficient for the guide wheels, which yields a resultant coefficient of 0.0039. On the other hand, the average rolling friction coefficient for the running wheels equals 0.0066. Based on these values it can be deduced that the the overall rolling friction coefficient is equal to 0.0053.



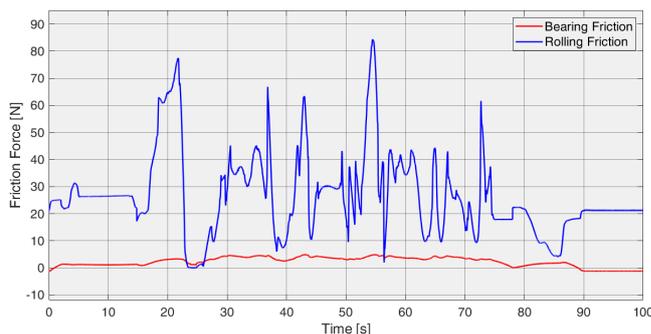
(a) Friction at a running wheel on the front-left bogie



(b) Friction at a running wheel on the front-right bogie

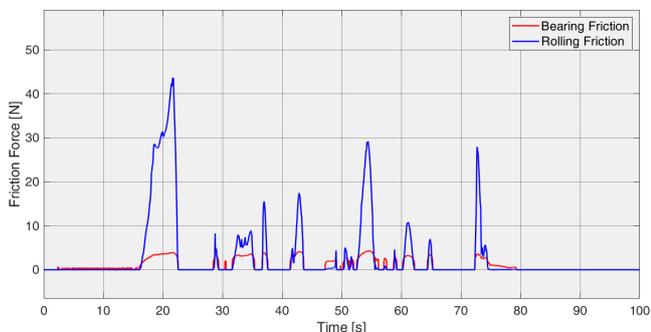


(c) Friction at a running wheel on the rear-left bogie

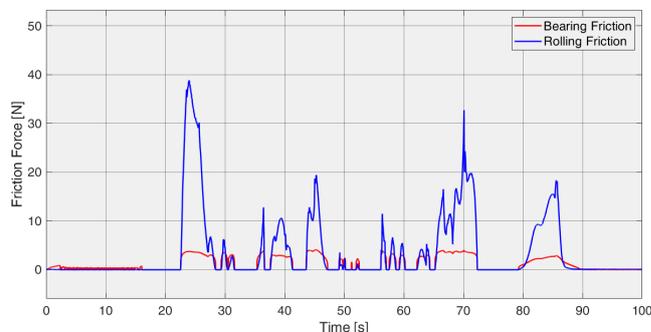


(d) Friction at a running wheel on the rear-right bogie

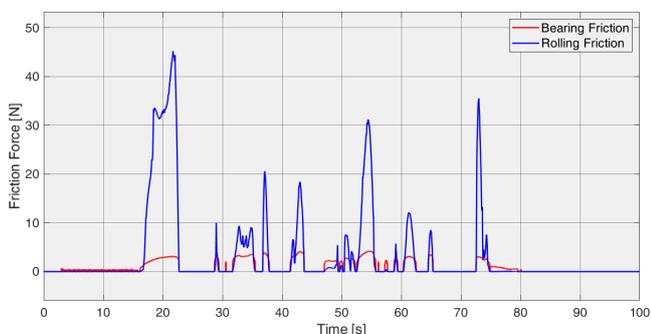
Figure 2.5: Rolling and bearing frictional force for a running wheel on each of the four bogies



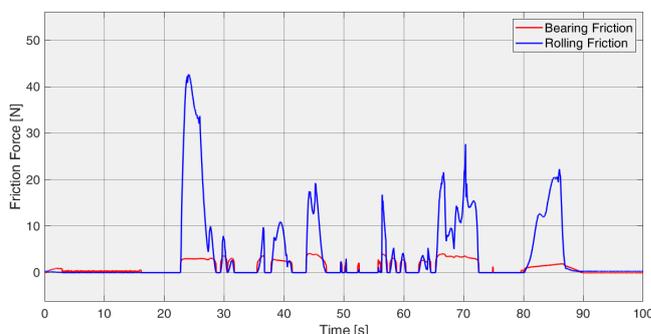
(a) Friction at a guide wheel on the front-left bogie



(b) Friction at a guide wheel on the front-right bogie



(c) Friction at a guide wheel on the rear-left bogie



(d) Friction at a guide wheel on the rear-right bogie

Figure 2.6: Rolling and bearing frictional force for a guide wheel on each of the four bogies

## 2.5 Modelling Air Resistance

A moving fluid such as an air flow exerts normal pressure forces and tangential shear forces on the surface of a body immersed in it. Both of these forces have components in the direction of flow, and thus the drag force is due to the combined effect of pressure and wall shear forces in the flow direction. The drag force  $F_D$  depends on the density  $\rho$  of the fluid, the upstream velocity  $v$ , and the size, shape and orientation of the body. The drag characteristics of a body is represented by the dimensionless drag coefficient  $C_D$ . The drag force can be calculated according to Equation 2.6.

$$F_D = \frac{1}{2}\rho C_D v^2 A_{front} \quad (2.6)$$

### 2.5.1 Frontal Surface

In Equation 2.6, the parameter  $A_{front}$  refers to the frontal surface of the body. For bodies that tend to block the flow, such as the rollercoaster vehicle depicted in Figure 2.2, the frontal surface is the area projected on a plane normal to the direction of flow. The orientations of the chassis, pivot axle, bogies and wheels remain constant with respect to the air flow during the ride due to their direct alignment with the track. The drag force that is exerted on these components can therefore be determined using a constant frontal surface of  $0.386 \text{ m}^2$  and presumed drag coefficient of 0.8. On the other hand, the orientation of the gondola continuously changes with respect to the direction of motion of the train during the ride, which implies that the gondola frontal surface should be regarded as a variable instead of a constant. The gondola frontal surface is shown as a function of the gondola rotation in Figure G.1 in Appendix G.2, while tabulated values can be found in Table G.1 in Appendix G.1. Six piece-wise cubic spline segments were used in Figure G.1 to interpolate between the seven gondola orientations for which the frontal surface has been manually determined. A detailed explanation on the topic of cubic spline interpolation can be found in Appendix G.2. Please note that it suffices to consider a ninety degree range instead of a full rotation when determining the gondola frontal surface, since the geometry of the gondola is characterized by two planes of symmetry. Hence, the interpolation functions between  $90^\circ$  and  $180^\circ$  in Figure G.1 are simply the mirrored equivalents of the cubic splines between  $0^\circ$  and  $90^\circ$ , while the interpolation functions between  $180^\circ$  and  $360^\circ$  would be identical to the interpolation splines depicted in Figure G.1.

### 2.5.2 Gondola Drag Coefficient

Numerous numerical simulations have been performed in Ansys Discovery AIM 19.2 (from this point on referred to as Ansys) to determine the drag force that is exerted by the air flow on the gondola. Aside from omitting any redundant details from the gondola geometry, the symmetry planes of the gondola were used to reduce the computational time of the simulations. The air flow around the simplified and halved gondola geometry can subsequently be modelled by creating an external body around the gondola, while the gondola itself is excluded from the physics region with boundary layer properties assigned to its surface. The frontal surface inlet of the enclosure is assigned with a certain upstream velocity, while the rear surface outlet is assigned a gauge static pressure equal to zero. A major disadvantage of halving the gondola geometry is the inability to model air flow around the gondola for orientations unequal to  $0^\circ$  or  $90^\circ$ . At these orientations the symmetry plane of the gondola is namely not aligned with the direction of the air flow. Considering the full geometry of the gondola during the numerical simulation comes at the cost of a drastically increased computational time though. Therefore, the air flow is only modelled around a gondola that has been halved at either two symmetry planes, which corresponds to gondola rotations of respectively  $0^\circ$  and  $90^\circ$ . Appendix G.3 provides a detailed description on the boundary conditions that were imposed on the model.

Fourteen simulations have been performed at two different gondola orientations and seven different upstream velocities, under the presumption of steady-state conditions and incompressible flow. Appendices G.4 and G.5 provide an extensive description of the reasoning behind the selected upstream velocities at which the air flows were modelled. Once a simulation is completed, the drag force exerted on the gondola

surface can be extracted from Ansys. The corresponding drag coefficient can subsequently be calculated by rewriting Equation 2.6 into the expression shown in Equation 2.7.

$$C_D = \frac{2 \cdot F_D}{\rho \cdot v^2 \cdot A_{front}} \quad (2.7)$$

The resultant drag coefficients and forces have been tabulated in Table 2.4, while Figure 2.7 provides a visual representation of the drag coefficient as a function of the upstream velocity and gondola orientation. It can be deduced from this figure that the drag coefficients are substantially higher for a  $90^\circ$  gondola rotation in comparison to a  $0^\circ$  gondola rotation. The total drag can almost entirely be assigned to the occurrence of pressure drag, where the majority of the pressure drag is presumably caused by contributions from the pressure difference between the front and rear of the gondola. When the air stream separates from the gondola surface, it forms a separated region between the gondola and the air stream with a reduced velocity relative to the upstream velocity. The larger the separated region, the larger the pressure drag is [7]. The velocity contours are shown in Appendix G.10 for the performed simulations. From these figures, it can be deduced that the separated region is typically larger for a  $90^\circ$  gondola rotation than for  $0^\circ$  rotation, which explains the higher drag force and drag coefficient for the former gondola orientation. The residuals of the simulations are presented and discussed in respectively Appendix G.9 and G.7.

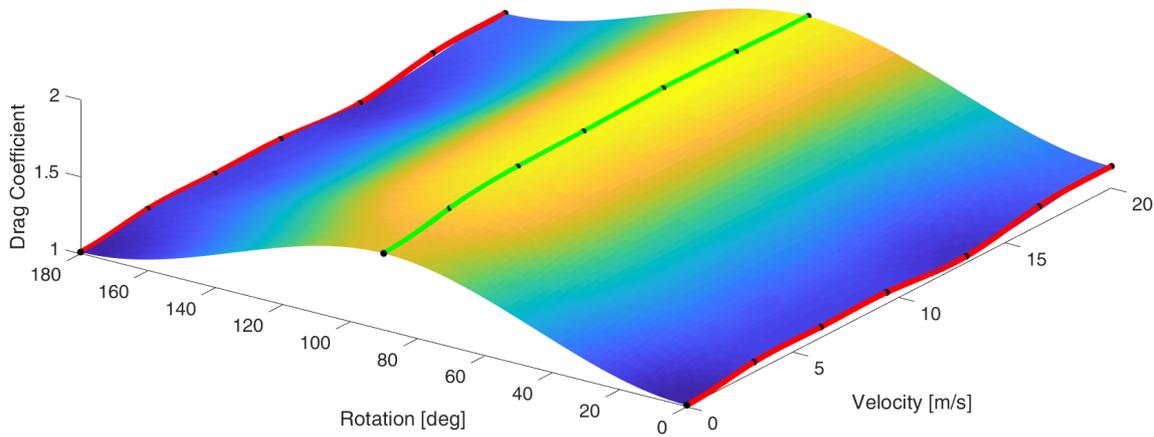


Figure 2.7: The drag coefficient as a function of the upstream velocity and the gondola rotation

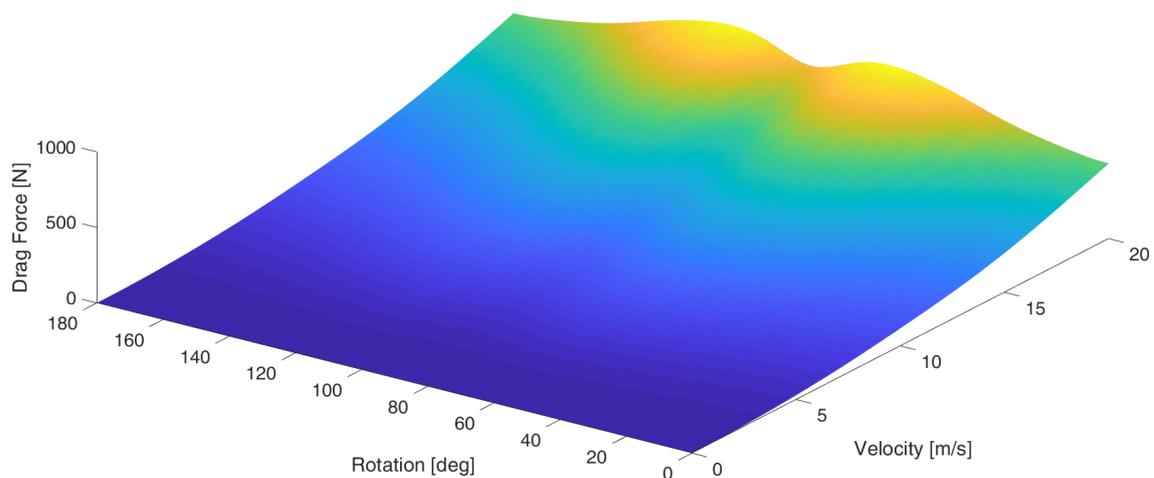


Figure 2.8: The drag force as a function of the upstream velocity and the gondola rotation

### 2.5.3 Gondola Drag Force

Calculating the frontal surface and drag coefficient according to the functions depicted in respectively Figure G.1 and 2.7 and substituting these results in Equation 2.6 yields the resultant gondola drag force, which has been plotted as a function of the upstream velocity and gondola rotation in Figure 2.8. The function `GondolaDragForce.m` in Appendix B describes the procedure of determining the drag force based on the upstream velocity and the rotation of the gondola. This function first determines the drag coefficient at a gondola rotation of respectively  $0^\circ$  and  $90^\circ$ . Subsequently, a cubic spline is formulated to interpolate between these values, so the drag coefficient can be determined at a specified gondola rotation angle. The slopes at the outer ends of this spline are equal to zero to ensure a smooth transition between the determined drag coefficient values at  $0^\circ$  and  $180^\circ$ .

### 2.5.4 Resultant Drag Forces

The resultant drag forces acting on the chassis and the gondola are presented in respectively Figure 2.9 and 2.10. The influence of the rotating gondola on the drag force can clearly be deduced from a comparison between these two figures. While the constant drag coefficient and frontal surface lead to a relatively smooth function for the drag force acting on the chassis, the rotational effect of the gondola increases the number of fluctuations in the gondola drag force plot. The gondola drag force is modelled as an external force acting on the gondola pivot point on the main chassis beam. Exerting the drag force directly on the gondola would namely be a rather complex procedure, since the orientations of the reference frames on the gondola continuously change along with the gondola rotation. On the other hand, the orientation of the chassis main beam is aligned with the direction of the air flow during the entire ride. Hence, the chassis drag force is modelled as an external force acting on the frontal surface of the main chassis beam.

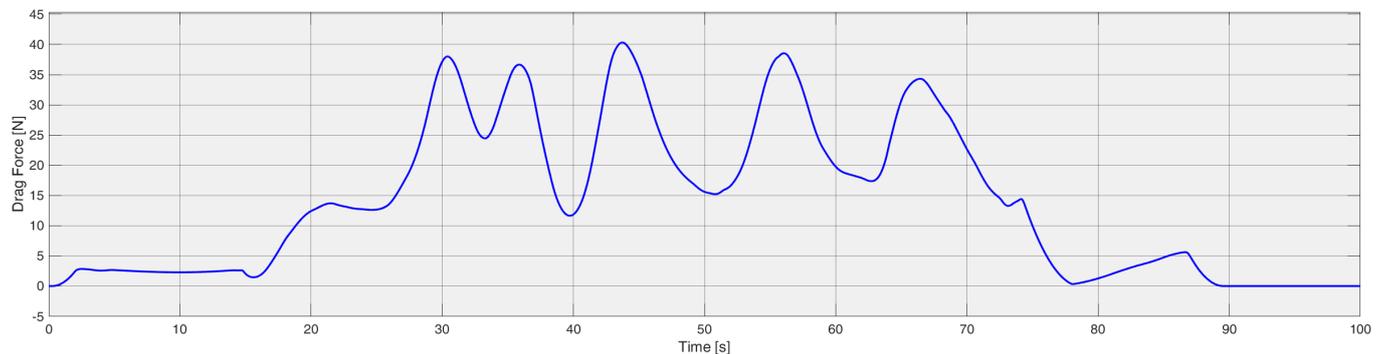


Figure 2.9: The resultant drag force acting on the chassis and bogies of the spinning rollercoaster vehicle

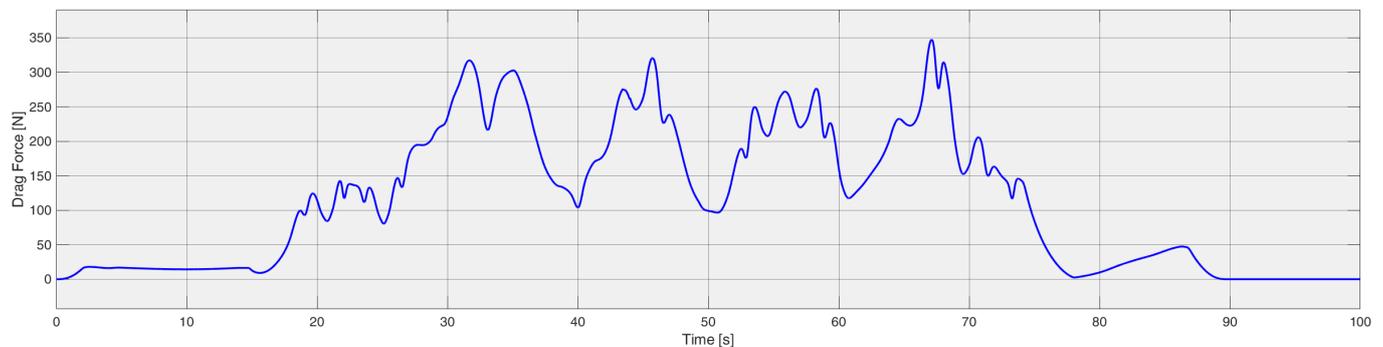


Figure 2.10: The resultant drag force acting on the gondola of the spinning rollercoaster vehicle

| $v$ [m/s] | $\alpha$   | $F_d$ [N] | $C_d$ |
|-----------|------------|-----------|-------|
| 0         | $0^\circ$  | 1.1016    | 1.011 |
| 0         | $90^\circ$ | 1.4985    | 1.499 |
| 3.15      | $0^\circ$  | 11.526    | 1.066 |
| 3.15      | $90^\circ$ | 15.581    | 1.571 |
| 6.3       | $0^\circ$  | 46.266    | 1.070 |
| 6.3       | $90^\circ$ | 63.975    | 1.613 |
| 9.4       | $0^\circ$  | 103.47    | 1.075 |
| 9.4       | $90^\circ$ | 143.17    | 1.621 |
| 13.15     | $0^\circ$  | 196.5     | 1.043 |
| 13.15     | $90^\circ$ | 282.72    | 1.636 |
| 16.575    | $0^\circ$  | 336.52    | 1.124 |
| 16.575    | $90^\circ$ | 447.35    | 1.629 |
| 20        | $0^\circ$  | 497.75    | 1.142 |
| 20        | $90^\circ$ | 647.21    | 1.619 |

Table 2.4: Drag forces and drag coefficients resulting from simulations in Ansys at various gondola orientations and upstream velocities

## 2.6 Motion Control

To create a kinematic model that is as realistic as possible, the Simulink model is expanded with the block schemes depicted in Figures A.15 till A.22. These schemes control the motion of the vehicle and the gondola on all track sections that do not belong to the gravity run. This implies that the motion of the vehicle is controlled between the start of the brake run and the end of the lifthill. Since an extensive explanation of each individual control scheme can be found in Appendix H, this section will explain the control schemes only in a general sense. The distance covered along the track by the vehicle is used to determine at which track section the vehicle is located at each time step. The distance ranges corresponding to each track section have been determined in advance, of which the results are presented in Table 2.5. For instance, the brake force is only applied to the vehicle when the distance covered by the vehicle falls within the range corresponding to the brake section. The translational motion control of the vehicle is generally explained in the following paragraph, while the subsequent paragraph elaborates on the rotational motion control of the gondola.

| Distance [m]       | Section Type  |
|--------------------|---------------|
| $0 \leq s < 5$     | Station exit  |
| $5 \leq s < 50$    | Lifthill      |
| $50 \leq s < 684$  | Gravity run   |
| $684 \leq s < 707$ | Brake-section |
| $707 \leq s < 737$ | Turnaround    |
| $737 \leq s < 746$ | Station entry |

Table 2.5: The distance ranges and corresponding section type for the track lay-out of interest

Upon leaving the station at the start of the simulation, the velocity of the vehicle should follow a reference profile with a slope of  $2 \text{ m/s}^2$ . As depicted in Figure 2.11, the vehicle should enter the lifthill with a velocity equal to the chain speed, which has been set equal to  $4 \text{ m/s}$ . Hence, the velocity reference profile that should be followed by the vehicle on the lifthill is a constant line at a magnitude equal to the chain speed. When the vehicle is released from the lifthill, the translational control of the vehicle is terminated. Until the vehicle reaches the brake section, its velocity results from a kinematic analysis of the model performed at each time step, without the application of any external forces that should either decelerate or accelerate the vehicle according to a prescribed velocity reference profile. At two instances on the right-half plane of Figure 2.11, a substantial braking force is applied to the vehicle to reduce its velocity. At the brake section, a reference profile with a slope equal to the desired deceleration of  $-2 \text{ m/s}^2$  should slow the vehicle down to a maximum velocity of  $1 \text{ m/s}$  upon leaving the brake section. Due to a slight decline on the subsequent turnaround, the vehicle again gradually accelerates before entering the station. However, at

the station entry the vehicle is braked according to a reference profile with a similar deceleration as on the brake section. In the station the vehicle comes to a standstill at approximately the same location as the initial position when it left the station. It can be deduced from Figure 2.11 that the vehicle does not move for the remainder of the simulation once it has come to a standstill.

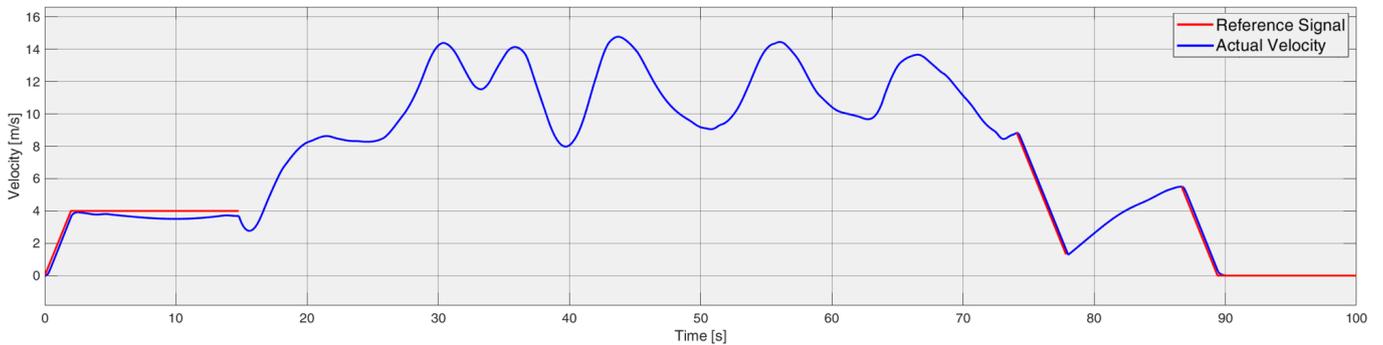


Figure 2.11: The actual and reference velocity signals for a vehicle with a fully-loaded gondola

The vehicle leaves the station at the start of the simulation with an initial gondola rotation angle equal to zero. By prescribing an angular velocity reference profile equal to zero, this initial gondola orientation should be maintained until the vehicle leaves the lifthill. Figure 2.12 and 2.13a show that on the lifthill the deviation of the actual gondola angular velocity from the reference profile is negligible. The rotational control of the gondola is terminated at the top of the lifthill, and the gondola is free to spin during the subsequent gravity run. Upon reaching the brake section, the angular velocity of the gondola is reduced to zero according to a reference profile with a delicately defined slope. The reference profile namely ensures that the gondola angular velocity and rotation angle simultaneously equal zero, so the vehicle leaves the brake section with a non-moving gondola that is orientated approximately at a zero rotation angle. At the subsequent turnaround between the brake section and the station, a reference profile equal to zero is defined. Hence, the rotation angle upon leaving the brake section should be maintained until the vehicle reaches the station entry, as shown in Figure 2.13b. Since it cannot be guaranteed that the vehicle enters the station with a gondola rotation angle equal to zero, a correction torque is required to rotate the gondola to its initial orientation. The effect of this correction torque on the angular velocity can be seen on the right-hand side of Figure 2.12 by means of a gradually increasing and decreasing angular velocity. The final gondola orientation is characterized by a rotation angle practically equal to zero for all passenger occupancy configurations (Figure 2.15). Hence, upon termination of the simulation the vehicle configuration in the station is almost identical to the initial configuration of the vehicle at the start of the simulation.

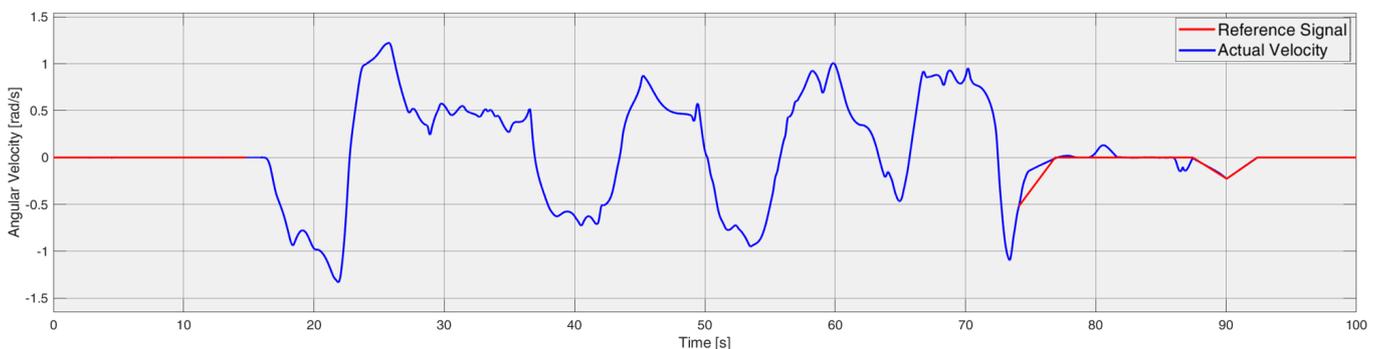
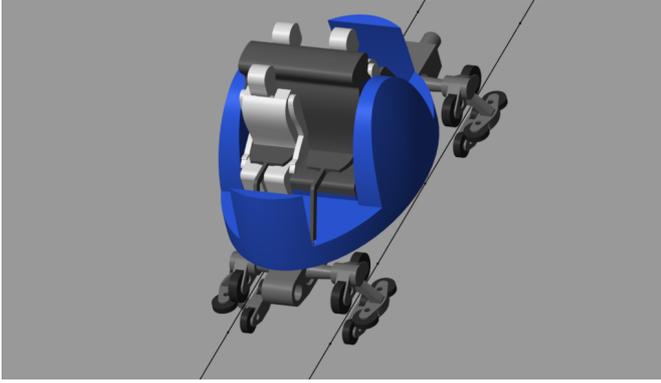
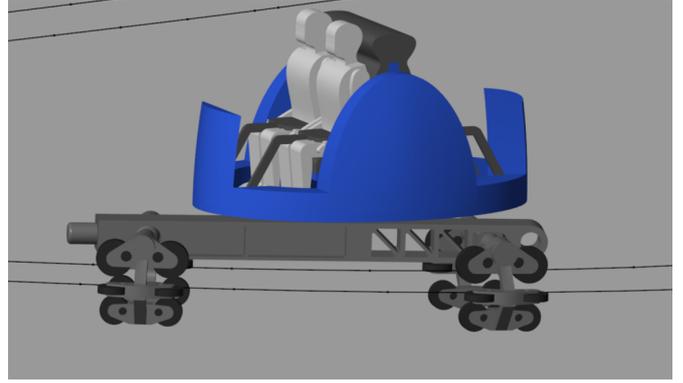


Figure 2.12: The actual and reference angular velocity signals for a vehicle with a fully-loaded gondola



(a) The gondola is kept at its initial orientation on the lifthill despite a mass imbalance of the gondola



(b) The rotational motion of the gondola is controlled at the turnaround between the brake section and station

Figure 2.13: Rotational motion control of the gondola

## 2.7 Tolerable Passenger Accelerations

The accelerations experienced by the passengers during a ride on the spinning rollercoaster are compared to the standardized norms for tolerable passenger accelerations. The linear accelerations resulting from the track lay-out to which the passengers may be subjected are namely restricted by certain regulations. It should be mentioned that these limits do not represent comfort tolerable limits to prevent for instance motion sickness. The passenger acceleration limits are of interest to ensure that the kinematic model includes a realistic track lay-out, so the forces and moments that are exerted on the main chassis beam are of realistic magnitude as well. The positive and negative directions of the accelerations are depicted in Figure 2.14 at the passenger measurement point. The mass of each individual passenger used in the kinematic model equals eighty kilograms. The passenger measurement point is a reference point whose location is specified by standardized norms, and the permissible acceleration values in each direction are prescribed at this point as well. These norms define a certain distance between the passenger measurement point and respectively the seat pan and seat back. This distance is typically formulated as a range, so a variety of passenger shapes and sizes can be considered. In practice, the passenger measurement point is located within the chest of the passenger, as can be seen in Figure 2.14. The permissible acceleration values at the passenger measurement point can be deduced from the graphs and scripts in

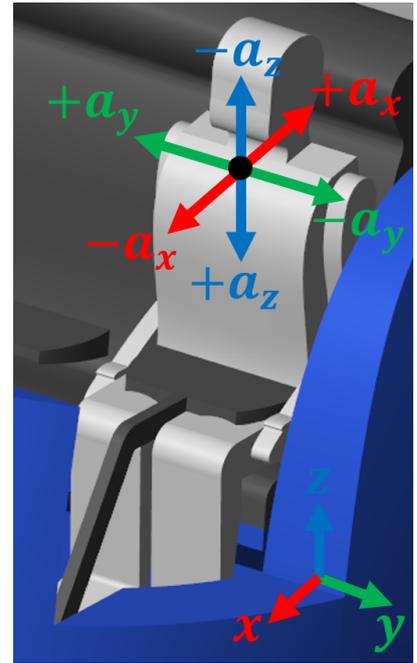


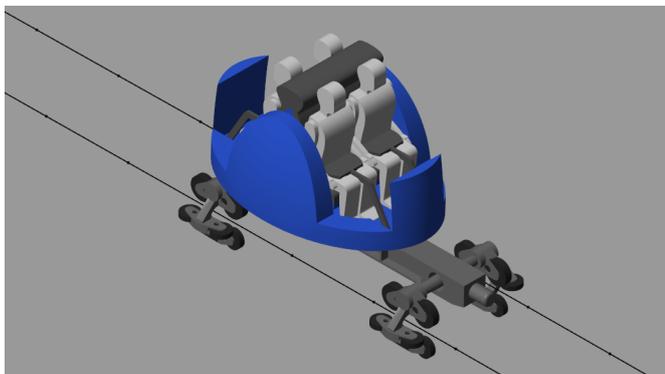
Figure 2.14: Directions of passenger accelerations

Appendix I, in which the tolerable passenger accelerations are presented as a function of time. Please note that the graphs have been truncated at  $t = 15$  s, since the acceleration limits remain constant for longer durations. Furthermore, the time duration limits are prescribed for  $t > 0.2$  s, since impacts with a duration less than 200 milliseconds are not addressed by the norms. The time duration limits for accelerations in  $x$ -direction are shown in Appendix I for a so-called base-case restraint. It is presumed that this type of restraint is used on the spinning rollercoaster. The acceleration limits for the base case restraint are relatively strict in comparison to other type of restraints such as an over-the-shoulder restraint, since the latter type of restraint minimizes forward motion of the passenger. Until the direction of the acceleration reverses, the permissible acceleration values decrease as time advances. However, when the direction of the acceleration

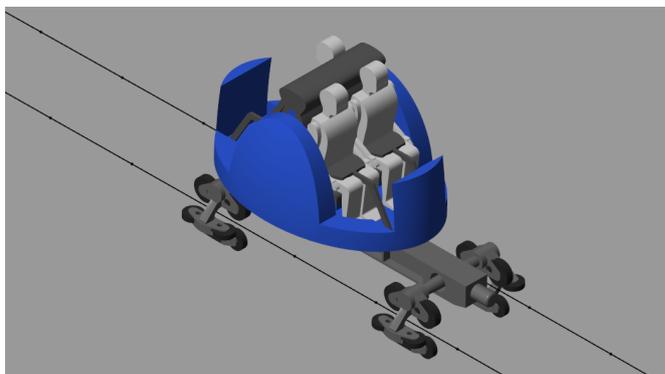
changes from positive to negative or vice versa, the duration is reset to zero. This implies that after an acceleration reversal, the permissible acceleration limits again start at their maximum values but with an opposite sign.

### 2.7.1 Passenger Occupancy Configurations

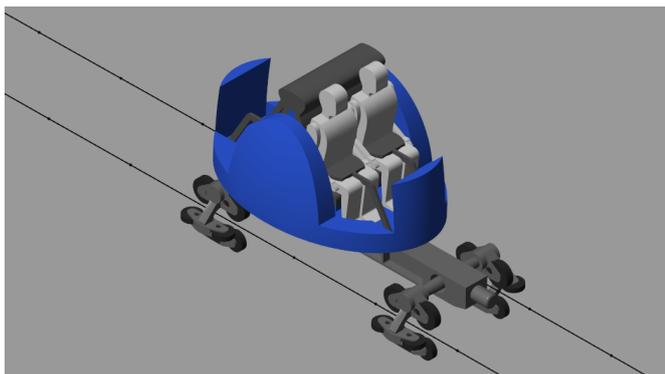
The primary incentive for developing the kinematic model is to determine the forces and moments that are exerted on the main chassis beam. Especially the most severe loadcases are of interest, since these cause the largest displacements and stresses in the main chassis beam. Beforehand it is unknown which passenger occupancy configuration leads to the most severe loadcases, and hence four unique configurations are considered, as depicted in Figure 2.15. The first configuration comprises four passengers, which results in the largest total mass of the rotating part of all considered configurations. Hence, the forces and moments exerted on the pivot point are expected to be relatively large for this configuration. On the other hand, the other configurations are characterized by a mass imbalance that could also lead to the application of substantial loads on the main chassis beam. It should be mentioned that an empty gondola is not believed to be a relevant passenger configuration. An empty gondola is namely not characterized by a mass imbalance, while the total mass of the rotating part is relatively small. Hence, a simulation with an empty gondola is not expected to result in more severe loadcases than a fully-loaded gondola.



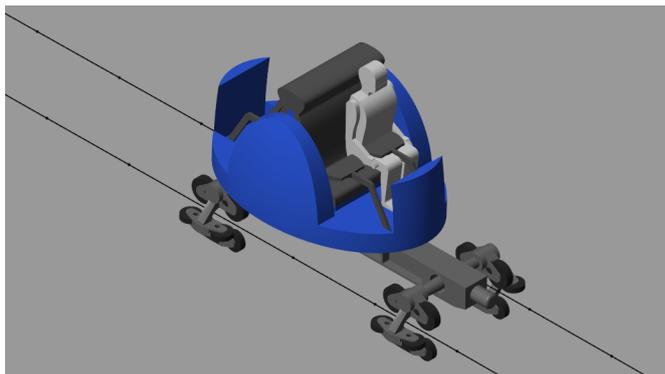
(a) Configuration 1



(b) Configuration 2



(c) Configuration 3



(d) Configuration 4

Figure 2.15: The four different passenger occupancy configurations

The passenger acceleration is determined by measuring the time-dependent relationship between the passenger measurement frame and the global frame. The rotation matrix and the linear accelerations between these two frames are computed at each time step, as shown in Figure A.19. The negative  $z$ -axis is specified as the direction of the gravitational acceleration, and hence gravity is defined as  $(0,0,-g)$  with respect to the global frame. Multiplying this vector with the rotation matrix results in the gravity components with respect to the passenger measurement frame. Summation of the gravity components and the relative accel-

erations yields the total accelerations experienced by the passengers. As can be deduced from the graphs in Appendix I.2, the passenger accelerations do not exceed the tolerable limits for any of the four considered passenger occupancy configurations.

### 2.7.2 Acceleration Combinations

Separate comparisons of the accelerations in  $x$ ,  $y$ , and  $z$ -direction with the corresponding limits is not sufficient for meeting the regulations, since the combination of accelerations should also be evaluated. When accelerations in different directions occur simultaneously, the combined effect of accelerations is not allowed to exceed specified limits either. The maximum values for the combined accelerations are specified by the following expressions. The admissible accelerations for selected durations need to be taken from the figures in Appendix I.

$$\left(\frac{a_x}{a_{adm,x}}\right)^2 + \left(\frac{a_y}{a_{adm,y}}\right)^2 \leq 1 \quad (2.8)$$

$$\left(\frac{a_x}{a_{adm,x}}\right)^2 + \left(\frac{a_z}{a_{adm,z}}\right)^2 \leq 1 \quad (2.9)$$

$$\left(\frac{a_z}{a_{adm,z}}\right)^2 + \left(\frac{a_y}{a_{adm,y}}\right)^2 \leq 1 \quad (2.10)$$

The combined effect of accelerations is extraordinarily relevant for a spinning coaster in particular. In case of a regular rollercoaster train, the passengers are typically subjected to significant lateral and vertical accelerations when the train traverses through a turnaround, while the effect of longitudinal accelerations is minor. On most other track sections, only one acceleration direction substantially contributes to the sum of accelerations experienced by the passenger, such as the vertical acceleration in a valley or the longitudinal acceleration on a launch. However, on a spinning rollercoaster any combination of passenger accelerations is possible at practically any section of track. Especially in a turnaround, the combination of vertical and longitudinal passenger accelerations would often exceed the corresponding combination limit presented in Equation 2.9 for the initial track lay-out design. The issue of exceeding particularly this combined acceleration limit is predominantly caused by the relatively strict acceleration limits in the longitudinal and vertical direction. For the base case restraint, the admissible acceleration in the negative  $x$ -direction may not exceed the -1.5 g-load limit. This acceleration direction corresponds to pushing the passenger out of the seat forward. The longitudinal passenger accelerations in this direction are usually small in case of a regular rollercoaster train. However, on a spinning rollercoaster the gondola could be orientated perpendicularly to the direction of travel of the train, which implies that the passenger could experience significant negative  $x$ -accelerations when the train traverses through a turnaround. For the spinning rollercoaster of interest, the passengers are not subjected to any negative vertical accelerations, which implies that the accelerations in positive  $z$ -direction may not exceed the 2.0 g-load limit when  $t > 12$  s. Please note that the 2.0 g-load limit also includes the contribution from the gravitational acceleration. Although the vertical direction does not account for the entire gravitational acceleration for most track sections, the effect of gravity still primarily contributes to the accelerations in the  $z$ -direction in turnarounds. Hence, the margin is small for the combination of positive  $z$ -accelerations and negative  $x$ -accelerations.

Several design iterations were required to improve the track lay-out up to a level where none of the time duration limits or combined acceleration limits were exceeded. The radii of the turnarounds were increased to reduce the accelerations to which the passengers were subjected. Figure 2.16 shows the combined acceleration factors for one of the passengers on a fully-loaded gondola. As can be deduced from this graph, the combined acceleration factor does not exceed the limit at any point along the ride. The improvements on the track lay-out have resulted in comparable combined acceleration factors for vertical accelerations in combination with respectively longitudinal or vertical accelerations.

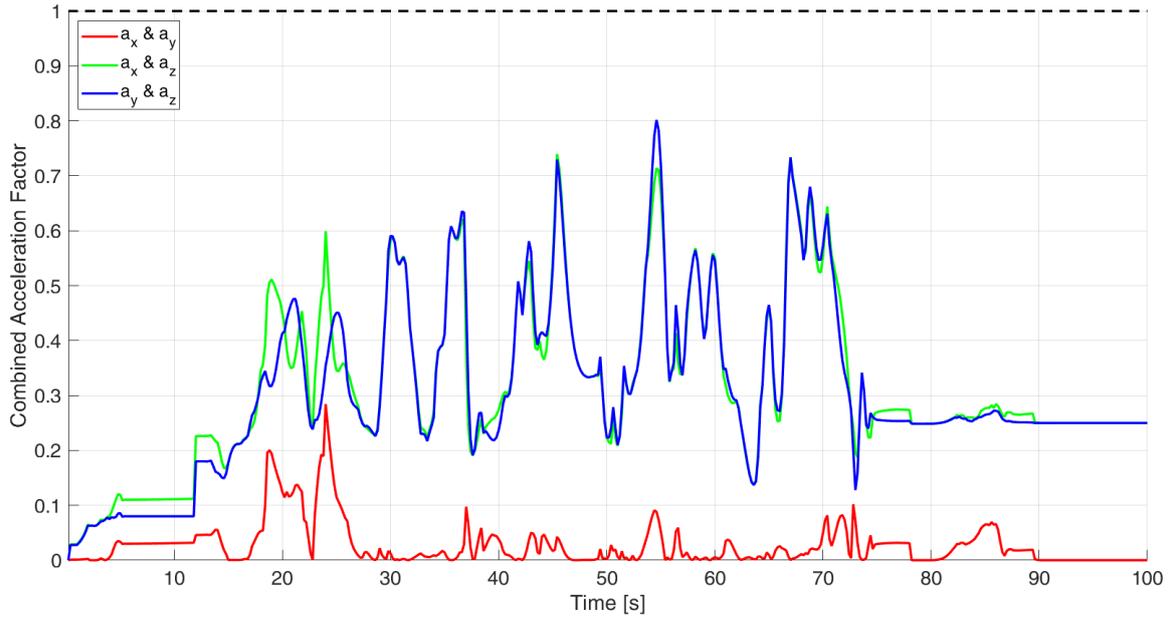


Figure 2.16: Combined acceleration factors for a passenger on a fully-loaded gondola

### 2.7.3 Acceleration Reversals

The norms also prescribe a minimum transition time between consecutive peaks of opposite sign in the  $x$  and  $y$ -acceleration signals. When the elapsed time between two consecutive acceleration peaks is less than 200 milliseconds, the allowable limit for the peak values becomes equal to half of the regular acceleration limit value. However, this regulation only applies to so-called sustained events, which have a duration of at least 0.2 second from zero-crossing to zero-crossing. Hence, the occurrence of non-sustained events can be neglected, since the impacts with a duration less than 200 milliseconds are not addressed by the norms. This implies that a violation of the time duration limits or the reversal norms by a non-sustained event can be ignored. The previously mentioned design iterations were also used to smooth transitions in the track lay-out, so the acceleration reversal norms were met. Evaluation of reversals in the  $z$ -acceleration data was not required, since the passengers are only subjected to positive vertical accelerations. The `Matlab`-script that checks all aspects of the passenger acceleration data with respect to the tolerable limits is presented in Appendix I.6. Please note that some sections of the script have been omitted to enhance readability.

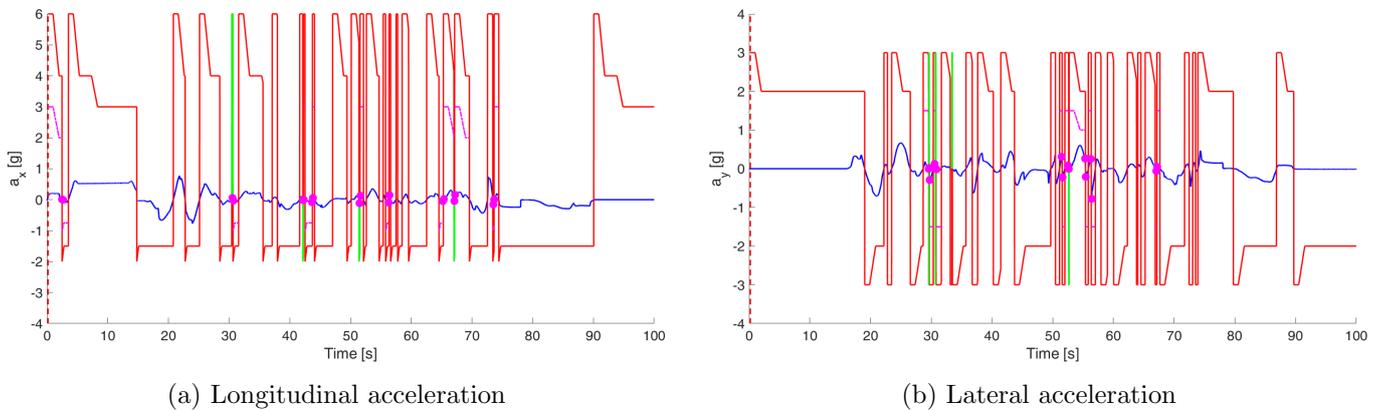


Figure 2.17: Comparison between the actual (blue) and admissible (red) accelerations for a passenger on a fully-loaded gondola. The acceleration limits for the sustained and non-sustained events are coloured respectively red and green, while the critical peak values and the corresponding limits are shown in purple

## 2.8 Model Validation

A validation of the kinematic model is presented in this section, which will be performed by means of a validation of the reaction forces sensed at the joints. A validation of the magnitude of these reaction forces should namely be possible based on the masses, inertias and accelerations of the bodies that comprise the vehicle multibody model. Please note that only the graphs corresponding to the first passenger occupancy configuration (Figure 2.15a) are presented in this chapter. The reader is referred to Appendix J for the graphs that depict the validation of forces and torques for other passenger occupancy configurations.

### 2.8.1 Forces at Pivot Joint

The forces exerted by the rotating part on the main chassis beam are sensed at the pivot joint. This joint constrains the translational motion of the gondola relative to the chassis beam, since a revolute joint only allows rotation around the  $z$ -axis. The rotating part includes all bodies that can rotate due to the allowance of  $z$ -rotation at the pivot joint. Aside from the gondola, seats, and the restraints, the rotating part also comprises the passengers. Hence, the total mass and centre of mass of the rotating part differ for each passenger occupancy configuration. This means that for each configuration, the accelerations need to be measured at a different centre of mass. Multiplication of these linear accelerations with the mass of the rotating part yields the total force that is exerted on the chassis beam at the pivot joint. This total force is decomposed in longitudinal ( $x$ ), lateral ( $y$ ), and vertical ( $z$ ) forces by means of a local frame at the pivot joint. As shown by Figure 2.18 for a fully-loaded gondola, the calculated forces are exactly aligned with the measured forces at the pivot joint at each time-step. These reaction forces are therefore of equal magnitude as the ones that can be expected based on the mass and accelerations of the rotating part.

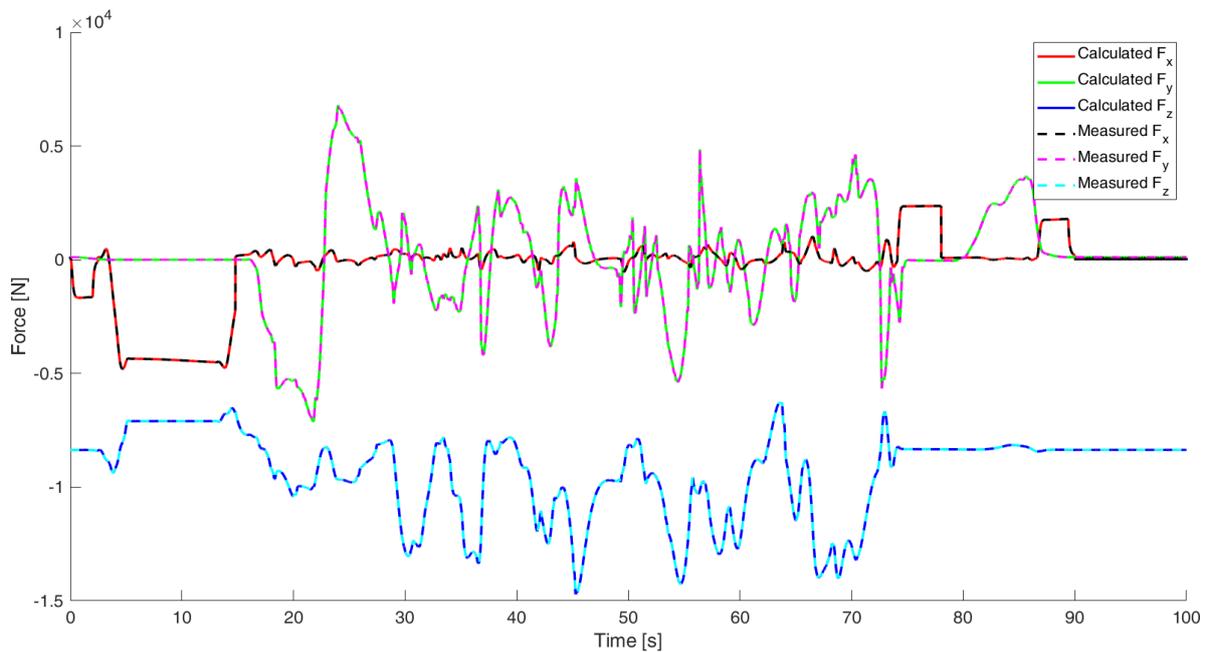


Figure 2.18: Comparison between the measured and calculated forces exerted on the main chassis beam by the rotating part

### 2.8.2 Torques at Pivot Joint

A similar analysis can be performed for the torques at the pivot joint. The previously-described forces should be multiplied with the distance between the pivot joint and the centre of mass of the rotating part. These distances function as a lever arm for the forces, and the relations presented in Equation 2.11 were found for the torques at the pivot joint. Forces in two separate directions contribute to  $T_x$  and  $T_y$  when multiplied

with a perpendicular lever arm. The distinct lever arm components result from a decomposition of the total distance between the pivot joint and the centre of mass, which differs for each passenger occupancy configuration. Hence, the values for  $r_x$ ,  $r_y$ , and  $r_z$  need to be determined for each individual configuration. The rotation around the  $z$ -axis is not constrained by the pivot joint, so based on the nature of the joint  $T_z$  is expected to equal zero at each time step.

$$T_x = (m \cdot a_z \cdot r_y) - (m \cdot a_y \cdot r_z) \quad T_y = (m \cdot a_x \cdot r_z) - (m \cdot a_z \cdot r_x) \quad T_z = 0 \quad (2.11)$$

where  $m$  represents the mass of the rotating part (including passengers),  $a$  symbolizes the acceleration at the centre of mass of the rotating part, and the lever arms are represented by the symbol  $r$ . Figure 2.19 depicts a comparison between the measured torques at the pivot joint and the torques that were calculated according to the expressions in Equation 2.11. In a similar manner as for the forces, also the magnitudes of the torques can be logically deduced from the forces at the pivot joint and the centre of mass of the rotating part.

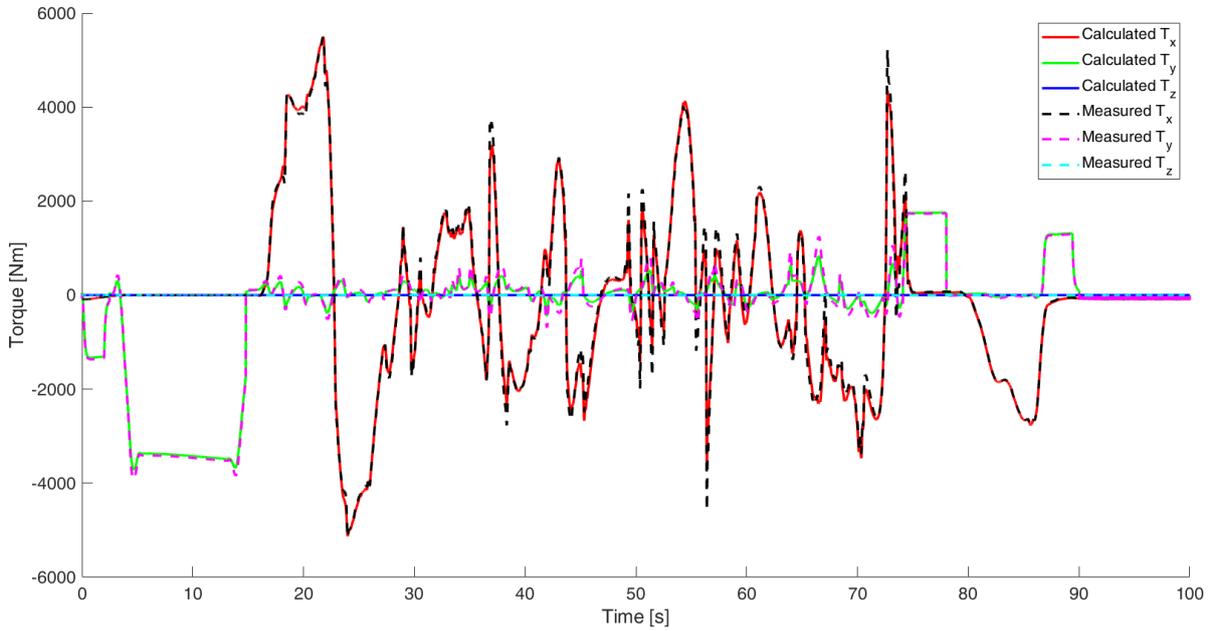


Figure 2.19: Comparison between the calculated torque and the torque measured at the pivot joint

### 2.8.3 Forces at Front and Rear Axle Joints

The main chassis beam and rotating part are carried by the front and rear axle. Hence, the reaction forces exerted by these axles on the main chassis beam can be substantial. The front axle is connected to the chassis beam by means of a spherical joint. This joint constrains all translations while the rotations are left unconstrained. On the other hand, the motion between the rear axle and the chassis beam is facilitated by a revolute joint. This type of joint constrains the translation along each of the three Cartesian axes, while rotational motion is only allowed around the vertical  $z$ -axis. The forces applied at the pivot joint have been validated at the previous section, while the accelerations at the centre of mass are also known. Hence, the forces at the spherical and revolute joint can be validated by evaluating the equilibrium of forces acting on the main chassis beam, as presented by Figure 2.20. The forces at the front axle, pivot joint, and rear axle act in opposite direction with respect to the accelerations due to the constraints imposed by the joints and the inertia of the bodies. Only the force acting in the  $x$ -direction on the underside of the chassis beam, which represents the sum of driving force and lift force, acts in a similar direction as the  $x$ -acceleration.

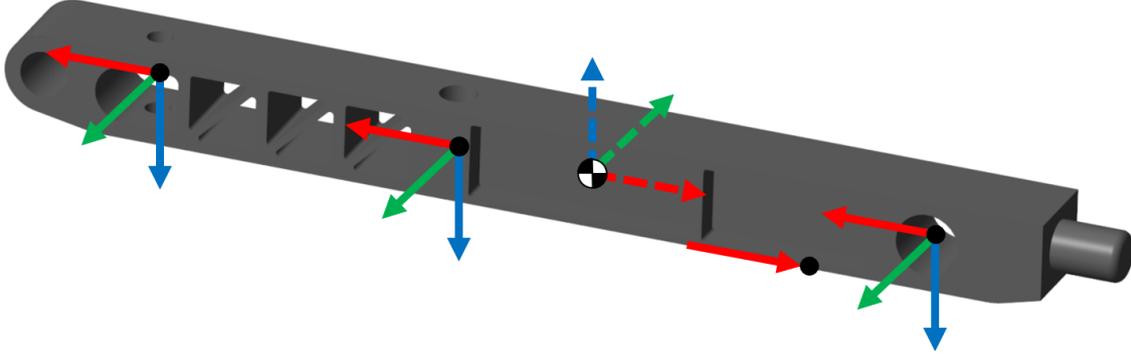


Figure 2.20: The forces acting on the main chassis beam have been decomposed in  $x$  (red),  $y$  (green), and  $z$ -direction (blue) and they are represented by a solid line, while the accelerations at the centre of mass are represented by a dashed line

Due to the application of a driving and lift force, separate validations of  $F_x$  at the front and rear axle is complicated. It is namely unknown at which ratio the reaction forces at either the front or rear axle compensate for the driving or lift force. Due to the dynamic nature of the analysis, a similar reasoning holds for the forces acting in the  $y$  and  $z$ -direction. A simple determination of the reaction forces at respectively the front and rear axle is consequently not possible. Hence, a validation is performed for the sum of forces in the  $x$ ,  $y$ , and  $z$ -direction, as shown in Equation 2.12.

$$\begin{aligned}
 ma_x &= -F_{x,axes} - F_{x,pivot} + F_{driving} + F_{lift} & \rightarrow & \quad F_{x,axes} = -ma_x - F_{x,pivot} + F_{driving} + F_{lift} \\
 ma_y &= -F_{y,axes} - F_{y,pivot} & \rightarrow & \quad F_{y,axes} = -ma_y - F_{y,pivot} \\
 ma_z &= -F_{z,axes} - F_{z,pivot} & \rightarrow & \quad F_{z,axes} = -ma_z - F_{z,pivot}
 \end{aligned}
 \tag{2.12}$$

where  $F_{x,axes}$ ,  $F_{y,axes}$ , and  $F_{z,axes}$  represent the sum of forces acting at the front and rear axle.

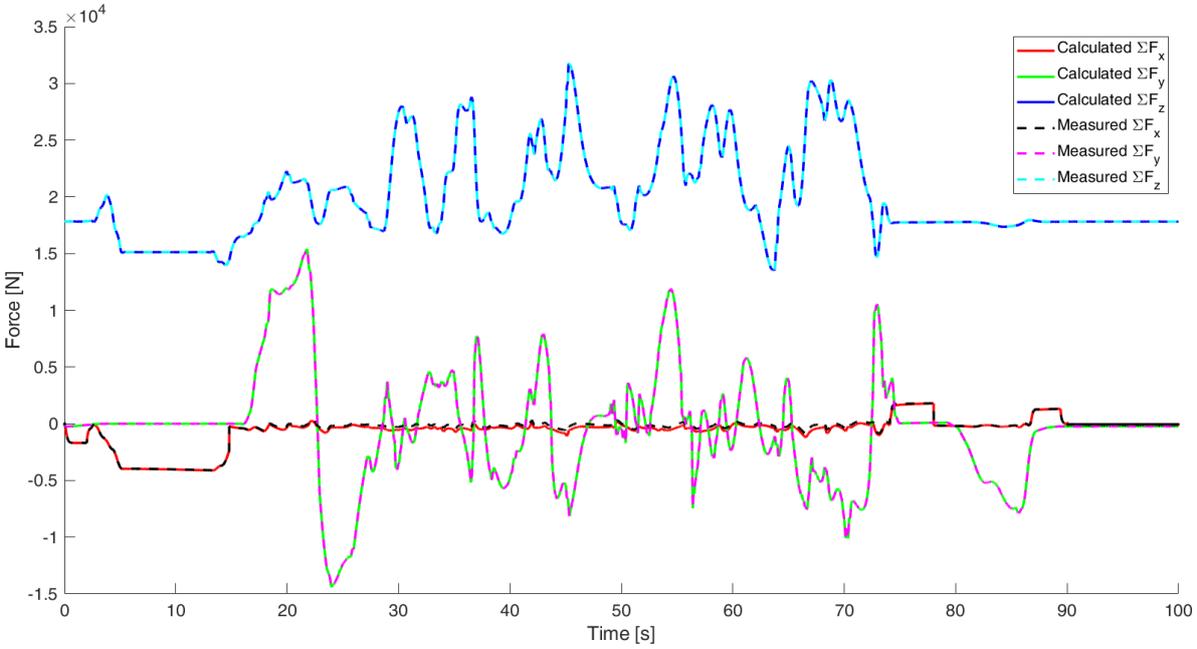


Figure 2.21: Comparison between the measured and calculated sum of forces exerted by the axles on the main chassis beam at the spherical joint (front) and revolute joint (rear)

The expressions on the left-hand side of Equation 2.12 are rewritten in such a way that all known quantities are on the right-hand side of the equations. The sum of reaction forces at the axles can be calculated using the mass of the chassis beam ( $m$ ), the accelerations at the centre of mass ( $a$ ), the validated forces at the pivot joint ( $F_{pivot}$ ), and the driving ( $F_{driving}$ ) and lift force ( $F_{lift}$ ). A comparison between the measured and calculated forces is presented in Figure 2.21. The alignment of the calculated and measured values proves that the expressions in Equation 2.12 are correct.

## 2.8.4 Torques at Rear Axle Joint

Since the front axle is connected to the main chassis beam by means of a spherical joint, reaction torques are only expected at the revolute joint between the rear axle and the chassis beam. The revolute joint allows rotation around the  $z$ -axis though, so only reaction torques around the  $x$  and  $y$ -axis are exerted at the revolute joint. The lever arms  $r_y$  and  $r_z$  of the spherical joint, centre of mass, and pivot joint with respect to the rear revolute joint are negligible or even equal to zero. The lever arm  $r_x$  with respect to the rear revolute joint is large, but as mentioned before this type of joint does not constrain rotation around the  $z$ -axis. Hence, the forces applied at the other joints and centre of mass do not induce a torque at the rear revolute joint. Evidently, this also applies to the forces that are applied at the rear revolute joint itself. This means that only the torques applied at the pivot joint could lead to a reaction torque at the rear revolute joint, as shown in Equation 2.13.

$$T_x = -T_{x,pivot} \cdot \cos(\theta) \quad T_y = -T_{x,pivot} \cdot \sin(\theta) \quad T_z = 0 \quad (2.13)$$

where  $\theta$  represents the angle between the rear axle and a reference line perpendicular to the main chassis beam. The orientation of the rear axle with respect to the chassis beam is of special interest for the validation of  $T_y$ . The bogies can hinge freely around the rear axle, and therefore no reaction torque around the  $y$ -axis is expected when the rear axle is orientated perpendicularly with respect to the chassis beam, which corresponds to  $\theta$  being equal to zero. On the other hand, when the angle  $\theta$  does not equal zero, a reaction torque  $T_y$  can be expected at the rear revolute joint. A comparison between the calculated and measured torques is depicted in Figure 2.22. The reader is referred to Appendix J for a validation of the reaction forces between the track and wheels, which were used for the calculation of the rolling and bearing frictional forces earlier in this report.

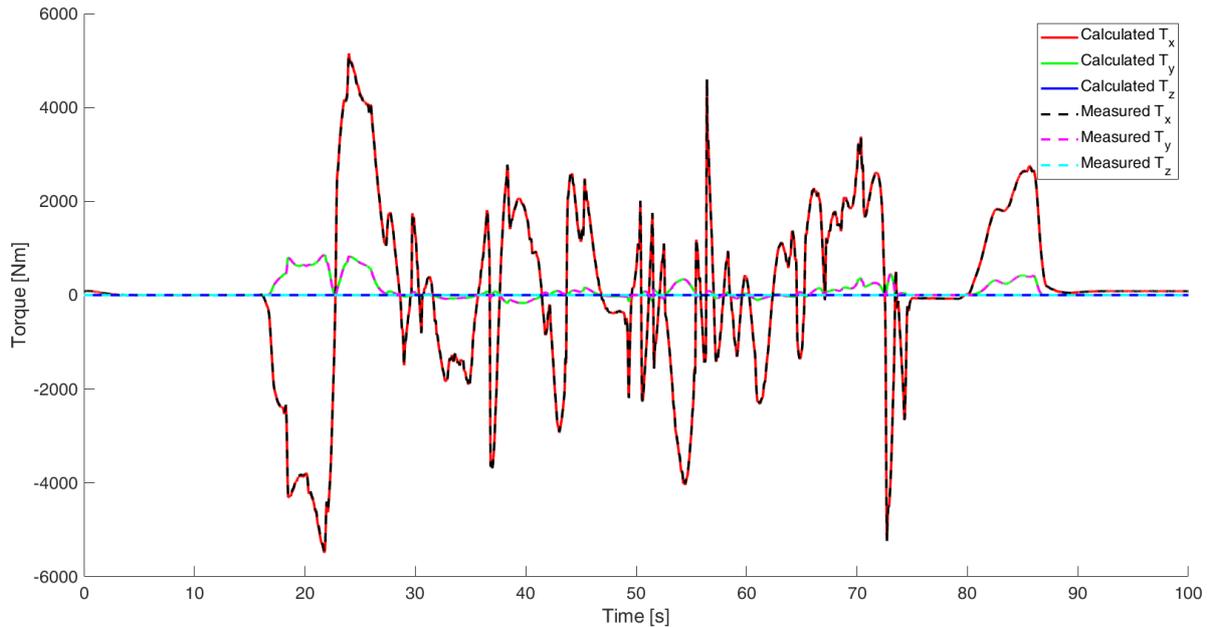


Figure 2.22: Comparison between the measured and calculated torques exerted by the rear axle on the main chassis beam at the revolute joint

# 3. Results

## 3.1 Loadcases

The forces and torques exerted on the main chassis beam have all been validated in the previous chapter, and hence these loads can be converted into loadcases that are required for a finite element analysis of the main chassis beam. The kinematic model is solved using a fixed time-step equal to 0.01 second, and at each time-step the forces, torques, and accelerations are known. Before the loadcases can be determined, the load data is first sampled with a sample time equal to 0.25 seconds. Prior to the finite element analysis it is unclear which passenger occupancy configuration results in the largest deformations and stresses. Therefore, the force and torque data of all four configurations depicted in Figure 2.15 is considered for the compilation of the loadcases. The duration of a single simulation equals 100 seconds, which implies that a data set with either accelerations, forces, or torques consists of 1604 values.

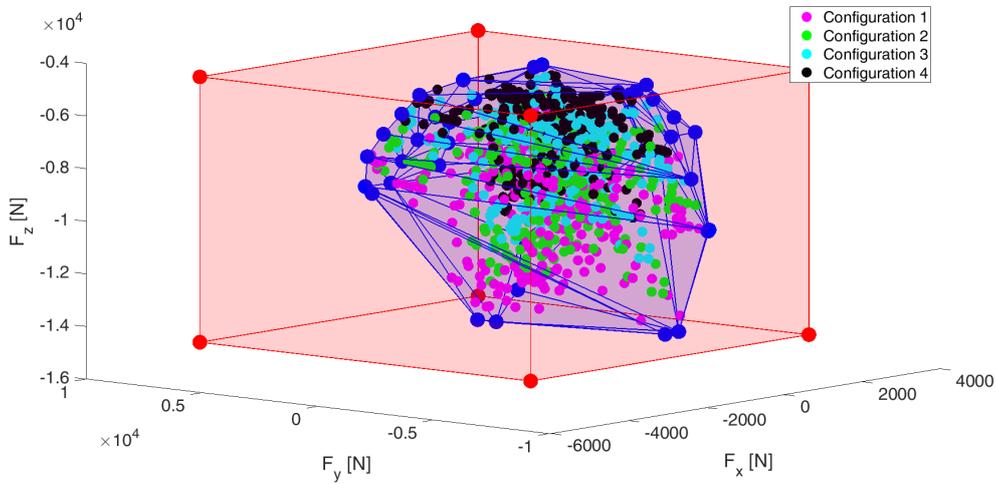


Figure 3.1: Forces exerted at the pivot joint for four different passenger occupancy configurations, which are used to compile the conventional (red) and optimized (blue) loadcases

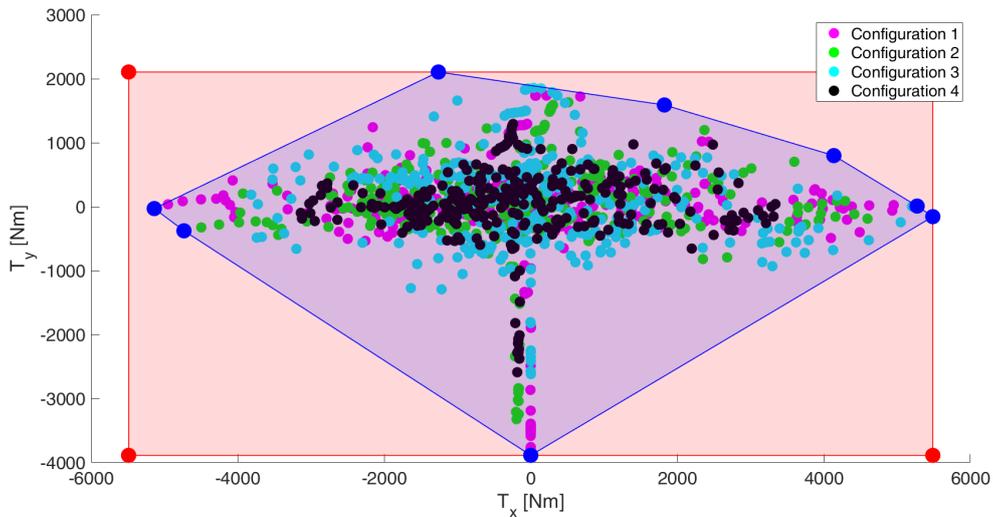


Figure 3.2: Torques exerted at the pivot joint for four different passenger occupancy configurations, which are used to compile the conventional (red) and optimized (blue) loadcases

The forces and torques exerted at the pivot joint are depicted in respectively Figure 3.1 and 3.2. According to the conventional methodology, the loadcases are described by the corners of the rectangular and cubical envelopes shown in red in both figures. These two envelopes enclose all forces and torques that are applied on the main chassis beam at the pivot joint, and hence these loadcases can be regarded as a sufficient test for a structural assessment of the chassis beam. However, the conventional loadcases are described by forces and torques with relatively large values in comparison to the actual values found in the data set. The structural integrity of the main chassis beam is therefore tested according to loadcases that are more severe than the actual loads induced on the beam. Hence, a different method is proposed that should result in a more accurate description of the loads by the loadcases. The optimized method is characterized by envelopes that coincide with the actual boundaries of the force and torque data, so the loadcases correspond to actual loads exerted on the pivot joint.

The corners of the cubical envelope in Figure 3.1 are defined by the extreme values in each dimension. For  $F_x$  and  $F_z$ , the lower and upper boundaries of the envelope are specified by respectively the minimum and maximum value. On the other hand, the minimum and maximum values of  $F_y$  are practically similar in magnitude, and hence the extreme value with the largest magnitude is selected as the boundary of the envelope in  $y$ -direction. The boundaries of the cubical envelope in the  $x-z$ -plane are therefore characterized by similar extreme values for  $F_y$ , aside from a different sign. The minimum and maximum values are also nearly similar in magnitude for  $T_x$  in Figure 3.2. Hence, the left and right-hand boundaries of the rectangular envelope are also described by the extreme value with the largest magnitude. On the other hand, the lower and upper boundaries of the rectangular envelope are described by respectively the minimum and maximum value of  $T_y$ , similar to  $F_x$  and  $F_z$  in Figure 3.1. The cubical and rectangular envelopes are defined by respectively eight and four corners, where each corner corresponds to a specific combination of forces or torques. Twelve unique loadcases result from different combinations of forces and torques for the conventional method.

The optimized loadcases are determined using a `Matlab` function named `boundary`, which results in an envelope that coincides with the boundaries of the data set. A shrink factor equal to zero has been specified for the `boundary` function, so the force and torque data is enclosed with the lowest possible number of corner points. The optimized method results nevertheless in a total of 39 unique loadcases, which is a substantially larger number than the number of loadcases for the conventional method. On the other hand, the most severe loads induced on the main chassis beam are most accurately described by the optimized method.

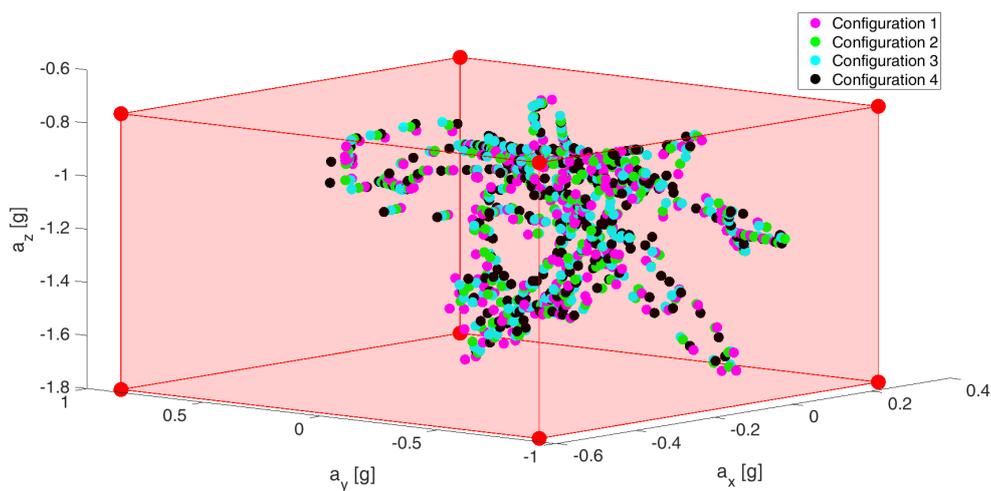


Figure 3.3: Linear accelerations at the centre of mass of the chassis beam for four different passenger occupancy configurations, where the corners of the red envelope prescribe the acceleration values used by the conventional method

Along with the forces and torques at the pivot joint, also the linear and angular accelerations at the centre of mass are required for an adequate finite element analysis. The finite element model is namely validated by means of the reaction forces at the front and rear axle, as will be elucidated at a later stage. Without a specification of especially the linear acceleration values, the reaction forces at the front and rear axle would not be similar for the kinematic model and the finite element model. For the optimized method, the points on the envelopes coincide with force and moment values that are actually exerted at the pivot joint. Hence, the linear and angular accelerations at the centre of mass of the chassis beam can be obtained from the kinematic model for each corresponding loadcase. With regards to the conventional methodology, the acceleration values are obtained using a different approach. In a similar manner as for the forces at the pivot joint, a cubical envelope is drawn that encloses all linear acceleration data, as shown in Figure 3.3. The linear acceleration values at the corners of the envelope are subsequently matched with each of the twelve conventional loadcases. A certain corner on the cubical force envelope corresponds to the same corner on the cubical acceleration envelope, since it is presumed that the chassis beam accelerates in the same direction as the direction in which the force is applied. Hence, each conventional loadcase is described by a combination of force, torque, and linear acceleration values. No clear relation can be found between the force and torque data of the conventional loadcases and the angular acceleration values though, and hence the angular acceleration is set to zero for the conventional method. Please note that the angular acceleration of the main chassis beam is typically small, so the effect of the angular acceleration on the results can be regarded as minor in comparison to the linear accelerations. The forces, torques, and linear accelerations for the conventional method are presented in Table 3.1, while the loadcases for the optimized method can be found in Table K.1 and K.2 in Appendix K.

| LC        | Pivot Joint |           |           |            |            | Centre of Mass            |                           |                           |
|-----------|-------------|-----------|-----------|------------|------------|---------------------------|---------------------------|---------------------------|
|           | $F_x$ [N]   | $F_y$ [N] | $F_z$ [N] | $T_x$ [Nm] | $T_z$ [Nm] | $a_x$ [m/s <sup>2</sup> ] | $a_y$ [m/s <sup>2</sup> ] | $a_z$ [m/s <sup>2</sup> ] |
| <b>1</b>  | 2363.39     | -4517.69  | -7131.68  | -5136.38   | -2105.14   | 2.79                      | -7.45                     | -8.74                     |
| <b>2</b>  | 2363.39     | -14602.00 | -7131.68  | -5136.38   | -2105.14   | 2.79                      | -17.63                    | -8.74                     |
| <b>3</b>  | 2363.39     | -4517.69  | 7131.68   | 5486.47    | -2105.14   | 2.79                      | -7.45                     | 8.74                      |
| <b>4</b>  | 2363.39     | -14602.00 | 7131.68   | 5486.47    | -2105.14   | 2.79                      | -17.63                    | 8.74                      |
| <b>5</b>  | 2363.39     | -4517.69  | -7131.68  | -5136.38   | 3886.92    | 2.79                      | -7.45                     | -8.74                     |
| <b>6</b>  | -4786.08    | -4517.69  | -7131.68  | -5136.38   | 3886.92    | -5.66                     | -7.45                     | -8.74                     |
| <b>7</b>  | 2363.39     | -14602.00 | -7131.68  | -5136.38   | 3886.92    | 2.79                      | -17.63                    | -8.74                     |
| <b>8</b>  | -4786.08    | -14602.00 | -7131.68  | -5136.38   | 3886.92    | -5.66                     | -17.63                    | -8.74                     |
| <b>9</b>  | 2363.39     | -4517.69  | 7131.68   | 5486.47    | 3886.92    | 2.79                      | -7.45                     | 8.74                      |
| <b>10</b> | -4786.08    | -4517.69  | 7131.68   | 5486.47    | 3886.92    | -5.66                     | -7.45                     | 8.74                      |
| <b>11</b> | 2363.39     | -14602.00 | 7131.68   | 5486.47    | 3886.92    | 2.79                      | -17.63                    | 8.74                      |
| <b>12</b> | -4786.08    | -14602.00 | 7131.68   | 5486.47    | 3886.92    | -5.66                     | -17.63                    | 8.74                      |

Table 3.1: Loadcases specified in terms of forces, torques, and accelerations for the conventional method with respect to the reference frame used in Ansys

## 3.2 Constraints

The finite element analysis of the main chassis beam is performed in Ansys Mechanical APDL. The point-and-click operations that are typically required by similar programs would be highly inconvenient due to the application of a large number of loadcases. Manually entering the forces, torques, and accelerations for each loadcase would namely be a very costly procedure in terms of time and effort. Ansys allows the user to import a text-file containing all the commands that should be executed by the program. Instead of entering all values by hand, loops can be scripted that dictate the program to repeat certain actions, such as running a large number of loadcases. However, before the forces and torques can be applied on the model, its constraints should be defined first.

Once the geometry of the main chassis beam has been imported in Ansys, it is meshed using **SOLID186** elements, which are higher order three-dimensional 20-node solid elements that exhibit quadratic displacement behaviour. Hence, this element is defined by twenty nodes having three degrees of freedom per node, which are the translations in the nodal  $x$ ,  $y$ , and  $z$ -directions [8]. The elasticity modulus ( $E = 200$  GPa), Poisson ratio ( $\nu = 0.3$ ), and density ( $\rho = 7500$  kg/m<sup>3</sup>) of steel are assigned to the solid elements.

Constraints should be defined at the spherical and revolute joint between the main chassis beam and respectively the front and rear axle. Both axles cannot translate in the  $y$  and  $z$ -direction with respect to the track, and hence the translation at the spherical and rear revolute joint should be constrained by prescribing a zero displacement in these directions. On the other hand, modelling similar  $x$ -translation constraints in Ansys as in the kinematic model appeared not be straightforward. Simultaneously constraining the  $x$ -translation at the front and rear axle, or definition of the  $x$ -translation at both axles as a degree of freedom, both proved to be incorrect approaches. An acceptable modelling technique for the  $x$ -translation constraints in Ansys is the prescription of a zero  $x$ -translation at the rear axle, while the translation in the  $x$ -direction is left unconstrained at the front axle. A spherical joint does not constrain any rotations, and hence only the  $y$  and  $z$ -translation are constrained at the front axle. The connection between the main chassis beam and the rear axle is facilitated by means of a revolute joint, which only allows rotation around the  $y$ -axis. Hence, aside from constraining all translations, also the rotations around the  $x$  and  $z$ -axis are constrained at the rear revolute joint. The rotation around the  $x$ -axis is constrained by prescribing a zero rotation, while the procedure for constraining rotation around the  $z$ -axis is explained in the next paragraph.

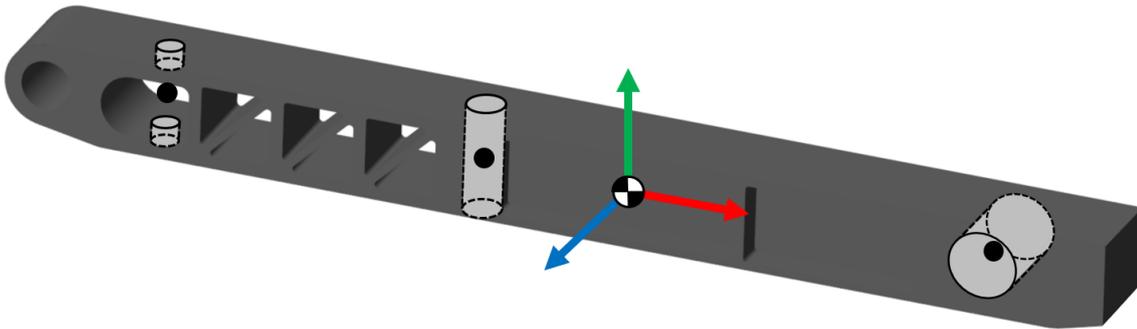


Figure 3.4: The target surfaces and pilot nodes are depicted in respectively grey and black, where the coordinate system in the centre of mass specifies the  $x$  (red),  $y$  (green) and  $z$ -direction (blue) used in Ansys

The constraints should be defined at the interface surfaces between the main chassis beam and the axles. Specification of a zero displacement or rotation for each node on these interface surfaces does not yield correct results though. For instance, constraining all translations at the rear revolute joint would automatically constrain all rotations as well, while the  $y$ -rotation should actually be left free. The nodes on the interface surface are namely all defined at different locations, which constrains any rotational motion if all nodal translations are set equal to zero. Therefore, the model comprises three so-called pilot nodes. As shown in Figure 3.4, the pilot nodes are located at the centre of the spherical joint, pivot joint, and rear revolute joint. The location of the pilot node is important when rotations or torques are specified, which explains the location of the pilot nodes at the centre of each joint. A **TARGE170** element is used to represent a three-dimensional target surface at the location of a pilot node for the associated contact elements. With regard to the joints on the main chassis beam, the target surface could therefore be regarded as an element that consists of a single node. The pilot node governs the motion of each corresponding rigid target surface, which means that forces, torques, displacements, and rotations for the target surface can therefore be prescribed on the pilot node [9]. Since the chassis beam geometry has been meshed with twenty-node solid elements, the interface surfaces are overlaid with eight-node **CONTA174** elements. These surface-to-surface contact elements are used to define a deformable surface at the interface surfaces [10]. The contact between the target surfaces

and the contact elements is prescribed as always bonded, which implies that a rigid connection is defined between the pilot nodes and the corresponding interface surfaces. The  $y$  and  $z$ -translation are set equal to zero for the pilot node at the centre of the spherical joint, while the  $x$ -translation and rotations are left unconstrained. In addition to the  $x$ -translation, the interface surface at the spherical joint can therefore freely pivot around the associated pilot node. This means that the behaviour of a spherical joint has been imitated in the finite element model. Furthermore, the forces and torques at the pivot joint are exerted on the associated pilot node. The application of the loads on the pivot joint is therefore modelled correctly for the conventional and optimized method. For the revolute joint at the rear axle, the translations and  $x$ -rotation of the associated pilot node are set equal to zero. One would also expect a zero  $z$ -rotation at this pilot node, since the revolute joint only allows rotation around the  $y$ -axis. However, the hinge joints between the rear axle and bogies allow the rear axle to rotate around its longitudinal axis with respect to the track. The longitudinal axis coincides with the  $z$ -axis at the rear revolute joint, and hence the  $z$ -rotation at the associated pilot node should not be fully constrained nor left unconstrained. It appears that the  $z$ -rotation at the rear revolute joint is modelled most realistically by direct application of a torque  $T_z$  at the associated pilot node. Under usual circumstances, the value of this torque is not known prior to the finite element analysis. However, from the validation it is known that this torque can be calculated according to the relation presented in Equation 3.1.

$$T_z = -T_{x,\text{pivot}} \cdot \sin(\theta) \quad (3.1)$$

The loadcases for the conventional and optimized method already include a value for  $T_{x,\text{pivot}}$ . Hence, this parameter can be regarded as a known quantity. The value for  $\theta$  can also be obtained for each loadcase of the optimized method, since each loadcase corresponds to an actual configuration of the kinematic model. However, with regard to the conventional method, the exact value for  $\theta$  is not known beforehand for each associated loadcase. Hence, the conventional method uses the maximum value of  $\theta$  to calculate  $T_z$ , which equals approximately 0.19 radians. The applied torques at the rear pilot node for each loadcase are shown in Table K.3, while an overview of the applied loads and degrees of freedom is depicted in Figure 3.5.

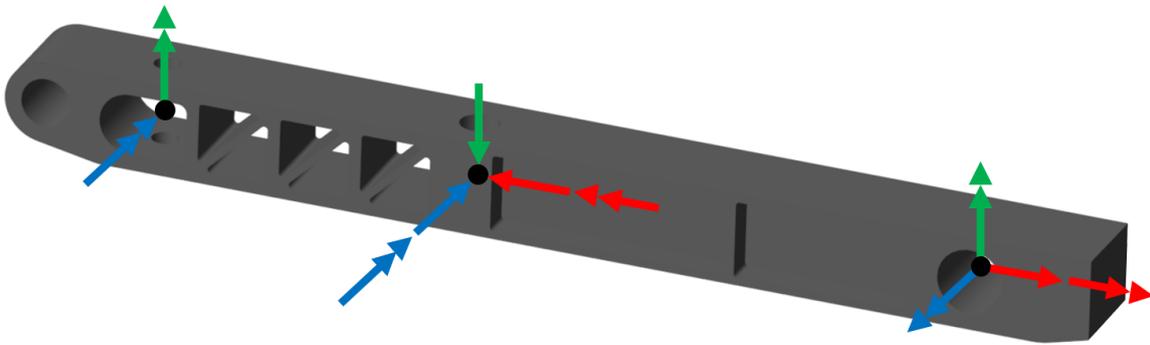


Figure 3.5: Overview of the applied loads (arrows pointing inwards) and degrees of freedom (arrows pointing outwards) of the pivot nodes, with the  $x$  (red),  $y$  (green) and  $z$ -direction (blue) indicated by colours

### 3.3 Reaction Forces

The finite element model is validated by means of a comparison between the reaction forces at the pilot nodes and at the joints in the kinematic model. Ansys namely allows the user to print the reaction forces at the constrained nodes. Only the reaction forces resulting from the optimized method can be compared to the reaction forces computed by Ansys, since the reaction forces at the joints in the kinematic model are known for the optimized loadcases only. As mentioned earlier, the method for constraining  $x$ -translations at the front and rear pilot node is not in full agreement with the kinematic model. This means that a direct comparison between the forces in  $x$ -direction at the front and rear pilot nodes is unfeasible. The values for

$F_x$  obtained by both the kinematic model and Ansys are therefore summed, so totals of the reaction forces in  $x$ -direction can be compared to each other rather than the individual contributions at the front or rear pilot nodes. External forces such as the brake or lift force are not modelled in the finite element model. To achieve an adequate comparison with the reaction forces in  $x$ -direction from the kinematic model, the external force should be added to the sum of  $F_x$  as obtained in Ansys. With the exception of  $T_z$  at the rear revolute joint, Table 3.2 shows a comparison between the reaction forces and torques that result from the kinematic model or Ansys. The value for  $T_z$  in Ansys can directly be computed according to the validated relation presented in Equation 3.1. In other words, a validation of the kinematic model has already proven that the torque applied at the rear pilot node in Ansys is equal to the reaction torque sensed by the kinematic model. A direct comparison between the reaction forces at the rear revolute joint could therefore be regarded as redundant.

The difference between the reaction forces in Simscape and Ansys can be quantified by means of a measure called the relative percentage difference (RPD). The expression used for calculating the RPD is shown in Equation 3.2.

$$\text{RPD} = \frac{|S - A|}{\frac{|S| + |A|}{2}} \quad (3.2)$$

where  $S$  and  $A$  represent the reaction force or torque from respectively the Simscape model and Ansys. The value for RPD can range between zero and two, where the latter magnitude is only obtained if the values for  $S$  and  $A$  are characterized by opposite signs. Large values for RPD are typically the result of very low absolute values of the reaction forces. Hence, a minor difference between  $S$  and  $A$  could already lead to a very large RPD value. Exceptionally large values for the relative percentage difference are usually encountered when the value for  $S$  or  $A$  approaches zero. Therefore, for the calculation of the average RPD values presented in Table 3.2, the values for the reaction forces were only considered if  $S$  and  $A$  complied to the condition shown in Equation 3.3. Please note that the threshold value in this condition has to some extent been chosen arbitrarily.

$$\frac{|S| + |A|}{2} > 5 \quad (3.3)$$

Table 3.2 also shows the average RPD values when all data would be taken into account, including the values that do not comply to the condition in Equation 3.3. In general, the enforcement of the condition in Equation 3.3 results in RPD values that are more representative for the differences between the reaction forces from the kinematic model and Ansys. The minimum and maximum RPD values are also presented in Table 3.2. The RPD values for all individual loadcases are shown in Appendix K.3.

| Load Type                              | Minimum RPD | Average RPD (All) | Average RPD (> 5) | Maximum RPD |
|--|-------------|-------------------|-------------------|-------------|
| $\mathbf{F}_{\mathbf{x},\text{total}}$ | 0.54 %      | 30.23 %           | 30.23 %           | 115.90 %    |
| $\mathbf{F}_{\mathbf{y},\text{front}}$ | 1.02 %      | 5.70 %            | 5.70 %            | 28.67 %     |
| $\mathbf{F}_{\mathbf{z},\text{front}}$ | 0.09 %      | 22.64 %           | 4.41 %            | 200.00 %    |
| $\mathbf{F}_{\mathbf{y},\text{rear}}$  | 0.12 %      | 2.80 %            | 2.80 %            | 9.47 %      |
| $\mathbf{F}_{\mathbf{z},\text{rear}}$  | 0.14 %      | 23.48 %           | 1.44 %            | 200.00 %    |
| $\mathbf{T}_{\mathbf{x},\text{rear}}$  | 0.00 %      | 0.17 %            | 0.03 %            | 2.06 %      |

Table 3.2: Relative percentage difference between the reaction forces obtained in Simscape and Ansys

It can be deduced from the values presented in Table 3.2 that the relative percentage difference is typically minor if the condition in Equation 3.3 is taken into account. However, for the sum of forces in  $x$ -direction the relative percentage difference remains substantial. It appears that the implemented modelling of the constraints in  $x$ -direction at the front and rear pilot nodes also leads to a flawed result for the sum of  $F_x$ , aside from incorrect results for the individual contributions. Only the reaction force in  $x$ -direction at the

rear pilot node contributes to the sum of  $F_x$ , and the absence of a  $x$ -translation constraint at the front pilot node consequently results in an improper total reaction force in  $x$ -direction. Constraining the  $x$ -translation at both pilot nodes is not deemed a viable alternative either, since it would lead to excessively large stresses in the main chassis beam. It should again be mentioned that acceptable relative percentage difference values were obtained for all other forces and torques. Overall, the reaction forces obtained with the kinematic model and finite element model show a sufficient level of similarity. The described procedure for applying the loads and constraints in the finite element model is therefore approved, despite the substantial relative percentage difference for the sum of reaction forces in the  $x$ -direction. The finite element model in Ansys is identical for the conventional and optimized method. Hence, the previously-described validation of reaction forces from the optimized method implies that both the conventional and optimized loadcases can be imposed on an approved finite element model.

### 3.4 Resultant Stresses and Displacements

Similar to the reaction forces, also the stresses and nodal displacements can be exported from Ansys for each loadcase. The resultant maximum stresses and displacements for the conventional and optimized method are presented in Table 3.3. The magnitudes of the stress and displacement values are relatively low due to the robust design of the main chassis beam. It is therefore more relevant to consider the proportional differences between the results instead of the absolute differences. Subtracting the result of the conventional method from the value associated with the optimized method and subsequently dividing this result by the respectively  $\sigma_{max}$  and  $d_{max}$  of the conventional method, yields the proportional difference between both methods. The resultant proportional differences are shown in Equation 3.4.

$$\frac{\sigma_{\max,O} - \sigma_{\max,C}}{\sigma_{\max,C}} = \frac{-2.99 \text{ [MPa]}}{30.21 \text{ [MPa]}} = -9.90\% \quad \frac{d_{\max,O} - d_{\max,C}}{d_{\max,C}} = \frac{-0.021 \text{ [mm]}}{0.238 \text{ [mm]}} = -8.82\% \quad (3.4)$$

where  $C$  and  $O$  denote respectively the conventional and optimized method. From the results in Equation 3.4 the conclusion can be drawn that for the same kinematic model, the maximum stress and displacement values predicted by the optimized method are nearly ten percent lower than the results obtained with the conventional method. Taking into consideration that the optimized method uses loads that are actually applied on the kinematic model, one could argue that the loadcases imposed on the finite element model by the conventional method are too severe. Earlier in the report it was stated that the conventional approach would be too conservative, since the loadcases of the conventional method are relatively stringent in comparison to the actual loads induced on the chassis beam. This statement is confirmed by the results presented in Table 3.3. The resultant stresses and displacements for each individual loadcase are shown in Table M.1 in Appendix M for both methods. The command lines required by Ansys for the performance of the conventional and optimized analyses are generated by the `Matlab` script in Appendix L.

|                | $\sigma_{\max}$ [MPa] |           | $d_{\max}$ [mm] |           |
|----------------|-----------------------|-----------|-----------------|-----------|
|                | Conventional          | Optimized | Conventional    | Optimized |
| <b>Minimum</b> | 24.78                 | 3.16      | 0.188           | 0.025     |
| <b>Average</b> | 27.61                 | 13.11     | 0.211           | 0.113     |
| <b>Maximum</b> | 30.21                 | 27.22     | 0.238           | 0.217     |

Table 3.3: Comparison between the maximum Von Mises stress values and nodal displacements of the conventional and optimized method

### 3.5 Transient Analysis

The previous section has shown that the optimized method provides a more accurate prediction of the stresses and displacements in the main chassis beam. The maximum stress and displacement values predicted by the optimized method are nearly ten percent lower than the values acquired with the conventional

method. However, the resultant stresses and nodal displacements obtained with the optimized method could be too optimistic, which could lead to an underestimation of the stress or strain levels in the main chassis beam. The loadcases of the optimized method are separately applied to the model, and hence the finite element analysis consists of distinct static structural analyses. On the other hand, a transient analysis also considers the deformation of the main chassis beam caused by previous loadcases. The deformation of the chassis beam and accumulated stress levels in the material are therefore taken into account by a transient analysis when a new loadcase is applied. From this perspective, a transient analysis could be regarded as a more realistic method for modelling the deformation of the main chassis beam. A comparison between the results from the optimized method and the transient analysis should reveal if too optimistic results are obtained by the optimized method.

The loadcases for the optimized and conventional method were obtained by creating two different envelopes around the load data, which was acquired from the kinematic model with a sample time of 0.25 seconds. These loadcases are representative for the entire load data set and all passenger occupancy configurations. It therefore sufficed to consider only a limited number of loadcases during the separate static analyses that were performed. On the other hand, a transient analysis is not characterized by a relatively small number of loadcases, since all loads should be evaluated in the correct sequence. Additionally, the loads induced on the main chassis beam should be evaluated by means of a separate transient analysis for each passenger occupancy configuration. The load data set contains 401 loadcases for each passenger occupancy configuration, and therefore 1604 loadcases should be evaluated in total by the transient analysis. This total number of loadcases is orders of magnitude larger than the number of loadcases that had to be evaluated in case of the conventional or optimized method.

| Load Type                     | Configuration | Minimum RPD | Average RPD (> 5) | Maximum RPD |
|-------------------------------|---------------|-------------|-------------------|-------------|
| $\mathbf{F}_{x,\text{total}}$ | <b>1</b>      | 0.13 %      | 56.09 %           | 200.00 %    |
|                               | <b>2</b>      | 0.36 %      | 58.85 %           | 200.00 %    |
|                               | <b>3</b>      | 0.20 %      | 59.32 %           | 200.00 %    |
|                               | <b>4</b>      | 0.03 %      | 61.86 %           | 200.00 %    |
| $\mathbf{F}_{y,\text{front}}$ | <b>1</b>      | 0.02 %      | 4.50 %            | 31.56 %     |
|                               | <b>2</b>      | 0.15 %      | 4.52 %            | 30.40 %     |
|                               | <b>3</b>      | 0.29 %      | 4.57 %            | 29.20 %     |
|                               | <b>4</b>      | 0.44 %      | 4.72 %            | 28.56 %     |
| $\mathbf{F}_{z,\text{front}}$ | <b>1</b>      | 0.00 %      | 16.93 %           | 200.00 %    |
|                               | <b>2</b>      | 0.01 %      | 9.11 %            | 200.00 %    |
|                               | <b>3</b>      | 0.08 %      | 11.38 %           | 200.00 %    |
|                               | <b>4</b>      | 0.04 %      | 14.48 %           | 200.00 %    |
| $\mathbf{F}_{y,\text{rear}}$  | <b>1</b>      | 0.01 %      | 1.59 %            | 19.21 %     |
|                               | <b>2</b>      | 0.00 %      | 1.59 %            | 19.26 %     |
|                               | <b>3</b>      | 0.01 %      | 1.65 %            | 19.31 %     |
|                               | <b>4</b>      | 0.00 %      | 1.70 %            | 18.71 %     |
| $\mathbf{F}_{z,\text{rear}}$  | <b>1</b>      | 0.01 %      | 11.79 %           | 200.00 %    |
|                               | <b>2</b>      | 0.00 %      | 5.83 %            | 200.00 %    |
|                               | <b>3</b>      | 0.01 %      | 8.57 %            | 200.00 %    |
|                               | <b>4</b>      | 0.00 %      | 9.45 %            | 200.00 %    |
| $\mathbf{T}_{x,\text{rear}}$  | <b>1</b>      | 0.00 %      | 8.58 %            | 200.00 %    |
|                               | <b>2</b>      | 0.00 %      | 0.78 %            | 35.50 %     |
|                               | <b>3</b>      | 0.00 %      | 3.25 %            | 200.00 %    |
|                               | <b>4</b>      | 0.00 %      | 0.87 %            | 155.45 %    |

Table 3.4: Relative percentage difference between the reaction forces obtained by the kinematic model in Simulink and a transient finite element analysis in Ansys

Except for the nature of the analysis, the finite element model used for the transient analysis is completely identical to the one previously used for the conventional and optimized methods. The loads and constraints are applied in a similar way as described in the previous chapters. Nevertheless, the reaction forces from the kinematic model and Ansys are compared to each other to verify if the transient analysis is conducted correctly. Table 3.4 shows the relative percentage difference between the reaction forces for all four passenger occupancy configurations. The RPD values for the sum of forces in  $x$ -direction are again substantial, and even twice the magnitude of the RPD values acquired for the optimized method. This significant difference could be caused by the application of loads on a deformed beam during a transient analysis. The kinematic model on the other hand always comprises an undeformed chassis beam, and the deformation of the beam during the transient finite element analysis could therefore affect the reaction force values. A comparison between the RPD values of both analyses also reveals significant differences for some other load types. Aside from the sum of forces in  $x$ -direction, also the RPD values for  $F_{z,\text{front}}$ ,  $F_{z,\text{rear}}$ , and  $T_{x,\text{rear}}$  significantly deviate from the RPD values acquired with the optimized method. The different reaction forces lead to RPD values that are multiple times larger for the transient finite element analysis. The forces applied in the  $y$ -direction are an exception to this general pattern, since the RPD values resulting from the transient analysis are comparable or even slightly lower than the RPD values shown in Table 3.2 for the optimized method. Furthermore, the extent to which the RPD values from the transient analysis deviate from the values of the optimized method also differs for each passenger occupancy configuration. The magnitude of the main chassis beam deformation apparently differs for each passenger occupancy configuration, which consequently leads to significantly different RPD values for each configuration. However, it cannot be stated with absolute certainty that the effect of the beam deformation fully accounts for the large difference in RPD values between both methods. Therefore, no clear explanation can be formulated that fully elucidates the origin of the increased RPD values for the transient analysis. Aside from the RPD values for the sum of forces in  $x$ -direction, the magnitudes of the RPD values for the other load types are in general acceptable though. The finite element model used for the transient analysis is therefore approved, despite the apparent increase in RPD values with respect to the optimized method. The acceptable RPD values for the majority of the load types namely outweigh the large RPD values for the sum of forces in  $x$ -direction.

The Von Mises stresses and nodal displacements resulting from the transient analyses are presented in respectively Table 3.5 and 3.6. The largest values for  $\sigma_{max}$  and  $d_{max}$  in these tables are observed for the first passenger occupancy configuration. This configuration is characterized by four passengers and thus the largest mass of the rotating part, which explains the relatively large stress and displacement values for this configuration. However, the mass of the rotating part is not the only parameter that contributes to the stress and displacement levels, as can be deduced from a comparison between the second and third configuration. The second passenger occupancy configuration includes three passengers, while two passengers are seated on one side of the gondola in case of the third configuration. The relatively large mass imbalance associated with the third passenger configuration leads to larger stresses and displacements than the second configuration, despite the lower total mass. The increased rotational velocity of the gondola due to the mass imbalance apparently leads to larger stress and displacement values. This observation indicates that the rotation of the gondola can significantly affect the stress and strain levels in the main chassis beam. Hence, it was correct to consider all passenger occupancy configurations during the conventional, optimized, and transient analysis.

|                | $\sigma_{max}$ [MPa] |                 |                 |                 |
|----------------|----------------------|-----------------|-----------------|-----------------|
|                | Configuration 1      | Configuration 2 | Configuration 3 | Configuration 4 |
| <b>Minimum</b> | 3.53                 | 3.47            | 2.80            | 3.03            |
| <b>Average</b> | 8.89                 | 8.15            | 7.50            | 6.85            |
| <b>Maximum</b> | 27.23                | 24.80           | 25.45           | 18.27           |

Table 3.5: The maximum Von Mises stress values determined with the transient analysis

|                | $d_{max}$ [mm]  |                 |                 |                 |
|----------------|-----------------|-----------------|-----------------|-----------------|
|                | Configuration 1 | Configuration 2 | Configuration 3 | Configuration 4 |
| <b>Minimum</b> | 0.032           | 0.033           | 0.029           | 0.028           |
| <b>Average</b> | 0.088           | 0.083           | 0.076           | 0.070           |
| <b>Maximum</b> | 0.217           | 0.197           | 0.199           | 0.160           |

Table 3.6: The maximum nodal displacements determined with the transient analysis

A comparison between the results obtained with the transient analysis and the optimized method shows that  $\sigma_{max}$  and  $d_{max}$  are practically similar for both analyses. Hence, the optimized method with 39 loadcases predicts similar maximum stress and displacement values as a transient analysis with 1604 loadcases. The evaluation of the 39 loadcases of the optimized method takes approximately fifteen minutes, while the total duration of calculating 1604 loadcases exceeds eight hours. If the prediction of the maximum stress and displacement is main objective of the analysis, it is advisory to use the optimized method instead of the conventional or transient analysis. The stress and displacement values obtained with the optimized method namely have a negligible error with respect to a transient analysis.

For both the transient analysis and the optimized method, a loadcase that leads to a maximum stress or strain level can be related to a certain passenger occupancy configuration and a specific moment in time. Given a maximum stress or displacement, it is therefore possible to determine the corresponding passenger occupancy configuration and the location of the vehicle along the track lay-out with both methods. Only the transient analysis enables a graphical plot of the stress or displacement level as a function of time though, as depicted in Figure 3.6 for the first passenger occupancy configuration. Appendix N presents the stress and displacement levels as a function of time for the other three configurations. It can be deduced from these graphs that aside from the maximum stress or displacement value, the passenger occupancy configuration also affects the development of the stress or displacement as a function of time. In general, the stress and displacement profiles are to a large extent similar. However, a comparison between these graphs also indicates that different passenger occupancy configurations can significantly impact the maximum stress and displacement levels at specific sections along the track lay-out. This again underscores the significance of loads induced on the main chassis beam by spinning gondolas.

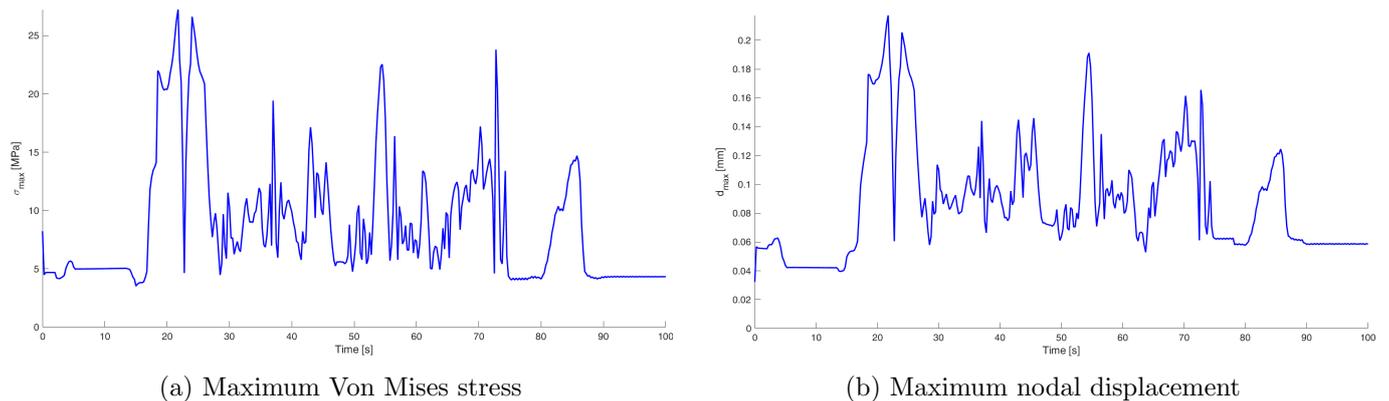
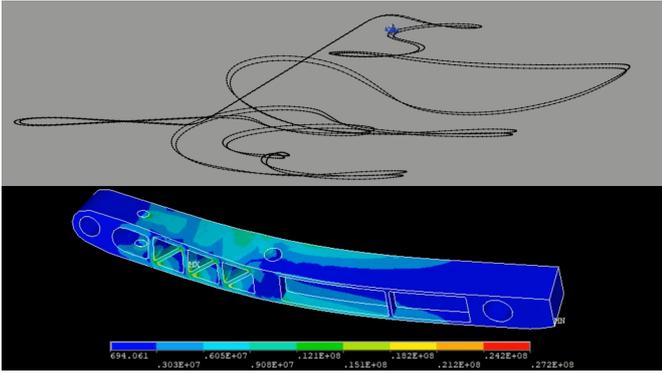
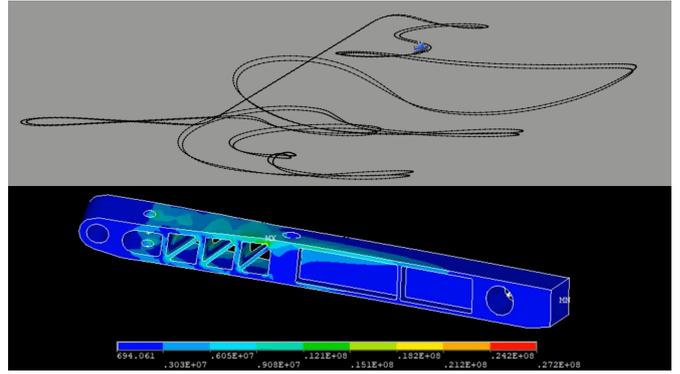


Figure 3.6: The maximum Von Mises stress and nodal displacement as a function of time for the first passenger occupancy configuration

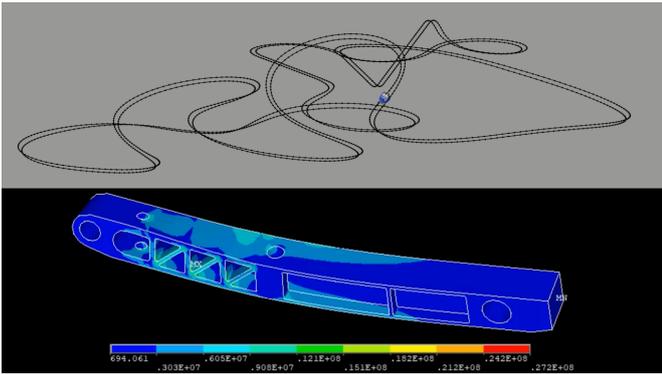
The deformation of the main chassis beam is shown alongside the kinematic model in Figure 3.7. The plot of the kinematic model namely provides an indication for the location of the vehicle along the track lay-out. The corresponding deformation plots depict the eight largest maximum Von Mises stress values found for the first passenger configuration. Hence, these maximum values correspond with the eight largest peaks in the stress profile depicted in Figure 3.6a. Please note that the deformation of the main chassis beam in these plots has been exaggerated by means of a scale factor.



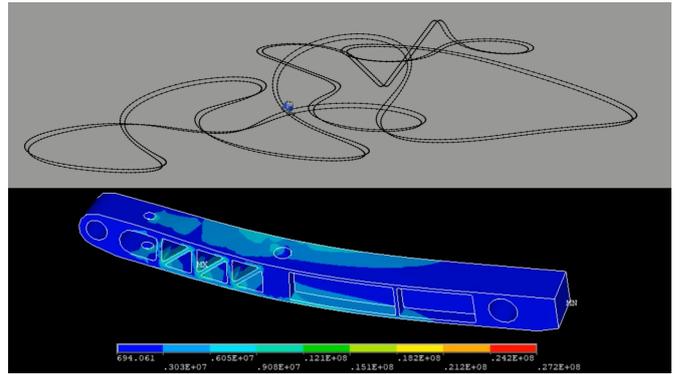
(a)  $t = 21$  s



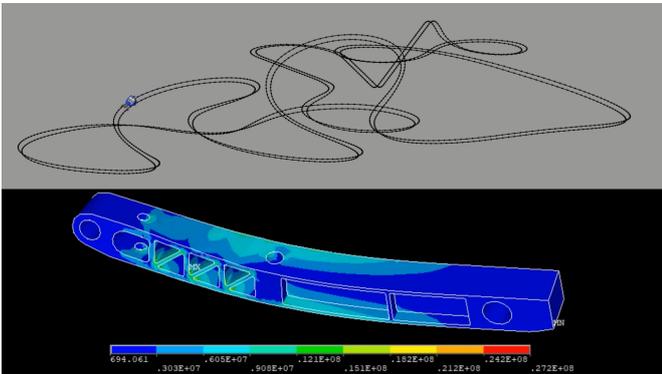
(b)  $t = 24$  s



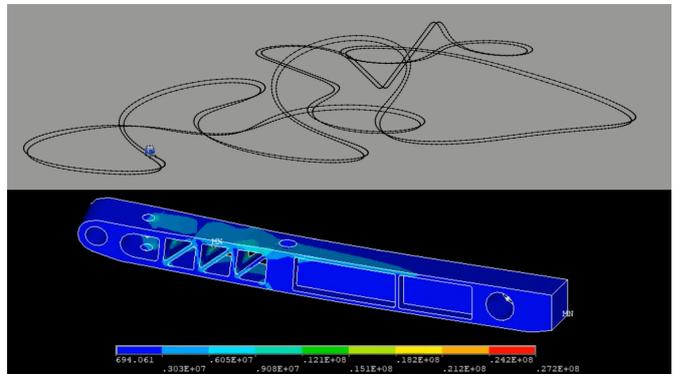
(c)  $t = 37$  s



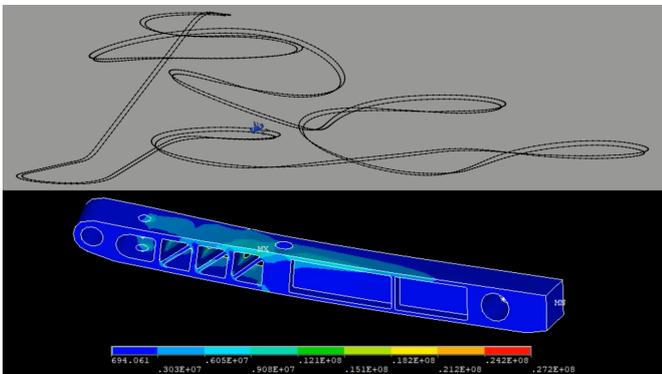
(d)  $t = 43$  s



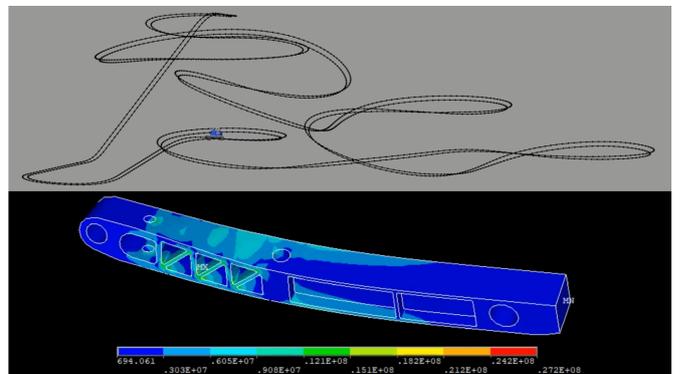
(e)  $t = 54$  s



(f)  $t = 56$  s



(g)  $t = 70$  s



(h)  $t = 73$  s

Figure 3.7: The deformation of the main chassis beam for the first passenger occupancy configuration

### 3.6 Mode-Superposition Method

A major advantage of the optimized method over the conventional method or a transient analysis is the ability to accurately predict the maximum stress and displacement values while the computational effort is limited. Aside from the maximum stress and displacement results, the optimized method also yields the stress and displacement values for a substantial number of other loadcases. However, under some circumstances the stress and displacement levels at each discrete time-step can be of interest. A preliminary stress analysis for each load in a certain loading cycle is for instance required by a fatigue analysis. An adequate fatigue analysis could be performed if this loading cycle is represented by a sufficiently large number of loadcases. A transient analysis consists of a vast number of load cases in chronological order, which implies that a certain load cycle can be more accurately described by a transient analysis than by the optimized method. A transient analysis namely results in a stress profile that can directly be used as input for a subsequent fatigue analysis. The substantial computational time associated with a transient analysis typically restrains the number of loadcases considered by such an analysis though. It would therefore be of great interest to reduce the computational time of a transient analysis at a minimum cost of accuracy.

When analyzing the response of linear structures such as the main chassis beam, mode-superposition is a powerful technique for reducing the computational time. The principle of mode-superposition is based on the assumption that the displacement of the structure can be written as a linear combination of its eigenmodes. Hence, this method requires a preliminary modal analysis, as the response of a structure can be approximated by superposition of a limited number of eigenmodes. Higher modes may have an insignificant influence on the dynamic behaviour of the structure, such that they could be neglected. Application of the mode-superposition technique requires a good deal of understanding of the problem though. The key to a successful performance of a mode-superposition analysis is to extract and consider all modes that may contribute to the dynamic response.

#### 3.6.1 Modal Analysis

The choice for a certain number of modes basically boils down to a trade-off between accuracy and computational time. To make a deliberate choice on the number of modes to consider, the eigenmodes of the main chassis beam are determined by means of a modal analysis. The model considered in the modal analysis is similar to the model used in the full transient analysis. Hence, the pilot nodes at the front spherical joint and rear revolute joint are constrained in a similar way as described earlier. The first three eigenmodes resulting from the modal analysis are depicted in Figure 3.8, while the remaining mode shapes associated with a frequency range up to a 1000 Hz are presented in Appendix O.1. The eigenmodes beyond the thirteenth eigenmode are therefore presumed to have an insignificant influence on the dynamic behaviour of the main chassis beam. The first eigenmode of the main chassis beam is described by transverse bending of the beam, while the second and third eigenmode correspond to respectively vertical bending and torsion of the beam around its longitudinal axis. Since these first three eigenmodes describe the first bending and torsion modes of the beam, the first mode-superposition analysis will comprise these three modes. The figures in Appendix O.1 show that higher bending or torsion modes can be identified at elevated frequencies. The next transverse bending mode, vertical bending mode, and torsion mode are found at respectively the fourth, sixth, and seventh eigenmode. The fifth eigenmode on the other hand, is characterized by longitudinal vibration of the

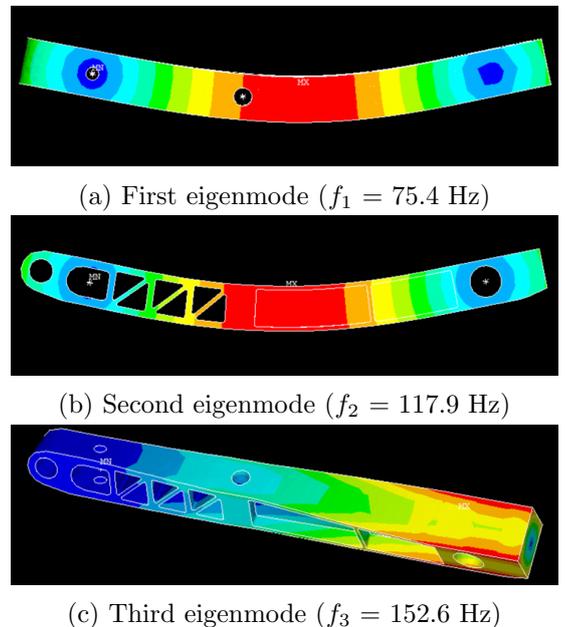


Figure 3.8: The first three eigenmodes

main chassis beam. The second mode-superposition analysis will be performed using the first seven eigenmodes. Higher bending and torsion mode shapes can be found up to the tenth and thirteenth eigenmode. Hence, the mode-superposition analysis will also be performed using respectively the first ten and thirteen eigenmodes.

### 3.6.2 Mode-Superposition Transient Analysis

The model that is considered during the mode-superposition transient analysis is the same as that described for the full method, including the constraints, prescribed accelerations, and applied loads. As mentioned earlier, four different mode-superposition transient analyses will be performed at which the modal expansion is truncated at respectively the third, seventh, tenth, and thirteenth eigenmode. Since the expansion is performed on these selected modes only, computational time will be saved in the subsequent mode-superposition transient analysis. The resultant maximum Von Mises stresses and nodal displacement sums for the first passenger occupancy configuration are presented in Figure 3.9 for the four mode-superposition transient analyses and the full transient analysis. The reader is referred to Appendix O.2 for a comparison between the results of the full and mode-superposition transient analyses for the other passenger occupancy configurations. From these comparisons it can be deduced that the mode-superposition method is incapable of correctly predicting the stress or displacement values when truncated at the third eigenmode. Both the predicted stress and displacement values are substantially lower than the results obtained with a full transient analysis. In terms of the predicted maximum nodal displacement sums, the results are significantly better for the mode-superposition transient analysis with seven eigenmodes. However, the difference in predicted maximum stress values with respect to the full transient analysis is still substantial. In comparison to the number of eigenmodes that are required for an adequate representation of the displacements, more eigenmodes must generally be used in the superposition to obtain appropriate stress results. Higher eigenmodes are typically characterized by more complex mode shapes, which means that the derivatives of the displacements and thus the strains are consequently higher as well. This explains the relatively low stress values that are predicted by the mode-superposition transient analysis when truncated at three or seven modes.

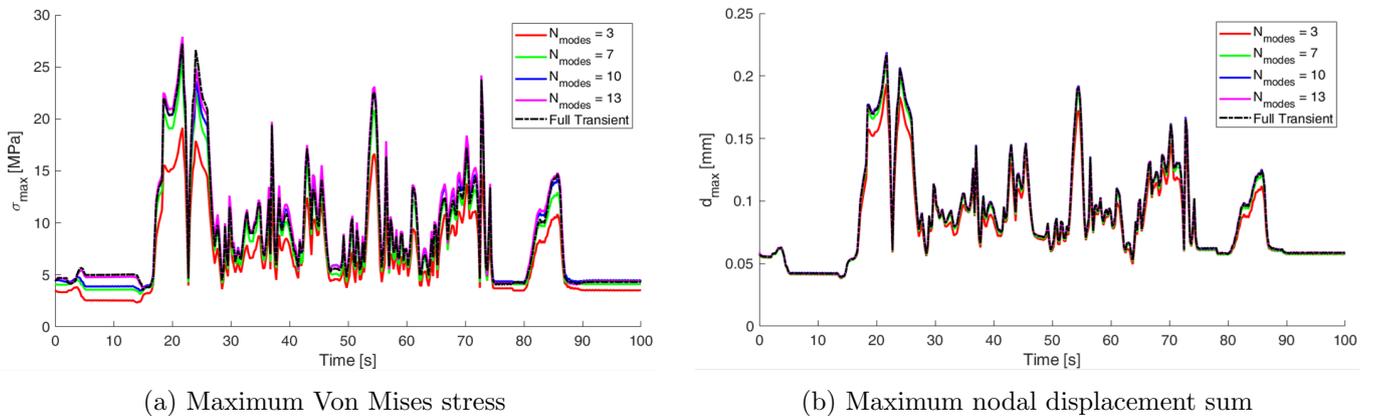


Figure 3.9: Comparison between the full and mode-superposition transient analyses for the first passenger occupancy configuration

For the mode-superposition transient analysis with ten or thirteen modes, the predicted stress and displacement levels approach the values predicted by the full transient analysis. The mean deviations from the full transient results are presented in Table 3.7. In agreement with the graphs in Figure 3.9, the mean deviation from the full transient stress and displacement results is substantial for the mode-superposition analyses with three or seven modes. These two mode-superposition analyses are therefore no longer regarded as relevant, and the focus is shifted to the mode-superposition analyses with ten or thirteen modes. The mean stress deviation is practically comparable for these two analyses, while the deviation from the displacement levels predicted by the full transient analysis is in both cases negligible. For the mode superposition analysis with ten modes, the mean deviation from the stress levels predicted by the full transient analysis is even

slightly lower. On the other hand, the maximum stress deviation is significantly larger in comparison to the analysis with thirteen eigenmodes. Hence, based on the results in Table 3.7, no clear preference can be expressed for either of the two mode superposition analyses. Since both analyses yield practically similar results, the computational time is the only remaining factor that can make a clear distinction between both mode-superposition analyses. The average computational time is three minutes longer for the mode-superposition analysis with thirteen modes, which corresponds to a proportional increase of 16.2% with respect to the analysis with ten eigenmodes. Therefore, the mode-superposition analysis with ten modes (up to  $f = 700$  Hz) appears to yield the best ratio between accuracy and computational time, while the analysis with thirteen eigenmodes (up to  $f = 1000$  Hz) gives slightly better overall results.

| $N_{\text{modes}}$ | $\Delta\sigma_{\text{eq,max}}$ [MPa] |         | $\Delta d_{\text{sum,max}}$ [mm] |                       | CPU Time [min] |
|--------------------|--------------------------------------|---------|----------------------------------|-----------------------|----------------|
|                    | Mean                                 | Maximum | Mean                             | Maximum               |                |
| <b>3</b>           | 1.84                                 | 9.08    | $5.088 \cdot 10^{-3}$            | $23.40 \cdot 10^{-3}$ | 13.75          |
| <b>7</b>           | 0.67                                 | 4.12    | $1.385 \cdot 10^{-3}$            | $5.270 \cdot 10^{-3}$ | 17.50          |
| <b>10</b>          | 0.37                                 | 2.97    | $0.352 \cdot 10^{-3}$            | $1.955 \cdot 10^{-3}$ | 18.50          |
| <b>13</b>          | 0.38                                 | 2.14    | $0.221 \cdot 10^{-3}$            | $2.010 \cdot 10^{-3}$ | 21.50          |

Table 3.7: Deviation of the results obtained with the mode-superposition method with respect to the results of a full-transient analysis, including the average computational time for each mode-superposition analysis

The maximum Von Mises stresses and nodal displacement sums are compared in Table 3.8 to the results from a full transient analysis. The maximum stress and displacement levels are relatively low for the mode-superposition analyses with three or seven modes, which is in alignment with the earlier observations in Figure 3.9. The results are again practically similar for the mode-superposition analyses with ten and thirteen eigenmodes. Only a minor deviation can be found from the full transient results for both the resultant maximum stress and displacement values. However, the predicted maximum stress and displacement levels are still not as accurate as the values predicted by the optimized method. Furthermore, the computational time associated with the optimized method is still substantially shorter than the computational times for any of the mode-superposition transient analyses. The optimized method is therefore preferred if predicting the maximum stress and displacement values is the primary objective of the analysis. On the other hand, the mode-superposition analysis with ten or thirteen modes is most appropriate when also the stress and displacement profiles are of interest. Both analyses can predict the stress and displacement profiles with only a minor error and a dramatic reduction in computational time with respect to a full transient analysis. The mode-superposition transient analysis with ten eigenmodes is especially suitable for loading cycles with a large number of load cases, since the stress and displacement profiles are computed relatively accurately and efficiently. The computational time reduces with over eighty percent, while the maximum stress and displacement levels are calculated with an error of less than one percent. Furthermore, the predicted stress and displacement profiles are characterized by a minor deviation from the results of a full transient analysis, as mentioned earlier. Please note that the time associated with data export is not included in computational times in Tables 3.7 and 3.8. The stress and displacement results of the mode-superposition transient analyses are presented in Appendix O.3 for each passenger occupancy configuration.

| $N_{\text{modes}}$ | $\sigma_{\text{eq,max}}$ [MPa] | $d_{\text{sum,max}}$ [mm] | CPU Time [min] |
|--------------------|--------------------------------|---------------------------|----------------|
| <b>Full</b>        | 27.23                          | 0.217                     | 110            |
| <b>3</b>           | 19.15 (-29.67%)                | 0.194 (-10.60%)           | 12 (-89.09%)   |
| <b>7</b>           | 25.51 (-6.32%)                 | 0.212 (-2.30%)            | 18 (-83.64%)   |
| <b>10</b>          | 27.04 (-0.70%)                 | 0.219 (+0.92%)            | 19 (-82.73%)   |
| <b>13</b>          | 27.90 (+2.46%)                 | 0.218 (+0.46%)            | 22 (-80.00%)   |

Table 3.8: Comparison between the mode-superposition methods and a full-transient analysis in terms of the maximum Von Mises stress, maximum displacement sum, and computational time

## 4. Conclusions

A kinematic model was developed in Simulink to determine the loads exerted on the main chassis beam of a spinning rollercoaster vehicle. To create a model analogous to an actual spinning rollercoaster, the correct masses and degrees of freedom are assigned to the bodies of the vehicle multibody model. After friction and drag forces were added to the model, the passenger accelerations were compared to the regulatory limits to improve the track lay-out. The compliance of the passenger accelerations to the corresponding limits implies that the forces and moments sensed at the joints are of realistic magnitude. These loads are subsequently validated before they can be applied on the finite element model of the main chassis beam. A successful validation of all reaction forces and moments could be performed based on the masses and the accelerations of the bodies that comprise the multibody model. Therefore, it can be concluded that the loads exerted on the main chassis beam are correct and of realistic magnitude.

The forces and moments at the pivot joint are converted into static loadcases by means of two different methodologies, namely the conventional and optimized method. The respectively twelve and thirty-nine distinct static loadcases of these methods prescribe different load combinations that are applied on the finite element model of the beam in Ansys. The constraints could only be prescribed correctly on so-called pilot nodes, which are located at the centres of respectively the front spherical and rear revolute joint. The loads are applied on an additional pilot node at the centre of the pivot joint. The finite element model is validated by means of a comparison between the reaction forces at the pilot nodes and at the joints in the kinematic model. The difference in terms of the reaction forces between both models is quantified by a measure called the relative percentage difference. When the contribution from outliers is omitted, the relative percentage difference values are generally acceptable and therefore the finite element model can be approved.

The static loadcases of the conventional and optimized method are applied on the model, and the resultant stress and displacement levels are compared to each other. The magnitudes of the stress and displacement values are relatively low due to the robust design of the main chassis beam. Nonetheless, for loads from the same kinematic model, the maximum stress and displacement values predicted by the optimized method are nearly ten percent lower with respect to the conventional method. To verify the resultant maximum stress and displacement values obtained with the optimized method, a transient analysis with a large number of loadcases is conducted. Contrary to the distinct static analyses of the conventional and optimized methods, a transient analysis also considers the deformation of the main chassis beam and accumulated stress levels caused by previous loadcases. A comparison between the results obtained with the optimized method and the transient analysis shows that the maximum equivalent stress and displacement sum are practically similar for both analyses. Thanks to a relatively accurate prediction of the maximum stress and displacement with limited computational effort, it is therefore advisory to use the optimized method instead of the conventional or transient analysis if these results are the primary objective of the analysis.

On the other hand, a transient analysis could be required if the resultant stress or displacement of each load in a certain load cycle is of interest, rather than solely the maximum values. Contrary to the conventional or optimized methods, a transient analysis can be used to perform a preliminary stress or displacement analysis for a large number of consecutive loadcases. The long computational times associated with a full transient analysis can be reduced by more than eighty percent if the mode-superposition technique is used. A preliminary modal analysis is required by a mode-superposition transient analysis, so all eigenmodes that may contribute to the dynamic response are linearly superpositioned. If ten or more eigenmodes of the main chassis beam are taken into account, the stress and displacement profiles are predicted with only a minor error and a dramatic reduction in computational time with respect to a full transient analysis. Therefore, given that a sufficiently large number of eigenmodes is considered, it can be concluded that a mode-superposition transient analysis yields a more favourable ratio between accuracy and computational time.

## 5. Recommendations

A general framework has been presented in which the loads from a kinematic model are converted into loadcases, which are subsequently applied on a finite element model to determine the maximum stress and displacement levels. The work presented in this thesis evidently leaves room for improvement, and hence recommendations for further research are presented in this chapter. Certain methodologies used in this research are also discussed in a general sense to clarify the main incentives for the recommended improvements.

**Modelling an entire train:** The current kinematic model comprises a single vehicle, whereas rollercoasters often feature an entire train. A train typically reaches higher velocities than a single vehicle due to the larger total mass, which could in turn lead to larger accelerations in case of an identical track lay-out. The accelerations experienced by the passengers is consequently expected to approach the regulatory limits more closely. The improvements to the track lay-out have fortunately resulted in a sufficiently large margin between the combined acceleration factor and the corresponding limit. The regulatory limits are therefore not expected to be violated in case of a short train consisting of only two or three coaches. However, certain adjustments to the track lay-out could be required if a longer train is used or if the mass properties of the vehicles or its passengers change. An extension of the kinematic model towards a full train could also reveal the effects of the interaction between the separate coaches. The forces and moments at the spherical joints between the coaches are not considered in the current finite element model, since the kinematic model consists of a single vehicle only. Hence, the performance of a finite element analysis for the chassis beam of each individual coach could indicate the influence of these loads on the stress and deformation levels of the beams. Expanding the kinematic model to a full train could therefore prove to be a valuable addition to the work presented in this thesis.

**Modelling wheel-rail contact:** The acceleration signals have to a certain extent been idealized due to the use of point-on-curve constraints between the wheels and the track. Vibrations caused by the wheel-rail contact do therefore not contribute to the acceleration signals. Hence, inclusion of the wheel-rail contact in the kinematic model is suggested if the accelerations experienced by the passengers are of primary interest. Additionally, the forces and moments exerted on the main chassis beam could also be affected by the modelling of play between the wheels and rails. Hence, the inclusion of the wheel-rail contact in the model might turn out to be relevant for the loads induced on the beam. For modern rollercoasters with a smooth track geometry, this effect is presumed to be minor though.

**More robust motion control:** The kinematic scheme has been expanded with a number of blocks that dictate the motion of the vehicle and gondola at certain track sections. The translational velocity of the vehicle and the angular velocity of the gondola should follow prescribed reference signals at the station, lifthill, and brake section. The loads on the main chassis beam are determined by the kinematic model for four different passenger occupancy configurations, and for all four configurations the vehicle should be capable of following the prescribed reference profiles. Determining the appropriate controller settings is a delicate process due to the different vehicle properties for each configuration, especially with respect to the gondola rotational inertia. These controller settings are to a large extent tailored to the vehicle properties and the track geometry, and finding properly functioning controller settings and reference profiles has therefore proven to be a trial-and-error process. It is undesirable if each adjustment to the vehicle geometry or track lay-out should be followed by a costly iterative determination of the appropriate settings. Therefore, if the loads for the entire ride are to be considered, more robust and sophisticated control schemes are required. On the other hand, it is sufficient to only model the gravity run if the loads induced on the beam at the lifthill, brake section, and station are known to be irrelevant. In that case, the vehicle can be released from the top of the lifthill with an initial velocity equal to the chain velocity on the lifthill. The simulation can subsequently be terminated when the vehicle reaches the brake section. Therefore, in light of the sensitive controller settings, it is recommended to carefully consider the necessity of prescribing the vehicle velocity at the sections outside the gravity run. Composing properly functioning control schemes has namely proven

to be an expensive procedure in terms of time and effort. The addition of motion control has fortunately turned out to be worthwhile, since the loads induced on the main chassis beam at for instance the brake section have contributed to the loadcases considered in the finite element analysis.

**Shorter sample time:** The conversion of forces and moments at the pivot joint into loadcases for a subsequent transient analysis occurred with a sample time equal to 0.25 seconds. The resultant stresses and displacements of the transient analysis are practically equal to the results obtained with the optimized method. The relatively long sample time could be a reason for this similarity, since the effect of transiently evaluating the loadcases might be insignificant for such a long duration between consecutive loadcases. In other words, the deformation of the beam and the accumulated stress levels could only affect the results for a shorter sample time. This would in turn lead to a larger difference between the optimized results and the results obtained with a more extensive transient analysis. However, the choice for the sample time used in this thesis has predominantly been based on the computational time and the available memory. Even with a relatively low sample frequency, the transient analysis features approximately 1600 loadcases in total when all four passenger occupancy configurations are considered. For every ten loadcases that are evaluated, one gigabyte of data is written to a results file. A transient analysis of all loadcases corresponding to a single passenger occupancy configuration therefore results in a data file of approximately forty gigabytes, whose size approaches the available disk space. Furthermore, the computational time of such a transient analysis equals approximately two hours. If a computer with superior specifications is available, it is therefore recommended to examine the influence of a shorter sample time on the results of a transient analysis.

**Using a finer mesh:** An increase in computational power could also provide an opportunity for choosing a finer mesh for the finite element model of the main chassis beam. A mesh refinement typically leads to more accurate results, especially when the mesh is locally refined at for instance gaps or corners. However, the mesh density of the finite element model in this thesis was predominantly restricted by a maximum number of nodes, as specified by the license. Therefore, it is recommended to perform the finite element analyses with software that does not impose restrictions on the maximum number of nodes in the model. A finite element model with a denser mesh is namely expected to result in a more accurate prediction of the stress and deformation levels.

**Expansion of the finite element model:** Increased computational power in combination with cancelled license restrictions could also enable an expansion of the model towards an assembly. A rotational constraint at the rear axle could not be modelled in exact agreement to the kinematic model. This issue was circumvented by a direct application of the reaction torque, whose relation with other parameters became clear from the model validation. An inclusion of the rear axle in the finite element model could provide a more elegant solution to this problem, as the joint between the beam and the rear axle would be modelled more accurately. The pilot nodes at the outer faces of the rear axle should only be assigned a rotational degree of freedom around the longitudinal axis. If the angle between the rear axle and beam is not perpendicular, the reaction torque at the rear pilot node is expected to be in agreement with the reaction torque from the kinematic model. Hence, a direct application of the torque at the rear axle is no longer required for an accurate description of the rear revolute joint. The model could be extended even further for the attainment of realistic reaction forces in longitudinal direction at both axles. The longitudinal translation is namely constrained at the rear axle and set as a degree of freedom at front axle, which results in flawed reaction forces. The complexity of the finite element model is expected to increase dramatically though, and the exact implementation is therefore not further elaborated on. Simpler methods could exist that also allow an accurate modelling of the constraints in longitudinal direction at the front and rear axle.

**Less robust beam design:** The difference between the resultant stress and displacement values for the conventional and optimized method is small due to the robust design of the main chassis beam. It is therefore suggested to perform a similar analysis with a less robust beam design, so the difference between both methods can be distinguished more clearly in terms of the results. Aside from an increase in the absolute

stress and displacement levels, the proportional difference between both methods might also increase. Considering a less robust beam design could reveal opportunities for a redesign of the main chassis beam, which would be a valuable addition to the work presented in this thesis.

**Including safety factors:** The multiplication of the forces and moments by safety factors is expected to result in increased stress and displacement levels. The safety factors were not considered to keep the comparison between both methods as genuine as possible. However, in reality the safety factors account for phenomena whose influence is unknown or hard to implement in the model, such as track irregularities due to production inaccuracies. Hence, the current research can be extended with the inclusion of safety factors, so their influence on the results can be examined.

**Modelling flexible bodies:** The modelling of the main chassis beam as a flexible body in the kinematic model would without doubt be a highly interesting addition to this research. The multibody model considered in this thesis namely assumes that the beam does not deform, since each body is treated as a rigid unit. The rigid-body approximation generally allows for relatively fast simulations with a very acceptable accuracy due to the minor deformation of most bodies. Although the transient analysis revealed that the deformation of the main chassis beam is relatively small as well, it could be interesting to examine the precise effect of the rigid-body assumption. In case of the transient analysis presented in this thesis, the loads from a rigid-body kinematic analysis are namely applied on the finite element model of the beam in its deformed state. This implies that the forces and moments exerted on the deformed beam could deviate slightly from the loads that would result from a flexible-body kinematic analysis. In other words, no coupling is present between the body deformation and the multibody dynamic analysis, since the deformation of the main chassis beam is only determined after the rigid-body kinematic analysis has been completed. Although the modelling of flexible bodies in the Simscape multibody package can prove to be a rather cumbersome procedure, the inclusion of flexibility in the multibody model is known to be possible. The deformed shape of a certain body, including the altered inertia properties, can therefore be directly considered in the dynamic analysis at each time-step. Determining the corresponding equivalent stress levels in the post-processing stage could prove to be a rather complex and computationally expensive procedure though. Certain software packages such as Adams fortunately feature a flexible body integration in a multibody dynamic analysis, so a multibody dynamic stress simulation is directly available to the user. The effect of the coupling can be assessed by comparing the resultant stress and deformation levels from a flexible analysis with the results presented in this thesis. Furthermore, the modelling of a flexible track could also prove to be a valuable addition to the work presented in this research. The current model namely assumes a rigid track, while an actual rollercoaster track is characterized by a finite stiffness. Including a flexible track in the model would reveal the effect of track flexibility on the results, since the coupling between track and vehicle dynamics would be present in the model. However, this coupling is also expected to substantially increase the computational time.

**Validation with measurement data:** The forces and torques at the joints are currently validated by evaluating their relation with the mass of each component and the acceleration at each centre of mass. Hence, this validation procedure focuses in essence on the existence of logical relations between various types of output from the same kinematic model. The validation procedure could therefore be improved with the use of actual acceleration measurement data from an existing spinning rollercoaster. This approach requires reliable acceleration measurements and a model with an accurate description of the spinning rollercoaster track geometry. The resultant acceleration values from these measurements can then be compared to the accelerations determined by the kinematic model. If the measured and calculated acceleration values are in agreement with each other, the accelerations and masses can be multiplied in a similar manner as presented in this report to validate the calculated forces and torques. However, it should be noted that an adequate comparison between the measured and calculated acceleration signals is challenging due to the influence of phenomena that are not accounted for by the model, such as track irregularities or wheel-rail contact.

# Bibliography

- [1] Silver Dollar City. Time Traveler, 2019.
- [2] Mathworks. Spline, 2018.
- [3] Joao Pombo and Jorge Ambrosio. Modelling tracks for roller coaster dynamics. *International Journal of Vehicle Design*, 45(4):470–500, 2007.
- [4] Rene Baptist. Modelling the Vibrational Behaviour of Roller Coaster Trains, 2018.
- [5] Andreas Simonis and Christian Schindler. Measuring the Wheel-Rail Forces of a Roller Coaster. *Journal of Sensors and Sensor Systems*, pages 469–479, 2018.
- [6] Mathworks. Point On Curve Constraint, 2018.
- [7] Yunus A. Cengel and Afshin J. Ghajar. *Heat and Mass Transfer Fundamentals and Applications*. McGraw-Hill Education, New York, 5th edition, 2015.
- [8] Ansys. SOLID186 Element Description, 2019.
- [9] Ansys. Defining the Target Surface, 2019.
- [10] Ansys. CONTA174 Element Description, 2019.
- [11] Anton van Beek. *Advanced Engineering Design: Lifetime Performance and Reliability*. Delft University of Technology, 1st edition, 2006.
- [12] B.F. Bloemendaal. Modelling and Measurement of Rolling Resistance on a Rollercoaster, 2009.
- [13] Millennium Elastomers. Para-Phenylene Diisocyanate, 2019.
- [14] Jignesh P. Sheth, Derek B. Klinedinst, Garth L. Wilkes, Iskender Yilgor, and Emel Yilgor. Role of Chain Symmetry and Hydrogen Bonding in Segmented Copolymers with Monodisperse Hard Segments, 2005.
- [15] H.J. Qi and M.C. Boyce. Stress-Strain Behavior of Thermoplastic Polyurethane, 2004.
- [16] SKF. The SKF Model for Calculating the Frictional Moment, 2019.
- [17] G.K. Batchelor. Introduction to Fluid Dynamics, 1967.

# Appendices

## A. Simulink Model

### A.1 Main Scheme

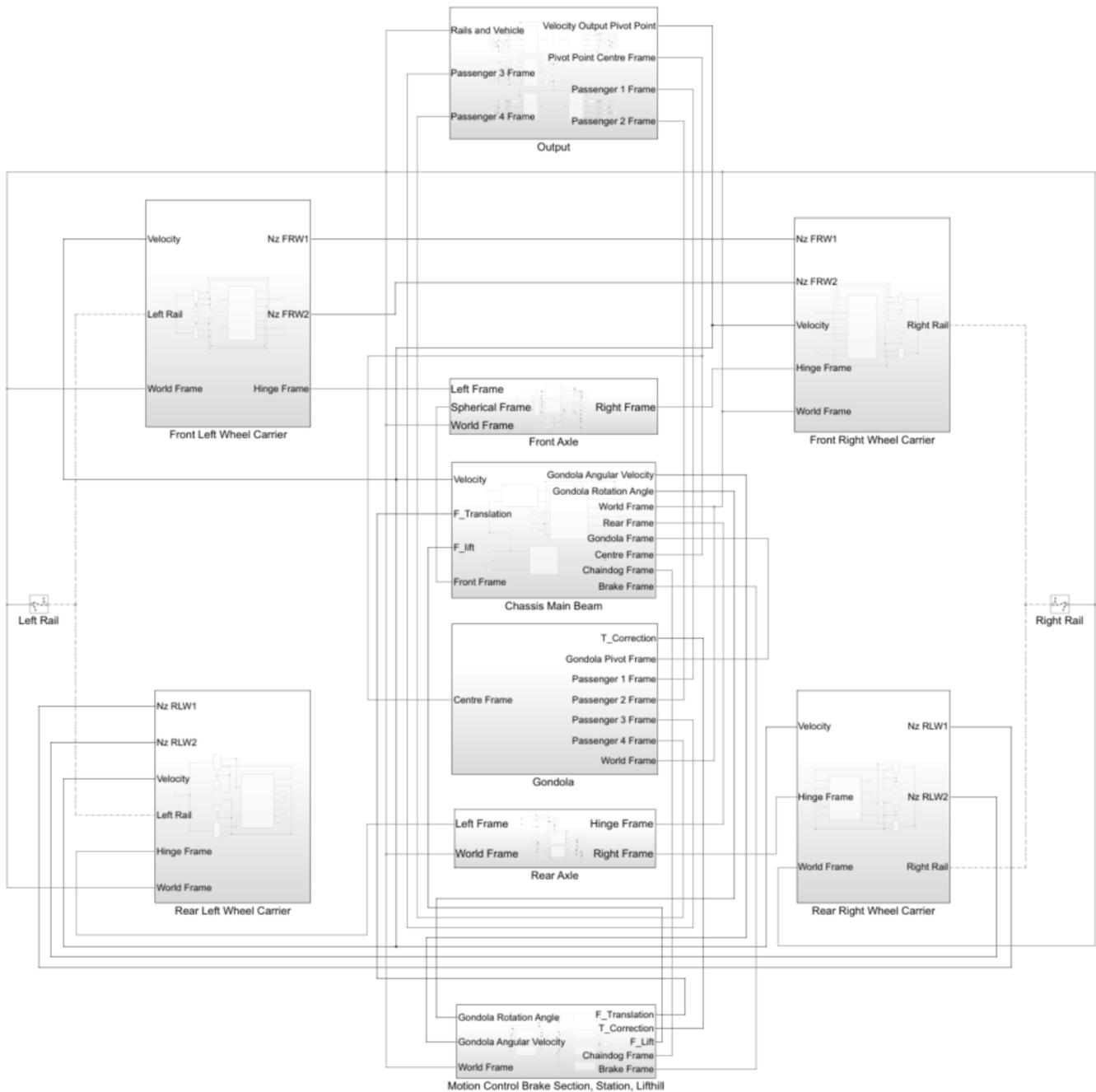


Figure A.1: Main scheme of the Simulink model

## A.2 Front-left Wheel Carrier Schemes

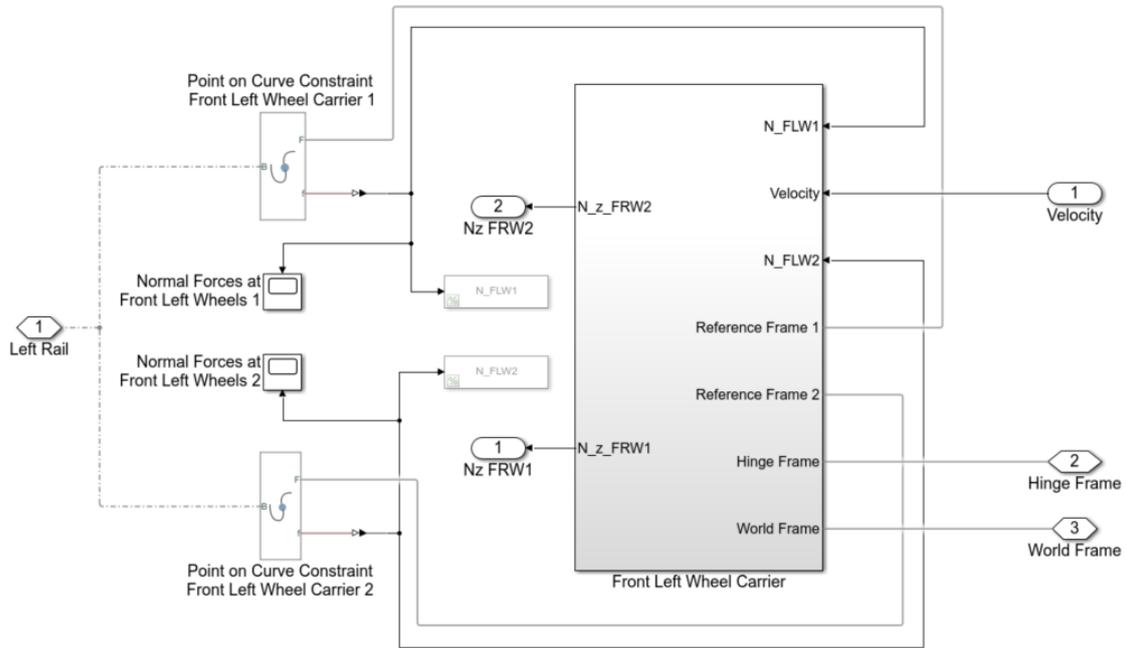


Figure A.2: Main Simulink scheme of the front-left wheel carrier, corresponding to the block named *Front Left Wheel Carrier* in the main scheme (Figure A.1)

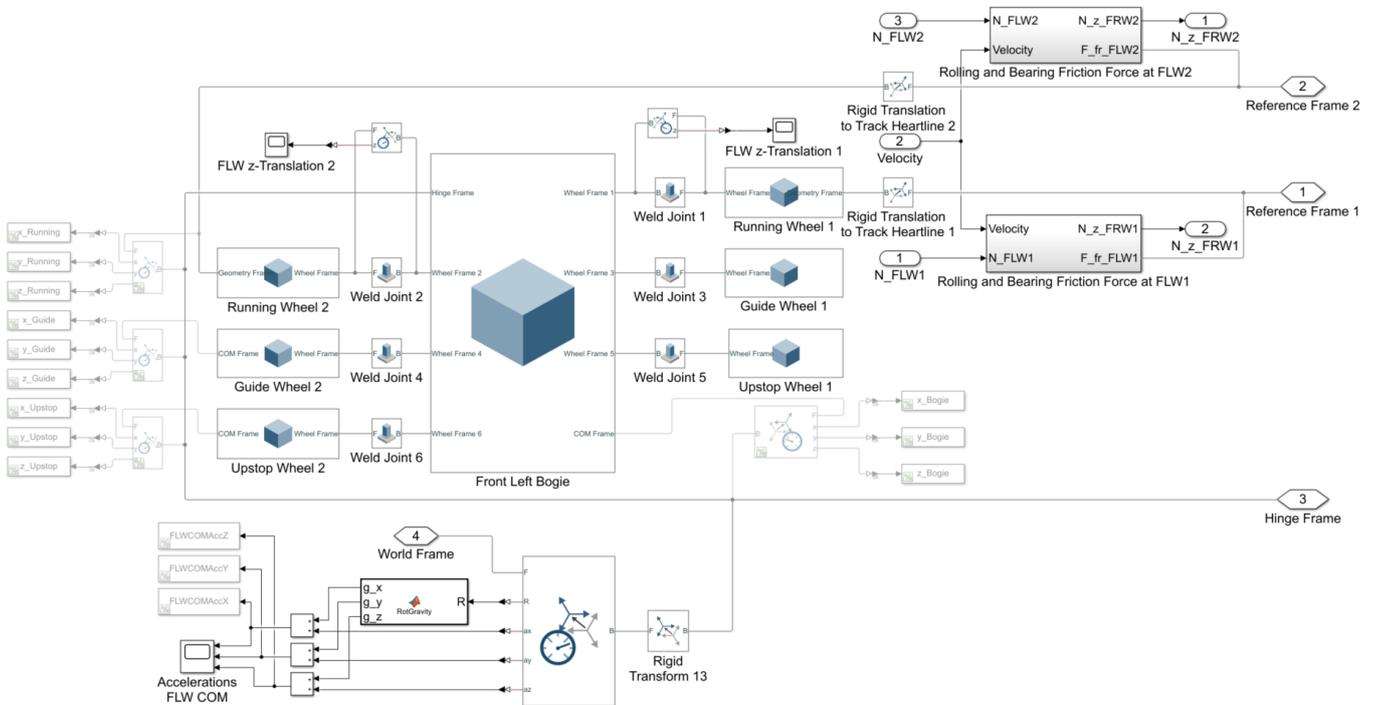


Figure A.3: Simulink sub-scheme of the front-left wheel carrier, corresponding to the block named *Front Left Wheel Carrier* in Figure A.2

### A.3 Front-right Wheel Carrier Schemes

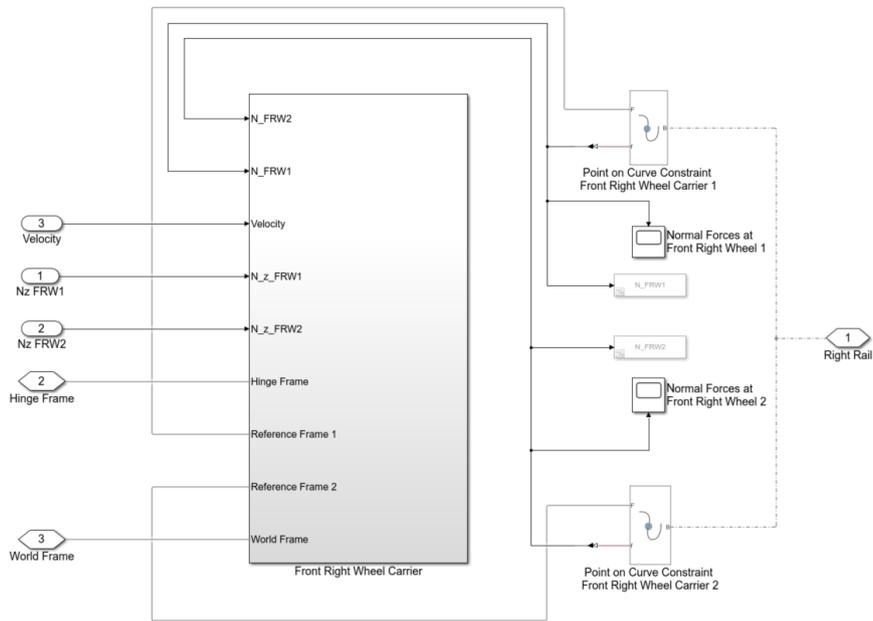


Figure A.4: Main Simulink scheme of the front-right wheel carrier, corresponding to the block named *Front Right Wheel Carrier* in the main scheme (Figure A.1)

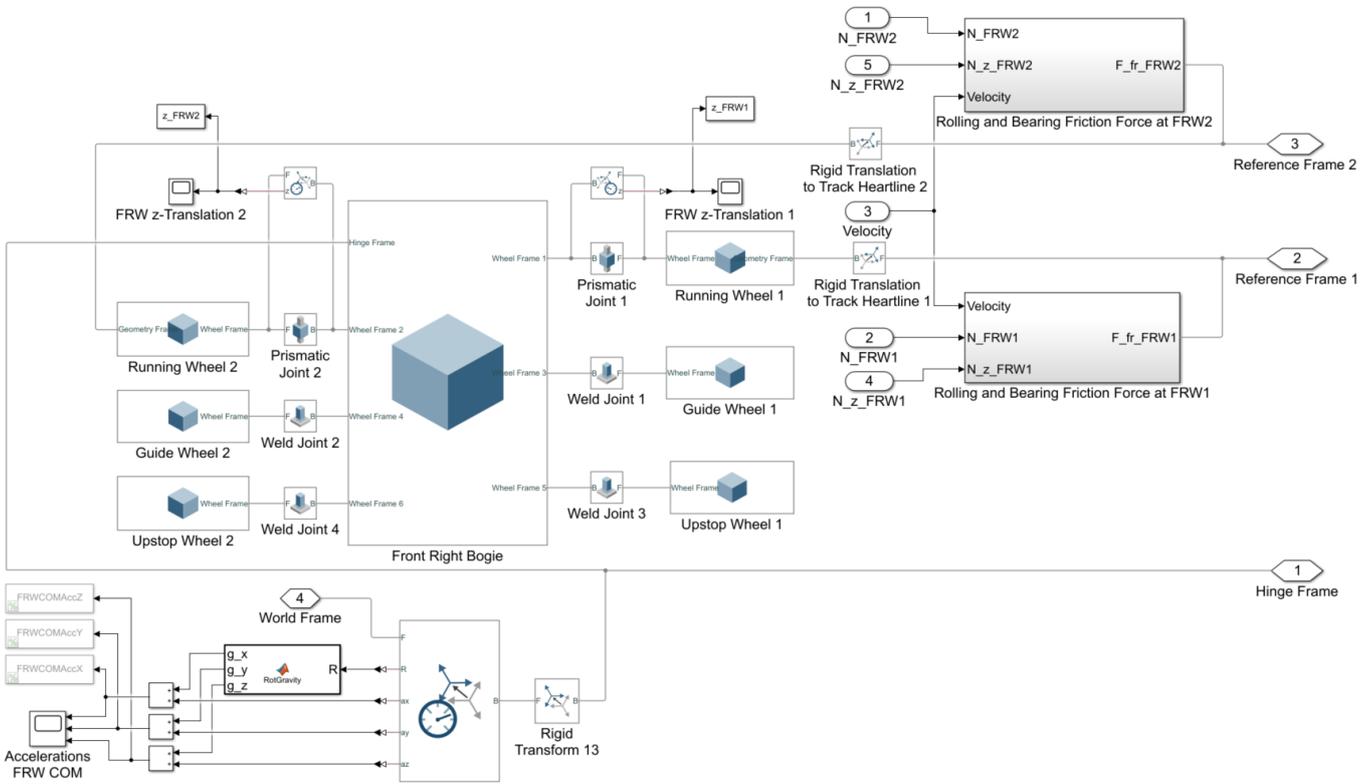


Figure A.5: Simulink sub-scheme of the front-right wheel carrier, corresponding to the block named *Front Right Wheel Carrier* in Figure A.4

## A.4 Rear-left Wheel Carrier Schemes

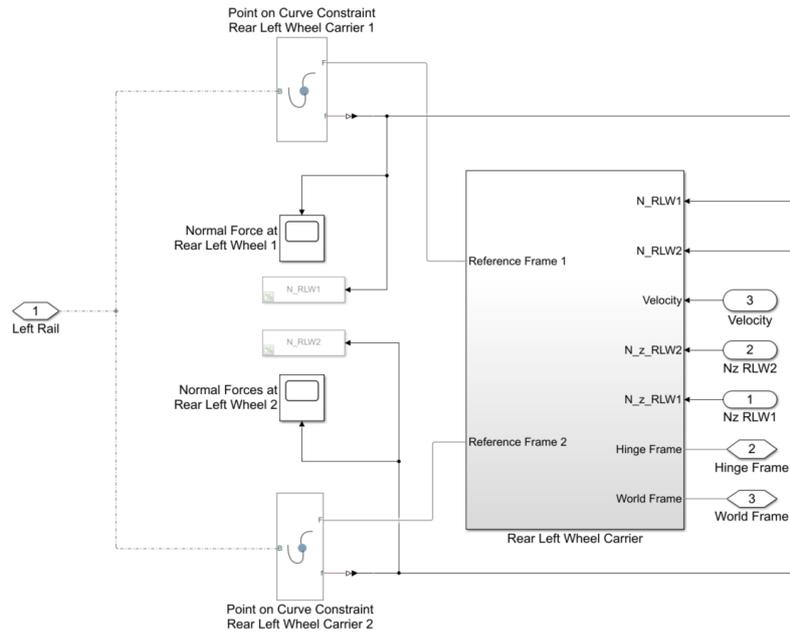


Figure A.6: Main Simulink scheme of the rear-left wheel carrier, corresponding to the block named *Rear Left Wheel Carrier* in the main scheme (Figure A.1)

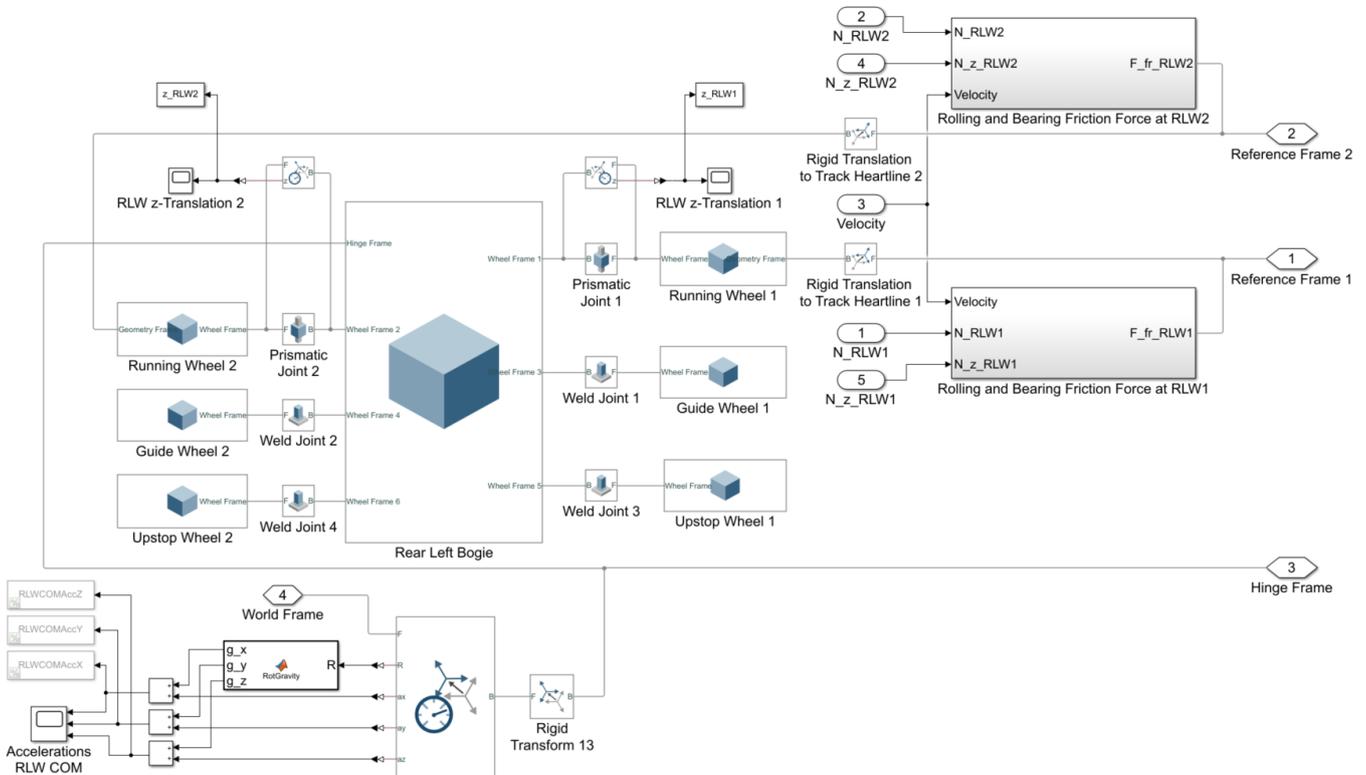


Figure A.7: Simulink sub-scheme of the rear-left wheel carrier, corresponding to the block named *Rear Left Wheel Carrier* in Figure A.6

## A.5 Rear-right Wheel Carrier Schemes

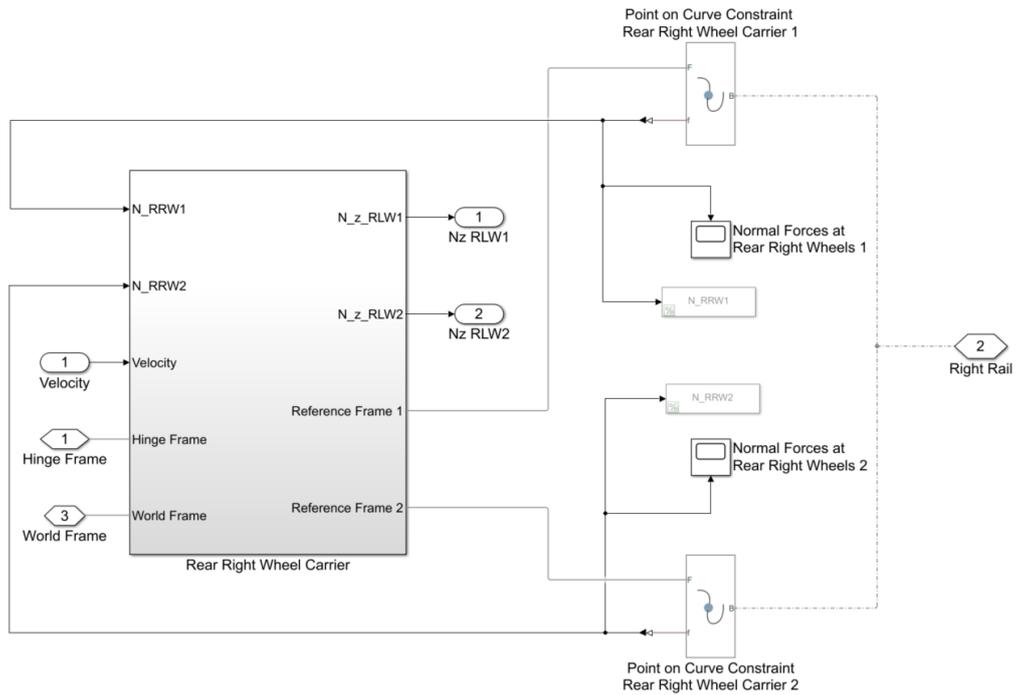


Figure A.8: Main Simulink scheme of the rear-right wheel carrier, corresponding to the block named *Rear Right Wheel Carrier* in the main scheme (Figure A.1)

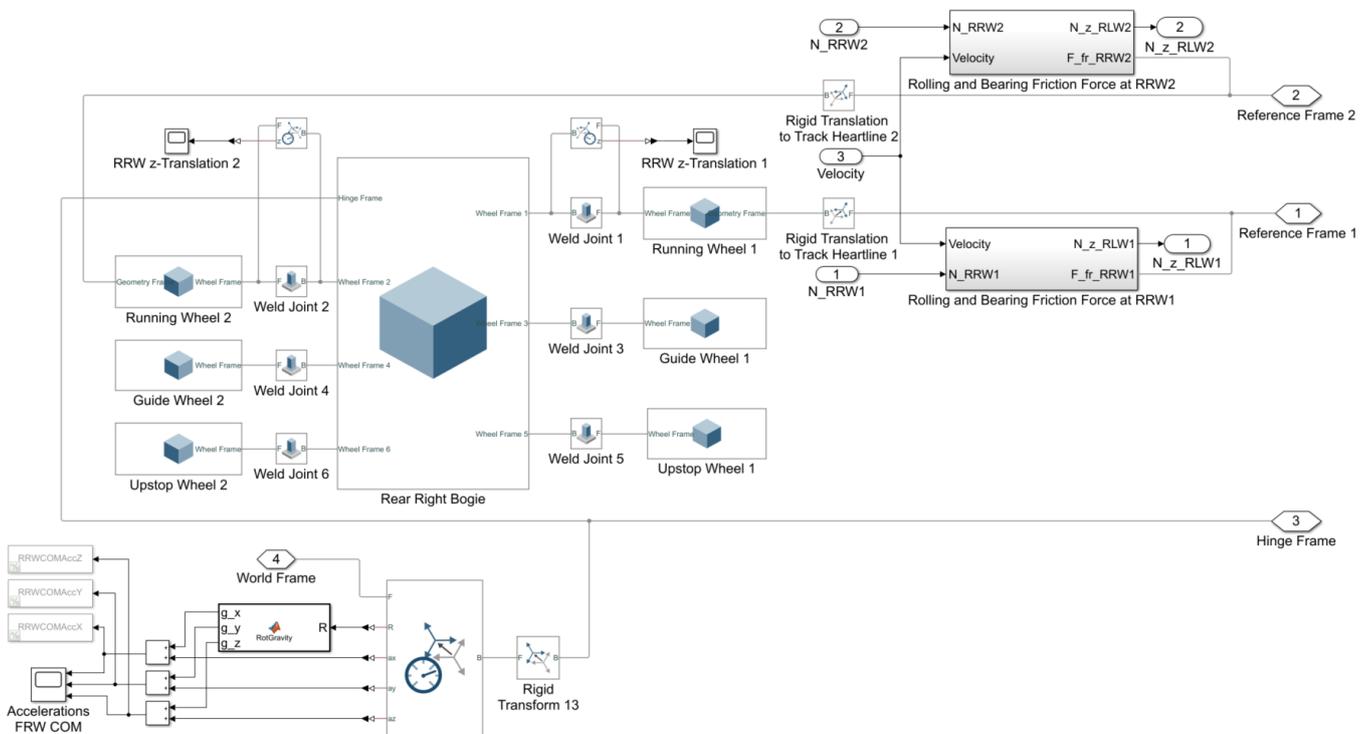


Figure A.9: Simulink sub-scheme of the rear-right wheel carrier, corresponding to the block named *Rear Right Wheel Carrier* in Figure A.8

## A.6 Friction Force Schemes

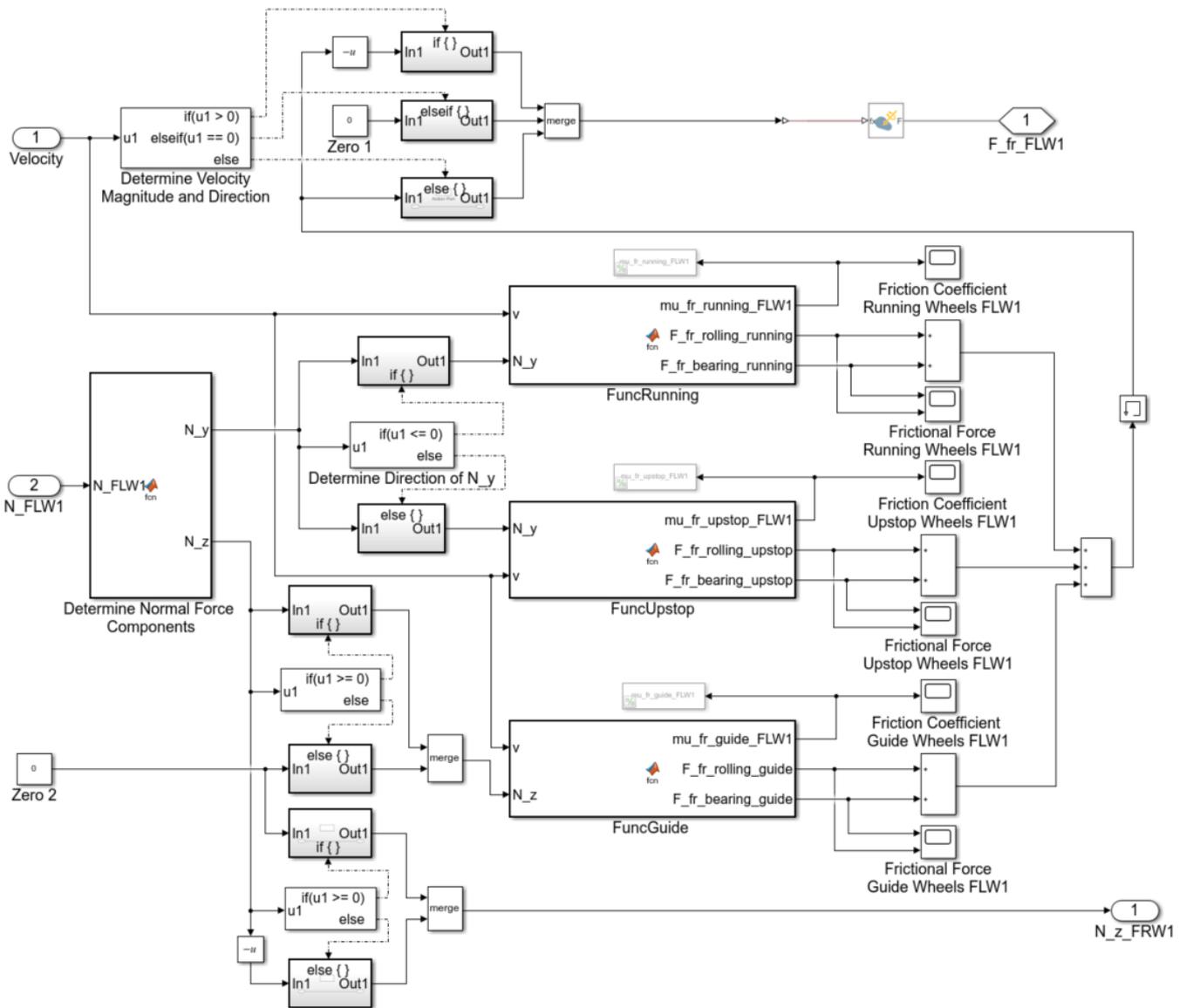


Figure A.10: Simulink scheme for the front-left bogie (corresponding to the block named *Rolling and Bearing Friction Force at FLW1* in Figure A.3) in which the frictional force is determined for a single running wheel, guide wheel, and upstop wheel (the schemes for the remaining wheels and the rear-right bogie are equivalent)

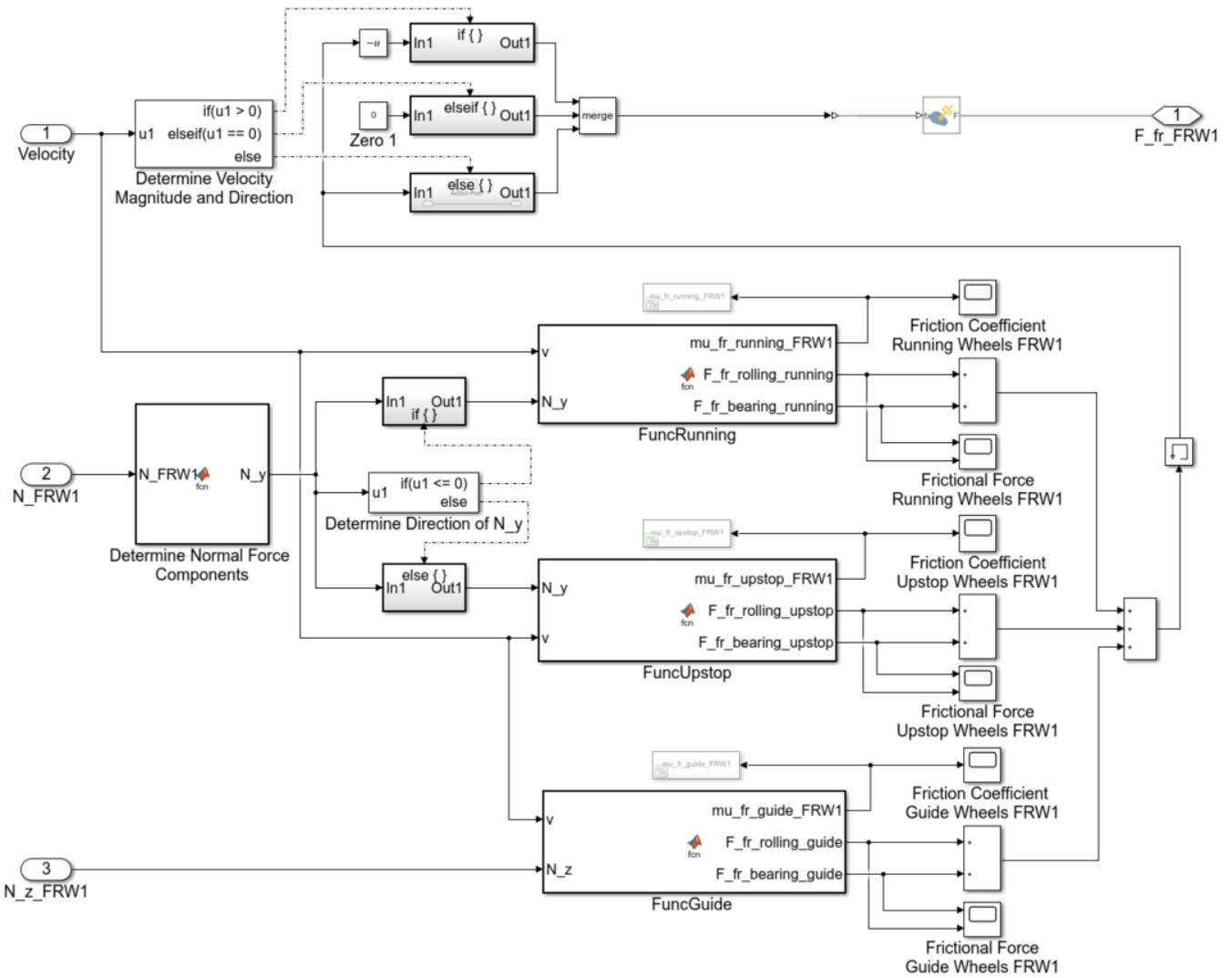


Figure A.11: Simulink scheme for the front-right bogie (corresponding to the block named *Rolling and Bearing Friction Force at FRW1* in Figure A.5) in which the frictional force is determined for a single running wheel, guide wheel, and upstop wheel (the schemes for the remaining wheels and the rear-left bogie are equivalent)

## A.7 Front Axle Scheme

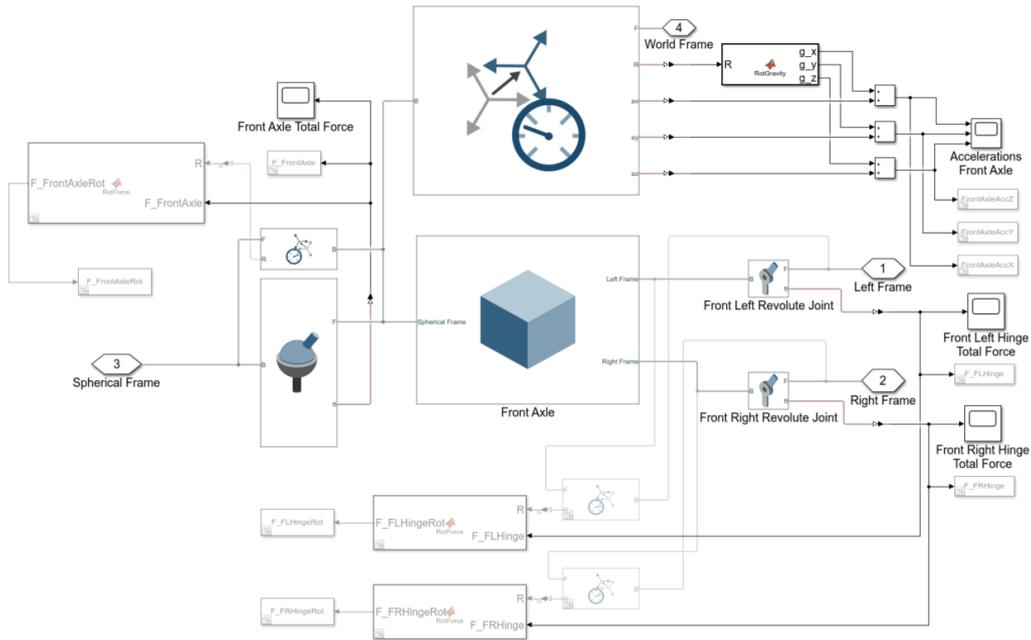


Figure A.12: Simulink scheme for the front axle, corresponding to the block named *Front Axle* in Figure A.1

## A.8 Rear Axle Scheme

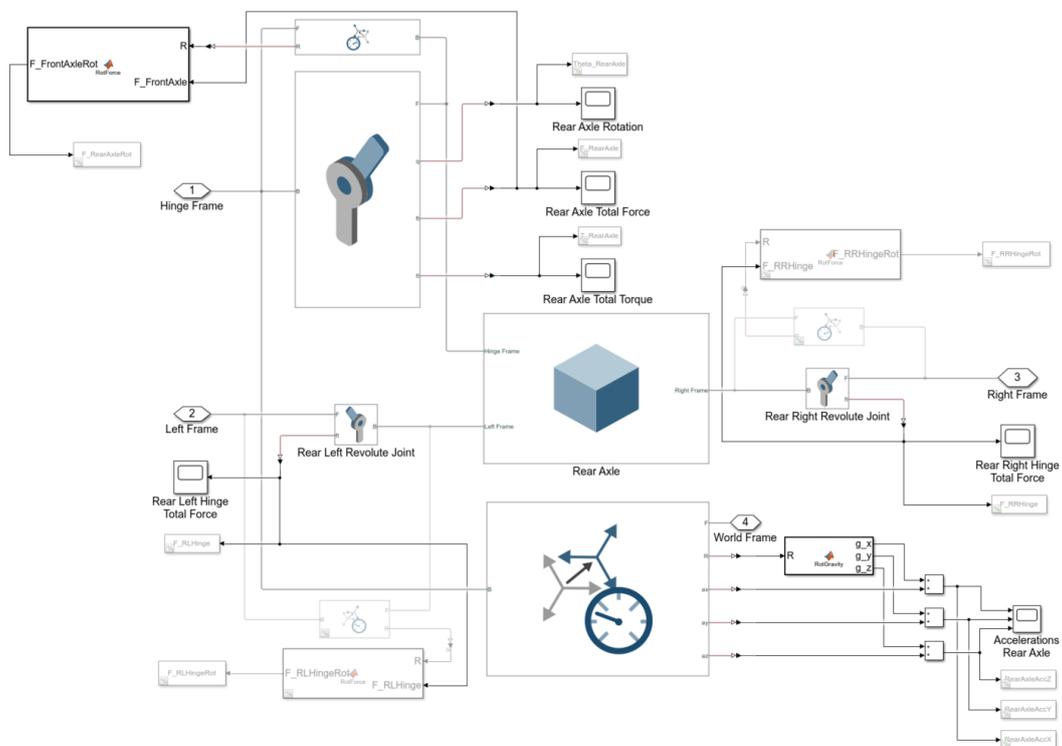


Figure A.13: Simulink scheme for the rear axle, corresponding to the block named *Rear Axle* in Figure A.1

## A.9 Gondola Scheme

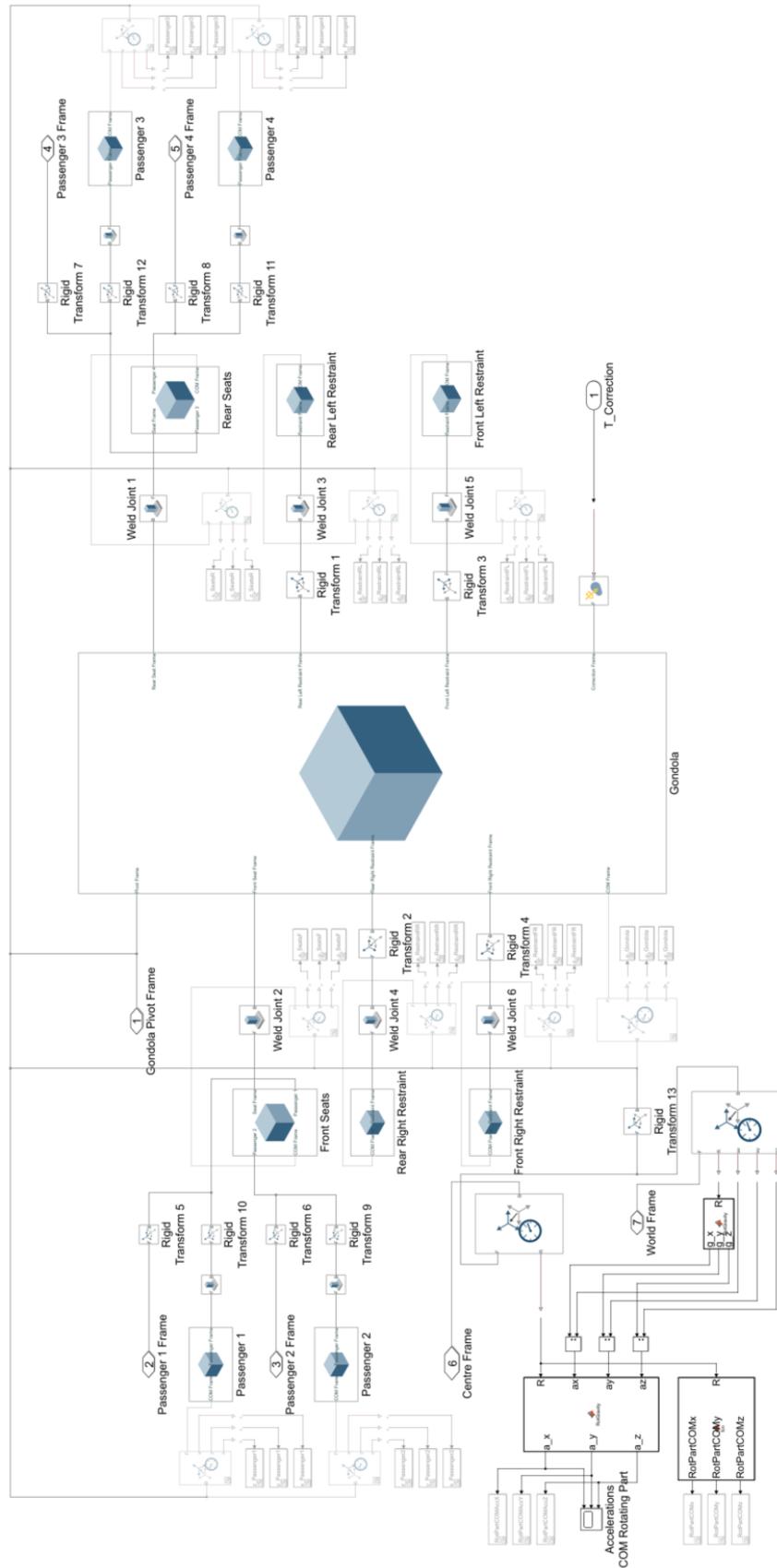


Figure A.14: Simulink scheme for the gondola, corresponding to the block named *Gondola* in Figure A.1

## A.10 Main Chassis Beam Scheme

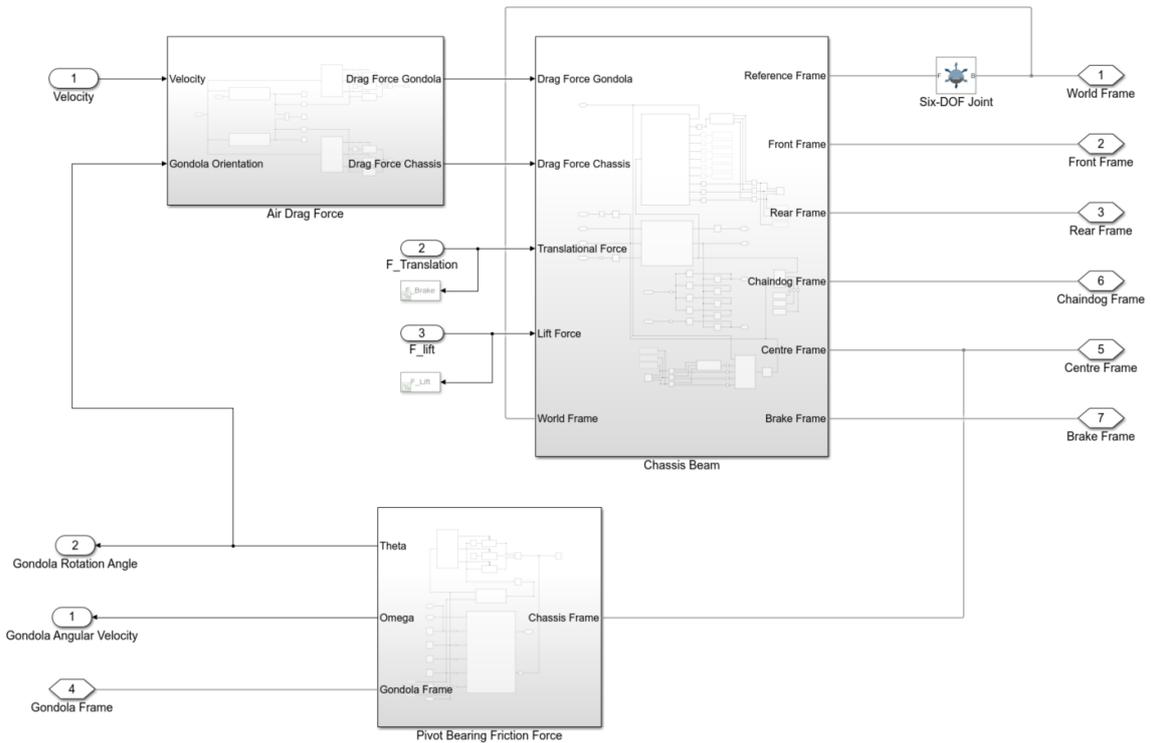


Figure A.15: Main Simulink scheme for the main chassis beam, corresponding to the block named *Main Chassis Beam* in Figure A.1

## A.11 Air Drag Force Scheme

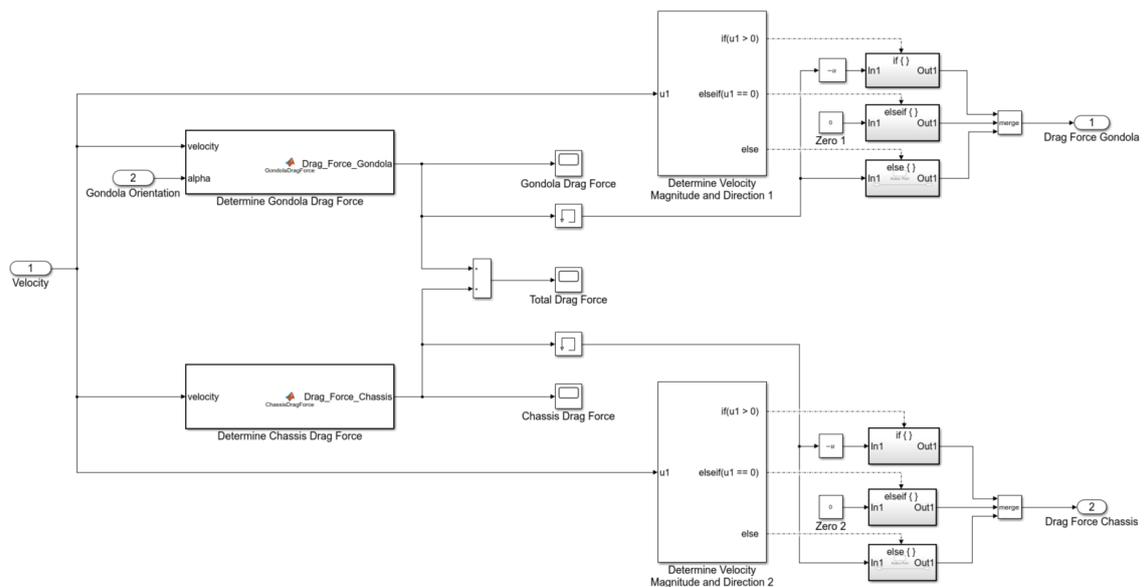


Figure A.16: Simulink scheme for determining the drag force, corresponding to the block named *Air Drag Force* in Figure A.15

## A.12 Beam Scheme

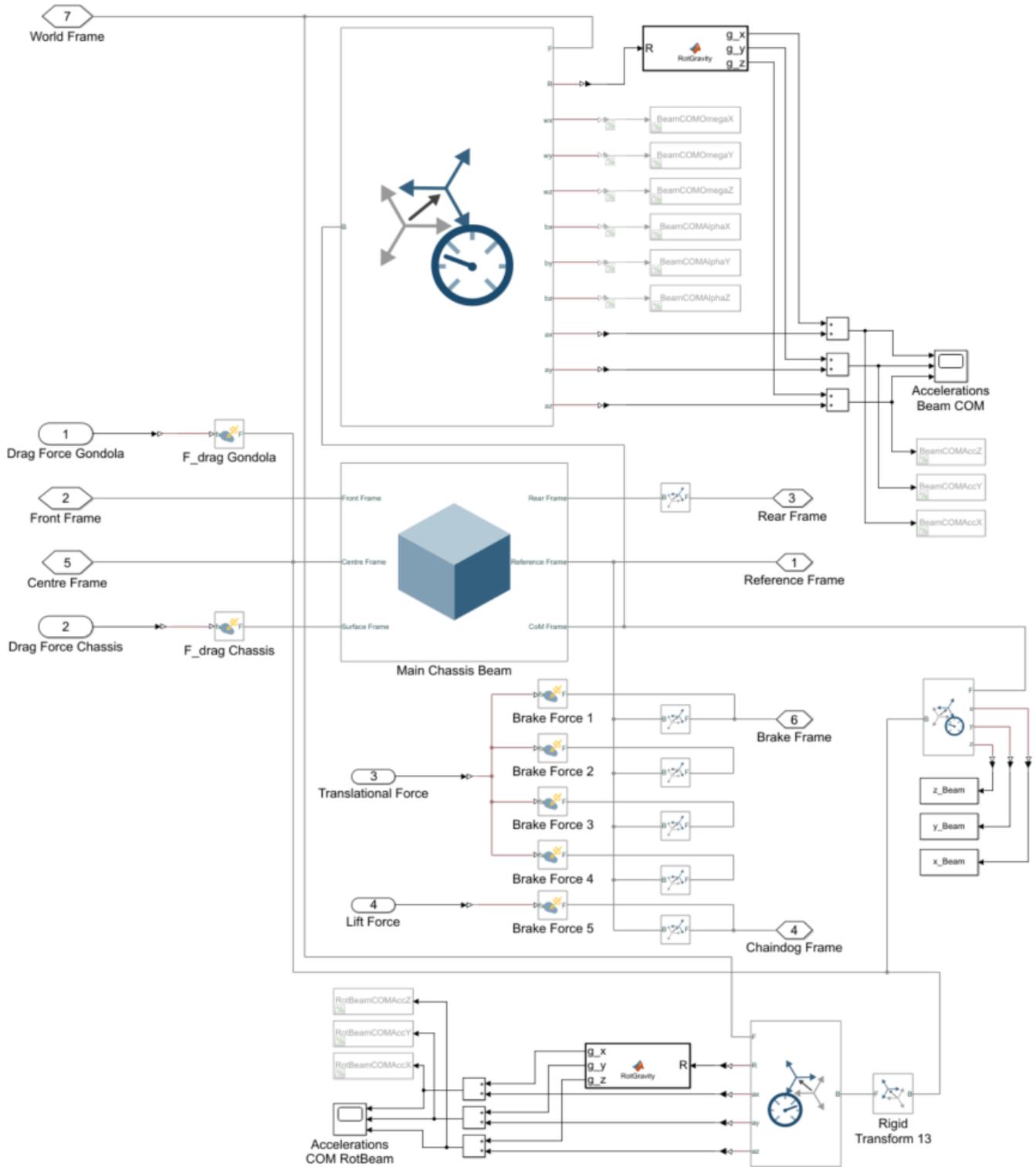


Figure A.17: Simulink sub-scheme for the main chassis beam, corresponding to the block named *Chassis Beam* in Figure A.15

## A.13 Pivot Bearing Friction Force Scheme

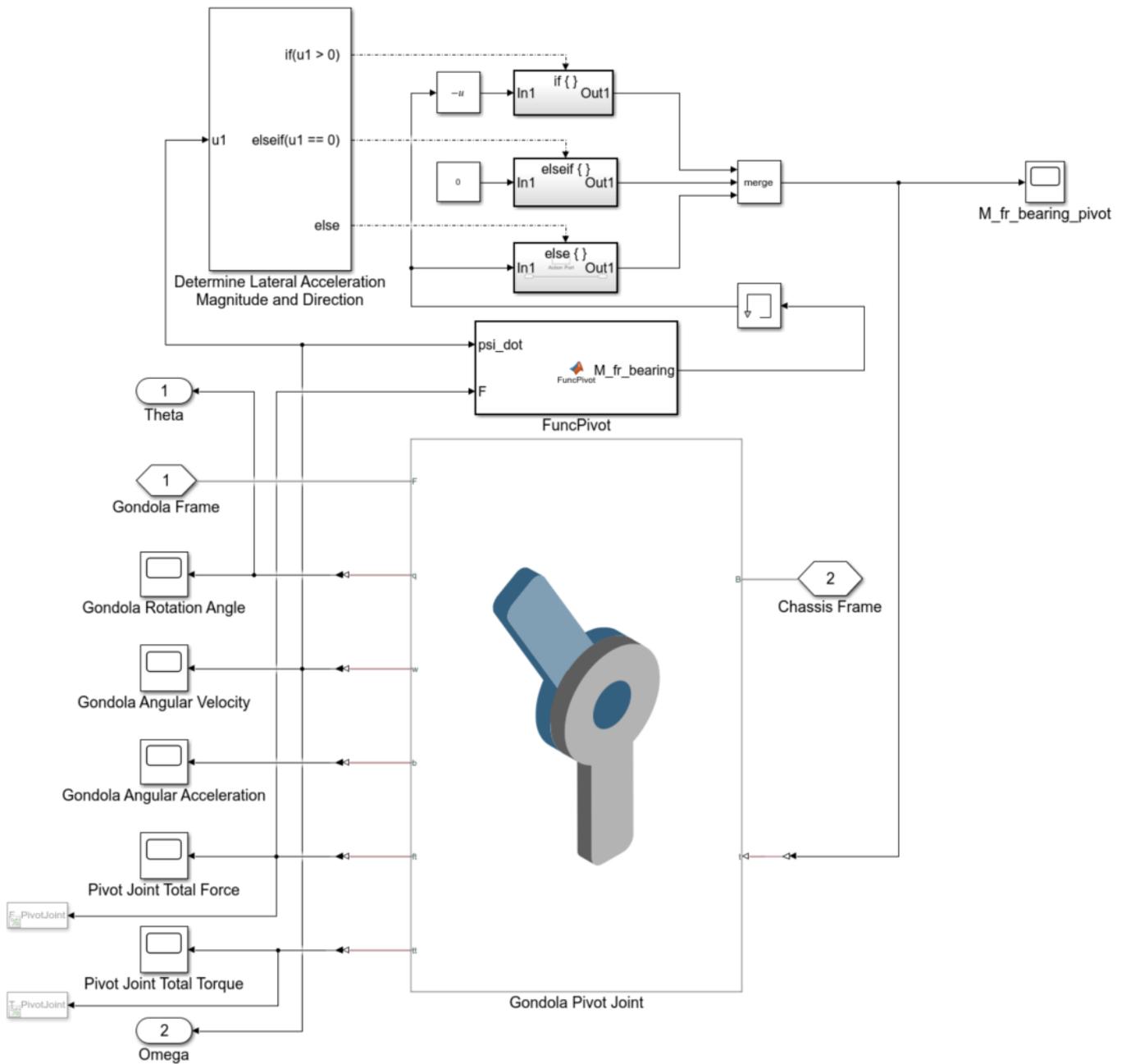


Figure A.18: Simulink scheme for determining the bearing friction force at the pivot joint, corresponding to the block named *Pivot Bearing Friction Force* in Figure A.15

## A.14 Output Scheme

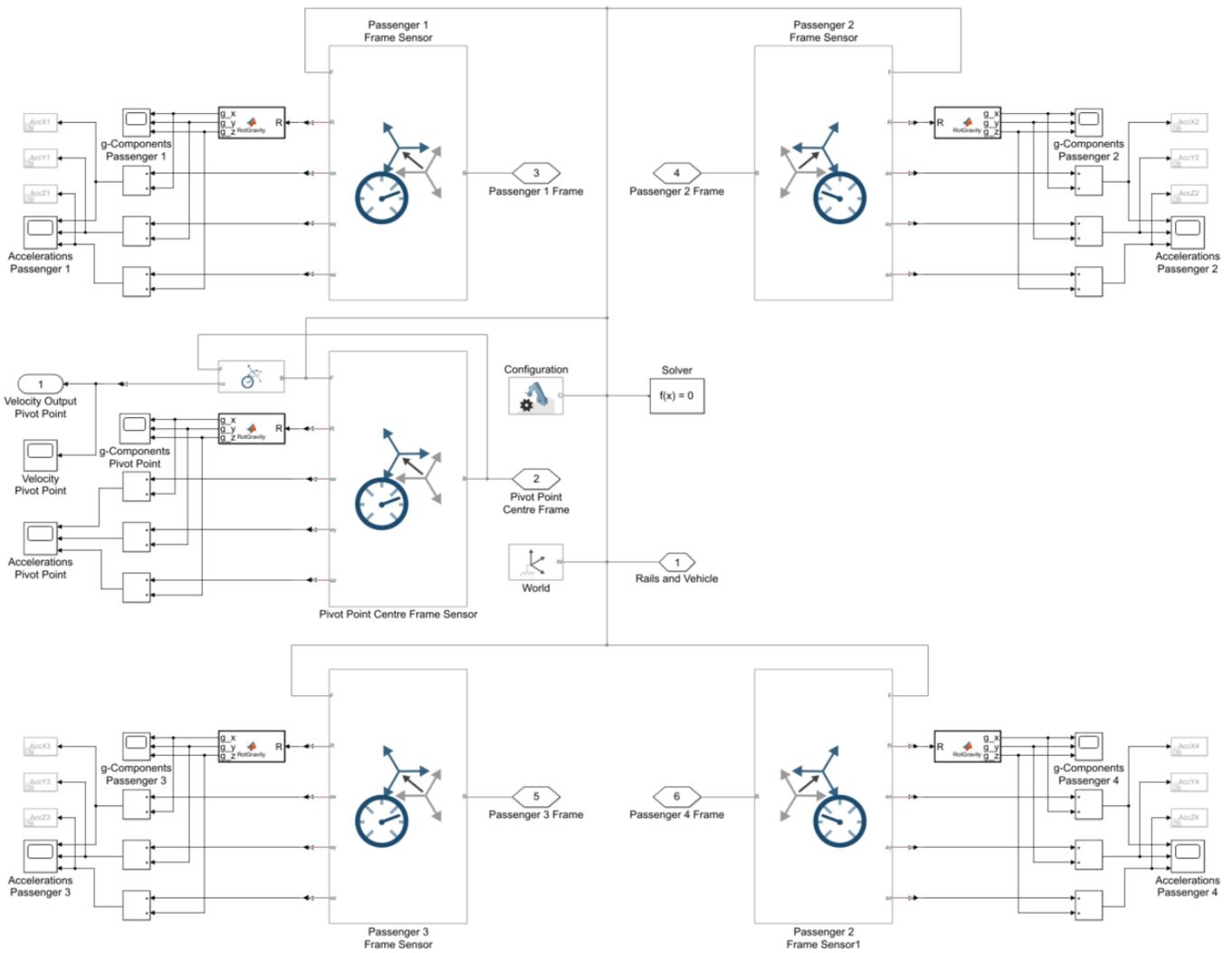


Figure A.19: Simulink scheme for determining the accelerations at the pivot point and the passengers, corresponding to the block named *Output* in Figure A.1

## A.15 Motion Control Scheme

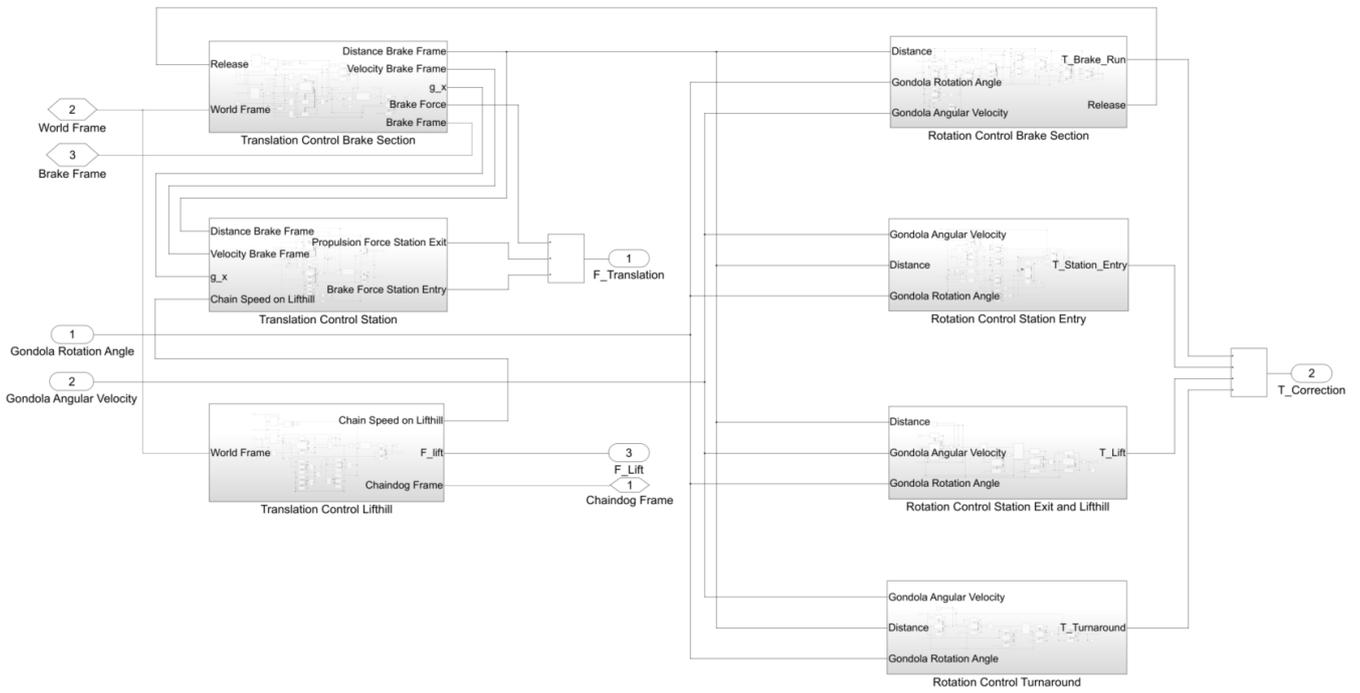


Figure A.20: Simulink scheme for motion control at various track segments, corresponding to the block named *Motion Control Brake Section, Station, Lifthill* in Figure A.1

## A.16 Translation Control Brake Section Scheme

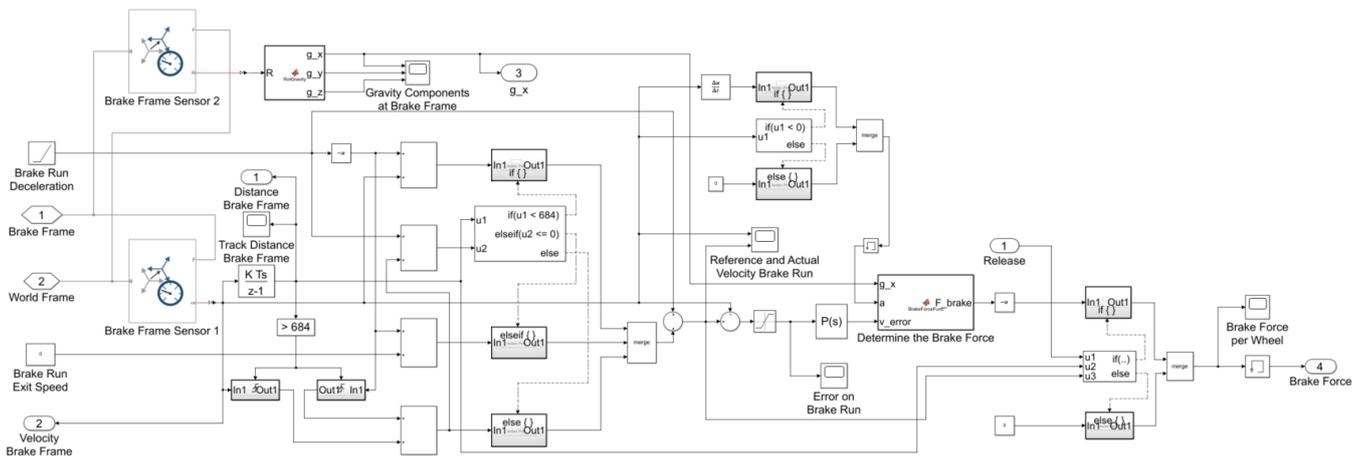


Figure A.21: Simulink scheme for translation control at the brake section, corresponding to the block named *Translation Control Brake Section* in Figure A.20

## A.17 Translation Control Station Scheme

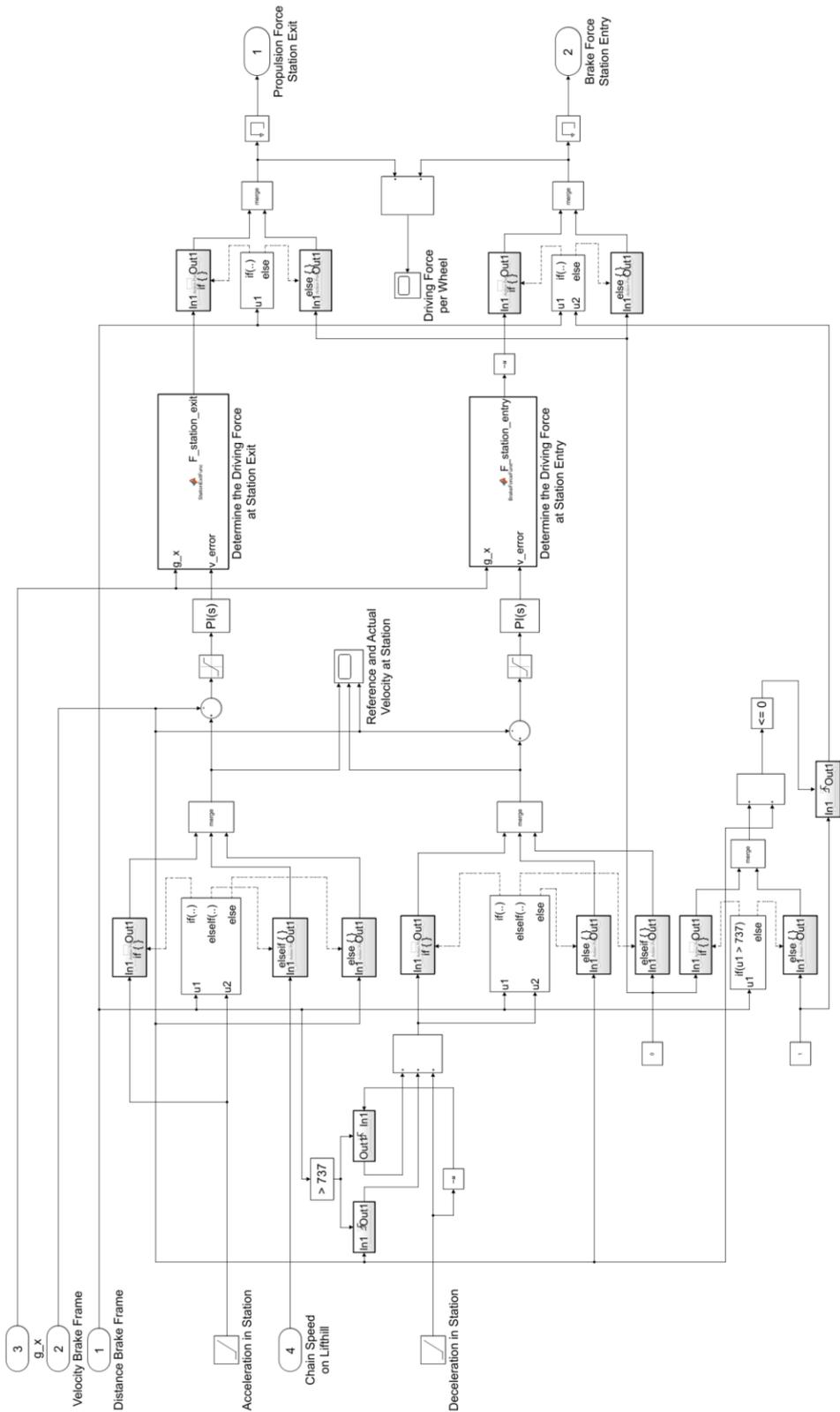


Figure A.22: Simulink scheme for translation control at the station, corresponding to the block named *Translation Control Station* in Figure A.20

## A.18 Translation Control Lifthill Scheme

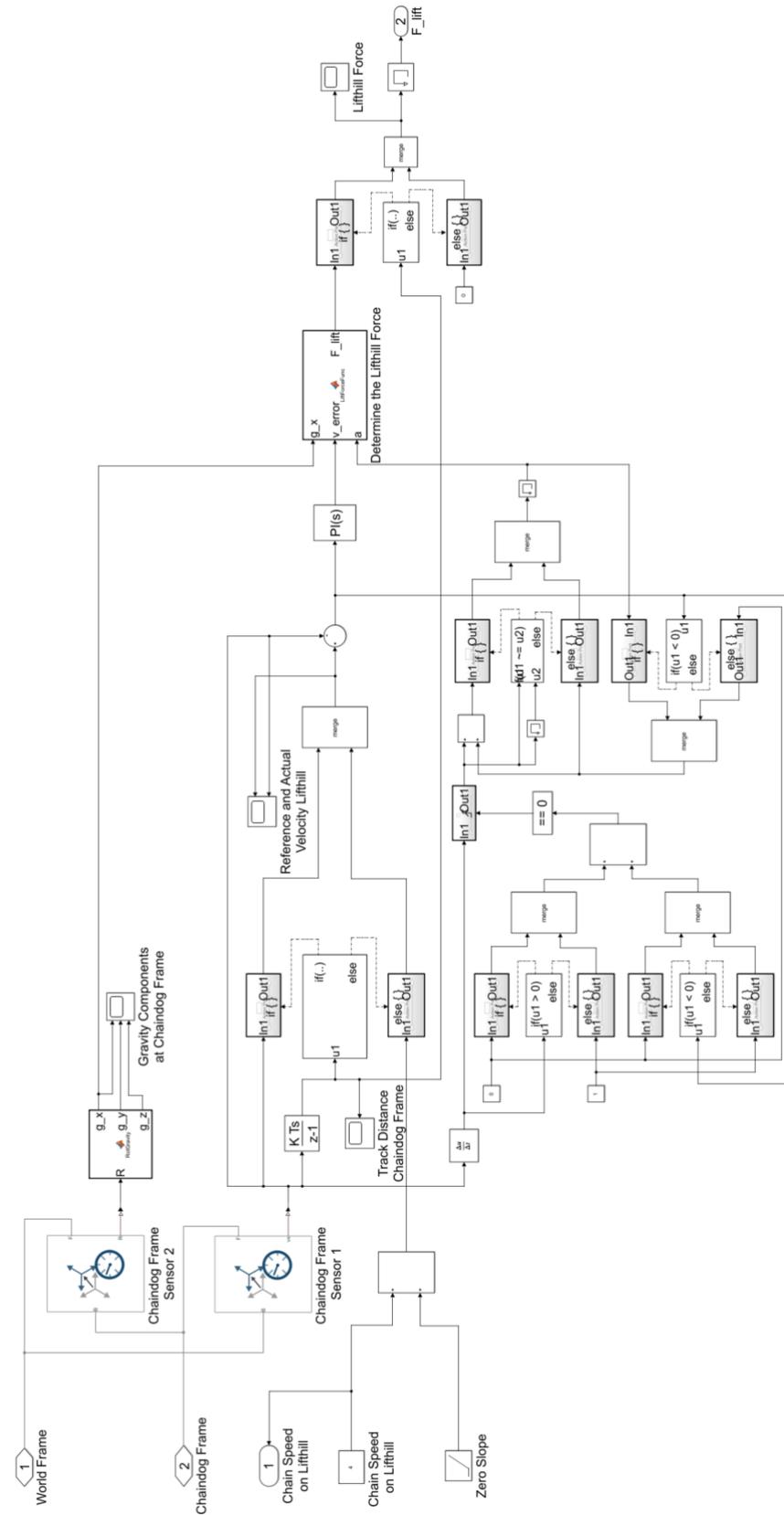


Figure A.23: Simulink scheme for translation control along the lifthill, corresponding to the block named *Translation Control Lifthill* in Figure A.20

## A.19 Rotation Control Brake Section Scheme

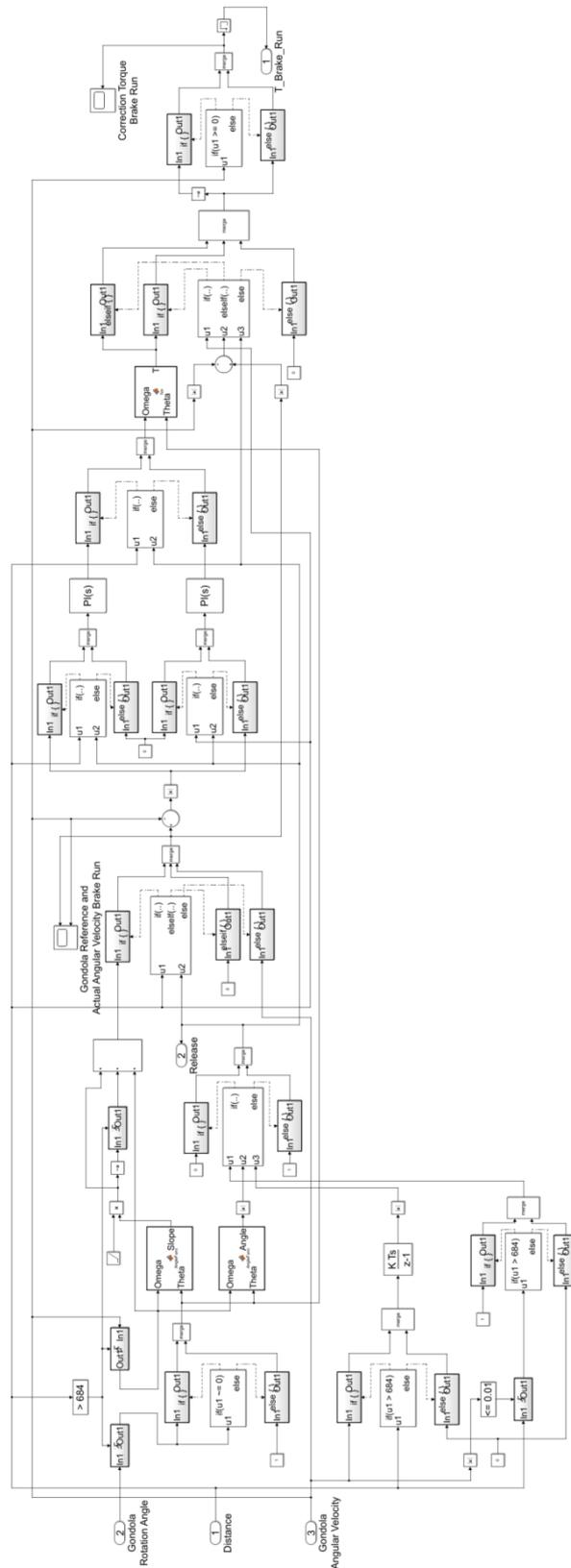


Figure A.24: Simulink scheme for rotation control at the brake section, corresponding to the block named *Rotation Control Brake Section* in Figure A.20

## A.20 Rotation Control Station Scheme

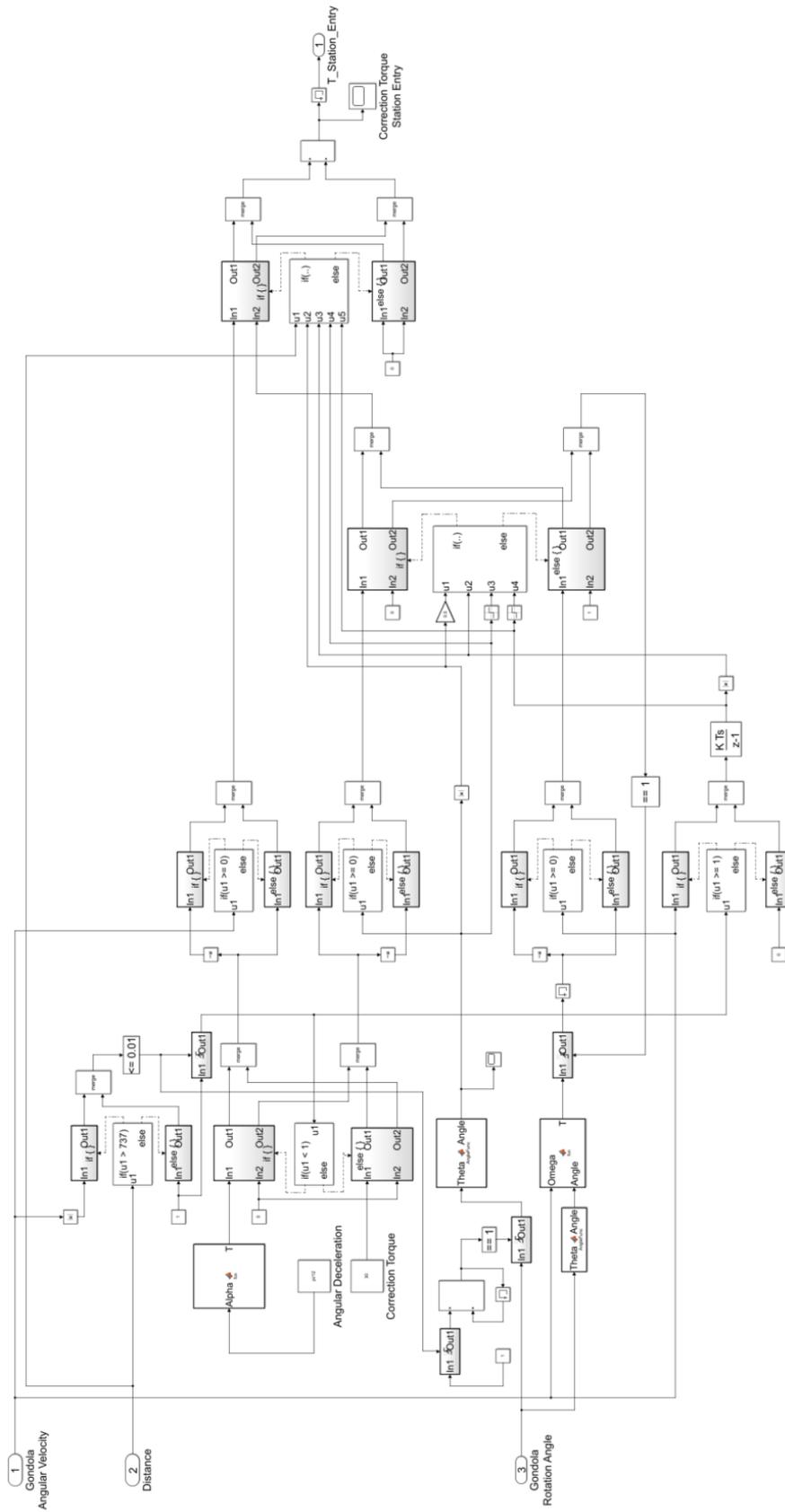


Figure A.25: Simulink scheme for rotation control at the station entry, corresponding to the block named *Rotation Control Station Entry* in Figure A.20

## A.21 Rotation Control Lifthill Scheme

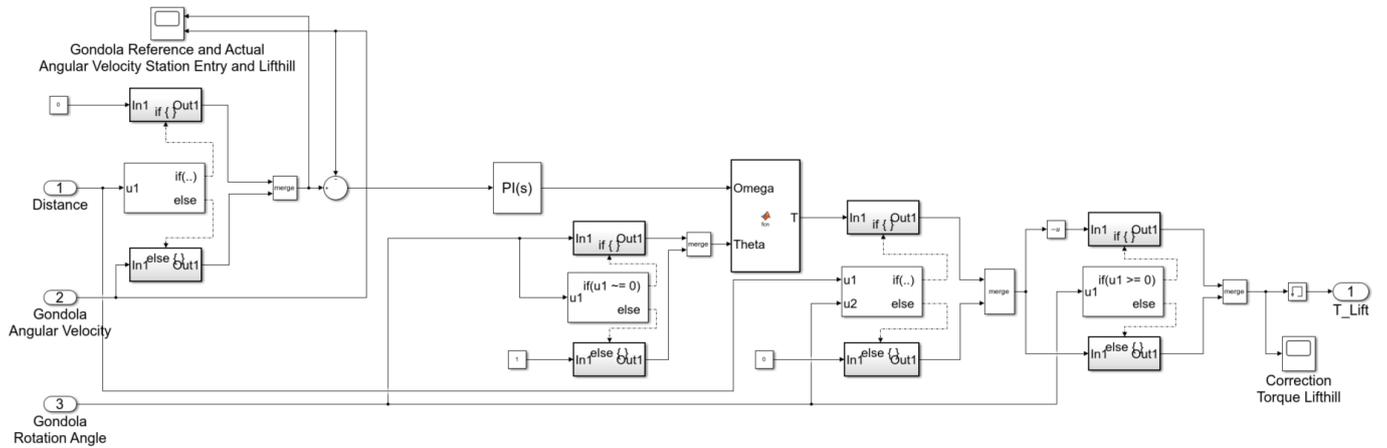


Figure A.26: Simulink scheme for rotation control at the station exit and lifthill, corresponding to the block named *Rotation Control Station Exit and Lifthill* in Figure A.20

## A.22 Rotation Control Turnaround

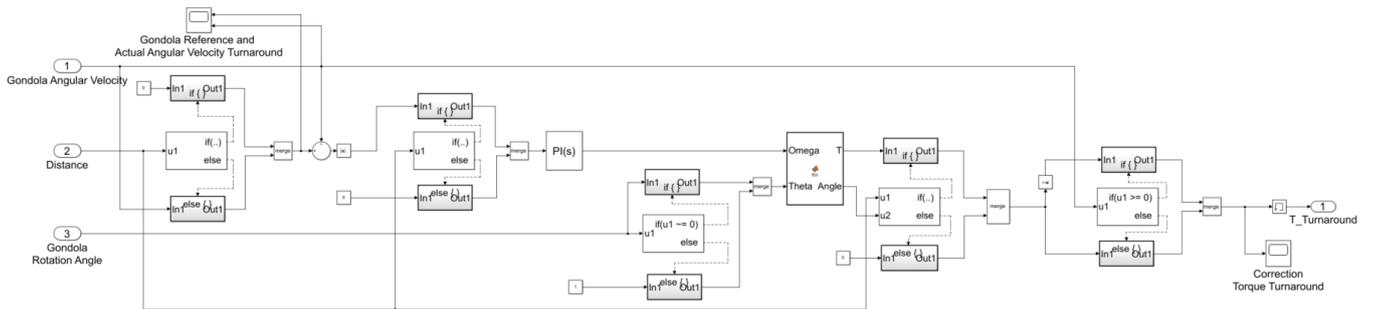


Figure A.27: Simulink scheme for rotation control at the turnaround between the brake section and the station, corresponding to the block named *Rotation Control Turnaround* in Figure A.20

# B. Matlab Functions in Simulink Model

## B.1 Copy of FuncRunning.m

The output of this function consists of the bearing and rolling frictional forces for a running wheel. The script in this function is based on the theory explained in Appendices E and F.

```
1 function [mu_fr_running_FLW1,F_fr_rolling_running,F_fr_bearing_running] = fcn(v,N-y)
2
3 % Define the y-component of normal force as the external radial load (N):
4 F_r = N-y;
5 % Set the axial load equal to zero (N):
6 F_a = 0;
7 % Specify the wheel radius (m):
8 R_x = 0.12625;
9
10 %% Calculation of the rolling frictional force
11
12 % Poisson's Ratio for Polyurethane PPDI:
13 nu_1 = 0.48;
14 % Elasticity Modulus for Polyurethane PPDI:
15 E_1 = 55e6; %Pa
16 % Poisson's Ratio for steel:
17 nu_2 = 0.3;
18 % Elasticity Modulus for steel (Pa):
19 E_2 = 200e9;
20 % Determine the inverse effective Elasticity Modulus:
21 E_ef_inv = ((1-nu_1^2)/(2*E_1)) + ((1-nu_2^2)/(2*E_2));
22 % Determine the effective Elasticity Modulus:
23 E_ef = 1/E_ef_inv; %Pa
24
25 % Track radius (m):
26 R_z = 0.0675;
27 % Determine the inverse effective radius:
28 R_ef_inv = (1/R_x) + (1/R_z);
29 % Determine the effective radius:
30 R_ef = 1/R_ef_inv;
31
32 % Determine the value for the parameter named lambda:
33 lambda = min([(R_x/R_z), (R_z/R_x)]);
34 % Determine the inverse elastic coefficient:
35 kappa_inv = 1 + sqrt(log(16/lambda)/(2*lambda)) - sqrt(log(4) + 0.16*log(lambda));
36 % Determine the elastic coefficient:
37 kappa = 1/kappa_inv;
38 % Determine the values for the asterisk semi-axes (m):
39 a_ast = kappa*nthroot((1 + ((2*(1-kappa^2))/(pi*kappa^2)) - 0.25*log(kappa)),3);
40 b_ast = a_ast/kappa;
41 % Determine the lengths of the semi-axes (m):
42 a = a_ast*nthroot(((3*F_r*R_ef)/E_ef),3);
43 b = b_ast*nthroot(((3*F_r*R_ef)/E_ef),3);
44
45 % Value for the logarithmic decrement is provided to reduce computational time:
46 lambda_s = 0.141300172376344;
47 % Determining the shear loss factor:
48 alpha_s = (1-exp(-lambda_s))/(1+exp(-lambda_s));
49 % The generalized compensation factor:
50 f_rs = 2.5;
51 % Determining the hysteresis loss factor:
52 alpha_r = f_rs*alpha_s;
53
```

```

54 % Determine the rolling friction coefficient:
55 mu_fr_running_FLW1 = (3*alpha_r*b)/(16*R_x);
56 % Determine the rolling friction force:
57 F_fr_rolling_running = F_r*mu_fr_running_FLW1;
58
59 %% Calculation of the bearing frictional force
60
61 % Bearing bore diameter (mm) of tapered roller bearings (single row) SKF 32912:
62 d = 60;
63 % Bearing outside diameter (mm) of tapered roller bearings (single row) SKF 32912:
64 D = 85;
65
66 % Estimation of actual operating viscosity of the oil or the base oil ...
67 % of the grease (mm^2/s), based on the base oil viscosity of ...
68 % Kluber Isoflex Topas NB 52 at 40 degrees Celsius:
69 nu = 30;
70 % Replenishment/starvation constant (for the assumption of a low level oil bath):
71 K_rs = 3e-8;
72 % Bearing type related geometric constant (for the assumption of tapered roller bearings):
73 K_z = 6;
74 % Geometric constant for rolling frictional moments (for bearing series 329):
75 R_1 = 2.31e-6;
76 % Geometric constant for rolling frictional moments (for bearing series 329):
77 R_2 = 10.9;
78 % Calculation factor (for SKF32912 bearings):
79 Y = 1.8;
80
81 % Sliding frictional moment (for bearing series 329):
82 S_1 = 0.009;
83 % Sliding frictional moment (for bearing series 329):
84 S_2 = 2;
85 % Constant depending on movement (for the rotational speed being larger than zero):
86 mu_bl = 0.12;
87 % Sliding friction coefficient in full-film conditions (for tapered roller bearings):
88 mu_EHL = 0.002;
89
90 % Bearing mean diameter:
91 d_m = 0.5*(d+D);
92 % Rotational speed (r/min):
93 n = (60*v)/(2*pi*R_x);
94 % Rolling frictional variable:
95 G_rr = R_1*(abs(d_m)^2.38)*(abs(F_r + (R_2*Y*F_a))^0.31);
96 % Inlet shear heating reduction factor:
97 phi_ish = 1/(1 + (1.84e-9)*(abs(n*d_m)^1.28)*(abs(nu)^0.64));
98 % Kinematic replenishment/starvation reduction factor:
99 phi_rs = 1/exp(K_rs*nu*n*(d+D)*sqrt(K_z/(2*(D-d))));
100 % Rolling frictional moment (Nmm):
101 M_rr = abs(phi_ish*phi_rs*G_rr*(nu*n))^0.6;
102
103 % Sliding frictional variable:
104 G_sl = S_1*(abs(d_m)^0.82)*(F_r + S_2*Y*F_a);
105 % Weighing factor for the sliding friction coefficient:
106 phi_bl = 1/exp(2.6e-8*(abs(n*nu)^1.4)*d_m);
107 % Sliding friction coefficient:
108 mu_sl = phi_bl*mu_bl + (1-phi_bl)*mu_EHL;
109 % Sliding frictional moment (Nmm):
110 M_sl = G_sl*mu_sl;
111
112 % Total bearing frictional moment (frictional moments of seals and drag losses ...
113 % are neglected):
114 M_fr_bearing_running = 2*(M_rr + M_sl)*1e-3; %Nm
115 % Total bearing frictional force (N):
116 F_fr_bearing_running = M_fr_bearing_running/(R_z+R_x);

```

The script in function `FuncRunning.m` applies to the first running wheel on the front-left bogie. For the other wheels and wheel carriers, the frictional values are determined analogous to the procedure shown in function `FuncRunning.m`. Certain input parameters such as the wheel radii or bearing dimensions can differ for each type of wheel, as shown in Appendices E and F. The function `FuncPivot.m` in the block named Pivot Bearing Friction Force is similar to the second half of `FuncRunning.m`.

## B.2 Copy of GondolaDragForce.m

The output of this function is the drag force acting on the gondola. The script in this function is based on the theory explained in Appendix G. The spline coefficients are already provided in the script to reduce the computational time.

```

1 function Drag_Force_Gondola = GondolaDragForce(velocity,alpha)
2
3 %% Input
4
5 % Specify the air density (kg/m^3):
6 rho = 1.2;
7
8 %% Gondola Frontal Surface Calculation
9
10 % Take the absolute value of the rotation:
11 rot = mod(abs((alpha/pi)*180),180);
12
13 % The frontal surface is specified for angles in the domain [0,360] with a
14 % stepsize of 15 degrees:
15 Rotation = [0:15:180]; %deg
16 A_front = [1.816,1.884,1.969,2.107,2.153,1.983,1.666, ...
17            1.983,2.153,2.107,1.969,1.884,1.816]; %m^2
18 % Use cubic spline data interpolation between the data points:
19 SplineDatacoefs = [-1.49310541310540e-05,0.000526188034188032,0,1.816;
20                    9.53390313390309e-06,-0.000145709401709400,0.00570717948717948,1.884;
21                    -1.25378917378918e-05,0.000283316239316239,0.00777128205128207,1.969;
22                    -2.34529914529908e-06,-0.000280888888888890,0.00780769230769230,2.107;
23                    -1.48216524216525e-05,-0.000386427350427349,-0.00220205128205128,2.153;
24                    8.20763532763534e-05,-0.00105340170940171,-0.0237994871794872,1.983;
25                    -8.20763532763534e-05,0.00264003418803419,-8.67361737988404e-19,1.666;
26                    1.48216524216525e-05,-0.00105340170940171,0.0237994871794872,1.983;
27                    2.34529914529908e-06,-0.000386427350427349,0.00220205128205128,2.153;
28                    1.25378917378918e-05,-0.000280888888888890,-0.00780769230769230,2.107;
29                    -9.53390313390309e-06,0.000283316239316239,-0.00777128205128207,1.969;
30                    1.49310541310540e-05,-0.000145709401709400,-0.00570717948717948,1.884];
31
32 % Determine the interval corresponding to the specified rotation angle:
33 diff = abs(Rotation-rot);
34 [minf1,idxf1] = min(diff);
35 if idxf1 == 1
36     idxf2 = idxf1+1;
37     minf2 = diff(idxf2);
38 elseif idxf1 == length(diff)
39     idxf2 = idxf1-1;
40     minf2 = diff(idxf2);
41 else
42     if diff(idxf1-1) <= diff(idxf1+1)
43         idxf2 = idxf1-1;
44         minf2 = diff(idxf2);
45     else
46         idxf2 = idxf1+1;
47         minf2 = diff(idxf2);
48     end

```

```

49 end
50
51 % Determine the index corresponding to the lower endpoint of the interval:
52 lowidxf = min(idxf1,idxf2);
53 lowvalf = Rotation(lowidxf);
54 % Define the polynomial coefficients corresponding to the interval:
55 a = SplineData0coefs(lowidxf,1);
56 b = SplineData0coefs(lowidxf,2);
57 c = SplineData0coefs(lowidxf,3);
58 d = SplineData0coefs(lowidxf,4);
59
60 % Calculate the frontal surface corresponding to the specified rotation angle:
61 Frontal_Surface = a*(rot-lowvalf)^3 + b*(rot-lowvalf)^2 + c*(rot-lowvalf) + d;
62
63 %% Gondola Drag Coefficient Calculation
64
65 % Specify the velocities at which the drag coefficients are known:
66 v = [0,3.15,6.3,9.4,13.15,20];
67 % Specify the corresponding drag coefficients for a rotation of zero degrees:
68 Cd0 = [1.011,1.066,1.070,1.075,1.043,1.142];
69 % Specify the corresponding drag coefficients for a rotation of ninety degrees:
70 Cd90 = [1.499,1.571,1.613,1.621,1.636,1.619];
71 % Store the cubic interpolation spline data for both orientations:
72 SplineData0coefs = [-0.00211273538432178,0.0121980743845239,0,1.011;
73                    0.00118717454690726,-0.00776727499731689,0.0139570180697022,1.066;
74                    -0.000983270192073580,0.00345152447095673,0.000362403911667667,1.07;
75                    0.00137961399857970,-0.00569288831532757,-0.00658582400588194,1.075;
76                    -0.00161376342356207,0.00982776916869401,0.00891997919424224,1.043;
77                    0.000761927724491951,-0.00675365000840627,0.0194488373182278,1.124];
78 SplineData90coefs = [-0.00198793059623423,0.0135182172058022,0,1.499;
79                    0.000396838375944305,-0.00526772692861127,0.0259890443731515,1.571;
80                    0.000277832189164854,-0.00151760427593759,0.00461525107882259,1.613;
81                    -0.000228578712461560,0.00106623508329556,0.00321600658163231,1.621;
82                    0.000131465071931999,-0.00150527543189699,0.00156960527437695,1.636;
83                    0.000146998732171020,-0.000154471817795695,-0.00411502905582050,1.629];
84
85 % Determine the interpolation interval corresponding to the specified velocity:
86 difd = abs(v-velocity);
87 [mind1,idxd1] = min(difd);
88 if idxd1 == 1
89     idxd2 = idxd1+1;
90     mind2 = difd(idxd2);
91 elseif idxd1 == length(difd)
92     idxd2 = idxd1-1;
93     mind2 = difd(idxd2);
94 else
95     if difd(idxd1-1) < difd(idxd1+1)
96         idxd2 = idxd1-1;
97         mind2 = difd(idxd2);
98     else
99         idxd2 = idxd1+1;
100        mind2 = difd(idxd2);
101    end
102 end
103
104 % Determine the index corresponding to the lower endpoint of the interval:
105 lowidxd = min(idxd1,idxd2);
106 % Determine the value at this index:
107 lowvald = v(lowidxd);
108 % Define the polynomial coefficients of the interval for a zero degree rotation:
109 a0 = SplineData0coefs(lowidxd,1);
110 b0 = SplineData0coefs(lowidxd,2);
111 c0 = SplineData0coefs(lowidxd,3);

```

```

112 d0 = SplineData0coefs(lowidxd,4);
113 % Define the polynomial coefficients of the interval for a ninety degree rotation:
114 a90 = SplineData90coefs(lowidxd,1);
115 b90 = SplineData90coefs(lowidxd,2);
116 c90 = SplineData90coefs(lowidxd,3);
117 d90 = SplineData90coefs(lowidxd,4);
118
119 % Calculate the drag coefficient corresponding to the specified velocity at
120 % respectively a zero and ninety degree rotation:
121 dragcoeff0 = a0*(velocity-lowvald)^3 + b0*(velocity-lowvald)^2 + c0*(velocity-lowvald) ...
    + d0;
122 dragcoeff90 = a90*(velocity-lowvald)^3 + b90*(velocity-lowvald)^2 + ...
    c90*(velocity-lowvald) + d90;
123
124 % Determine the cubic interpolation spline for the data at the specified velocity:
125 SplineDatav = spline([0,90,180],[0,dragcoeff0,dragcoeff90,dragcoeff0,0]);
126
127 % Define the polynomial coefficients for the first interval:
128 av1 = SplineDatav.coefs(1,1);
129 bv1 = SplineDatav.coefs(1,2);
130 cv1 = SplineDatav.coefs(1,3);
131 dv1 = SplineDatav.coefs(1,4);
132 % Define the polynomial coefficients for the second interval:
133 av2 = SplineDatav.coefs(2,1);
134 bv2 = SplineDatav.coefs(2,2);
135 cv2 = SplineDatav.coefs(2,3);
136 dv2 = SplineDatav.coefs(2,4);
137
138 % Calculate the drag coefficient for the specified velocity and rotation:
139 if (0 ≤ rot) && (rot ≤ 90)
140     Drag_Coefficient = av1*rot^3 + bv1*rot^2 + cv1*rot + dv1;
141 else
142     Drag_Coefficient = av2*(rot-90)^3 + bv2*(rot-90)^2 + cv2*(rot-90) + dv2;
143 end
144
145 %% Gondola Drag Force Calculation
146
147 Drag_Force_Gondola = 0.5*Drag_Coefficient*rho*(velocity^2)*Frontal_Surface;

```

### B.3 Copy of ChassisDragForce.m

The output of this function is the drag force acting on the chassis, bogies, and wheels. The script in this function is based on the theory explained in Appendix G. The functions `GondolaDragForce` and `ChassisDragForce` can both be found in the block named Air Drag Force in Figure A.15.

```

1 function Drag_Force_Chassis = ChassisDragForce(velocity)
2
3 %% Input
4
5 % Specify the air density (kg/m^3):
6 rho = 1.2;
7 % Presumed drag coefficient of chassis (equal to drag coefficient of a cube):
8 Cd = 0.8;
9
10 %% Chassis Drag Force Calculation
11
12 % Frontal surface of chassis:
13 S_front = 0.386; %m^2
14 % Drag Force of the main chassis beam and bogies:
15 Drag_Force_Chassis = 0.5*Cd*rho*(velocity^2)*S_front;

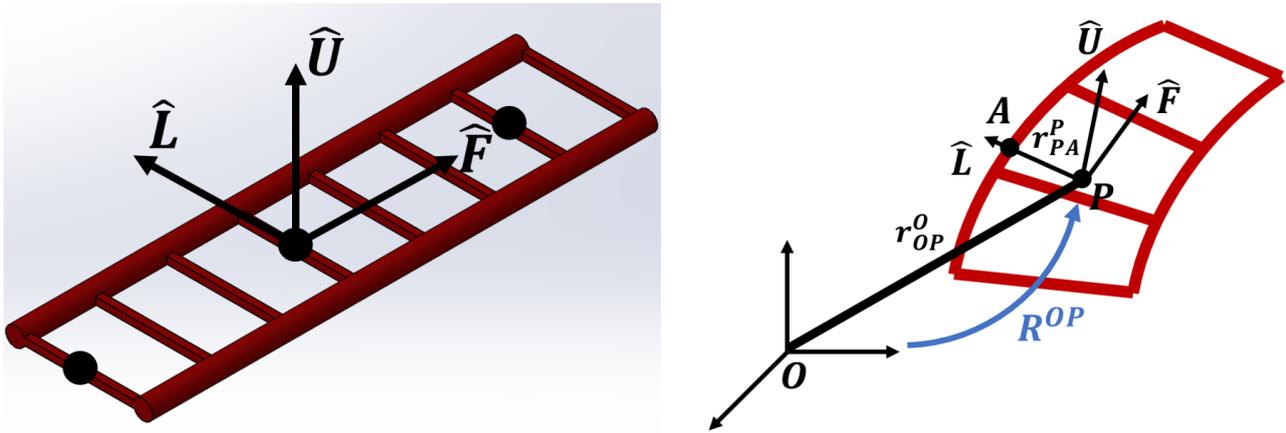
```

# C. Track Generator

## C.1 NoLimits Track Data to Rail Coordinates

This appendix describes the conversion of NoLimits track data to the Cartesian coordinates that correspond to the left and right rail of the track. The professional version of NoLimits 2 allows the user to export and save track data in CSV-format. The user can specify the distance between the nodes along the track curvature, and each line in the CSV-file corresponds with a node on the centre spline of the track. The centre spline can either be chosen equal to the editor spline or the centre of the rails, where the editor spline is unequal to the centre of rails for most track types due to the addition of for instance a backbone that strengthens the track. However, on the spinning rollercoaster the editor spline and centre of rails coincide with each other thanks to the simple track geometry.

The CSV-file that is exported from NoLimits consists of thirteen columns. The first column specifies the node number, while the next three columns specify the Cartesian coordinates of the nodes with respect to a global reference frame. The remaining nine columns specify the  $x$ -,  $y$ -, and  $z$ -components of respectively the front  $\hat{F}$ , left  $\hat{L}$ , and upper vector  $\hat{U}$  that start at the corresponding centre node, as depicted in Figure C.1a. Each of these three vectors are unit vectors that are decomposed in Cartesian components with respect to a global reference frame. Together the  $\hat{F}$ ,  $\hat{L}$ , and  $\hat{U}$  vectors can be regarded as a local reference frame, since the three vectors are always mutually perpendicular. The origin of this local reference frame is located at point P in Figure C.1b.



(a) Specification of the vectors as exported by NoLimits (b) The frames and parameters used during the analysis

Figure C.1: The vectors and frames used for converting the NoLimits track data to Cartesian coordinates of the left and right rail

Prior to calculating the coordinates of the left and right rail, the data should first be converted from millimeters to meters. Any track width can be specified by the user, but in case of the spinning rollercoaster the width of the track is set equal to 0.9 meters. In Figure C.1b,  $r_{PA}^P$  refers to the distance between P and either the left or the right rail, formulated with respect to the local reference frame. The value for  $r_{PA}^P$  equals 0.45 meters in case of the left rail, and -0.45 meters for the right rail. The direction of  $\hat{F}$  and  $\hat{U}$  can solely be determined based on the direction of  $\hat{L}$ , thanks to the orthogonality between the three vectors. Therefore, it suffices to only use  $\hat{L}$  for determining the rail coordinates. The position of a certain rail coordinate can be computed with the expression in Equation C.1.

$$r_{OA}^O = r_{OP}^O + R^{OP} r_{PA}^P \quad (C.1)$$

where  $r_{OA}^O$  is the distance between point O and point A formulated with respect to the global reference

frame in point O. The distance  $r_{OP}^O$  is equal to the position of the centre node, and hence the value for this parameter is known. The rotation matrix  $R^{OP}$  between the global reference frame at point O and the local reference frame at point P is depicted in blue in Figure C.1b. A rotation matrix between any two frames is defined as the matrix that relates the orientation of the axes of both frames. Taking into account that  $\hat{\mathbf{L}}$  is a vector with unit length, the rotation matrix is determined between  $\hat{\mathbf{L}}$  and a local vector  $[1,0,0]$  for each node along the track. This could also be regarded as the specification of a unit vector along the  $x$ -axis of the local reference frame at point P, after which a multiplication with the rotation matrix leads to a decomposition of this vector in Cartesian coordinates with respect to the global reference frame at point O. The rotation between the two vectors can be calculated using the built-in function `vrrotvec` in `Matlab`. The output is a four-element rotation row vector, where the first three elements specify the rotation axis and the last element defines the angle of rotation. The rotation row vector can subsequently be represented in matrix-form by using the `Matlab`-function `vrrotvec2mat`, which results in the rotation matrix  $R^{OP}$  with three rows and three columns. The left and right rail coordinates can subsequently be determined with respect to the global reference frame by multiplying  $R^{OP}$  with  $r_{PA}^P$ , after which  $r_{OP}^O$  is added to the result. Please note that  $r_{PA}^P$  is defined as  $[0.45,0,0]$  or  $[-0.45,0,0]$  for respectively the left and right rail, whose direction is in agreement with the local vector  $[1,0,0]$  that was previously defined.

Performing the computation for each node results in a list of  $xyz$ -coordinates for the left and right rail. Plotting these coordinates along with the coordinates of the centre rail results in the plot shown in Figure C.2. It should be noted that the global reference frames in `NoLimits` and `Simulink` are orientated differently with respect to each other. Hence, the left and right rail coordinates should first be transformed from one coordinate system to the other before the rail data can be used in `Simulink`. The user can choose a convenient location for the global reference frame in `Simulink`, and subsequently subtract the coordinates of the chosen location from the rail coordinates. This procedure can for instance be used to assign the station as the start position of the rollercoaster train. If the default zero-length initial train position vector is used in `Simulink`, the train always starts at the origin of the global reference frame which coincides with the station. Once the transformation has been completed and the global reference frame origin has been specified, the rail coordinates are saved in two separate text files e.g. `LeftRail.txt` and `RightRail.txt`. These files specify the  $xyz$ -coordinates of the left and right rail, which are used by `Simulink` to define two cubic interpolation splines that form the track. A copy of the `Matlab`-script that has been described in this appendix can be found in Section C.2.

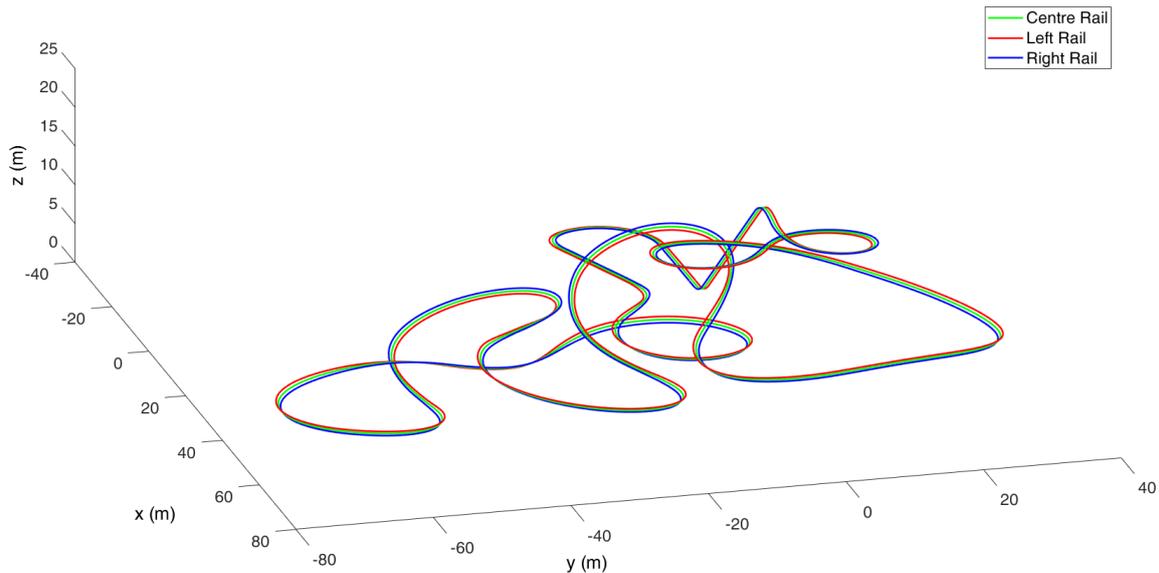


Figure C.2: The track geometry of the spinning rollercoaster lay-out

## C.2 Copy of TrackGenerator.m

```
1 % 1. Open a blank workbook in Excel
2 % 2. Click on 'From Text/CSV' under the tab 'Data'
3 % 3. Select the right CSV file and click on 'Import'
4 % 4. Click on 'Load' and save the file
5 % 5. Specify the directory path and file name on line 18
6
7 % Specify the track width:
8 TrackWidth = 0.9; %m
9
10 % Define the text-file names to which the left and right rail coordinates will be written:
11 fileID1 = fopen('LeftRail.txt','w');
12 fileID2 = fopen('RightRail.txt','w');
13
14 % Import the Excel sheet with track data from NoLimits:
15 Data = xlsread('C:\Users\Name\Documents\TrackConverter\ExcelFiles\CentreRails.xlsx');
16
17 % Each node on the centre rail is numbered according to the variable below:
18 Nodenr = Data(:,1);
19 % The position of each node is specified according to Cartesian
20 % coordinates in the global reference frame used in Nolimits:
21 PosX = Data(:,2)*1e-6;
22 PosY = Data(:,3)*1e-6;
23 PosZ = Data(:,4)*1e-6;
24 % The Cartesian coordinates of the normalized vector pointing to the left,
25 % again specified according to the global reference frame used in NoLimits:
26 LeftX = Data(:,8)*1e-6;
27 LeftY = Data(:,9)*1e-6;
28 LeftZ = Data(:,10)*1e-6;
29
30 % Preallocation to increase computational speed:
31 LeftRailX = zeros(length(Nodenr),1);
32 LeftRailY = zeros(length(Nodenr),1);
33 LeftRailZ = zeros(length(Nodenr),1);
34 RightRailX = zeros(length(Nodenr),1);
35 RightRailY = zeros(length(Nodenr),1);
36 RightRailZ = zeros(length(Nodenr),1);
37
38 for n = 1:length(Nodenr)
39     % In the new local reference frame, the Left-vector should coincide ...
40     % with the local x-axis:
41     LocalVec = [1;0;0];
42     % Determine the rotation vector between the Left-vector specified
43     % in respectively the local and global reference frame:
44     RotVec{n,1} = vrrotvec(LocalVec, [LeftX(n,1);LeftY(n,1);LeftZ(n,1)]);
45     % Conversion to a rotation matrix:
46     RotMat{n,1} = vrrotvec2mat(RotVec{n,1});
47     % The distance between the track centre and the left rail specified ...
48     % in the local reference frame:
49     LocalLeft = [TrackWidth/2;0;0];
50     % Multiplication with the rotation matrix leads to a specification
51     % of this distance in terms of the global reference frame:
52     GlobalLeft{n,1} = RotMat{n,1}*LocalLeft;
53     % The distance between the track centre and the right rail specified ...
54     % in the local reference frame:
55     LocalRight = [-TrackWidth/2;0;0];
56     % Multiplication with the rotation matrix leads to a specification
57     % of this distance in terms of the global reference frame:
58     GlobalRight{n,1} = RotMat{n,1}*LocalRight;
59
```

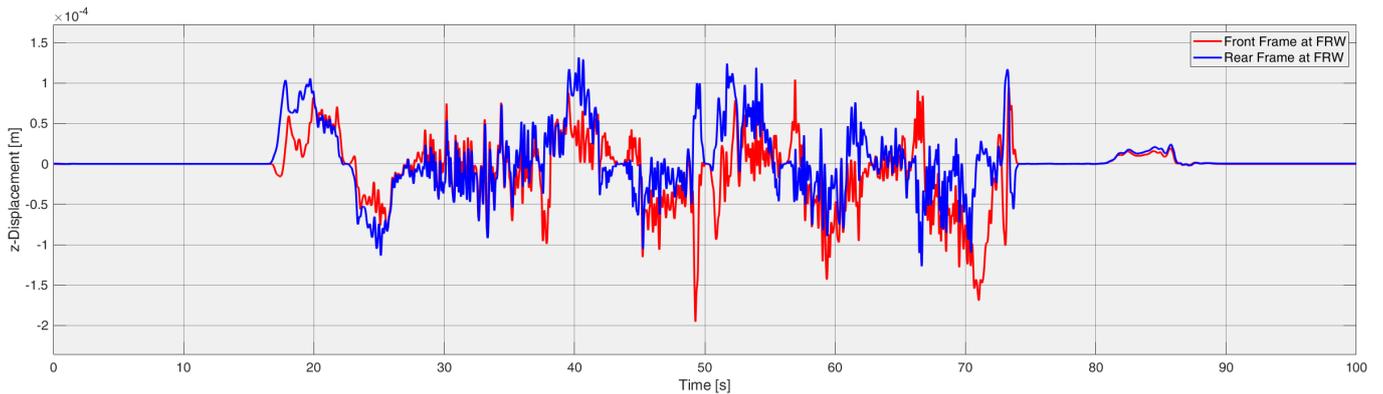
```

60 % The Cartesian coordinates of the respectively left and right rail are determined ...
61 % by a superposition of the centre node position and the distance between the ...
62 % centre node and the rail node:
63 LeftRailX(n,1) = PosX(n,1) + GlobalLeft{n,1}(1,1);
64 LeftRailY(n,1) = PosY(n,1) + GlobalLeft{n,1}(2,1);
65 LeftRailZ(n,1) = PosZ(n,1) + GlobalLeft{n,1}(3,1);
66 RightRailX(n,1) = PosX(n,1) + GlobalRight{n,1}(1,1);
67 RightRailY(n,1) = PosY(n,1) + GlobalRight{n,1}(2,1);
68 RightRailZ(n,1) = PosZ(n,1) + GlobalRight{n,1}(3,1);
69 end
70
71 % Conversion to the global reference frame used in Simulink:
72 CentreXSim = -PosZ(:,1);
73 CentreYSim = -PosX(:,1);
74 CentreZSim = PosY(:,1);
75 LeftXSim = -LeftRailZ(:,1);
76 LeftYSim = -LeftRailX(:,1);
77 LeftZSim = LeftRailY(:,1);
78 RightXSim = -RightRailZ(:,1);
79 RightYSim = -RightRailX(:,1);
80 RightZSim = RightRailY(:,1);
81
82 % Plot the centre track (green), left rail (red), and right rail (blue) in a figure:
83 figure(1), clf(1), hold on
84 plot3(CentreXSim(:,1),CentreYSim(:,1),CentreZSim(:,1),'-g')
85 plot3(LeftXSim(:,1),LeftYSim(:,1),LeftZSim(:,1),'-r')
86 plot3(RightXSim(:,1),RightYSim(:,1),RightZSim(:,1),'-b')
87 xlabel('x (m)')
88 ylabel('y (m)')
89 zlabel('z (m)')
90 view(-98,27)
91 hold off
92
93 % Define the top of the lifthill as the start position of the train:
94 [StartZ,StartNode] = max(CentreZSim);
95 StartX = CentreXSim(StartNode,1);
96 StartY = CentreYSim(StartNode,1);
97
98 % Write the coordinates of the left and right rail to the text-files:
99 formatSpec1 = ' [%8.4f %8.4f %8.4f ; \r\n';
100 fprintf(fileID1,formatSpec1,LeftXSim(1,1)-StartX,LeftYSim(1,1)-StartY,LeftZSim(1,1)-StartZ);
101 fprintf(fileID2,formatSpec1,RightXSim(1,1)-StartX,RightYSim(1,1)-StartY, ...
102 RightZSim(1,1)-StartZ);
103 for n = 2:(length(Nodenr)-1)
104     formatSpec2 = '%8.4f %8.4f %8.4f ; \r\n';
105     fprintf(fileID1,formatSpec2,LeftXSim(n,1)-StartX,LeftYSim(n,1)-StartY, ...
106 LeftZSim(n,1)-StartZ);
107     fprintf(fileID2,formatSpec2,RightXSim(n,1)-StartX,RightYSim(n,1)-StartY, ...
108 RightZSim(n,1)-StartZ);
109 end
110 formatSpec3 = '%8.4f %8.4f %8.4f]';
111 fprintf(fileID1,formatSpec3,LeftXSim(length(Nodenr),1)-StartX, ...
112 LeftYSim(length(Nodenr),1)-StartY,LeftZSim(length(Nodenr),1)-StartZ);
113 fprintf(fileID2,formatSpec3,RightXSim(length(Nodenr),1)-StartX, ...
114 RightYSim(length(Nodenr),1)-StartY,RightZSim(length(Nodenr),1)-StartZ);

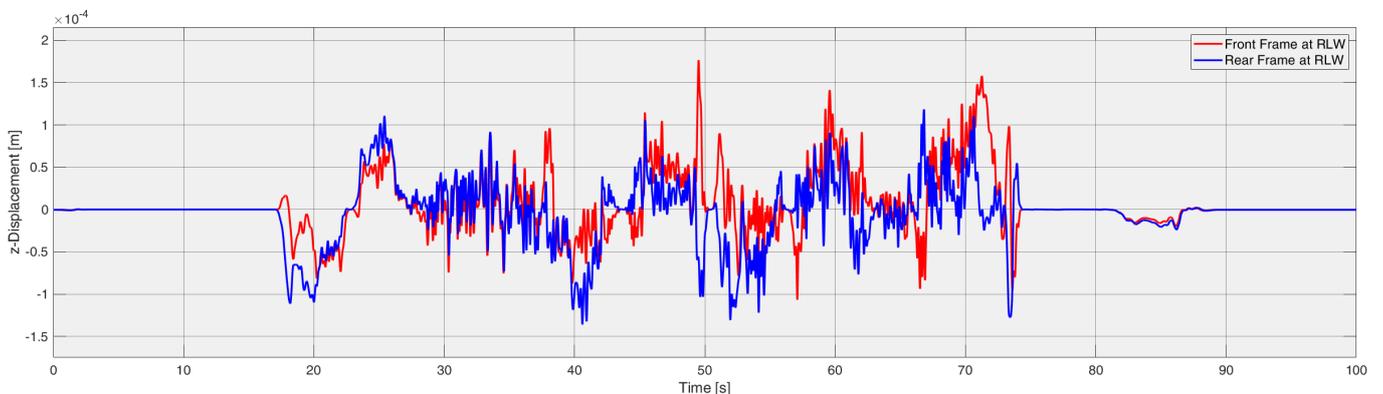
```

## D. Track Oscillations

An advantage of using piece-wise polynomial schemes is the exhibition of local geometrical control, which means that a variation in the control point position only affects the neighbourhood of that point while leaving the rest of the curve unchanged. However, when cubic splines are used to interpolate a set of control points, undesired oscillations are typically noticed in the track model. Especially in the vicinity of transition regions between for instance a straight and a circular segment, a curve is obtained that oscillates about the original line. No physical meaning can be assigned to the oscillatory behaviour, since the small perturbations of the track are inherent to the interpolation process. Since the undesired oscillations could be confused with track irregularities during a dynamic analysis, analytical segments are occasionally implemented instead of cubic interpolation. The implementation of straight, circular, and transition curves does not introduce unwanted oscillations in the track model. However, analytical segments do typically not prove to be an alternative method for defining the track geometry, since it is rather complex to describe fully spatial geometries such as a rollercoaster track [3]. Hence, track parameterisation with cubic splines is still preferred over analytical segments for describing the track geometry of the spinning rollercoaster. Furthermore, since a vibration analysis of the wheel-rail contact is beyond the scope of this research, the impact of the apparent oscillatory behaviour on the model can be regarded as negligible. Figure D.1 shows the track oscillations sensed at the front-right and rear-left wheel carrier for a control point increment  $\Delta s$  of one meter. From these figures, it can be deduced that the extent of the track oscillations is small with a maximum displacement in  $z$ -direction of only 0.195 millimeters.



(a) Track oscillations sensed by the front (red) and rear (blue) follower frame at the front-right wheel carrier



(b) Track oscillations sensed by the front (red) and rear (blue) follower frame at the rear-left wheel carrier

Figure D.1: Track perturbations sensed in the  $z$ -direction at the front-right and rear-left wheel carrier

## E. Hertzian Elliptic Contact Model

This appendix presents the analytical modelling of frictional rolling contact on a roller coaster. Although a variety of models is available to describe the frictional rolling contact between two bodies, the Hertzian contact model provides the basic contact mechanics in many advanced contact problems. The theory devised by Hertz provides the contact area shape and stress distribution for contact between different body geometries with purely elastic surfaces. The Hertzian contact theory is a basic theory that only holds under the restriction of several criteria, which implies that the application of the Hertzian contact model can only be justified if the contact surfaces are imposed to the following mathematical boundary conditions:

- There is no penetration by the contact surfaces (the contact pressure is never negative)
- There are not tangential stresses along the contact surfaces
- The displacements are equal to zero where a contact surface goes to infinity
- The contact force is defined by the surface integral of the stresses in the contact area

It should be explicitly mentioned that the Hertzian contact model is only valid for rolling contact between two purely elastic bodies in free rolling motion, which is the most basic form of rolling contact. The distance travelled by the free rolling body after a full rotation equals the circumference of the rolling body, which implies that there is no relative linear motion between the contact surfaces on macro scale. However, the rollercoaster wheels are allowed some lateral movement and they are always in some misalignment with the direction of motion of the train. Although lateral sliding and slip are therefore likely to be present to some degree, free rolling motion is usually considered in models of undriven wheels. Hence, it is presumed that the rolling contact between the track and a rollercoaster wheel can safely be approximated as pure rolling motion, since the wheels are not actively driven, braked or steered. The expressions for calculating the contact pressures and dimensions of the elliptic contact area were derived by Hertz on the basis of the following presumptions [11]:

- Elastic deformation
- The stress distribution below the surface is not affected by the finite dimensions of the contacting bodies
- The surfaces are assumed to be frictionless so that only a normal pressure is transmitted

### E.1 Modelling Rolling Contact

For properties such as the modulus of elasticity or the surface geometry, a collective equivalent parameter can be defined for the bodies in contact. The effective modulus of elasticity  $E_{ef}$  concerns the combined material elasticity, while the effective radius  $R_{ef}$  concerns the combined surface geometry. The effective radius  $R_{ef}$  is a measure of conformity between the bodies and therefore determines the shape and size of the contact area. The algebraic equations for  $E_{ef}$  and  $R_{ef}$  are given in Equation E.1 and E.2 respectively. The subscripts 1 and 2 in these equations refer to the two bodies in contact, where subscript 1 concerns the wheel while subscript 2 refers to the track. In both equations,  $E$  and  $\nu$  symbolize respectively the elasticity modulus and the Poisson ratio.

$$\frac{1}{E_{ef}} = \frac{1 - \nu_1^2}{2E_1} + \frac{1 - \nu_2^2}{2E_2} \quad (\text{E.1})$$

$$\frac{1}{R_{ef}} = \frac{1}{R_x} + \frac{1}{R_z} \quad (\text{E.2})$$

The considered contact area is assumed to be small with respect to the dimensions of the contacting bodies. Even at curved track sections, the radius of curvature of the track is typically orders of magnitude larger than the track or wheel radius. It is presumed that the curvature of the track has a negligible effect on the contact between the rollercoaster wheel and the track, and hence an infinite radius of curvature of the track is assumed. Furthermore, the running surface of the wheel is presumed to be flat instead of conic, which implies that the radius of curvature of the wheel cover is taken equal to infinity as well. The implications of these assumptions with respect to the parameters  $R_x$  and  $R_z$  are demonstrated in respectively Equation E.3 and E.4. From these equations and Figure E.1 it can be concluded that  $R_x$  equals the wheel radius ( $R_{1,x}$ ), while  $R_z$  is equivalent to the track radius ( $R_{2,z}$ ).

$$\frac{1}{R_x} = \frac{1}{R_{1,x}} + \frac{1}{R_{2,x}} = \frac{1}{R_{1,x}} + \frac{1}{\infty} = \frac{1}{R_{1,x}} \quad (\text{E.3})$$

$$\frac{1}{R_z} = \frac{1}{R_{1,z}} + \frac{1}{R_{2,z}} = \frac{1}{\infty} + \frac{1}{R_{2,z}} = \frac{1}{R_{2,z}} \quad (\text{E.4})$$

The concentrated load in the wheel-rail interface on a roller coaster is characterized by an elliptic contact area. The Hertzian elliptic contact model is capable of describing contact between two elastic cylindrical bodies, where the shape of the elliptic contact area is defined by the semi-axes  $a$  and  $b$ . The definitions of the semi-axes  $a$  and  $b$  are presented by Equations E.5 and E.6, in which the normal force on the bodies in contact is defined by  $N$ .

$$a = a^* \sqrt[3]{\frac{3NR_{ef}}{E_{ef}}} \quad \text{with} \quad a^* = \kappa \sqrt[3]{1 + \frac{2(1 - \kappa^2)}{\pi\kappa^2} - 0.25\ln(\kappa)} \quad (\text{E.5})$$

$$b = b^* \sqrt[3]{\frac{3NR_{ef}}{E_{ef}}} \quad \text{with} \quad b^* = \frac{a^*}{\kappa} \quad (\text{E.6})$$

The value of the parameter  $\kappa$  can be computed according to the expression in Equation E.7.

$$\frac{1}{\kappa} = 1 + \sqrt{\frac{\ln(\frac{16}{\lambda})}{2\lambda}} - \sqrt{\ln(4) + 0.16\ln(\lambda)} \quad \text{with} \quad \lambda = \min \left[ \frac{R_x}{R_z}, \frac{R_z}{R_x} \right] \quad (\text{E.7})$$

## E.2 Modelling Viscoelasticity

For the elastic contact surfaces that are considered by the Hertzian contact model, the experience in rolling contact is that indentation of the surfaces takes place at the front of the contact area, while relaxation occurs behind the contact area. Although elastic material is regarded, the rate of deformation is unequal to the rate of recovery due to elastic hysteresis. Elastic material namely suffers from small energy losses during deformation and recovery due to internal friction in the material. The work that is provided to the contact material is partly transformed into heat during the deformation process. Hence, part of the physical work is not regained, and a torque is experienced as a resistance to rolling. Hysteresis can typically be regarded as a major contributor to the total rolling resistance, since rolling friction is mainly caused by internal friction. The percentage of the total rolling resistance accounted for by hysteresis is commonly defined by the hysteresis loss factor of rolling motion  $\alpha_r$ . The viscoelastic model that is discussed in this section combines the elastic contact definition with this loss factor. Instead of modelling rolling resistance by explicitly defining a viscoelastic stress-strain relationship, the deformations in the contact area are modelled by regarding an elastic rolling contact that is adjusted by the hysteresis loss factor. In purely elastic rolling contact, compression and relaxation of the contact materials in front and behind the contact area are equal. The hysteresis loss factor accounts for the mechanical energy that is dissipated into heat in a viscoelastic contact. The predominant advantage of this approach is the reduced complexity of calculations on linear-elastic material in comparison with calculations on viscoelastic material. A friction force  $P$  can be defined

by equating the hysteresis energy loss to the distance travelled in rolling motion  $s$ , as shown in Equation E.8.

$$P \cdot s = \alpha_r \cdot W \quad (\text{E.8})$$

where  $\alpha_r$  is the hysteresis loss factor for a rolling motion that defines the percentage of work  $W$  spent on deforming the contact bodies that is not regained after relaxation. For the Hertzian elliptic contact model, Equation E.8 can be rewritten to the expression shown in Equation E.9.

$$P = \alpha_r \frac{3}{16} \frac{N}{R} b = \mu N \quad \text{with} \quad \mu = \left( \alpha_r \frac{3}{16} \frac{b}{R} \right) \quad (\text{E.9})$$

where  $R$  is the undeformed outer radius of the rolling body and  $\mu$  symbolizes the friction coefficient. The hysteresis loss factor  $\alpha_r$  is a material property that is typically determined by multiplying the equivalent loss factor for pure shear  $\alpha_s$  by a compensating factor  $f_{rs}$ . The values for  $\alpha_s$  and subsequently  $\alpha_r$  can be computed using Equation E.10, in which  $\lambda_s$  is the logarithmic decrement of the function that bounds the damped oscillations of a vibration. The empirical compensating factor  $f_{rs}$  depends on the ratio of elliptic semi-axes  $a$  and  $b$ . The compensating factor  $f_{rs}$  is equal to 2.5 for the track radius and wheel radii considered in this analysis.

$$\alpha_r = f_{rs} \cdot \alpha_s \quad \text{with} \quad \alpha_s = \frac{1 - e^{-\lambda_s}}{1 + e^{-\lambda_s}} \quad (\text{E.10})$$

The logarithmic decrement is related to the mechanical loss factor  $\tan(\delta)$  by Equation E.11. The property  $\tan(\delta)$  can be regarded as a generalized measure of viscoelasticity [12].

$$\tan(\delta) = \frac{\lambda_s}{\pi} \left( 1 - \frac{\lambda_s^2}{4\pi^2} \right)^{-1} \quad (\text{E.11})$$

A rollercoaster wheel is typically composed of a polyurethane tire that is bonded around the outside diameter of an aluminum hub. The wheel cover is presumed to be made from polyurethane PPDI, since its low hysteric energy and superior dynamic properties make this material ideally suited for applications like high speed rollercoaster wheels [13]. The value of the mechanical loss factor for polyurethane PPDI is presented in [14] as a function of temperature. The minimum and maximum temperature under which a rollercoaster is expected to operate are respectively  $-10^\circ\text{C}$  and  $40^\circ\text{C}$ . At the average operating temperature of  $T = 15^\circ\text{C}$  the mechanical loss factor equals 0.045. Substituting this value for  $\tan(\delta)$  into Equation E.11 and solving for  $\lambda_s$  yields  $\lambda_s = 0.1413$ , which can subsequently be substituted in the expressions in Equation E.10 to determine the hysteresis loss factor. The elasticity modulus and Poisson ratio of polyurethane PPDI equal respectively 55 MPa ( $E_1$ ) and 0.48 ( $\nu_1$ ) [15]. Since rollercoaster track is typically composed of steel tubes,  $E_2$  and  $\nu_2$  are set equal to respectively 200 GPa and 0.30. The input parameters that are used in the rolling friction analysis are summarized in Table E.1. Substituting these values and a certain normal load into the expressions presented in this appendix and solving the equations will ultimately yield the value of the corresponding friction coefficient and friction force.

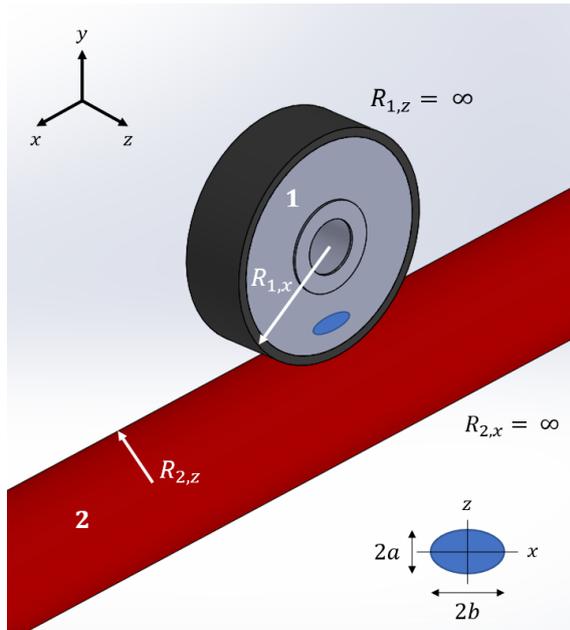


Figure E.1: Specification of dimensions used in the rolling friction analysis

| Property              | Value     |
|-----------------------|-----------|
| $E_1$                 | 55 MPa    |
| $\nu_1$               | 0.48      |
| $E_2$                 | 200 GPa   |
| $\nu_2$               | 0.30      |
| $\tan(\delta)$        | 0.045     |
| $R_x$ (running wheel) | 0.12625 m |
| $R_x$ (guide wheel)   | 0.1025 m  |
| $R_x$ (upstop wheel)  | 0.1 m     |
| $R_z$                 | 0.0675 m  |

Table E.1: Input parameters used in the rolling friction analysis and their corresponding values

## F. Bearing Friction Calculation

A smooth motion between separate components is facilitated by the use of bearings, which are characterized by a non-constant frictional moment that depends on certain tribological phenomena. Although the frictional moment induced by the bearings is typically small in comparison to other sources of resistance such as rolling friction or air drag, its effect is assigned a sufficient level of significance for it to be included in the model. The frictional moment induced by the bearings is modelled for two distinct contact areas, namely at the revolute joints between the wheels and the accompanying hubs and at the revolute joint between the gondola and the main chassis beam. The frictional moment is not modelled for the spherical and revolute joint between respectively the front and rear axle and the chassis main beam, since the degree of relative motion between these components is small in comparison to the other components.

The SKF model is used for calculating the frictional moment induced by the bearings, since it closely follows the real behaviour of the bearing. All contact areas and design changes made to SKF bearings over recent years are considered by this model, including internal and external influences. The SKF model uses Equation F.1 for calculating the total frictional moment.

$$M = M_{rr} + M_{sl} + M_{seal} + M_{drag} \quad (\text{F.1})$$

where  $M$  is the total frictional moment,  $M_{rr}$  is the rolling frictional moment,  $M_{sl}$  is the sliding frictional moment,  $M_{seal}$  is the frictional moment of the seals, and  $M_{drag}$  is the frictional moment of drag losses, churning, splashing, and other similar physical phenomena. It should be noted that the SKF model is derived from more advanced computational models and that it is valid for grease or oil lubricated bearings. Although the model only provides approximate reference values, it is believed to be sufficiently accurate for modelling the bearing frictional moment between the mentioned components.

### F.1 Rolling Frictional Moment

The first term on the right-hand side of Equation F.1 can be calculated using the expression in Equation F.2.

$$M_{rr} = \phi_{ish} \phi_{rs} G_{rr} (\nu n)^{0.6} \quad (\text{F.2})$$

where  $M_{rr}$  is the rolling frictional moment in Nmm,  $\phi_{ish}$  is the inlet shear heating reduction factor,  $\phi_{rs}$  is the kinematic replenishment reduction factor,  $G_{rr}$  is a variable that depends on the bearing type and load,  $n$  is the rotational speed in r/min, and  $\nu$  is the actual operating viscosity of the lubrication oil in  $\text{mm}^2/\text{s}$ . The inlet shear heating reduction factor captures the effect of a reverse flow caused by repelling oil close to the contact area. This reverse flow shears the lubricant and generates heat, which lowers the oil viscosity and reduces the film thickness and rolling friction. The inlet shear heating reduction factor can be estimated using the expression in Equation F.3.

$$\phi_{ish} = \frac{1}{1 + 1.84 \cdot 10^{-9} (n d_m)^{1.28} \nu^{0.64}} \quad (\text{F.3})$$

where  $d_m$  is the bearing mean diameter in millimeters. The velocity of the train can be translated to a rotational speed according to Equation F.4, where the values for  $R_x$  can be found in Table E.1 for each wheel type.

$$n = \frac{60 \cdot v}{2\pi \cdot R_x} \quad (\text{F.4})$$

The bearing mean diameter can be calculated using the bearing bore diameter  $d$  and the bearing outside diameter  $D$ , as shown in Equation F.5.

$$d_m = 0.5(d + D) \quad (\text{F.5})$$

It is presumed that both the wheel axles and the gondola pivot joint are fitted with two tapered roller bearings each, since these bearings are designed to accommodate simultaneously acting radial and axial loads. The use of two tapered roller bearings constrains motion in all directions except for the desired rotation around the rotational axis of the wheels or gondola. The selection of the tapered roller bearings is predominantly based on the dimensions of the axles and the rotating components, while also taking the limiting rotational speed and load limit into account. The main dimensions of each selected bearing type are shown in Table F.1. Using these dimensions and Equation F.5, the bearing mean diameter  $d_m$  can be calculated for each type of tapered roller bearing.

| Component      | Bearing   | Width [mm] | Bore Diameter [mm] | Outside Diameter [mm] |
|----------------|-----------|------------|--------------------|-----------------------|
| Running Wheels | SKF 32912 | 17         | 60                 | 85                    |
| Guide Wheels   | SKF 32009 | 20         | 45                 | 75                    |
| Upstop Wheels  | SKF 32009 | 20         | 45                 | 75                    |
| Gondola        | SKF 32922 | 25         | 110                | 130                   |

Table F.1: Dimensions of the tapered roller bearings

The viscosity of the lubricant  $\nu$  is presumed to equal  $30 \text{ mm}^2/\text{s}$  at a temperature of  $40^\circ$  Celsius. For low level oil bath lubrication, continuous over-rolling displaces excess lubricants from the raceways. In applications where the viscosity or rotational speeds are high, the lubricant may not have sufficient time to replenish the raceways, causing a kinematic starvation effect. Kinematic starvation reduces the thickness of the hydrodynamic film and rolling friction, and the the kinematic replenishment reduction factor can be estimated using Equation F.6.

$$\phi_{rs} = \frac{1}{\exp\left(K_{rs} \nu n (d + D) \sqrt{\frac{K_Z}{2(D-d)}}\right)} \quad (\text{F.6})$$

where  $K_{rs}$  represents the replenishment constant, which equals  $3 \cdot 10^{-8}$  for the low level oil bath that is assumed in this analysis. The term  $K_Z$  is a bearing type related geometric constant, which is equal to 6 for tapered roller bearings. The value that should be assigned to the variable  $G_{rr}$  in case of tapered roller bearings can be calculated according to Equation F.7.

$$G_{rr} = R_1 d_m^{2.38} (F_r + R_2 Y F_a)^{0.31} \quad (\text{F.7})$$

The values for the geometric constants  $R_1$  and  $R_2$  and the calculation factor  $Y$  are listed in Table F.2 for each of the relevant tapered roller bearing types. Furthermore,  $F_a$  and  $F_r$  represent respectively the external axial and radial load applied on the bearings.

| Bearing   | $R_1$                | $R_2$ | $S_1$ | $S_2$ | $Y$ |
|-----------|----------------------|-------|-------|-------|-----|
| SKF 32912 | $2.31 \cdot 10^{-6}$ | 10.9  | 0.009 | 2     | 1.8 |
| SKF 32009 | $2.38 \cdot 10^{-6}$ | 10.9  | 0.014 | 2     | 1.5 |
| SKF 32922 | $2.31 \cdot 10^{-6}$ | 10.9  | 0.009 | 2     | 1.7 |

Table F.2: Values for the geometric constants and calculation factor used for computing the rolling and sliding frictional moments of tapered roller bearings

## F.2 Sliding Frictional Moment

The sliding frictional moment can be calculated using the expression in Equation F.8.

$$M_{sl} = G_{sl} \mu_{sl} \quad (\text{F.8})$$

where  $G_{sl}$  is a variable that depends on the bearing type and load, while  $\mu_{sl}$  represents the sliding friction coefficient. For tapered roller bearings, the value of  $G_{sl}$  can be computed according to Equation F.9.

$$G_{sl} = S_1 d_m^{0.82} (F_r + S_2 Y F_a) \quad (\text{F.9})$$

where the values for the geometric constants  $S_1$  and  $S_2$  and the calculation factor  $Y$  can again be found in Table F.2. The sliding friction coefficient  $\mu_{sl}$  for full-film and mixed lubrication conditions can be estimated using Equation F.10.

$$\mu_{sl} = \phi_{bl} \mu_{bl} + (1 - \phi_{bl}) \mu_{EHL} \quad (\text{F.10})$$

where  $\phi_{bl}$  represents the weighting factor for the sliding friction coefficient, whose value can be computed with the expression in Equation F.11.

$$\phi_{bl} = \frac{1}{\exp(2.6 \cdot 10^{-8} (n \nu)^{1.4} d_m)} \quad (\text{F.11})$$

where  $\mu_{bl}$  equals 0.12 for  $n \neq 0$  and 0.15 for  $n = 0$ , where the former value is considered in this analysis. Furthermore, the value for  $\mu_{EHL}$  equals 0.002 for tapered roller bearings [16].

## F.3 Frictional Moment of Seals and Drag Losses

Since the selected tapered roller bearings are not fitted with contact seals, the term  $M_{seal}$  in Equation F.1 is set to zero. The contribution of the term  $M_{drag}$  to the total frictional moment is typically very small in comparison to the contributions from the rolling and sliding frictional moments. Hence, the frictional moment induced by drag losses is neglected in this analysis. Consequently, the total frictional moment can be calculated by adding the contributions from the rolling and sliding frictional moments, as shown in Equation F.12. Hence, all expressions have been provided at this point that are required for calculating the total frictional moment.

$$M = M_{rr} + M_{sl} \quad (\text{F.12})$$

# G. Modelling Air Resistance

A stationary fluid exerts only normal pressure forces on the surface of a body immersed in it, such as the gondola geometry surrounded by air. However, a moving fluid such as an air flow also exerts tangential shear forces on the gondola surface because of the no-slip condition caused by viscous effects. Both of these forces have components in the direction of flow, and thus the drag force is due to the combined effects of pressure and wall shear forces in the flow direction. The drag force  $F_D$  depends on the density  $\rho$  of the fluid, the upstream velocity  $v$ , and the size, shape, and orientation of the body. The drag characteristics of a body is represented by the dimensionless drag coefficient  $C_D$ . The drag force can be calculated according to Equation G.1.

$$F_D = \frac{1}{2} \rho C_D v^2 A_{front} \tag{G.1}$$

## G.1 Frontal Surface Values

In Equation G.1, the parameter  $A_{front}$  refers to the frontal surface of the body. For bodies that tend to block the flow, such as a rollercoaster train, the frontal surface is the area projected on a plane normal to the direction of flow. Since the effect of wind is disregarded in this analysis, the direction of motion of the rollercoaster train always directly opposes the relative direction of the air flow. The orientation of the chassis, pivot axle, bogies and wheels with respect to the air flow remains constant during the ride due to their direct alignment with the track. The drag force that is exerted on these components can therefore be determined using a constant drag coefficient and frontal surface, where a drag coefficient of 0.8 is presumed for the entire assembly of chassis, pivot axle, bogies and wheels. The combined frontal surface of these components is shown in Figure G.6 in Section G.8. On the other hand, the orientation of the gondola continuously changes with respect to the direction of motion of the train during the ride, which implies that the gondola frontal surface should be regarded as a variable instead of constant. The gondola frontal surface has been determined manually at seven equidistant gondola orientations between  $0^\circ$  and  $90^\circ$ . Please note that it suffices to consider a ninety degree range instead of a full rotation, since the geometry of the gondola is characterized by two planes of symmetry. The gondola orientation can be represented as an angle of attack of the air flow, which is symbolized by  $\alpha$ . The frontal surface at the seven different gondola orientations is shown in Figures G.6 and G.7 in Section G.8. The corresponding values of the frontal surfaces are given in Table G.1.

| Component | Orientation         | $A_{front}$ [m <sup>2</sup> ] |
|-----------|---------------------|-------------------------------|
| Chassis   | $\alpha = 0^\circ$  | 0.386                         |
| Gondola   | $\alpha = 0^\circ$  | 1.816                         |
| Gondola   | $\alpha = 15^\circ$ | 1.884                         |
| Gondola   | $\alpha = 30^\circ$ | 1.969                         |
| Gondola   | $\alpha = 45^\circ$ | 2.107                         |
| Gondola   | $\alpha = 60^\circ$ | 2.153                         |
| Gondola   | $\alpha = 75^\circ$ | 1.983                         |
| Gondola   | $\alpha = 90^\circ$ | 1.666                         |

Table G.1: Frontal area of the chassis and the gondola at different orientations

## G.2 Cubic Interpolation Spline

As mentioned in the previous subsection, the gondola frontal surface has been determined for seven different orientations. Interpolation between the frontal surface values in Table G.1 is required to determine the

frontal surface for the entire range of gondola orientations. The implementation of an interpolation spline is preferred over the use of a regression model, since the latter results in a function that does not necessarily intersect with the data points. For more than two data points, the interpolation spline consists of two or more piece-wise polynomials. Local coefficients are determined for each interval between a pair of data points, and the lower endpoint of the corresponding interval should be subtracted to use the coefficients in a conventional polynomial equation. For instance, the polynomial for the coefficients  $[a, b, c, d]$  on the interval  $[x_1, x_2]$  is described by Equation G.2. The third power in this expression indicates that the interpolation spline is a cubic one.

$$f(x) = a(x - x_1)^3 + b(x - x_1)^2 + c(x - x_1) + d \quad (\text{G.2})$$

The built-in `Matlab`-function `spline` can be used to determine the polynomial coefficients for each interval. The lower endpoint and the polynomial coefficients can subsequently be substituted in Equation G.2 to determine the function that describes the cubic interpolation spline on a specific interval. Hence, when the corresponding interval is known for a certain gondola orientation, the correct function can be used to determine the frontal surface of the gondola. The frontal surface values and the piece-wise cubic interpolation splines are shown in Figure G.1. Please note that the interpolation functions between  $90^\circ$  and  $180^\circ$  are the mirrored equivalents of the cubic interpolation splines that have been calculated for the domain between  $0^\circ$  and  $90^\circ$ .

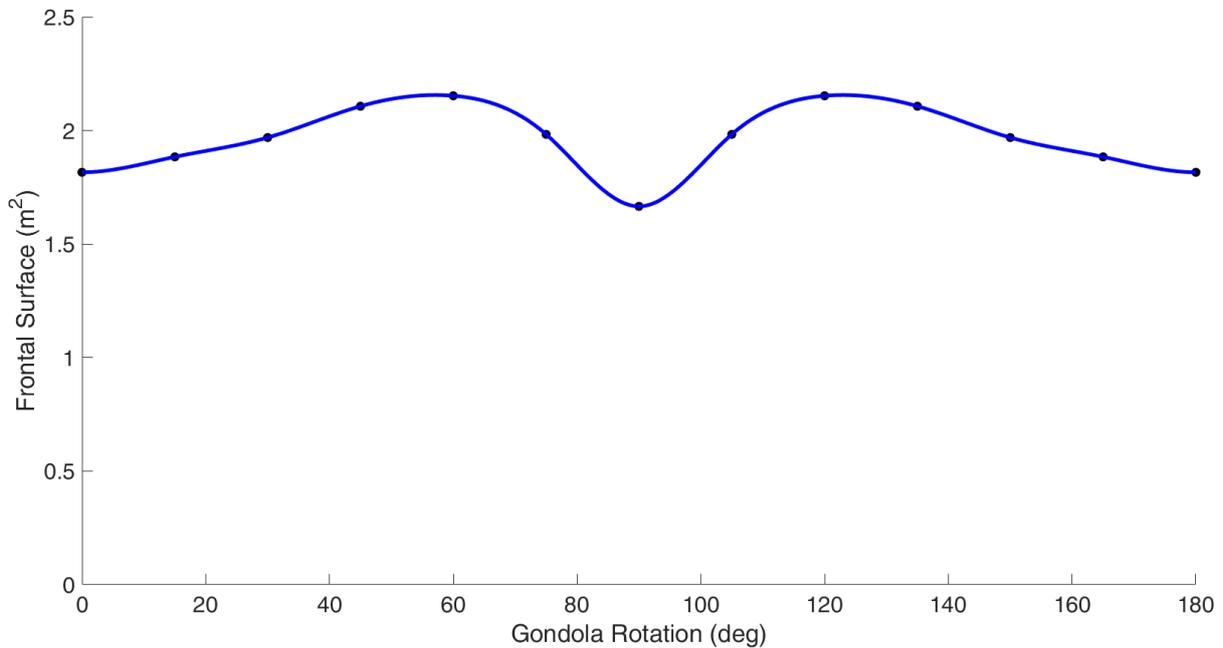


Figure G.1: The frontal surface as a function of the gondola orientation

### G.3 Boundary Conditions

Numerous numerical simulations have been performed in Ansys Discovery AIM 19.2 (from this point on referred to as Ansys) to determine the drag force that is exerted by the air on the gondola. One of the capabilities of this program is the feature that allows the user to model a fluid flow around an imported geometry. Several aspects of the gondola geometry were altered to maintain the computational time at a sufficient level of practicality. For instance, the restraints and other detailed surfaces were omitted from the gondola design, so only simple and large surfaces would remain part of the gondola geometry. Including bodies such as the restraints would namely lead to a rather complex mesh with a large number of nodes, while their effect on the air drag is relatively small in comparison to larger surfaces on the gondola. The

large number of nodes would drastically increase the required computational time for each simulation with only limited effect on the results, which underscores the significance of simplifying the gondola geometry. Another simple yet very effective measure to reduce the computational time is to use one of the symmetry planes on the gondola geometry. By only modelling half of the gondola the number of nodes is evidently reduced with a factor two, which could lead to a reduction in computational time of a factor eight.

The air flow around the simplified and halved gondola geometry can be modelled by creating an external body around the gondola. The cushion of the enclosure equals five meters at the front and the back of the gondola, while the left and top clearance equal 2.5 meters each. No clearance is included in the model at the symmetry plane and the bottom of the gondola. The gondola is excluded from the physics region with boundary layer properties assigned to its surface. The frontal surface inlet of the enclosure is assigned with a certain upstream velocity, while the rear surface outlet is assigned a gauge static pressure equal to zero. The upper surface and the side surface away from the gondola geometry are given the properties of an opening, with an entrainment pressure equal to zero. At last, the other side view is assigned with the symmetry condition, while the remaining bottom surface has the properties of a wall. The described configuration is clarified in Figure G.2.

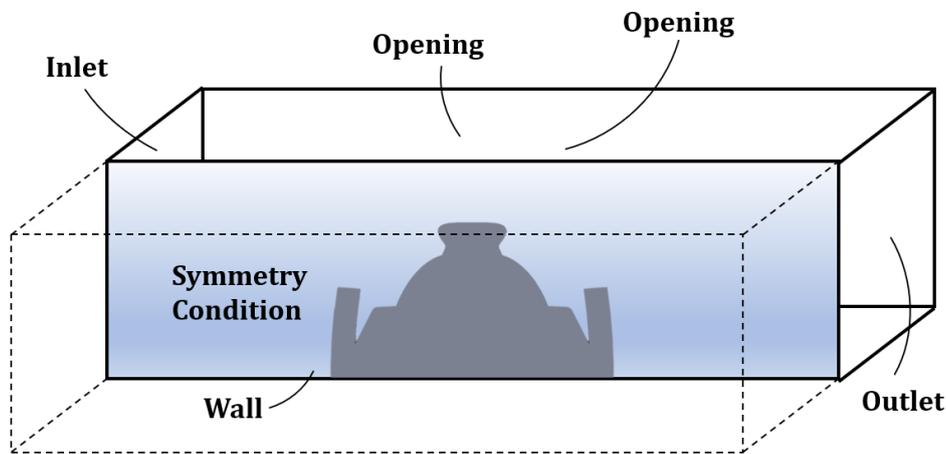


Figure G.2: The properties of the enclosure surfaces around the gondola

A major disadvantage of halving the gondola geometry is the inability to model air flow around the gondola with a value for  $\alpha$  unequal to  $0^\circ$  or  $90^\circ$ . At  $0^\circ < \alpha < 90^\circ$  the symmetry plane of the gondola is namely not aligned with the direction of the air flow. Hence, it would be incorrect to model the air flow around the halved gondola at for instance  $\alpha = 30^\circ$  or  $\alpha = 45^\circ$ . A conceivable solution to this problem would be to position the gondola either parallel or perpendicular to the symmetry plane of the enclosure, while specifying a fluid flow with a certain angle  $\alpha$  with respect to the gondola. However, since the symmetry condition also applies to the air flow, the air flows would start at opposite angles on both sides of the symmetry plane and realign again when they approach the outlet. The correct modelling of the air flow around the gondola for  $0^\circ < \alpha < 90^\circ$  is only possible if the full geometry of the gondola is considered during the numerical simulation. However, this would come at the cost of a drastically increased computational time, which is unfeasible in light of the relatively large number of simulations that have to be performed. Therefore, the decision is taken to only model air flow around a gondola that has been halved at its two symmetry planes. The reader is referred to Figure G.3 for clarification on the approach that has been used for modelling the air flow around the gondola.

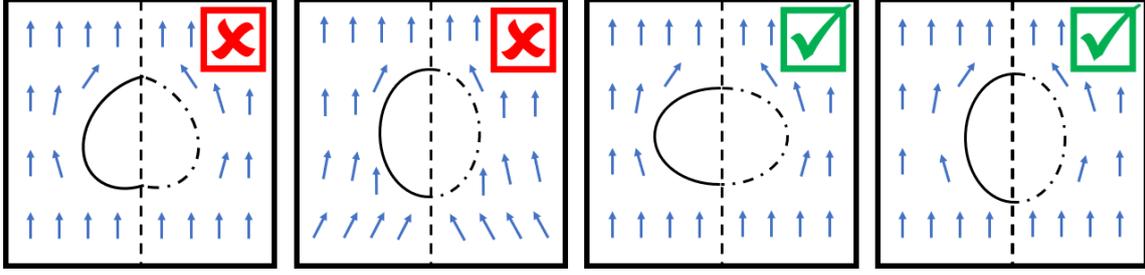


Figure G.3: Incorrect and correct modelling of the air flow around the gondola in Ansys

## G.4 Reynolds Number

The flow regime around an object depends mainly on the ratio of the inertia forces to the viscous forces in the fluid. The dimensionless quantity that represents this ratio is called the Reynolds number, which is expressed in Equation G.3 for external flow.

$$Re = \frac{\rho \cdot v \cdot L_c}{\mu} \quad (\text{G.3})$$

where  $\rho$  is the density of air, which equals  $1.2 \text{ kg/m}^3$  at a temperature of  $293.15 \text{ K}$ . Additionally,  $v$  represents the upstream velocity (in  $\text{m/s}$ ),  $L_c$  is the characteristic length of the geometry (in  $\text{m}$ ), and  $\mu$  is the dynamic viscosity of the air (in  $\text{kg/ms}$ ). The latter can be determined using Sutherland's law, which relates the dynamic viscosity to the absolute temperature of an ideal gas. Sutherland's law is defined by Equation G.4.

$$\mu = \mu_0 \frac{T_0 + S}{T + S} \left( \frac{T}{T_0} \right)^{3/2} \quad (\text{G.4})$$

where  $T_0$  is the reference temperature of  $273.15 \text{ K}$  and  $\mu_0$  is the viscosity at the reference temperature, which is equal to  $1.716 \cdot 10^{-5} \text{ kg/ms}$ . Additionally, the Sutherland temperature  $S$  is equal to  $110.4 \text{ K}$ . At a temperature of  $293.15 \text{ K}$ , the dynamic viscosity  $\mu$  equals:

$$\mu = 1.716 \cdot 10^{-5} \frac{273.15 + 110.4}{293.15 + 110.4} \left( \frac{293.15}{273.15} \right)^{3/2} = 1.813 \cdot 10^{-5} \text{ kg/ms} \quad (\text{G.5})$$

Although the drag coefficient depends on the dynamic properties of the air flow, the coefficient may typically be regarded as a constant for a Reynolds number larger than  $Re = 10^6$  [17]. It will be verified by means of numerous simulations if this statement also applies to the air flow around the gondola. The characteristic length equals the length of the gondola in longitudinal direction for  $\alpha = 0^\circ$  ( $L_c = 2.4 \text{ m}$ ), while the characteristic length is equal to the width of the gondola ( $L_c = 1.6 \text{ m}$ ) for  $\alpha = 90^\circ$ . The velocities corresponding to  $Re = 10^6$  can be calculated for both cases, as shown in respectively Equation G.6 and Equation G.7.

$$\alpha = 0^\circ : v = \frac{Re \cdot \mu}{\rho \cdot L_c} = \frac{10^6 \cdot 1.813 \cdot 10^{-5}}{1.2 \cdot 2.4} = 6.3 \text{ m/s} = 22.7 \text{ km/h} \quad (\text{G.6})$$

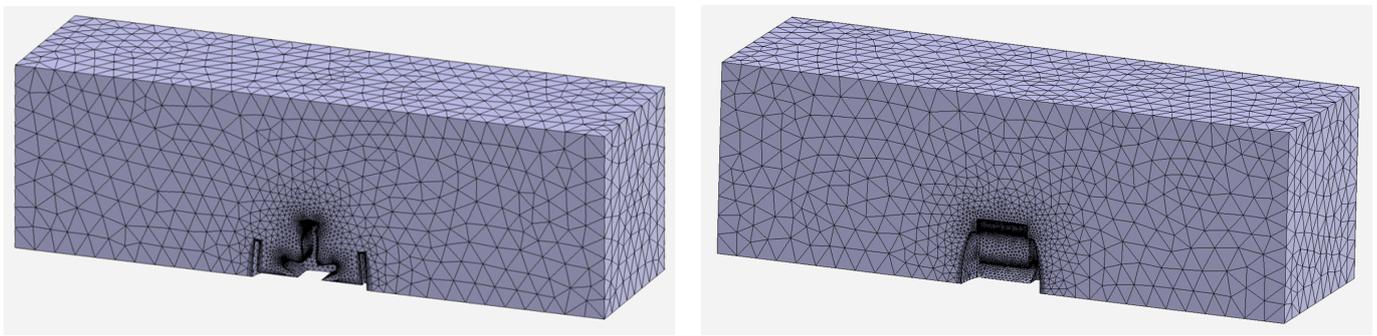
$$\alpha = 90^\circ : v = \frac{Re \cdot \mu}{\rho \cdot L_c} = \frac{10^6 \cdot 1.813 \cdot 10^{-5}}{1.2 \cdot 1.6} = 9.4 \text{ m/s} = 33.8 \text{ km/h} \quad (\text{G.7})$$

This would imply that the drag coefficient should be regarded as a variable for velocities lower than respectively  $6.3 \text{ m/s}$  or  $9.4 \text{ m/s}$ , while the drag coefficient could be held constant for respectively  $v > 6.3 \text{ m/s}$  and  $v > 9.4 \text{ m/s}$ .

## G.5 Initialization

The maximum velocity during a ride on the spinning rollercoaster is estimated to be  $v_{max} = 20 \text{ m/s}$ . Hence, to verify the statement of a constant drag coefficient for  $v > 6.3 \text{ m/s}$ , six simulations are performed at  $\alpha =$

$[0^\circ, 90^\circ]$  and  $v = [6.3, 9.4, 20]$  m/s. Please note that the minimum required number of simulations equals four, but that the two additional simulations allow a better comparison between the results. The adopted meshes for respectively  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  are shown in Figure G.4a and G.4b respectively. The mesh has been refined near the gondola surface, while the mesh is relatively coarse further away from the gondola geometry. An equivalent mesh has been used for each simulation despite the varying boundary conditions that were applied to the different models. Once a simulation is completed, the drag force exerted on the gondola surface can be extracted from Ansys, of which the results are shown in Table G.2. The drag force is defined as the net force exerted by a fluid on a body in the direction of flow due to the combined effects of wall shear and pressure forces. The part of the drag that is due directly to wall shear stress is called the skin friction drag since it is caused by frictional effects, whereas the part that is due directly to the pressure is called the pressure drag. When the friction and pressure drag are known, the total drag force can be calculated by directly adding them. For the six simulations that were performed in Ansys, the pressure drag accounts on average for 99.5 percent of the total drag, while only the remaining 0.5 percent can be assigned to frictional effects.



(a) The adopted mesh of the fluid volume for  $\alpha = 0^\circ$

(b) The adopted mesh of the fluid volume for  $\alpha = 90^\circ$

Figure G.4: The adopted meshes during the simulations in Ansys

The drag coefficient could be calculated by rewriting Equation G.1 into the expression shown in Equation G.8. Substituting the corresponding velocity and frontal surface into this formula yields the drag coefficient, of which the results are presented in Table G.2.

$$C_d = \frac{2 \cdot F_d}{\rho \cdot v^2 \cdot A_{front}} \quad (\text{G.8})$$

From the values in Table G.2 it can be deduced that the drag coefficient is substantially higher for  $\alpha = 90^\circ$  in comparison to the drag coefficients found for  $\alpha = 0^\circ$ . The total drag can almost entirely be assigned to the occurrence of pressure drag for both gondola orientations. Hence, the underlying cause of the different drag coefficients is also likely to originate from the occurrence of pressure drag. The pressure drag is proportional to the frontal area and to the difference between the pressures acting on the front and back of the gondola. From Table G.1 it can be deduced that the ratio between the frontal surfaces of both gondola orientations equals 1.09. However, the average ratio between the drag coefficients of both values for  $\alpha$  equals 1.48, which is substantially larger. Hence, the majority of the pressure drag is presumably caused by contributions from the pressure difference between the front and rear of the gondola. When the air stream separates from the gondola surface, it forms a separated region between the gondola and the air stream. The larger the separated region, the larger the pressure drag is. The effects of flow separation are felt far downstream in the form of reduced velocity relative to the upstream velocity [7]. The velocity contours are shown in Section G.10 for the performed simulations. From these figures, it can be deduced that the separated region is typically larger for  $\alpha = 90^\circ$  than for  $\alpha = 0^\circ$ , which explains the higher drag force and drag coefficient for the former gondola orientation.

The statement of a constant drag coefficient for  $v > 9.4$  m/s appears to hold for a gondola orientation of  $90^\circ$ . However, from Table G.2 it can also be deduced that the drag coefficient is substantially larger for

| $\alpha$   | $\mathbf{v}$ [m/s] | $\mathbf{F}_d$ [N] | $\mathbf{C}_d$ |
|------------|--------------------|--------------------|----------------|
| $0^\circ$  | 6.3                | 46.266             | 1.070          |
| $0^\circ$  | 9.4                | 103.47             | 1.075          |
| $0^\circ$  | 20                 | 497.75             | 1.142          |
| $90^\circ$ | 6.3                | 63.975             | 1.613          |
| $90^\circ$ | 9.4                | 143.17             | 1.621          |
| $90^\circ$ | 20                 | 647.21             | 1.619          |

Table G.2: Drag forces and drag coefficients at various gondola orientations and upstream velocities

air flows at the maximum velocity than at the velocity that corresponds to  $Re = 10^6$ . Hence, additional simulations are required to accurately determine the drag coefficient as a function of the upstream velocity.

## G.6 Air Flow Simulations

The upstream velocities that are modelled in the additional simulations are distributed as evenly as possible on the domains  $[0, 6.3]$  and  $[6.3, 20]$  m/s. Fourteen simulations have been performed in total under the presumption of steady-state conditions and incompressible flow. The resultant drag forces and coefficients are presented in Table 2.4. Please note that it would be impossible to determine the value for the drag coefficient in case of a zero upstream velocity. Therefore, an upstream velocity equal to unity has been simulated for both gondola orientations, and the resultant drag forces and coefficients are presumed similar to their equivalents at a zero upstream velocity. A cubic spline as described in Section G.2 is used to interpolate between the drag coefficient values. The data points and the interpolation splines are presented in Figure G.5.

From Figure G.5 the conclusion can be drawn that the drag coefficient is practically constant for  $v > 9.4$  m/s, as was anticipated based on the results in Section G.5. On the other hand, Figure G.5 shows a significant fluctuation in the value for the drag coefficient between  $v = 6.3$  m/s and  $v_{max}$ . Hence, the statement of a constant drag coefficient for Reynolds numbers larger than  $10^6$  does not apply to the gondola orientation with  $\alpha = 0^\circ$ . The difference between the drag coefficients at  $v = 6.3$  m/s and  $v_{max}$  equals 6.7 percent, of which the significance justifies the additional simulations. The underlying cause for the increasing drag coefficient for Reynolds numbers larger than  $10^6$ , in contrast to the constant  $C_d$  for  $\alpha = 90^\circ$ , remains unknown up to this point.

## G.7 Discussion on the Residuals

The problem of air flow around the gondola geometry is highly non-linear, and the governing Navier-Stokes equations can only be solved analytically if restrictive conditions are imposed. Hence, the computational fluid dynamics solutions must be calculated iteratively. The Navier-Stokes equations are always solved together with the continuity equation, where the former represents the conservation of momentum, while the latter represents conservation of mass. The mass continuity equation simplifies to a volume continuity equation for incompressible flow, as shown in Equation G.9.

$$\nabla \cdot \mathbf{u} = 0 \tag{G.9}$$

The residuals are presented in Section G.9 for the performed simulations. Each graph shows the residual continuity and velocity errors for both gondola orientations at a certain upstream velocity. The residual is an important measure for judging the convergence of the iterative solution, as it directly quantifies the error encountered while solving the system of equations. For instance, the continuity residual is the numerical error that remains after solving the mass continuity equation. The lower the residual value, the more numerically accurate the obtained solution is. Special attention should be paid to the judgment of the residuals for the performed simulations, since the solutions rarely converged before the maximum number

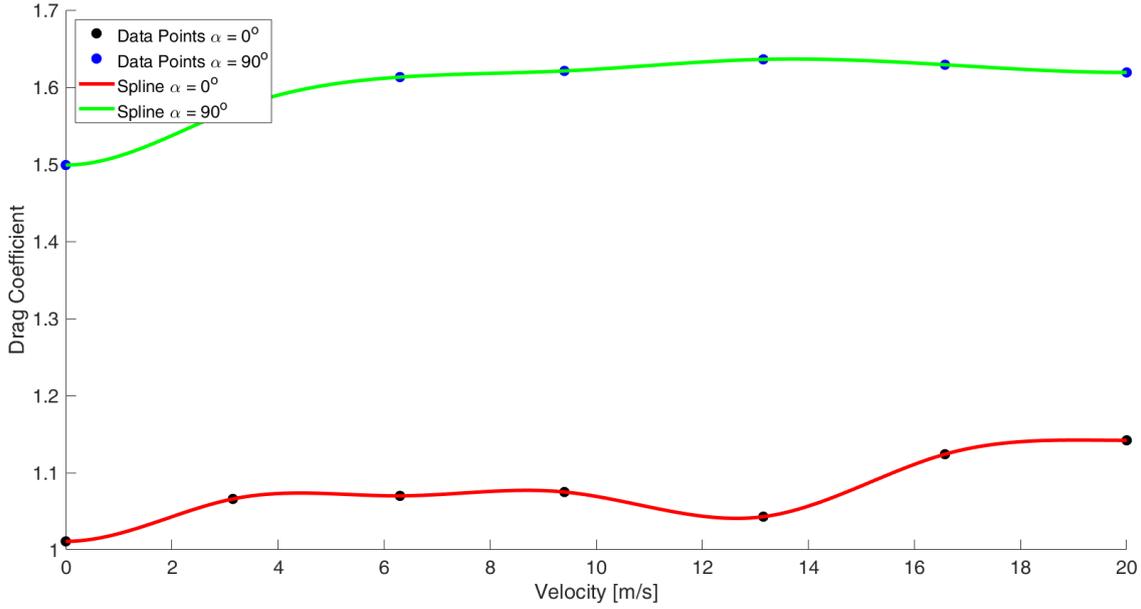
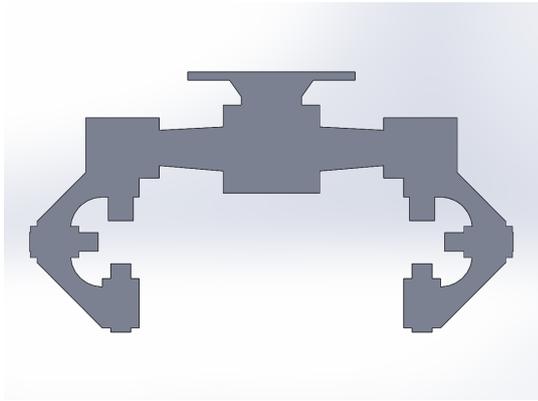


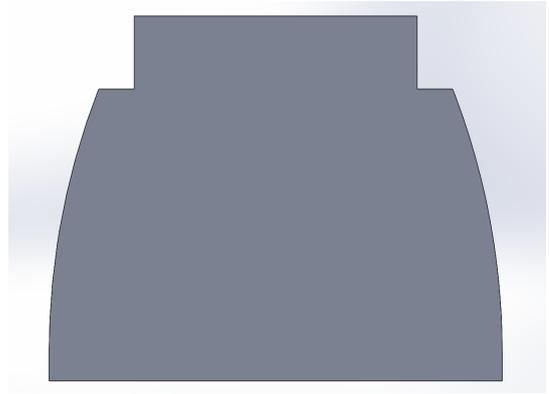
Figure G.5: The drag coefficient values and the cubic interpolation splines for both gondola orientations

of iterations was reached. Aside from the continuity residual, only the residual error for the velocity in  $x$ -direction is shown in the graphs in Section G.9, since the residuals were practically equal for the  $x$ -,  $y$ -, and  $z$ -direction. Furthermore, the air flow at the inlet has been specified in the direction perpendicular to the inlet surface, which corresponds to the  $x$ -direction. From the graphs it can be deduced that both the velocity and continuity residual are typically lower for  $\alpha = 90^\circ$  than for  $\alpha = 0^\circ$ . The size of the continuity residual error for the former gondola orientation can typically be found between  $10^{-5}$  and  $10^{-6}$ , which can be regarded as well converged. The continuity residual error for  $\alpha = 0^\circ$  is orders of magnitude larger, and often coincides with the velocity residuals for  $\alpha = 90^\circ$  between  $10^{-3}$  and  $10^{-4}$ . The velocity residual error for  $\alpha = 0^\circ$  is the largest of all plotted residuals, ranging from a minimum of approximately  $10^{-3}$  up to a relatively large numerical error of  $10^{-2}$ . These residuals are considered to be loosely converged. Especially for the gondola orientation corresponding to  $\alpha = 0^\circ$ , smaller residual values would be preferred. A finer mesh generally results in more numerically accurate results. However, this also comes at the cost of longer computational times. The choice for a certain mesh density is predominantly a consideration based on a trade-off between numerical accuracy and computational time. Given the considerable total computational time for the fourteen simulations, it is believed that a reasonable trade-off between numerical accuracy and computational time has been made. Hence, the results are presumed to be sufficiently accurate for use in the Simulink model.

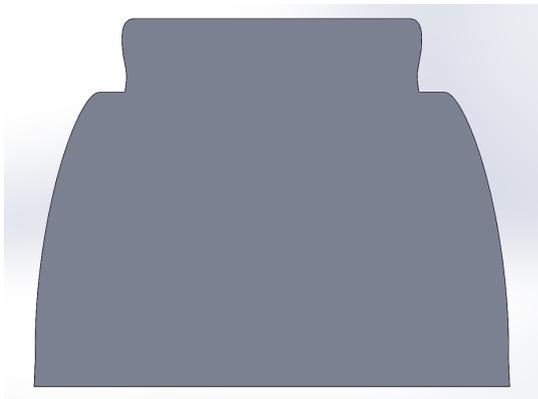
## G.8 Frontal Surfaces used in Drag Analysis



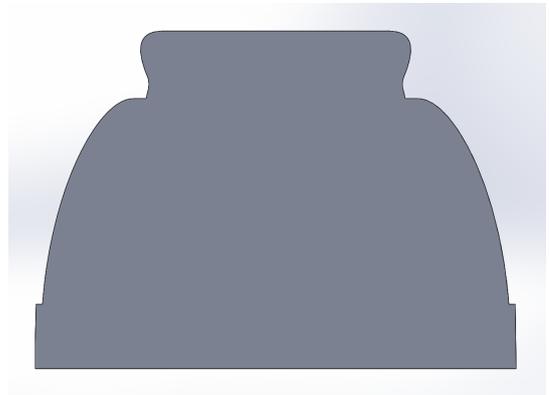
(a) Frontal area of chassis, pivot axle, and bogies



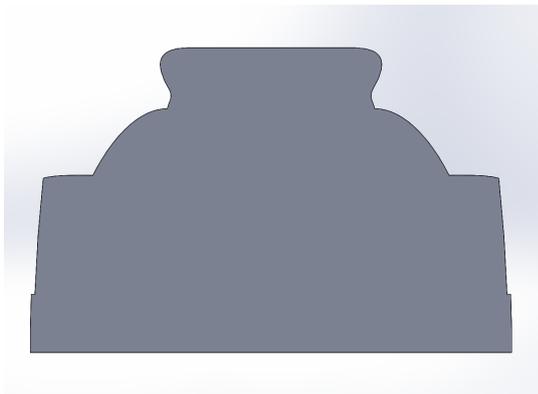
(b) Gondola frontal area at  $\alpha = 0^\circ$



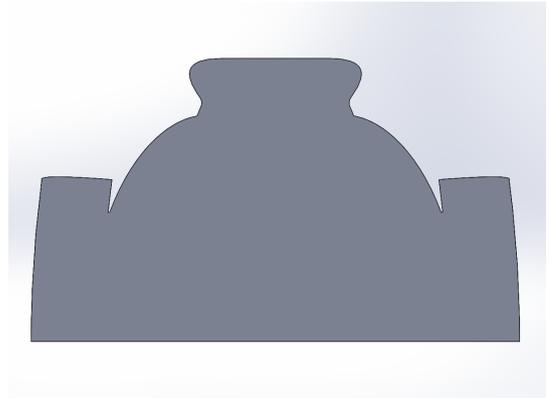
(c) Gondola frontal area at  $\alpha = 15^\circ$



(d) Gondola frontal area at  $\alpha = 30^\circ$

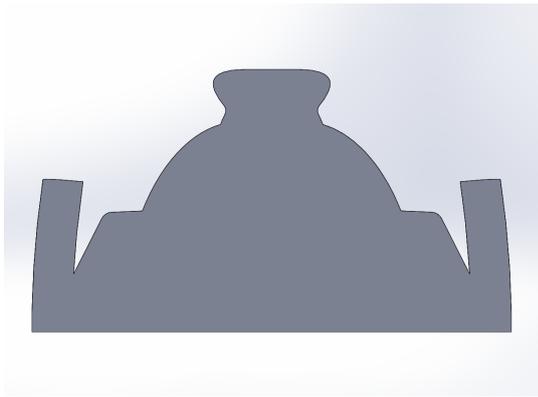


(e) Gondola frontal area at  $\alpha = 45^\circ$

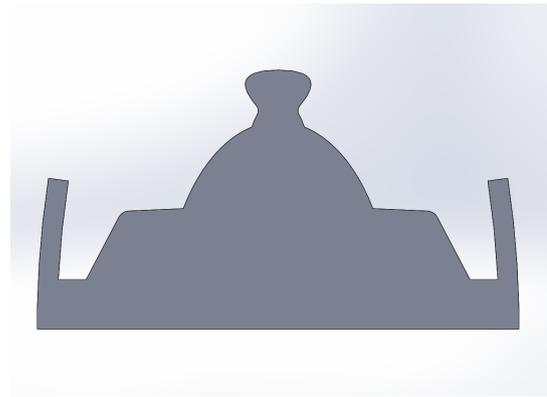


(f) Gondola frontal area at  $\alpha = 60^\circ$

Figure G.6: Frontal area for the assembly of chassis and bogies and for the gondola at five different orientations



(a) Gondola frontal area at  $\alpha = 75^\circ$



(b) Gondola frontal area at  $\alpha = 90^\circ$

Figure G.7: Frontal area of the gondola at the two remaining orientations

## G.9 Residuals Plots from Drag Analysis

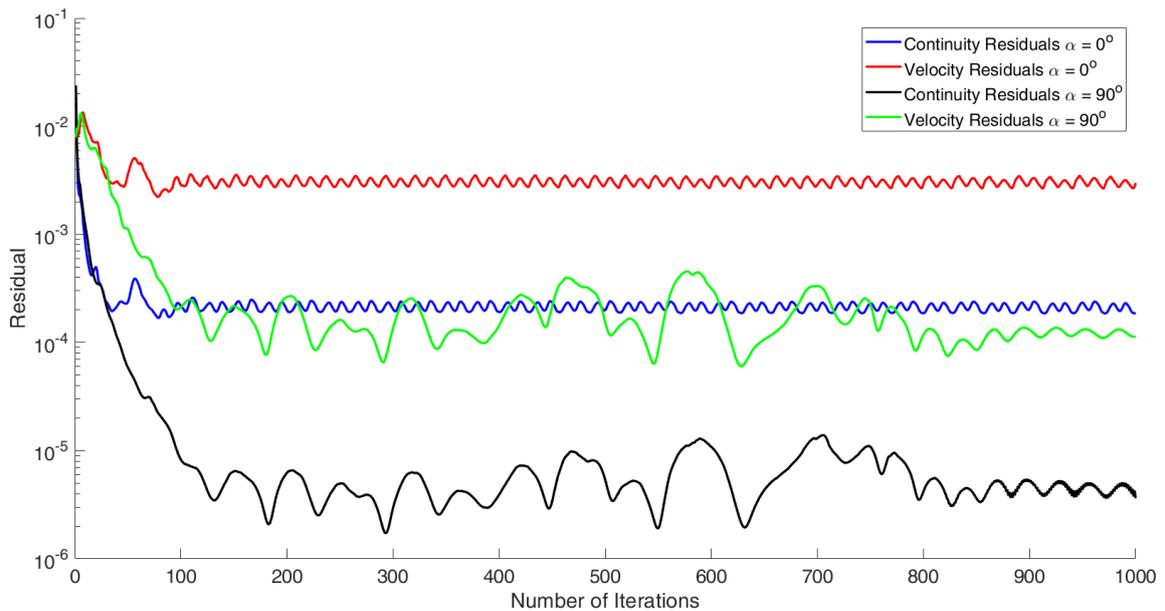


Figure G.8: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 1$  m/s

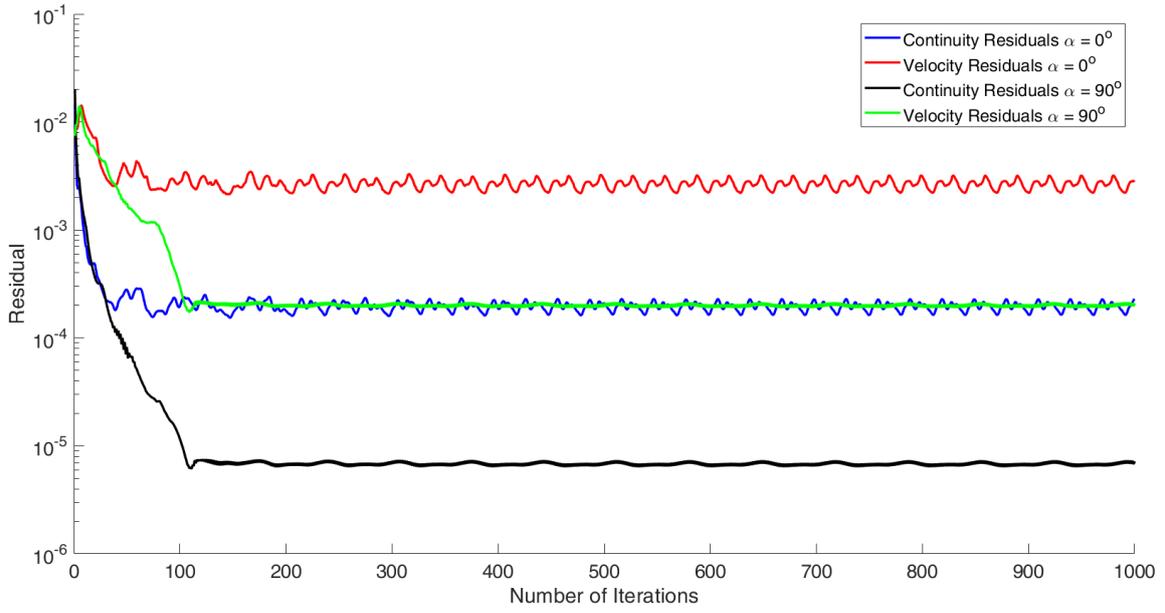


Figure G.9: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 3.15$  m/s

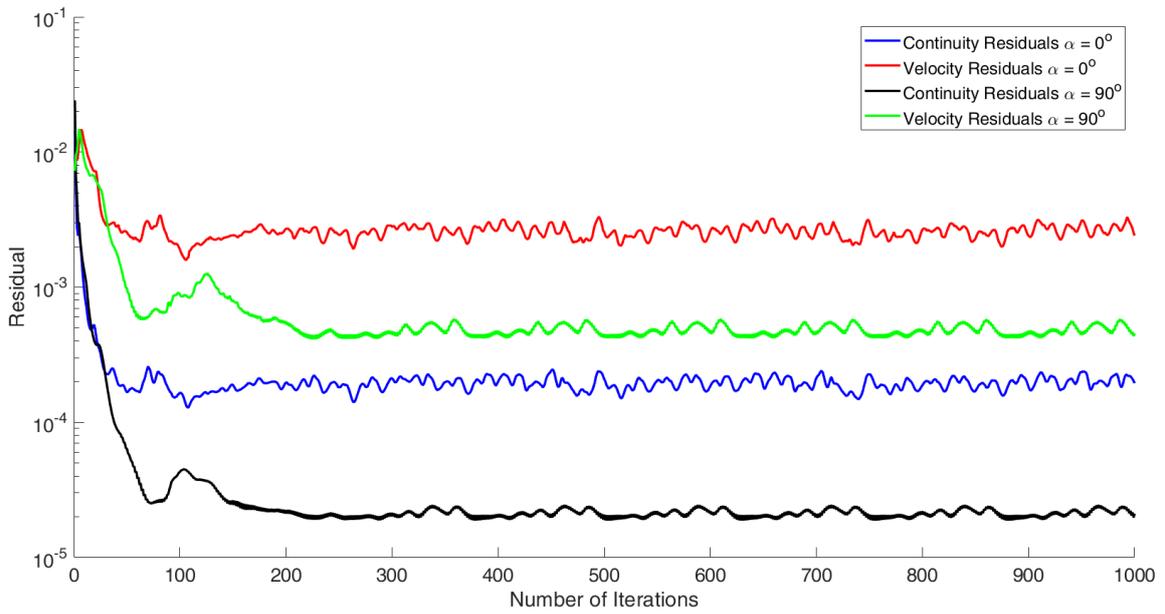


Figure G.10: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 6.3$  m/s

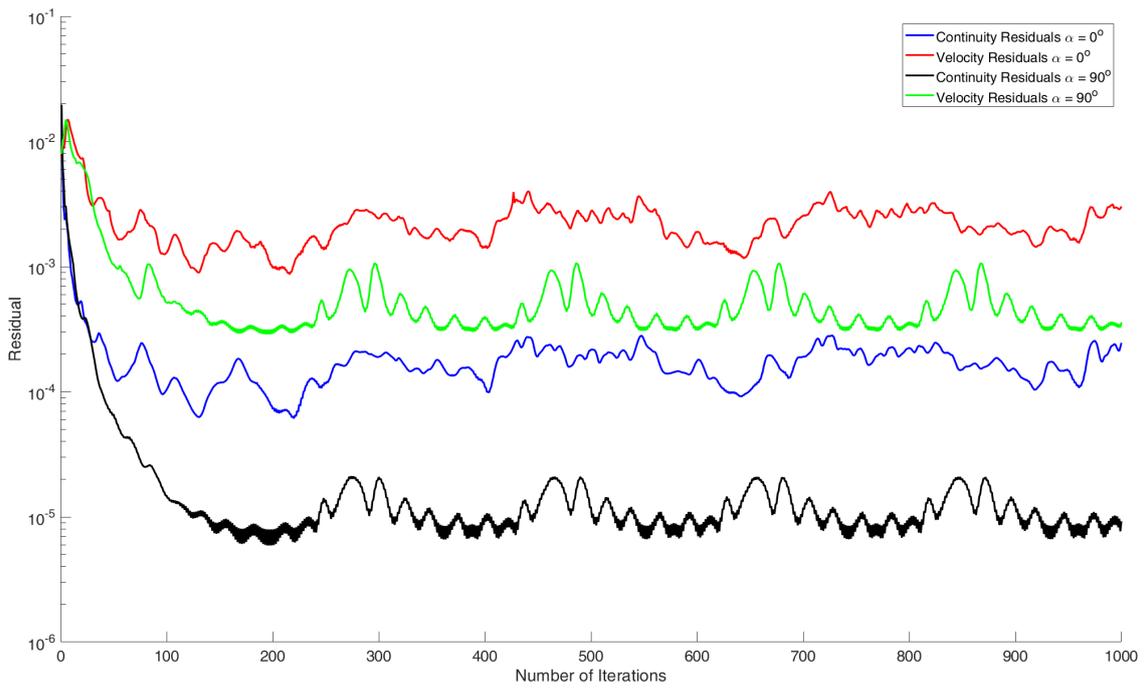


Figure G.11: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 9.4$  m/s

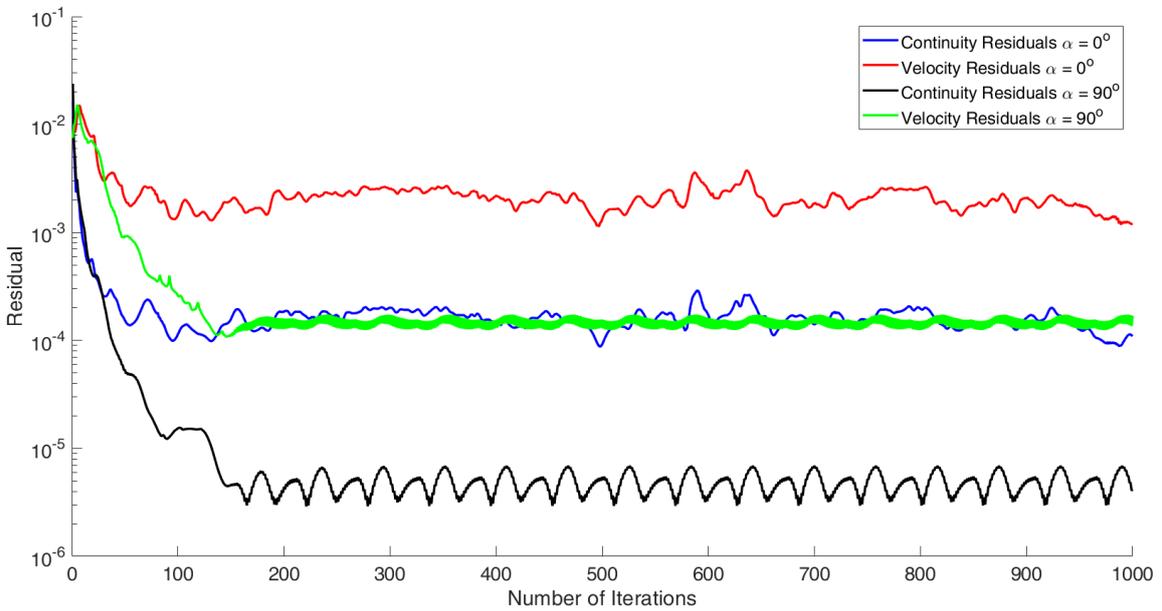


Figure G.12: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 13.15$  m/s

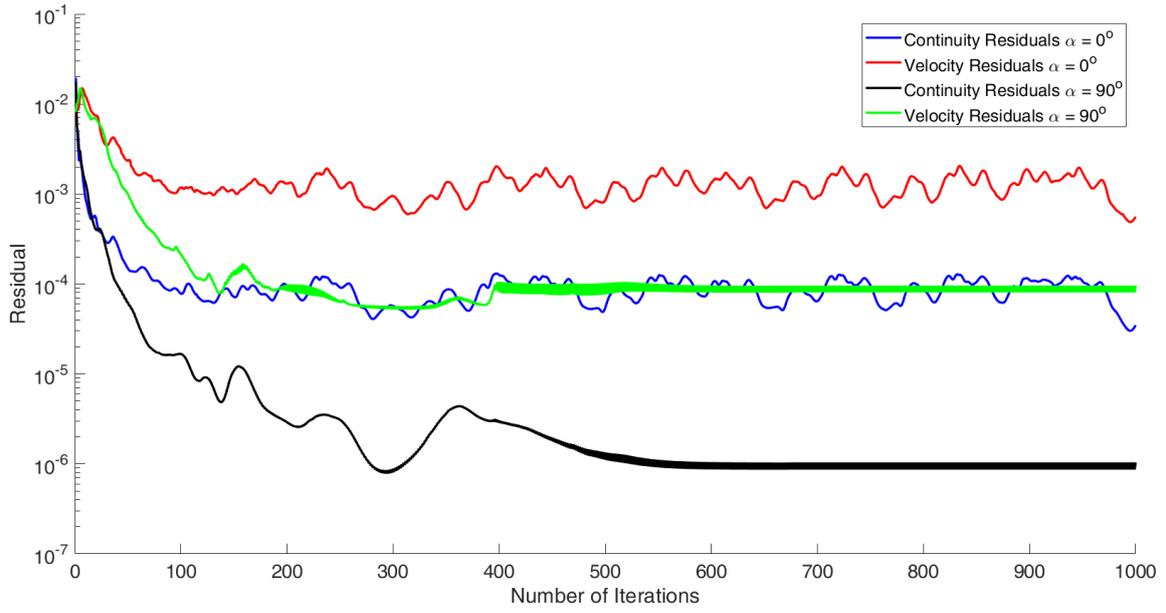


Figure G.13: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 16.575$  m/s

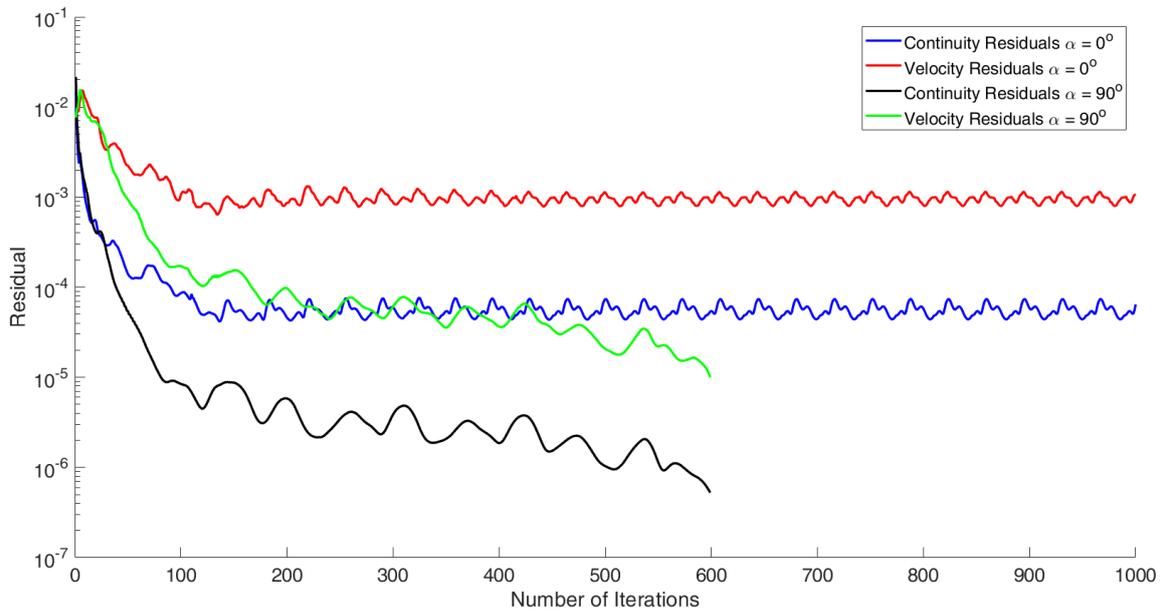


Figure G.14: Continuity and velocity residuals for  $\alpha = 0^\circ$  and  $\alpha = 90^\circ$  at  $v = 20$  m/s

## G.10 Air Velocity Contour Plots from Drag Analysis

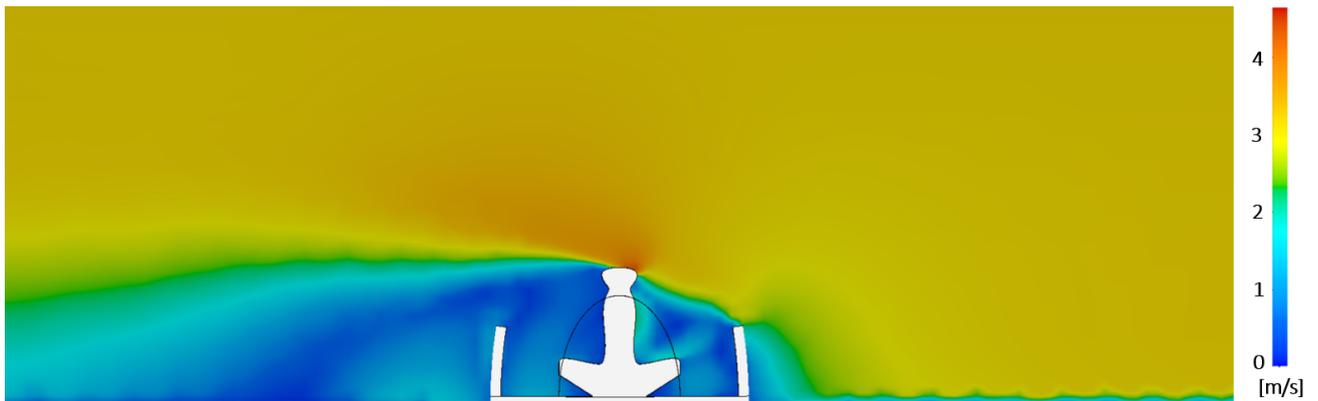


Figure G.15: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 3.15 m/s ( $Re = 5 \cdot 10^5$ )

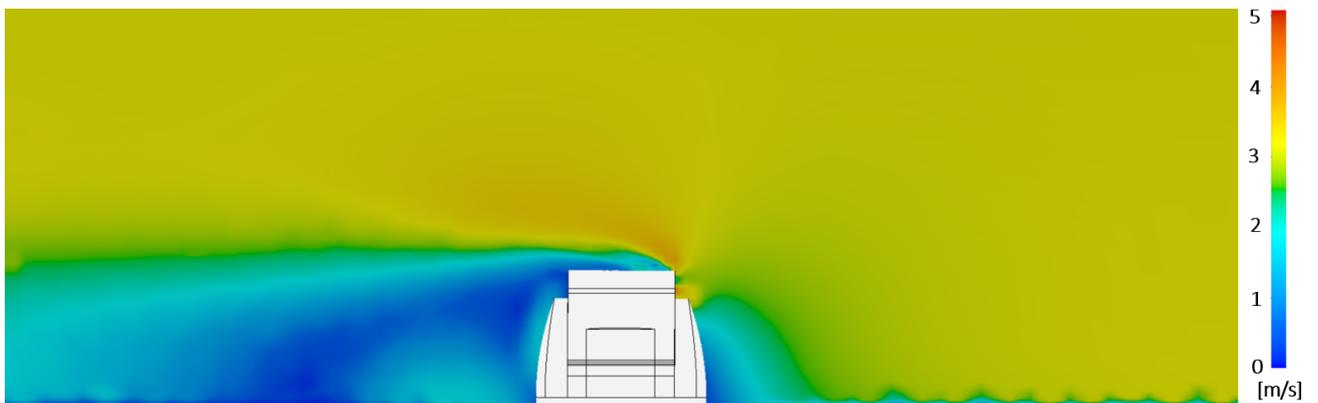


Figure G.16: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 3.15 m/s

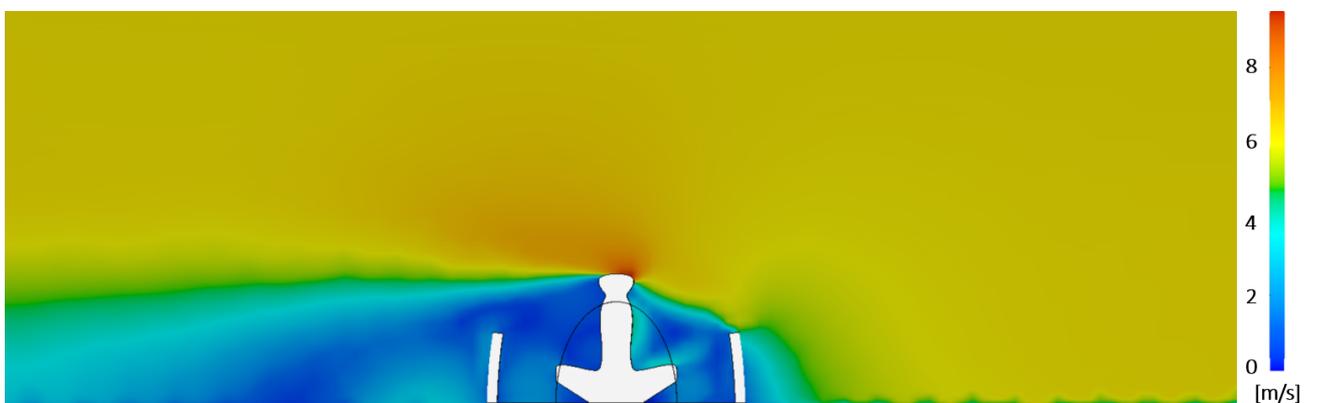


Figure G.17: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 6.3 m/s ( $Re = 10^6$ )

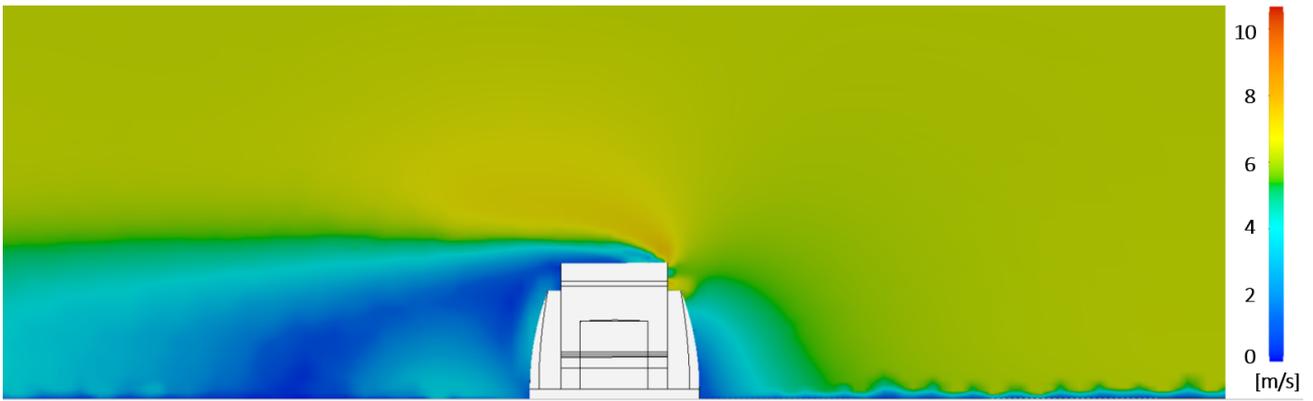


Figure G.18: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 6.3 m/s

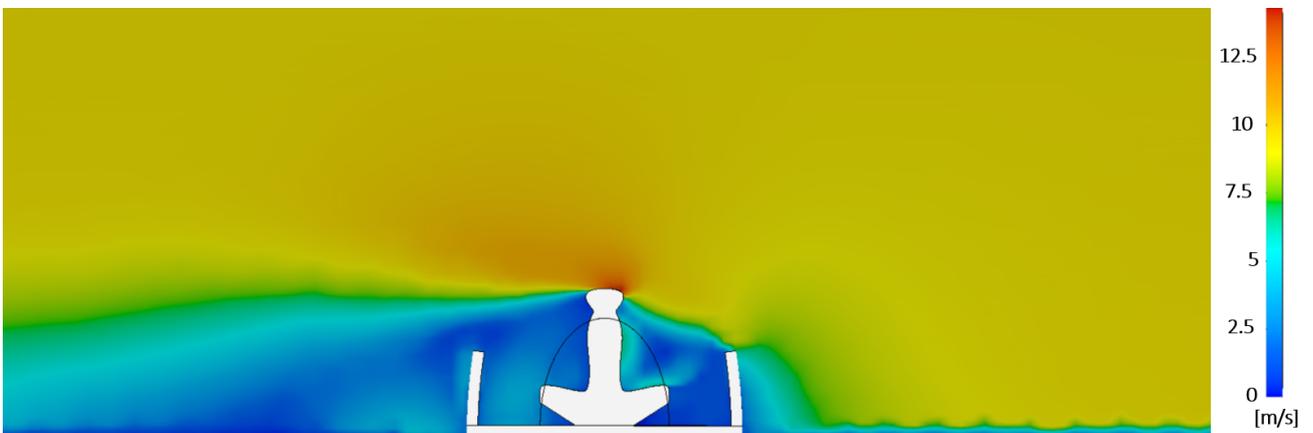


Figure G.19: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 9.4 m/s

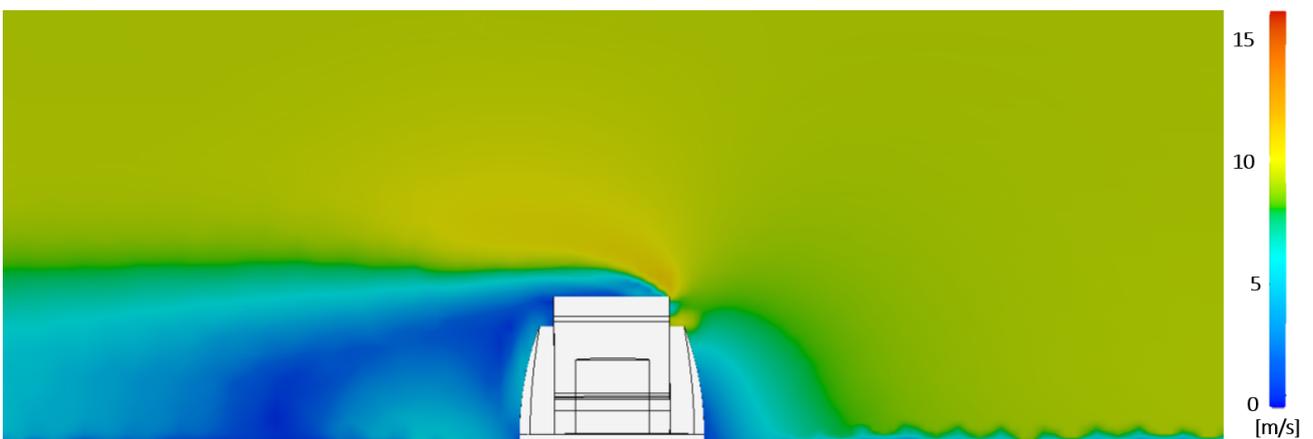


Figure G.20: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 9.4 m/s ( $Re = 10^6$ )

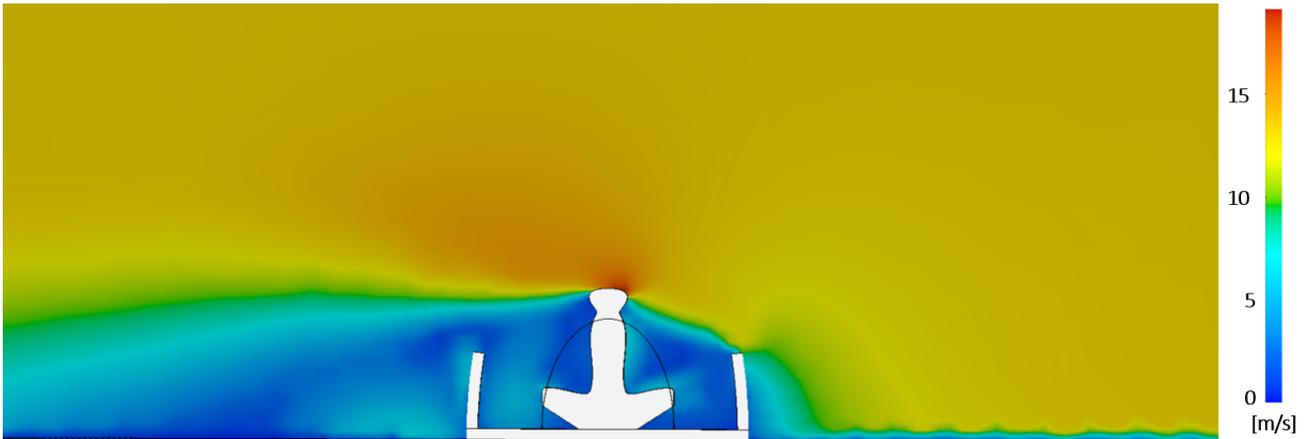


Figure G.21: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 13.15 m/s

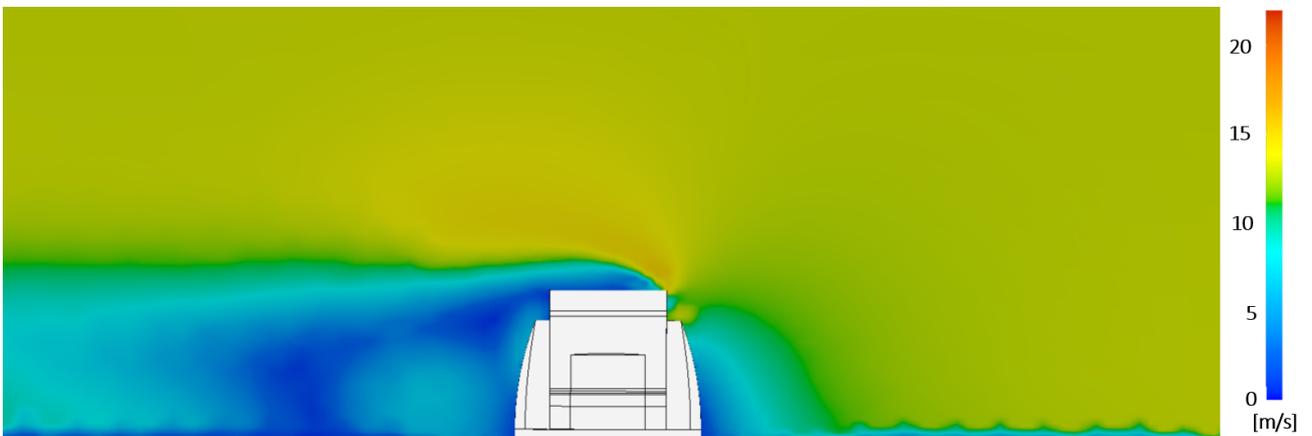


Figure G.22: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 13.15 m/s

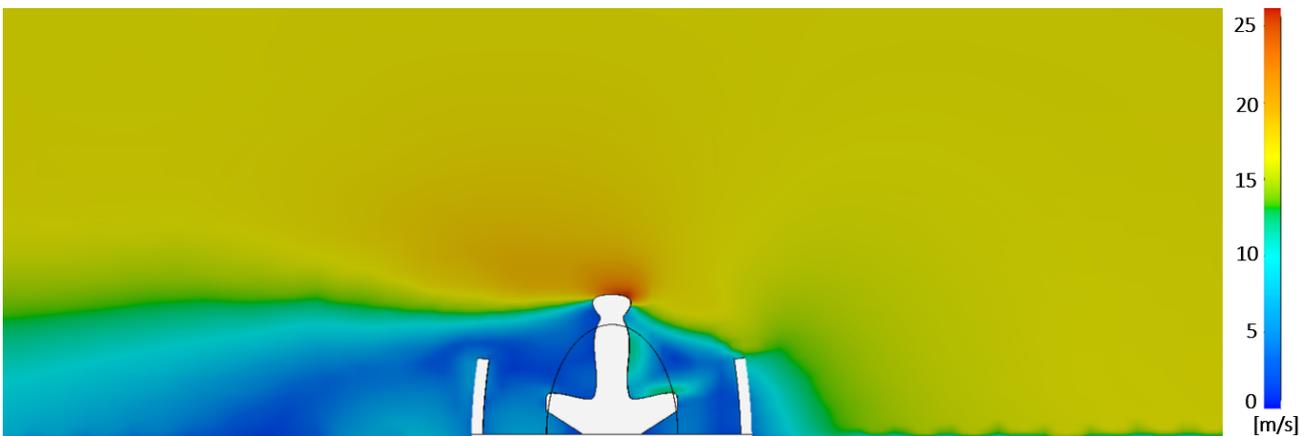


Figure G.23: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 16.575 m/s

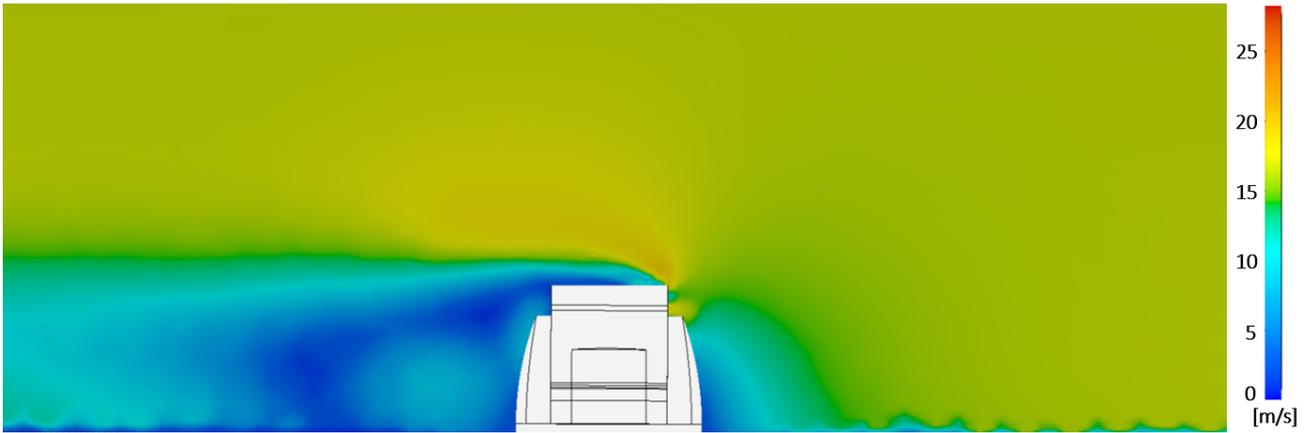


Figure G.24: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 16.575 m/s

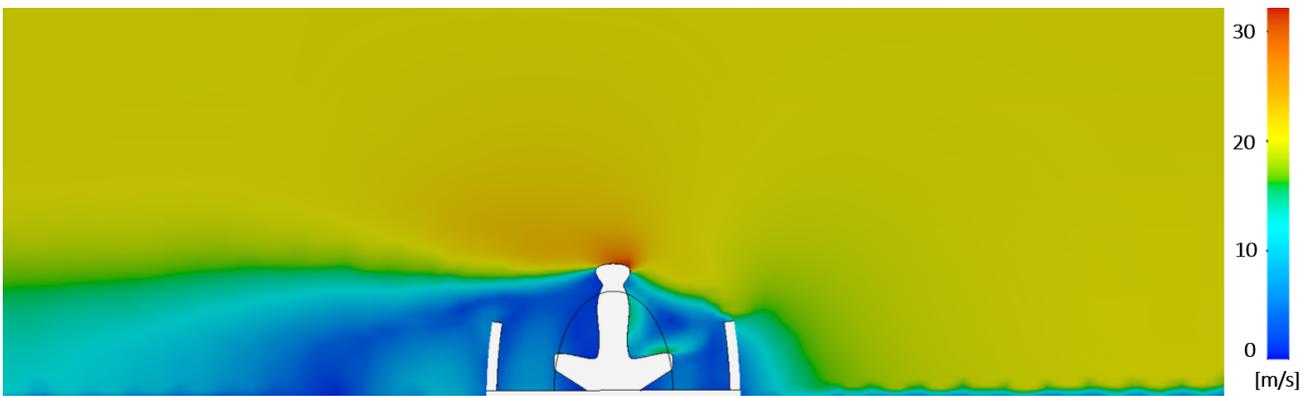


Figure G.25: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $0^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 20 m/s

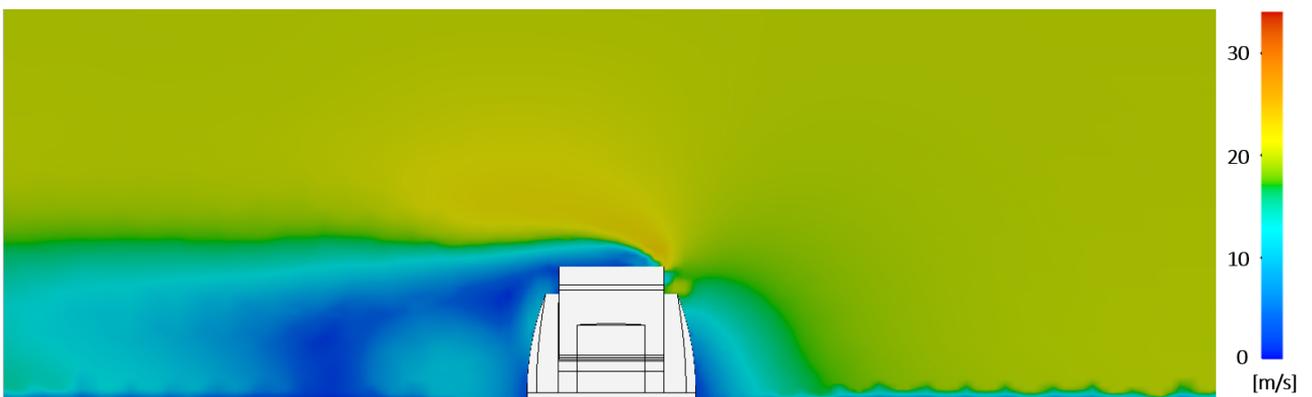


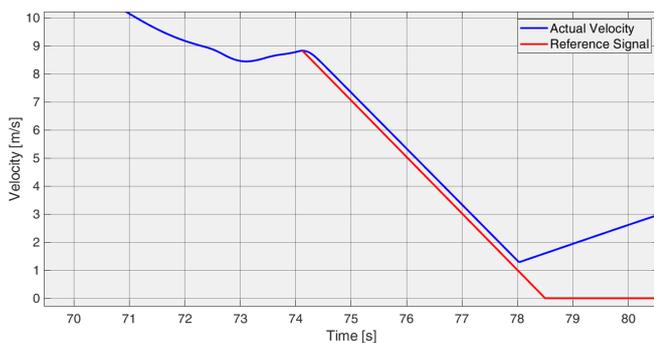
Figure G.26: Velocity contour of the airflow over the gondola for a relative angle  $\alpha$  equal to  $90^\circ$  between the direction of the airflow and the gondola orientation and an upstream velocity  $v$  of 20 m/s

# H. Motion Control

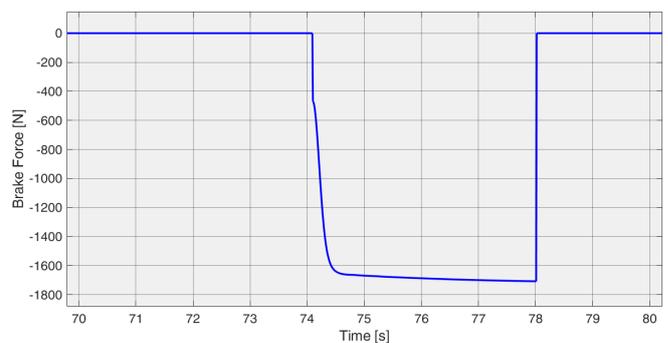
The motion of the vehicle is controlled at the brake section, subsequent turnaround, station, and on the lifthill. The first three sections in this Appendix discuss the schemes used for translational control of the vehicle at respectively the brake section, station, and lifthill. Rotational control of the gondola is achieved with the schemes discussed in the four subsequent sections. The reader is referred to Appendix 5 for screen captures of the motion control schemes.

## H.1 Translation Control Brake Section

The physical modelling connection port named **Brake Frame** (with port number 1) on the left-hand side in Figure A.21 refers to a measurement frame that is located at the bottom of the main chassis beam. The motion of this frame is measured with respect to the world frame by the transform sensor named **Brake Frame Sensor 1**. The relative velocity between these two frames is subsequently integrated in time by a discrete time integrator block, which yields the distance covered along the track by the brake frame. As can be deduced from Table 2.5, the coach enters the brake section when  $s = 684$  meter. The ramp block on the left-hand side of the scheme outputs a ramp signal, where the slope represents the prescribed deceleration of the vehicle on the brake section, which equals  $-2 \text{ m/s}^2$ . Two triggered subsystems are used to determine the velocity and the cumulative quantity of the inverted ramp when the vehicle reaches the start of the brake section. Summation of this result and subtraction of the actual ramp signal results in a reference signal for the deceleration on the brake section. The combination of **If**, **Merge**, and **Sum** blocks at the centre of the scheme dictate by means of the first **if** expression that the reference signal should coincide with the actual velocity when the vehicle has not reached the brake section yet. On the other hand, the **else** expression of the **If** block is valid when the vehicle is on the brake section and the reference signal has not reached the desired brake section exit velocity of  $0 \text{ m/s}$  yet. The corresponding reference signal is characterized by a ramp with a slope of  $-2$  and an initial output equal to the velocity at  $s = 684$  meter. At last, the **elseif** expression dictates that once the reference signal has reached zero, it should stay equal to zero. The actual and reference velocity are depicted in Figure H.1a.



(a) The actual and reference velocity signals



(b) The brake force per wheel

Figure H.1: The velocity profiles and brake force with regard to the vehicle when located on the brake section

The error between the actual velocity and the reference value is obtained by subtracting the velocity signal from the reference signal at each time-step. The error signal enters a **Saturation** block, which bounds the upper limit of the error to zero. In practice this implies that the error equals zero when the actual velocity is lower than the reference value. Hence, in case of a negative error, the wheels on the brake section would simply stop braking the vehicle instead of driving it. The error signal subsequently proceeds to a **PID** control block with a proportional gain equal to 30, after which the amplified error signal is used as input in a **Matlab Function** block that computes the brake force. Additionally, the gravity component in

the direction of travel of the vehicle is also needed as input in the `Matlab Function` block, since the brake section is slightly declined. The  $x$ -component of gravity is computed using the rotation matrix between the `Brake Frame` and the `World Frame`, which is obtained by `Brake Frame Sensor 2`. At last, the brake frame acceleration at the previous time-step is the only remaining input parameter required by the the function block. The acceleration signal is obtained by taking the numerical derivative of the velocity to time. The actual acceleration value is only used as long as the velocity of the vehicle is positive. In any other case, the acceleration is set equal to zero to prevent a reversal of the vehicle direction of motion. The brake force is computed based on the energetic balance presented in Equation H.1.

$$ma\Delta s = \frac{1}{2}m(\Delta v)^2 + mg_x\Delta s - F_{\text{brake}}\Delta s \quad (\text{H.1})$$

where  $m$  represents the mass of a fully-loaded gondola ( $m = 2550.9$  kg),  $a$  symbolizes the net acceleration of the vehicle (in  $m/s^2$ ), the length of brake section (23 meters) is represented by  $\Delta s$ ,  $\Delta v$  symbolizes the error between the reference and actual velocity, and  $g_x$  represents the gravity component in longitudinal direction. Solving this expression for  $F_{\text{brake}}$  yields the total brake force. The presumption is made that four equidistantly spaced braking wheels are simultaneously in contact with the bottom of the main chassis beam. Hence, to obtain the brake force per wheel, the total brake force is divided by four. The brake force required for braking a fully-loaded gondola equals approximately 1700 N per wheel, as shown in Figure H.1b. Please note that the negative brake force in this graph stems from application of the brake force in opposite direction to the local  $x$ -axis, which has been aligned with the direction of travel. The `If` block on the right-hand side of the scheme states that the wheels keep braking the vehicle as long as the vehicle is on the brake section, the velocity of the vehicle is larger than 1 m/s, and the gondola is not yet at its equilibrium position. Please note that the latter condition is prescribed by the `inport` named `Release`, which will later be elaborated on. To avoid the existence of an algebraic loop in the Simulink scheme, the `Memory` block applies a one integration step time delay to the brake force signal.

## H.2 Translation Control Station

The scheme depicted in A.22 controls the translational motion of the vehicle at the station exit and entry. The top half of this scheme is dedicated to the computation of the driving force at the station exit, while the lower half is used to determine the brake force at the station entry.

### H.2.1 Station Exit

The desired acceleration at the station entry is set equal to  $2$  m/s<sup>2</sup>. When the vehicle leaves the station and enters the lifthill at  $s = 5$  meters, the velocity of the vehicle should equal the velocity of the chain on the lifthill (4 m/s). The `if` expression of the top-left `If` block states that as long as the velocity of the vehicle is not equal to the chain speed, the reference signal should be a ramp with a positive slope of 2 and an initial output value equal to zero. Hence, the reference signal dictates that the vehicle should accelerate from standstill with the desired acceleration of  $2$  m/s<sup>2</sup>. Once the magnitude of the ramp signal has reached the chain speed value, the `elseif` expression prescribes that the reference signal should be a constant line with a magnitude equal to the chain speed, so the velocity of the vehicle is maintained at 4 m/s until it reaches the lifthill. Once the vehicle has left the station, the `else` expression of the `If` block dictates that the reference signal should coincide with the actual velocity of the vehicle. The previously-described reference signal is shown in Figure H.2a.

The error between the actual velocity and the reference value is obtained by subtracting the velocity signal from the reference signal. The subsequent `Saturation` block bounds the lower limit of the error to zero, so the error equals zero when the actual velocity is larger than the reference value. In case of an error equal to zero, the driving wheels would therefore simply stop accelerating the vehicle instead of braking it. The error signal subsequently proceeds to a PID control block with a proportional gain of 10 and an integral gain equal to 1. The amplified error signal is one of the two input parameters required by the `Matlab Function` block

to compute the driving force at the station exit. The other required input parameter is the longitudinal component of gravity acting on the vehicle, whose contribution is minor thanks to the flat track in the station. The driving force at the station exit can be computed according to Equation H.2.

$$F_{\text{driving}} = \frac{m(\Delta v)^2}{2\Delta s} - mg_x \quad (\text{H.2})$$

where  $\Delta s$  equals 5 meters. The top-right **If** block states that the driving force should be applied to the vehicle as long as the vehicle is at the station exit, whereas the driving force should equal zero when the vehicle has left the station.

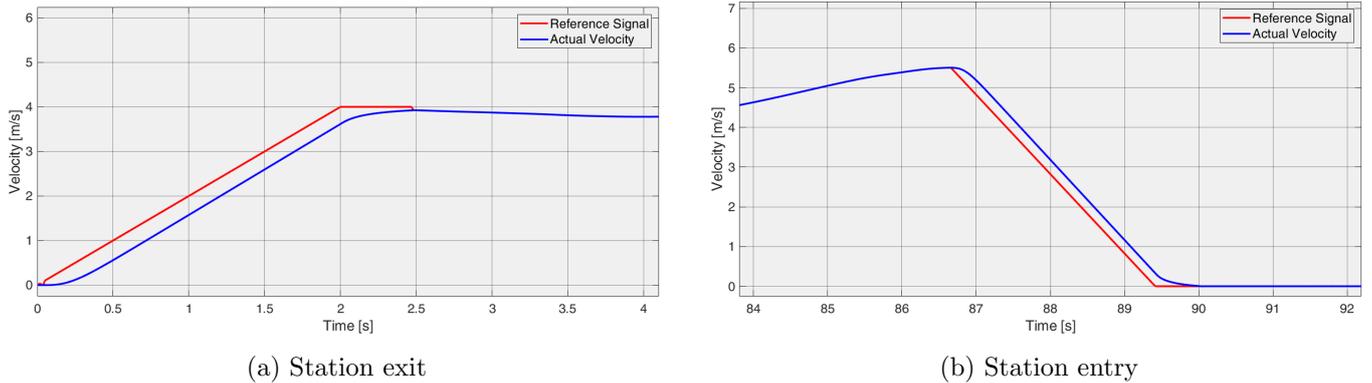


Figure H.2: The actual and reference velocity signals for the vehicle when located at the station

## H.2.2 Station Entry

At the station entry the desired deceleration is set equal to  $-2 \text{ m/s}^2$ . The procedure for acquisition of the reference signal is similar to the approach explained in Section H.1 for translational control on the brake section. Hence, the scheme for translation control at the station entry is not elaborated on in detail in this paragraph. The **else** expression in the **If** block dictates that the reference signal should coincide with the actual velocity of the vehicle as long as the vehicle has not reached the station entry yet. When the vehicle is at the station entry but the vehicle has not come to a standstill yet, the reference signal prescribed by the **if** expression is characterized by a ramp with a slope of  $-2$  and an initial output equal to the vehicle velocity at  $s = 737$  meter. The **else** expression in the **If** block dictates a reference signal equal to zero once the vehicle has come to a standstill.

Subtracting the velocity signal from the reference signal yields the error value, which proceeds to a **Saturation** block with an upper limit equal to zero. Hence, the braking wheels simply stop braking the vehicle when its velocity is lower than the reference value instead of driving it to approach the reference signal again. The error signal subsequently proceeds to a **PID** control block with a proportional and integral gain equal to respectively 15 and 1. The expression used for calculating the braking force at station entry is to a large extent similar to the expression in Equation H.2, as shown in Equation H.3.

$$F_{\text{brake}} = \frac{m(\Delta v)^2}{2\Delta s} + mg_x \quad (\text{H.3})$$

A minus sign is added to the calculated braking force, since this force acts in opposite direction with respect to the direction of travel of the vehicle. Among the blocks at the bottom of the scheme is a **Triggered Subsystem** block that outputs a one when the vehicle has come to a standstill, while the output equals zero when the vehicle is still at motion. The bottom-right **If** block uses this signal to prescribe a brake force that remains equal to zero once the vehicle has come to a standstill in the station. The **if** expression in this **If** block namely passes the calculated brake force to the subsequent **Merge** block as long as the triggered subsystem outputs a zero. The system is only triggered when the velocity of the vehicle at the station entry is equal to zero, and the unity output subsequently ensures that a zero brake force is

passed on to the `Merge` block. Since the `Triggered Subsystem` cannot be triggered again, the brake force remains equal to zero for the remainder of the simulation. The drive force at station exit and the brake force at station entry are shown in Figure H.3. The required driving force and braking force at respectively the station exit and station entry are practically similar, which could be expected based on the equal magnitudes of respectively the acceleration and deceleration.

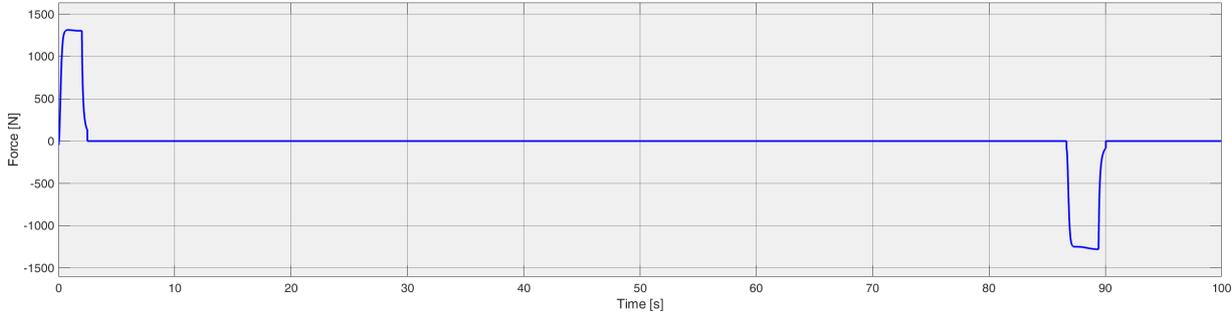
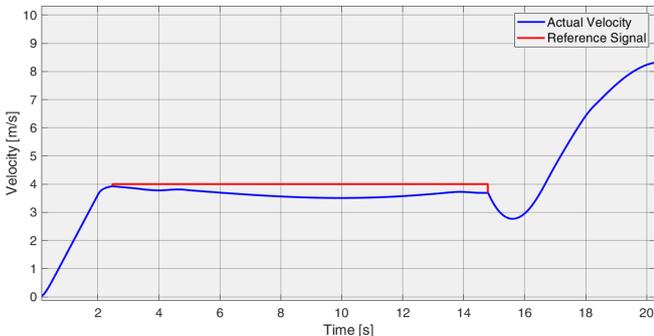


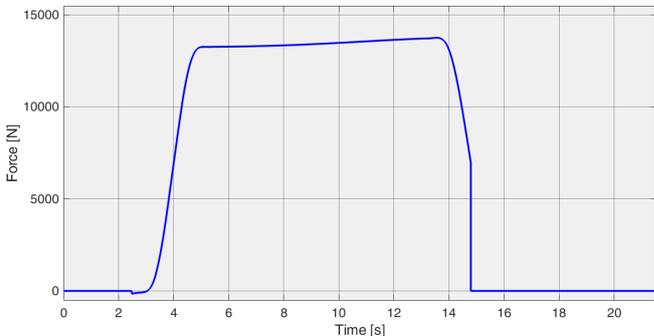
Figure H.3: The brake force per wheel required for braking the vehicle at the station entry

### H.3 Translation Control Lifthill

The chaindog is a metal component that is engaged by the chain so the vehicle is pulled up the lifthill. It has been mounted on the underside of the main chassis beam in the proximity of the front axle, and hence a local frame named `Chaindog Frame` has been defined at this position. The relative velocity between the chaindog and `World Frame` is measured by the transform sensor named `Chaindog Frame Sensor 1`. The longitudinal velocity of the chaindog frame with respect to the world frame is subsequently integrated in time by a discrete time integrator block, which results in the distance covered along the track by the chaindog frame. The `If` block at the top-left of the scheme in Figure A.23 states that the reference signal should equal the desired chain speed of 4 m/s when the vehicle is on the lifthill ( $5 \leq s \leq 50$ ). On the other hand, the reference signal should equal the actual velocity of the vehicle when the vehicle is not located on the lifthill. The actual and reference velocity signals are depicted in Figure H.4a, which shows that the vehicle moves up the lifthill at a relatively constant pace. Subtracting the vehicle velocity from the reference signal yields the error between the actual and reference velocity. The error signal is subsequently amplified by respectively a proportional and an integral gain both equal to unity. Aside from the error value, the longitudinal component of the gravitational acceleration is also required as an input parameter by the `Matlab Function` on the right-hand side of the scheme. The  $x$ -component of gravity is computed in a similar manner as described before, using the rotation matrix between the chaindog frame and the world frame.



(a) The actual and reference velocity signals



(b) The lift force exerted on the chaindog

Figure H.4: The velocity profiles and brake force force with regard to the vehicle when pulled up the lifthill

The acceleration of the vehicle at the previous time-step is the last input parameter required to calculate the force that is needed to pull the vehicle up the lifthill. The calculation of the vehicle acceleration is performed at the lower half of the scheme. The acceleration value is used together with the other input parameters to calculate the lift force (in N) according to Equation H.4.

$$F_{\text{lift}} = \frac{m(\Delta v)^2}{2\Delta s} - ma - mg_x \quad (\text{H.4})$$

where  $\Delta s$  represents the length of the lifthill (45 m). The **If** expression of the **If** block at the right-hand side of the scheme prescribes a lift force equal to the computed value when the vehicle is on the lifthill. On the other hand, when the vehicle is not on the lifthill, the **else** expression dictates a lift force equal to zero. The computed lift force is exerted at the chaindog frame with a time delay of one integration step to avoid the occurrence of an algebraic loop. Figure H.4b shows the lift force as a function of the simulation time.

## H.4 Rotation Control Brake Section

The distance covered along the track by the brake frame is compared to the constant value of 684 meters, which corresponds to the start of the brake section. The vehicle is located on the brake section when  $684 \leq s \leq 707$ , and two **Triggered Subsystems** in the top-left corner of Figure A.24 determine respectively the gondola rotation angle and angular velocity at the moment when the vehicle enters the brake section at  $s = 684$  meters. These two values are used by the first **Matlab Function** to compute the gradient angle of the ramped reference signal. A copy of the function that is used for calculating the slope of the reference signal is presented below.

```

1 function Slope = AngleFunc(Omega, Theta)
2
3 if Theta > 0
4     if Omega >= 0
5         MinRot = pi - mod(Theta, pi);
6     else
7         MinRot = mod(Theta, pi);
8     end
9 else
10    if Omega <= 0
11        MinRot = -pi + mod(abs(Theta), pi);
12    else
13        MinRot = mod(abs(Theta), pi);
14    end
15 end
16
17 dt = 1;
18
19 if MinRot > 0
20     Angle = MinRot + floor(mod((dt*abs(Omega))-abs(MinRot), pi)/pi)*pi;
21 else
22     Angle = MinRot - floor(mod((dt*abs(Omega))-abs(MinRot), pi)/pi)*pi;
23 end
24
25 Slope = sign(Omega)*(abs(Omega)^2)/(2*abs(Angle));

```

The parameter **MinRot** in the above script denotes the smallest possible angle between the equilibrium orientation of the gondola and the gondola rotation corresponding to  $s = 684$  meters. The calculation of **MinRot** depends on the direction in which gondola rotates when entering the brake section, so the residual angle with respect to the equilibrium orientation is in accordance with the direction of rotation. Additionally, the angular velocity of the gondola on the brake section is taken into account for the first second to prevent an inordinate angular deceleration of the gondola. Hence, the angular deceleration of the gondola to the initial

orientation should last at least one second. Please note that the slope of the reference signal should be similar to the desired angular deceleration of the gondola. After multiplication of the ramp signal with the calculated gradient angle, the combination of subsequent `Triggered Subsystem` and `Sum` block is used to define a ramped reference signal. This reference signal is characterized by a slope equal to the calculated angular deceleration and an initial value that is similar to the gondola angular velocity at the start of the brake section.

The `If` block assigned with determining a so-called `Release` indicator can be found on the left-hand side of the scheme. This block uses three input parameters, and the value of the first parameter depends on the angular velocity of the gondola. If the gondola angular velocity is practically equal to zero while the vehicle is on the brake section, the value of the first parameter would cause the `If` block to assign a unity value to the release indicator. A release indicator equal to unity denotes that the translational and rotational control of the vehicle on the brake section can be terminated. However, the value assigned to the release indicator also depends on two other parameters. The second parameter is similar to the value for `Angle` in the script, while the rotation of the gondola after the vehicle has entered the brake section is kept record of by the third input parameter. A comparison between these two parameters also determines if the vehicle can be released from the brake section without any further motion control. The release indicator is assigned a value equal to unity if the accumulated gondola rotation on the brake section equals the absolute value of the parameter `Angle`.

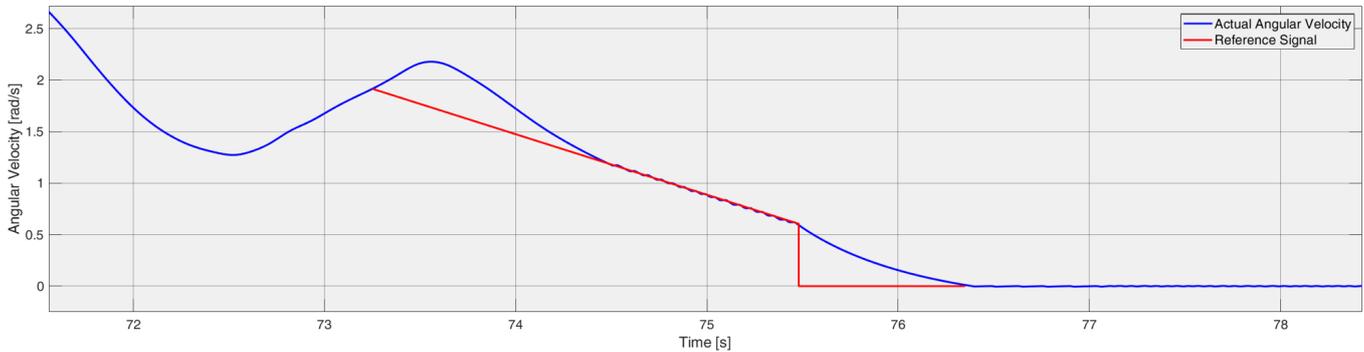


Figure H.5: The actual and reference angular velocity signals for the rotating part when the vehicle is on the brake section

The release indicator is used as input for the subsequent `If` block that prescribes the reference signal. If the vehicle is on the brake section and the motion should still be controlled, the reference signal should coincide with the ramped reference signal described earlier. On the other hand, when the vehicle has passed the brake section entry and the release indicator equals unity, the `elseif` expression states that the reference signal for the gondola angular velocity should equal zero. At last, the `else` expression dictates that the reference signal should coincide with the actual angular velocity when the vehicle has not yet reached the brake section. The error is subsequently obtained by subtracting the actual angular velocity from the reference signal. Two different PID controller blocks are used for controlling the motion of the gondola while the vehicle is on the brake section. The actual error is transferred to the controller on top as long as the rotational control of the gondola cannot be terminated yet. An error equal to zero is transferred to the controller block once the release indicator equals unity. The proportional and integral gain of this controller equal respectively 25 and 50. On the other hand, the lower PID controller block amplifies the error with a proportional and integral gain both equal to 25 once the vehicle can be released from the brake section. Prior to the termination of motion control on the brake section, the error transferred to the controller equals zero. The `Matlab Function` on the right-hand side of the scheme is allocated with the calculation of the braking torque. While the amplified error is the first input parameter required for determining the braking torque, the other one is the gondola orientation angle  $\theta$  when the vehicle enters the brake section. An average moment of inertia  $I$  equal to  $240 \text{ kg m}^2$  has been presumed for the four passenger occupancy configurations.

The braking torque  $T$  (in Nm) can be computed according to Equation H.5.

$$T_{\text{brake}} = \frac{I \cdot \Delta\omega^2}{2\theta} \quad (\text{H.5})$$

where  $\Delta\omega$  represents the error between the reference and actual angular velocity. The blocks at the right-most part of the scheme ensure an application of the braking torque in opposite direction with respect to the gondola angular velocity. Figure H.5 shows the actual and reference angular velocity signals for a passenger occupancy configuration with two passengers seated on one side of the gondola, which corresponds to the configuration with the largest mass imbalance of the rotating part.

## H.5 Rotation Control Station Entry

The rotation of the gondola at the station entry is controlled by the scheme depicted in Figure A.25. Only a general explanation of this scheme is presented in this section, since a clear explanation is hardly possible due to the high level of complexity. A constant angular deceleration of  $\frac{\pi}{12}$  rad/s<sup>2</sup> has been specified in case the gondola angular velocity does not approach zero. Multiplication of this angular deceleration by the moment of inertia yields a torque that is exerted on the pivot axle in opposite direction with respect to the angular velocity. On the other hand, when the angular velocity of the gondola is practically zero, a constant torque of 30 Nm is used to compensate the overshoot of the previous torque. The gondola is kept at the equilibrium position once this desired final orientation has been reached. Hence, the orientation of gondola at the end of the simulation is similar to its initial orientation for all four separate passenger occupancy configurations.

## H.6 Rotation Control Lifthill

The **If** block on the left-hand side of the scheme in Figure A.26 defines a reference angular velocity equal to zero for  $0 \leq s \leq 50$ . Hence, this reference signal applies to the gondola orientation when the vehicle is located at either the station exit or the lifthill. On the other hand, the reference signal coincides with the actual angular velocity once the vehicle has left the lifthill. The error between the reference and actual angular velocity is amplified with a proportional and integral gain of respectively 0.5 and 10. The error is subsequently used as input in a **Matlab Function** that outputs the torque required to maintain a zero gondola rotation angle. The required torque can be computed according to an expression similar to the one presented in Equation H.5. The subsequent **If** block specifies that the calculated torque should be applied on the pivot axle when the vehicle is at the station exit or lifthill and when the gondola rotation angle is unequal to zero. In any other case, the torque should be equal to zero. The torque should be applied in opposite direction with respect to the gondola rotation angle and with a time delay of a single integration step.

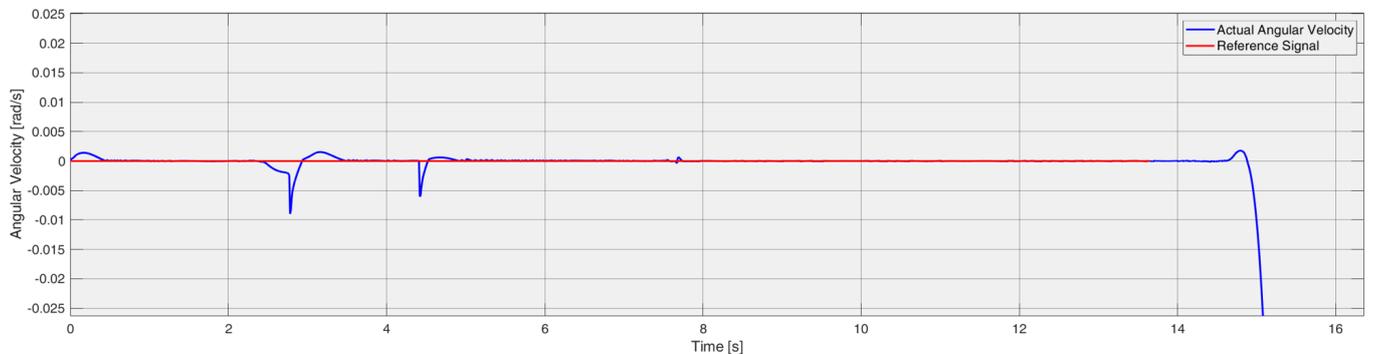


Figure H.6: The actual and reference angular velocity signals for the rotating part when the vehicle is located on the lifthill

From Figure H.6 it can be deduced that the deviation of the actual angular velocity from the reference signal is only minor. Hence, the gondola rotation angle is practically equal to zero when the vehicle is at the station exit and on the lifthill for all passenger occupancy configurations.

## H.7 Rotation Control Turnaround

The control scheme depicted in Figure A.27 is practically similar to the scheme used for controlling the motion at the station exit and on the lifthill (Figure A.26). Hence, the description presented in the above section also applies to the scheme used for controlling the gondola rotation at the turnaround between the brake section and the station. However, there are two notable differences between both schemes that are worth mentioning. First, the PID controller block in Figure A.27 comprises a proportional and integral gain equal to respectively 1 and 5. The second difference can be found in the conditions formulated in the subsequent `If` block, where the computed torque may only proceed if the parameter `Angle` is unequal to zero. This parameter is in essence similar to the gondola rotation angle though. When the gondola is in alignment with the equilibrium position, no torque is required to correct the position of the gondola. Figure H.7 shows the actual and reference angular velocity signals for a vehicle occupied by two passengers on a single side of the gondola. The constant need for a correction torque can especially be deduced from the right-half plane of this graph, which is partly caused by the mass imbalance on the rotating part that continuously initiates a rotary motion of the gondola.

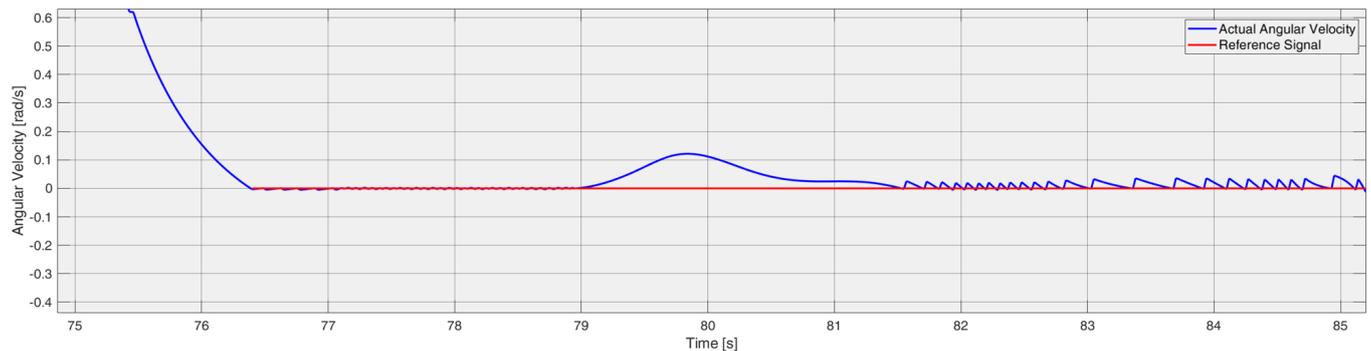


Figure H.7: The actual and reference angular velocity signals for the rotating part when the vehicle traverses through the turnaround between the brake section and station

# I. Acceleration Effects on Passengers

## I.1 Time Duration Limits for Accelerations

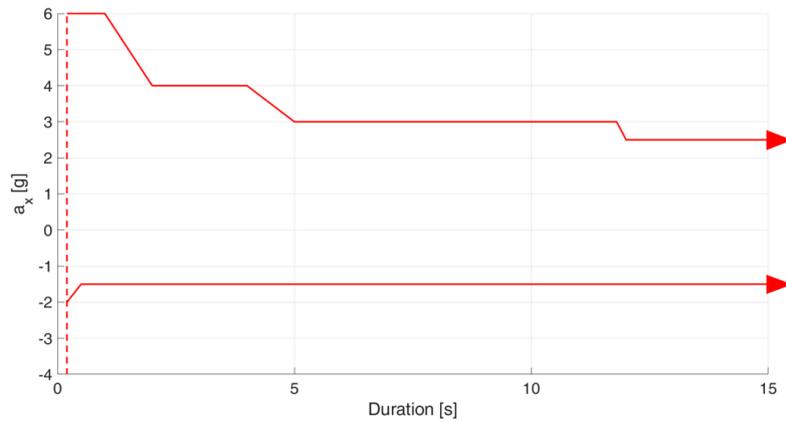


Figure I.1: Time duration limits for accelerations in  $x$ -direction

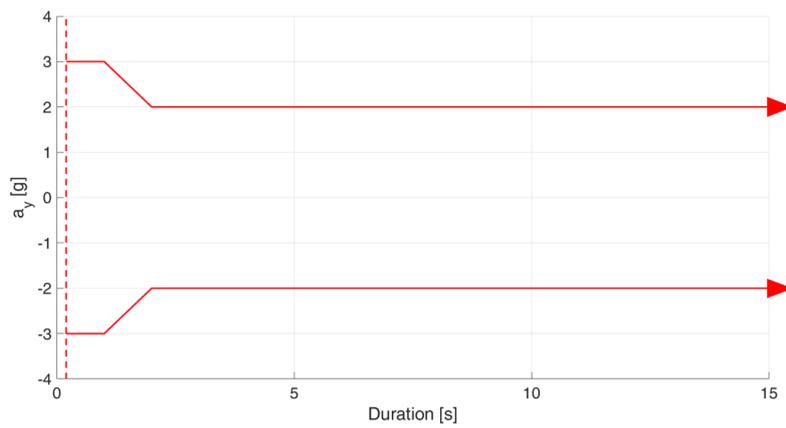


Figure I.2: Time duration limits for accelerations in  $y$ -direction

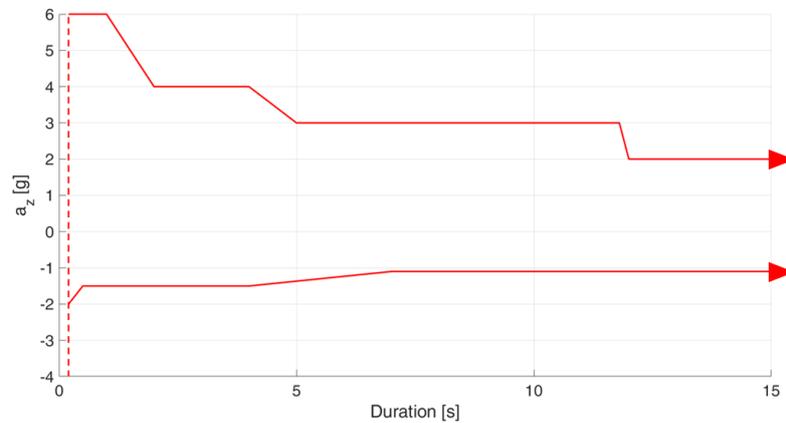
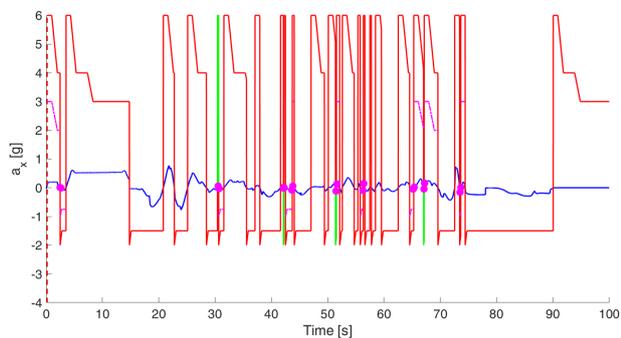


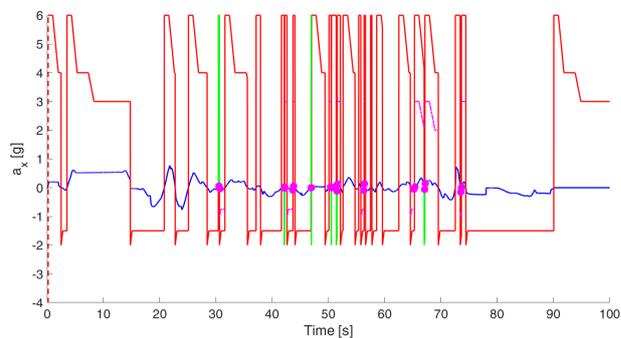
Figure I.3: Time duration limits for accelerations in  $z$ -direction

## I.2 Resulting Passenger Accelerations

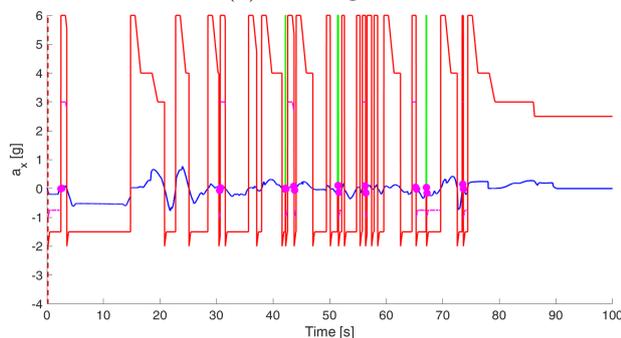
### I.2.1 Configuration 1



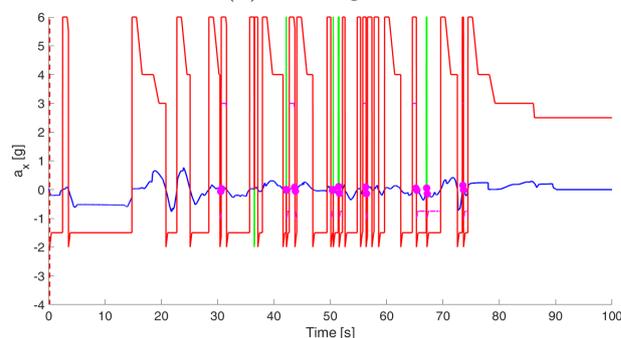
(a) Passenger 1



(b) Passenger 2

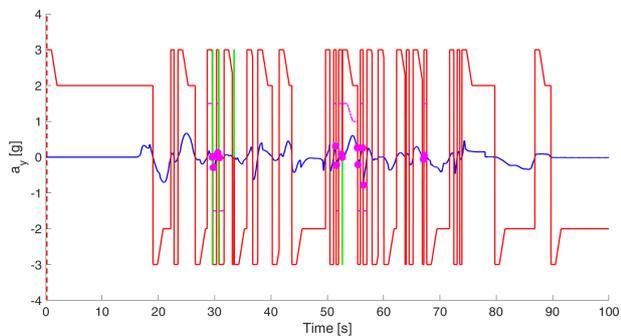


(c) Passenger 3

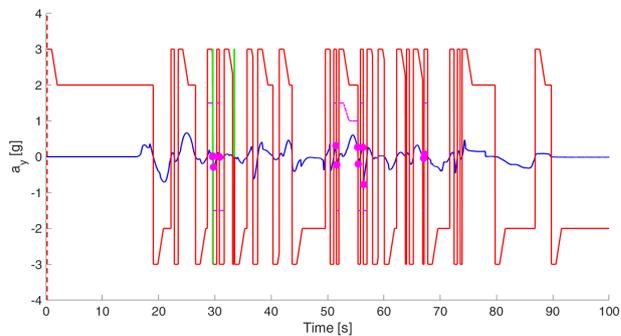


(d) Passenger 4

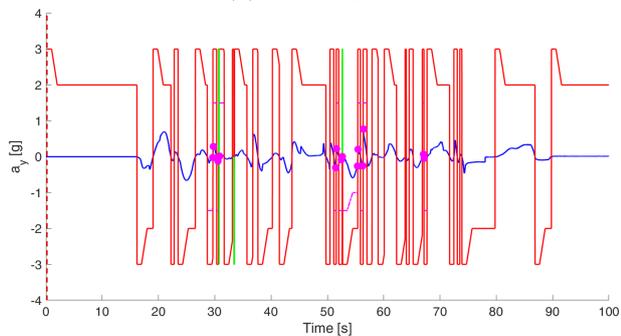
Figure I.4: Actual (blue) and admissible (red) accelerations in the  $x$ -direction



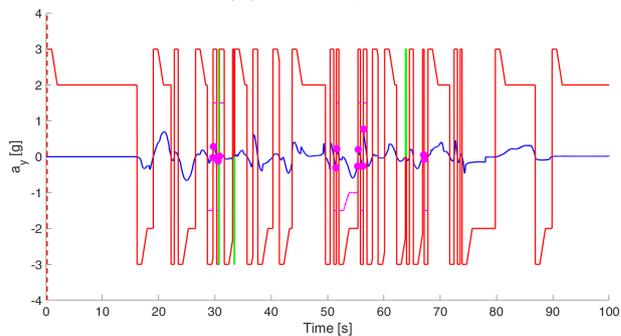
(a) Passenger 1



(b) Passenger 2

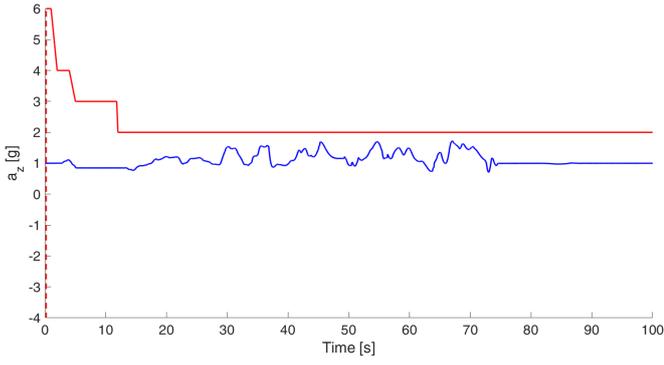


(c) Passenger 3

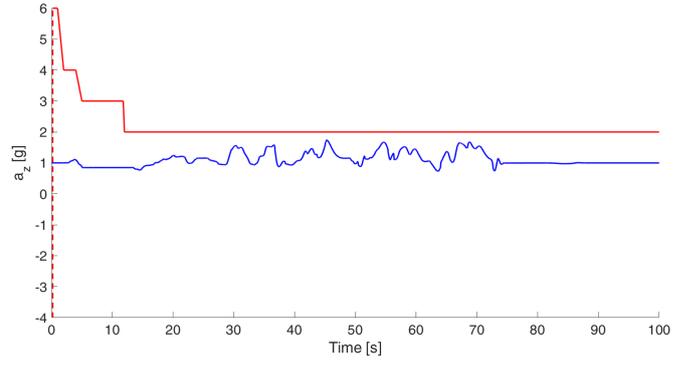


(d) Passenger 4

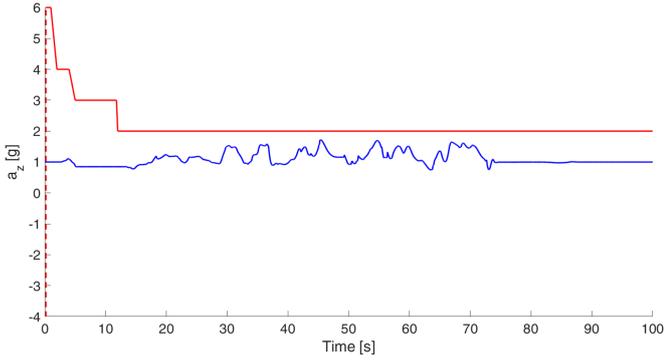
Figure I.5: Actual (blue) and admissible (red) accelerations in the  $y$ -direction



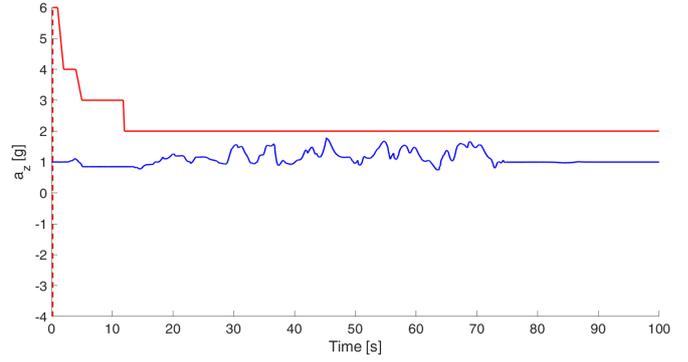
(a) Passenger 1



(b) Passenger 2

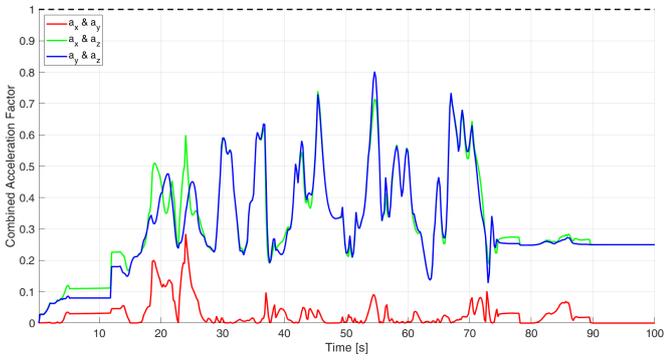


(c) Passenger 3

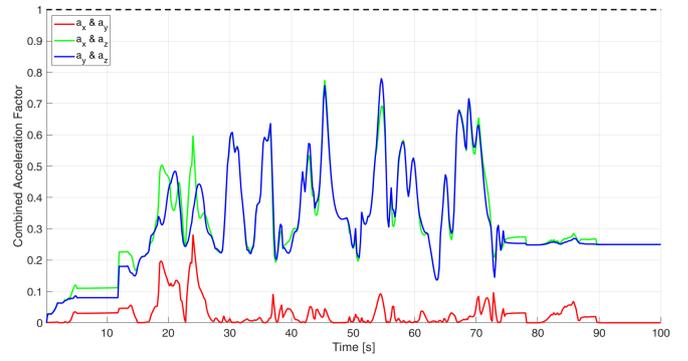


(d) Passenger 4

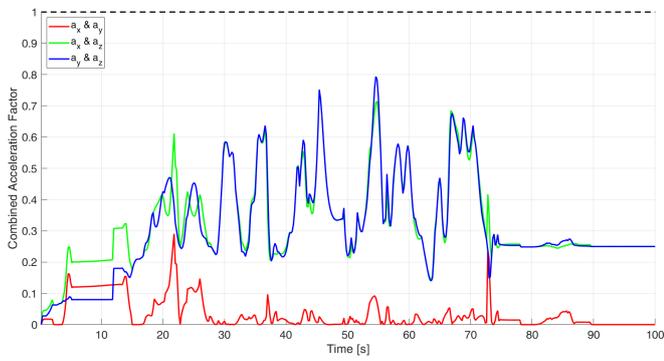
Figure I.6: Actual (blue) and admissible (red) accelerations in the  $z$ -direction



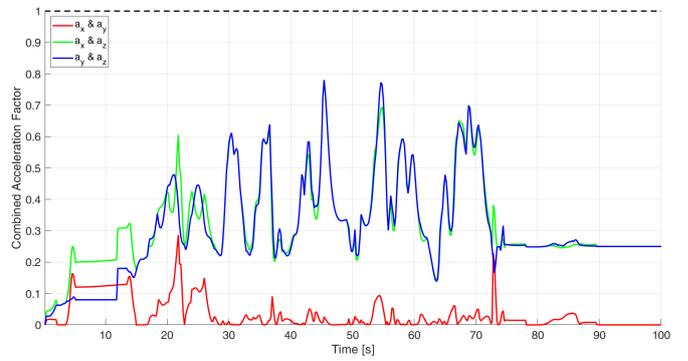
(a) Passenger 1



(b) Passenger 2



(c) Passenger 3



(d) Passenger 4

Figure I.7: Combined acceleration factors with respect to the combination limit

## I.2.2 Configuration 2

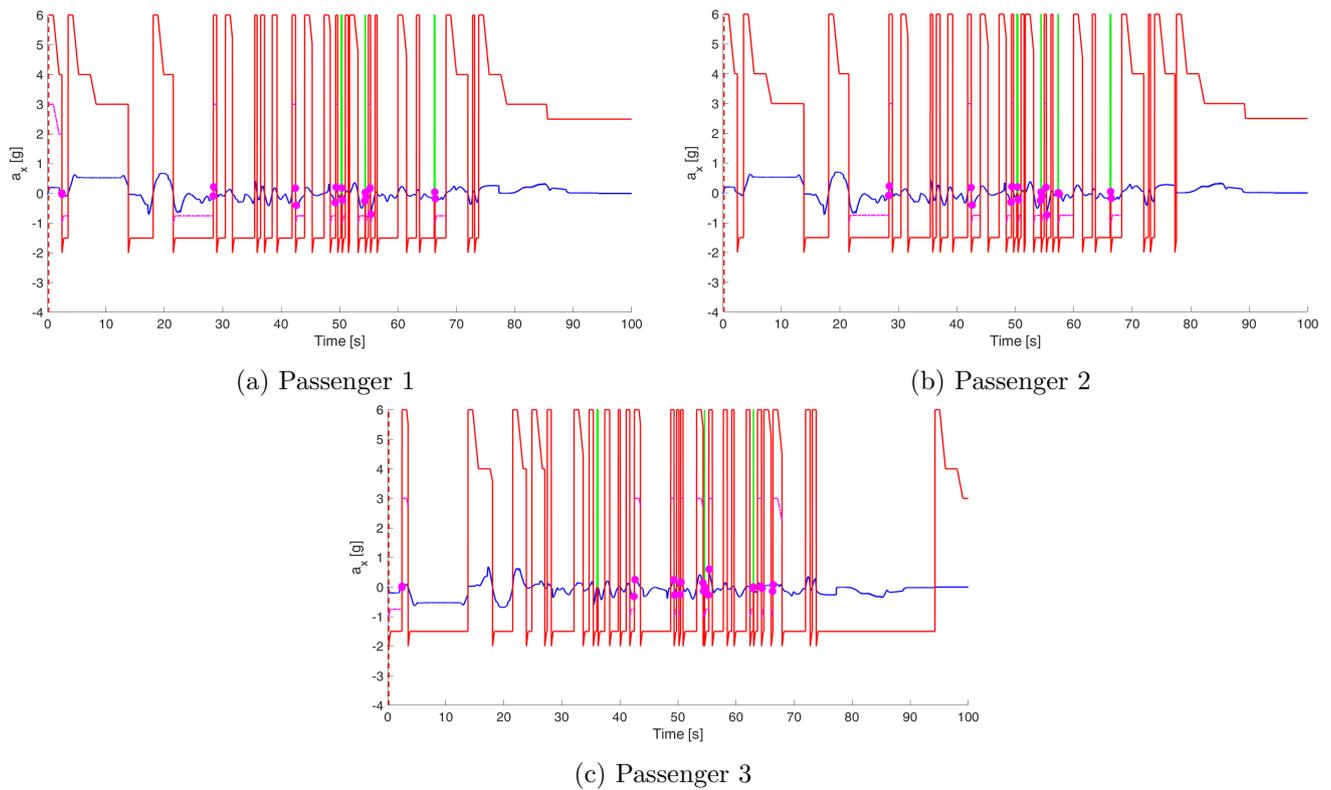


Figure I.8: Actual (blue) and admissible (red) accelerations in the  $x$ -direction

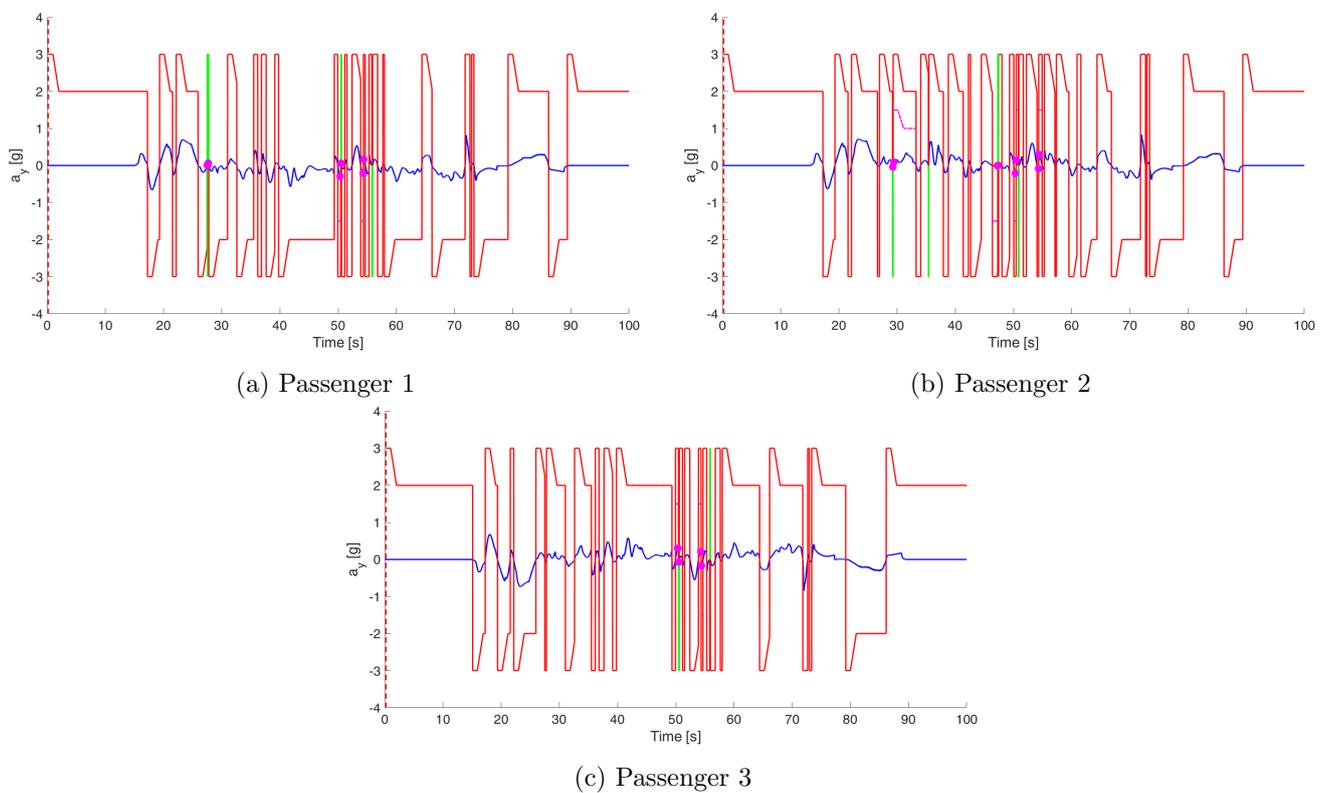
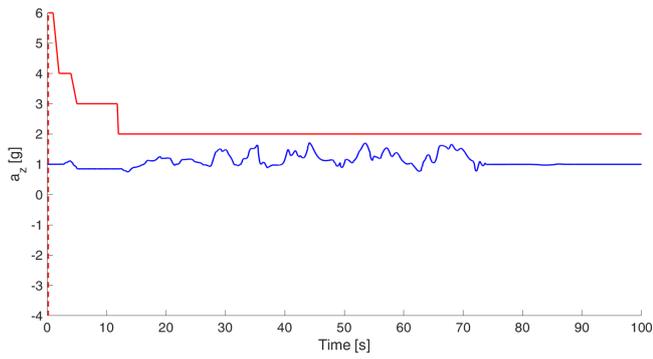
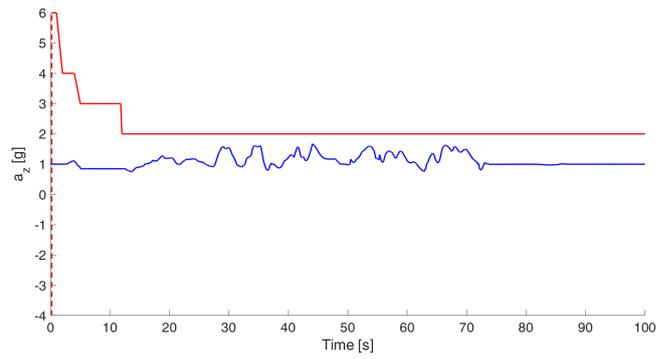


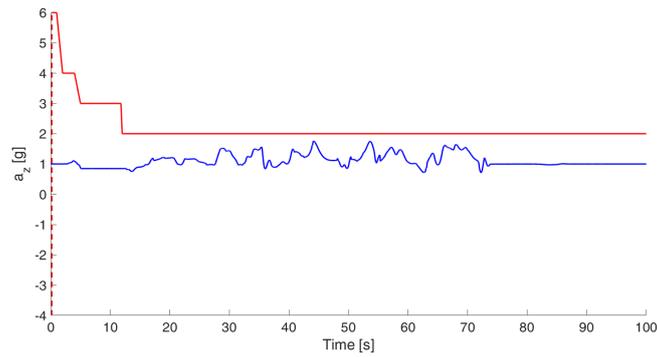
Figure I.9: Actual (blue) and admissible (red) accelerations in the  $y$ -direction



(a) Passenger 1

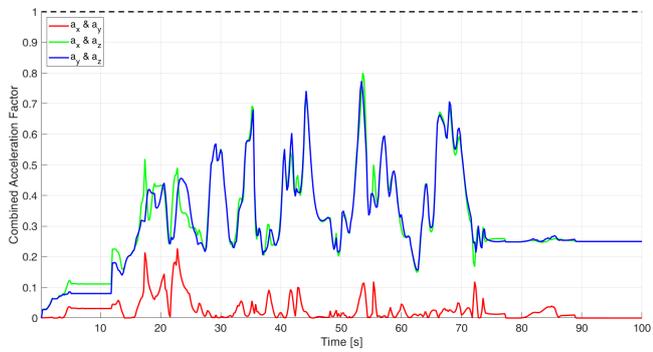


(b) Passenger 2

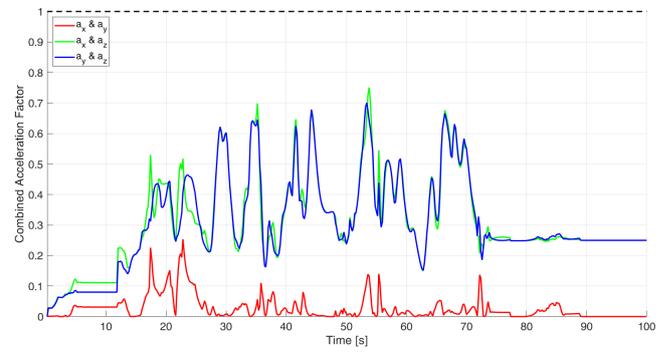


(c) Passenger 3

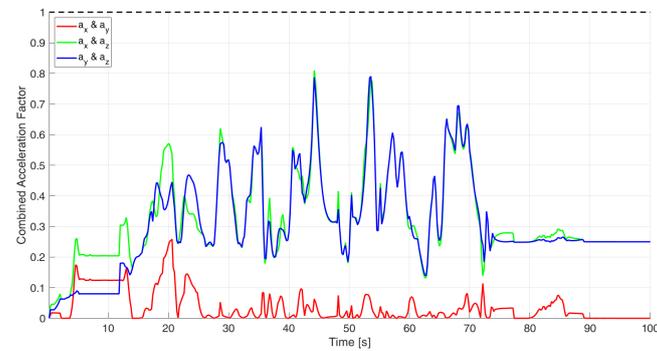
Figure I.10: Actual (blue) and admissible (red) accelerations in the  $z$ -direction



(a) Passenger 1



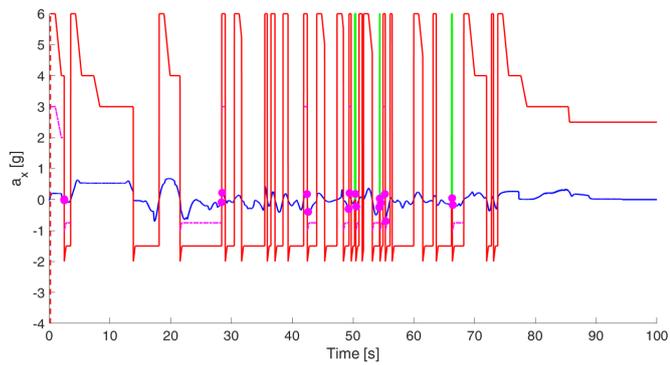
(b) Passenger 2



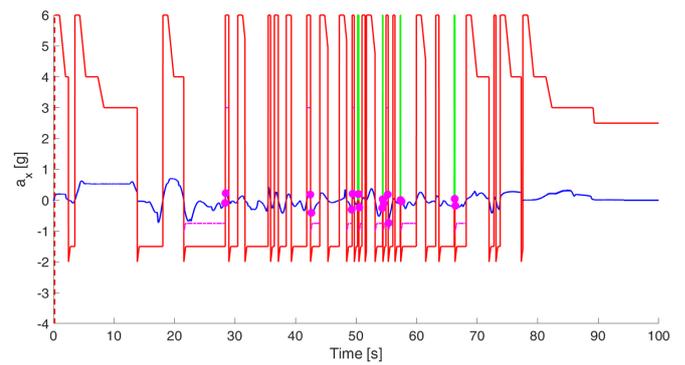
(c) Passenger 3

Figure I.11: Combined acceleration factors with respect to the combination limit

### I.2.3 Configuration 3

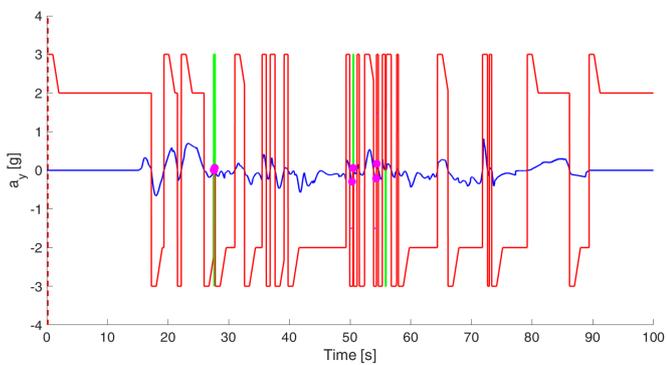


(a) Passenger 1

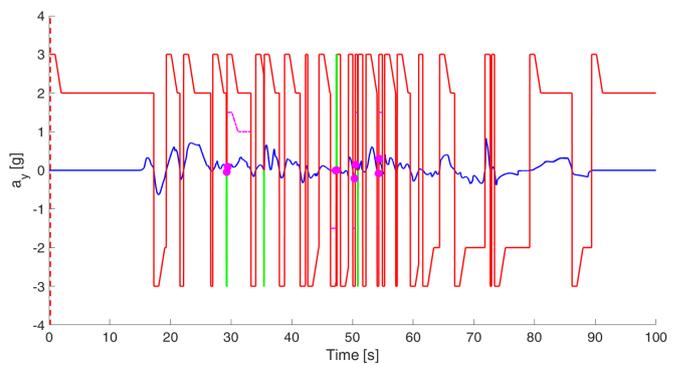


(b) Passenger 2

Figure I.12: Actual (blue) and admissible (red) accelerations in the  $x$ -direction

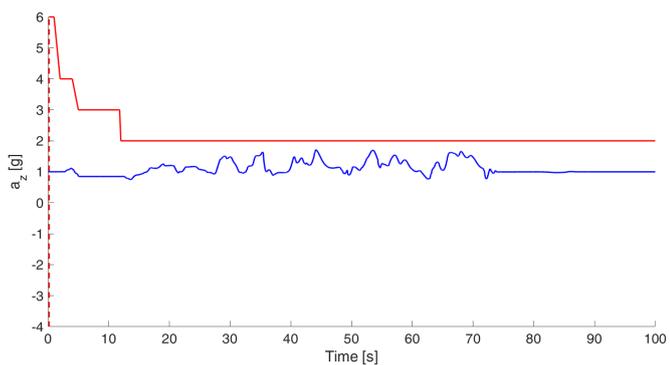


(a) Passenger 1

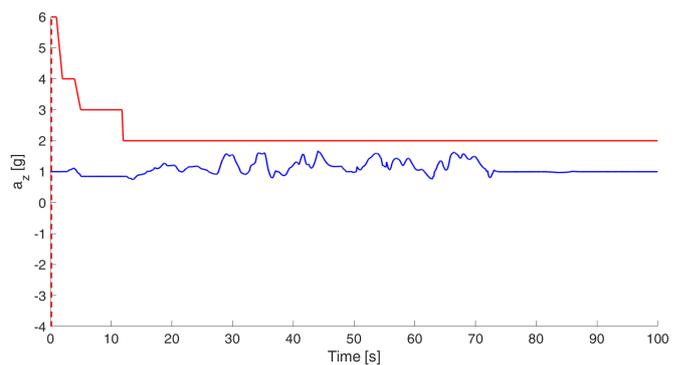


(b) Passenger 2

Figure I.13: Actual (blue) and admissible (red) accelerations in the  $y$ -direction

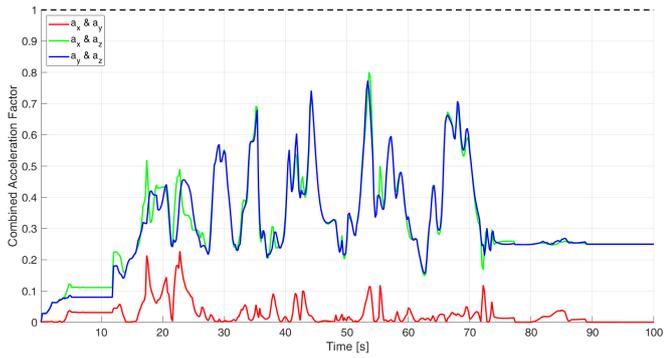


(a) Passenger 1

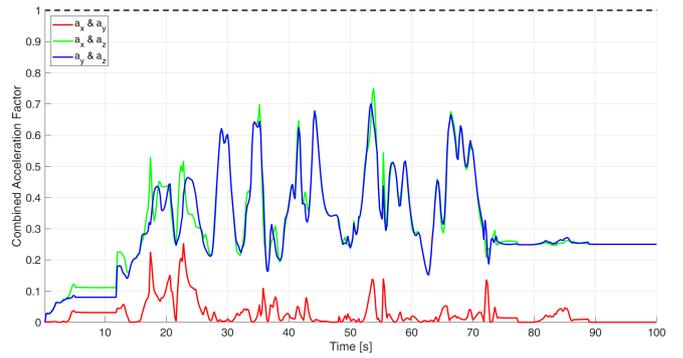


(b) Passenger 2

Figure I.14: Actual (blue) and admissible (red) accelerations in the  $z$ -direction



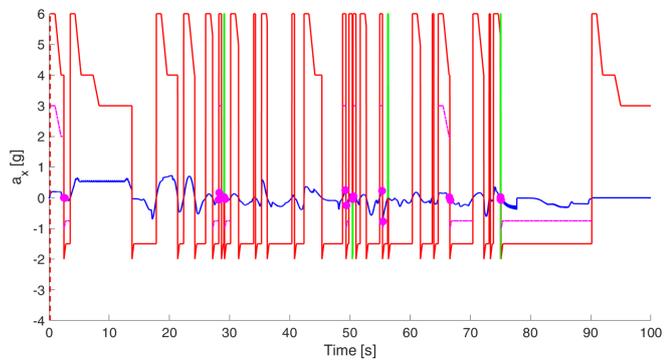
(a) Passenger 1



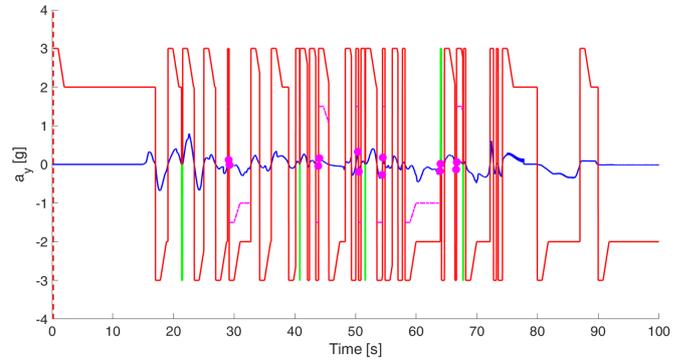
(b) Passenger 2

Figure I.15: Combined acceleration factors with respect to the combination limit

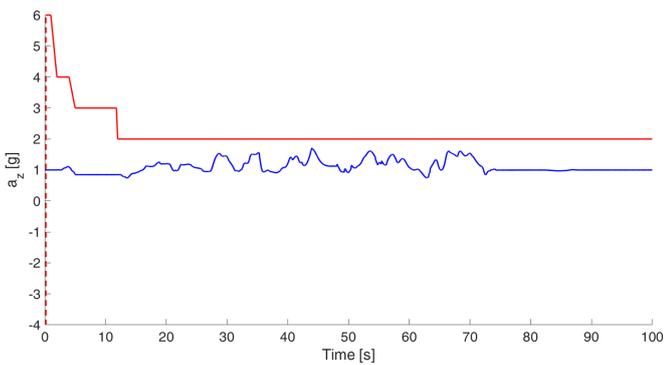
### I.2.4 Configuration 4



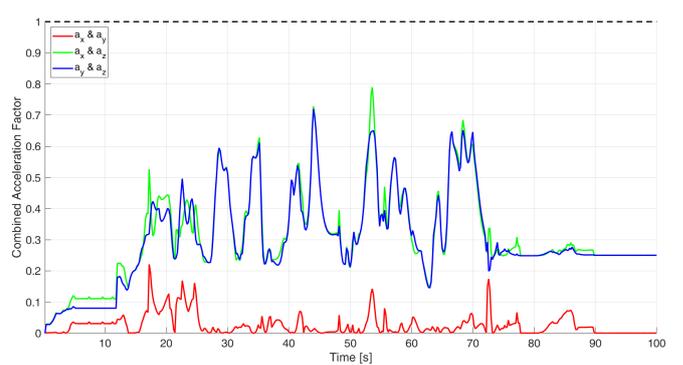
(a) Actual (blue) and admissible (red) accelerations in the  $x$ -direction for Passenger 1



(b) Actual (blue) and admissible (red) accelerations in the  $y$ -direction for Passenger 1



(a) Actual (blue) and admissible (red) accelerations in the  $z$ -direction for Passenger 1



(b) Combined acceleration factors with respect to the combination limit for Passenger 1

### I.3 Copy of AdmissibleX.m

```
1 function [AdmAccX] = AdmissibleX(t,AccX)
2
3 if AccX ≥ 0
4     if 0.2 ≤ t && t ≤ 1
5         AdmAccX = 6;
6     elseif 1 < t && t ≤ 2
7         AdmAccX = 8 - 2*t;
8     elseif 2 < t && t ≤ 4
9         AdmAccX = 4;
10    elseif 4 < t && t ≤ 5
11        AdmAccX = 8 - t;
12    elseif 5 < t && t ≤ 11.8
13        AdmAccX = 3;
14    elseif 11.8 < t && t ≤ 12
15        AdmAccX = 32.5 - 2.5*t;
16    else
17        AdmAccX = 2.5;
18    end
19 else
20     if 0.2 ≤ t && t ≤ 0.5
21         AdmAccX = -(7/3) + (5/3)*t;
22     else
23         AdmAccX = -1.5;
24     end
25 end
```

### I.4 Copy of AdmissibleY.m

```
1 function [AdmAccY] = AdmissibleY(t,AccY)
2
3 if AccY ≥ 0
4     if 0.2 ≤ t && t ≤ 1
5         AdmAccY = 3;
6     elseif 1 < t && t ≤ 2
7         AdmAccY = 4 - t;
8     else
9         AdmAccY = 2;
10    end
11 else
12     if 0.2 ≤ t && t ≤ 1
13         AdmAccY = -3;
14     elseif 1 < t && t ≤ 2
15         AdmAccY = -4 + t;
16     else
17         AdmAccY = -2;
18     end
19 end
```

## I.5 Copy of AdmissibleZ.m

```
1 function [AdmAccZ] = AdmissibleZ(t,AccZ)
2
3 if AccZ ≥ 0
4     if 0.2 ≤ t && t ≤ 1
5         AdmAccZ = 6;
6     elseif 1 < t && t ≤ 2
7         AdmAccZ = 8 - 2*t;
8     elseif 2 < t && t ≤ 4
9         AdmAccZ = 4;
10    elseif 4 < t && t ≤ 5
11        AdmAccZ = 8 - t;
12    elseif 5 < t && t ≤ 11.8
13        AdmAccZ = 3;
14    elseif 11.8 < t && t ≤ 12
15        AdmAccZ = 62 - 5*t;
16    else
17        AdmAccZ = 2;
18    end
19 else
20     if 0.2 ≤ t && t ≤ 0.5
21         AdmAccZ = -(7/3) + (5/3)*t;
22     elseif 0.5 < t && t ≤ 4
23         AdmAccZ = -1.5;
24     elseif 4 < t && t ≤ 7
25         AdmAccZ = -(61/30) + (2/15)*t;
26     else
27         AdmAccZ = -1.1;
28     end
29 end
```

## I.6 Copy of AdmAccelerations.m

```
1 %% Save the acceleration data
2 clc
3
4 % Save the acceleration data of passenger 1 to .mat-files in the working directory:
5 save('AccX1.mat','AccX1');
6 save('AccY1.mat','AccY1');
7 save('AccZ1.mat','AccZ1');
8 % Save the acceleration data of passenger 2 to .mat-files in the working directory:
9 save('AccX2.mat','AccX2');
10 save('AccY2.mat','AccY2');
11 save('AccZ2.mat','AccZ2');
12 % Save the acceleration data of passenger 3 to .mat-files in the working directory:
13 save('AccX3.mat','AccX3');
14 save('AccY3.mat','AccY3');
15 save('AccZ3.mat','AccZ3');
16 % Save the acceleration data of passenger 4 to .mat-files in the working directory:
17 save('AccX4.mat','AccX4');
18 save('AccY4.mat','AccY4');
19 save('AccZ4.mat','AccZ4');
20
21 %% Data Pre-Processing
22 clear all, clc
23
24 % Specify gravitational acceleration:
```

```

25 g = 9.80665; %m/s^2
26
27 % Load the passenger acceleration data of interest:
28 load('AccX1.mat')
29 load('AccY1.mat')
30 load('AccZ1.mat')
31
32 % Convert unit for acceleration from m/s^2 to g-loads:
33 AccX = -AccX1/g;
34 AccY = -AccY1/g;
35 AccZ = -AccZ1/g;
36
37 % Specify sample frequency (Hz) and kinematic model time-step (s):
38 fn = 5;
39 dt = 0.01;
40
41 %% Time Duration Limit for Accelerations in X-Direction
42
43 % End-time and time array:
44 t_end = 15; %s
45 t_vec = 0.2:dt:t_end;
46
47 % Determine the positive limit for the x-acceleration:
48 for i = 1:length(t_vec)
49     AdmAccXPos(i) = AdmissibleX(t_vec(i),1);
50 end
51 % Determine the negative limit for the x-acceleration:
52 for i = 1:length(t_vec)
53     AdmAccXNeg(i) = AdmissibleX(t_vec(i),-1);
54 end
55
56 %% Time Duration Limit for Accelerations in Y-Direction
57
58 % Determine the positive limit for the y-acceleration:
59 for i = 1:length(t_vec)
60     AdmAccYPos(i) = AdmissibleY(t_vec(i),1);
61 end
62 % Determine the negative limit for the y-acceleration:
63 for i = 1:length(t_vec)
64     AdmAccYNeg(i) = AdmissibleY(t_vec(i),-1);
65 end
66
67 %% Time Duration Limit for Accelerations in Z-Direction
68
69 % Determine the positive limit for the z-acceleration:
70 for i = 1:length(t_vec)
71     AdmAccZPos(i) = AdmissibleZ(t_vec(i),1);
72 end
73 % Determine the negative limit for the z-acceleration:
74 for i = 1:length(t_vec)
75     AdmAccZNeg(i) = AdmissibleZ(t_vec(i),-1);
76 end
77
78 %% Combined Acceleration Factors
79
80 % Initialization
81 k = 1;
82 t(k) = 1/fn;
83 index = ((1/fn)/dt):(1/fn)/dt:length(AccX);
84 AdmAccX1(k) = AdmissibleX(t,AccX(index(k)));
85 AdmAccY1(k) = AdmissibleY(t,AccY(index(k)));
86 AdmAccZ1(k) = AdmissibleZ(t,AccZ(index(k)));
87

```

```

88 for k = 2:length(index)
89     % Determine the admissible accelerations for positive x-accelerations:
90     if AccX(index(k)) ≥ 0
91         if AccX(index(k-1)) ≥ 0
92             t(k) = t(k-1) + (1/fn);
93         else
94             t(k) = (1/fn);
95         end
96     % Determine the admissible accelerations for negative x-accelerations:
97     else
98         if AccX(index(k-1)) < 0
99             t(k) = t(k-1) + (1/fn);
100        else
101            t(k) = (1/fn);
102        end
103    end
104    AdmAccX1(k) = AdmissibleX(t(k),AccX(index(k)));
105
106    % Determine the admissible accelerations for positive y-accelerations:
107    if AccY(index(k)) ≥ 0
108        if AccY(index(k-1)) ≥ 0
109            t(k) = t(k-1) + (1/fn);
110        else
111            t(k) = (1/fn);
112        end
113    % Determine the admissible accelerations for negative y-accelerations:
114    else
115        if AccY(index(k-1)) < 0
116            t(k) = t(k-1) + (1/fn);
117        else
118            t(k) = (1/fn);
119        end
120    end
121    AdmAccY1(k) = AdmissibleY(t(k),AccY(index(k)));
122
123    % Determine the admissible accelerations for positive z-accelerations:
124    if AccZ(index(k)) ≥ 0
125        if AccZ(index(k-1)) ≥ 0
126            t(k) = t(k-1) + (1/fn);
127        else
128            t(k) = (1/fn);
129        end
130    % Determine the admissible accelerations for negative z-accelerations:
131    else
132        if AccZ(index(k-1)) < 0
133            t(k) = t(k-1) + (1/fn);
134        else
135            t(k) = (1/fn);
136        end
137    end
138    AdmAccZ1(k) = AdmissibleZ(t(k),AccZ(index(k)));
139
140    % Determine the combined accelerations:
141    CombinAccXY(k) = (AccX(index(k))/AdmAccX1(k))^2 + (AccY(index(k))/AdmAccY1(k))^2;
142    CombinAccXZ(k) = (AccX(index(k))/AdmAccX1(k))^2 + (AccZ(index(k))/AdmAccZ1(k))^2;
143    CombinAccYZ(k) = (AccY(index(k))/AdmAccY1(k))^2 + (AccZ(index(k))/AdmAccZ1(k))^2;
144 end
145
146 % Specify the end-time and an accompanying time array:
147 t_end = length(index)*(1/fn);
148 t_vec = (1/fn):(1/fn):t_end;
149
150 %% Reversals in X and Y

```

```

151
152 % Specify a time array:
153 t_vec = 0:dt:t_end;
154
155 % Initialization:
156 k = 20;
157 SignChangeX(1:k) = 0;
158 SignChangeY(1:k) = 0;
159
160 % Determine the zero-crossings for the x-acceleration and y-acceleration data:
161 for k = 21:length(t_vec)
162     if sign(AccX(k-1)) ≠ sign(AccX(k))
163         SignChangeX(k) = 1;
164     else
165         SignChangeX(k) = 0;
166     end
167     if sign(AccY(k-1)) ≠ sign(AccY(k))
168         SignChangeY(k) = 1;
169     else
170         SignChangeY(k) = 0;
171     end
172 end
173
174 % Determine the location of the non-sustained events for the x-acceleration data:
175 zcridxX = find(SignChangeX == 1);
176 zcrttimeX = (zcridxX-1.5)*dt;
177 iX = 0;
178 for k = 1:(length(zcrttimeX)-1)
179     difzcrX(k) = zcrttimeX(k+1) - zcrttimeX(k);
180     if (difzcrX(k) < (1/fn)) && (zcrttimeX(k) > (1/fn))
181         iX = iX+1;
182         % Determine the lower and upper boundary of the non-sustained events:
183         lownonsusX(iX) = zcrttimeX(k);
184         highnonsusX(iX) = zcrttimeX(k+1);
185     end
186 end
187
188 % Determine the location of the non-sustained events for the y-acceleration data:
189 zcridxY = find(SignChangeY == 1);
190 zcrttimeY = (zcridxY-1.5)*dt;
191 iY = 0;
192 for k = 1:(length(zcrttimeY)-1)
193     ...
194 end
195
196 % Initialization
197 k = 1;
198 tX(k) = 1/fn;
199 tY(k) = 1/fn;
200 AdmAccX2(k) = AdmissibleX(t(k),AccX((1/fn)/dt));
201 AdmAccY2(k) = AdmissibleY(t(k),AccY((1/fn)/dt));
202 t_vec = (1/fn):dt:t_end;
203
204 for k = 2:length(t_vec)
205     % Determine the admissible accelerations for positive x-accelerations:
206     if AccX(((1/fn)/dt) + k) ≥ 0
207         if AccX(((1/fn)/dt) + k-1) ≥ 0
208             tX(k) = tX(k-1) + dt;
209         else
210             tX(k) = (1/fn);
211         end
212     % Determine the admissible accelerations for negative x-accelerations:
213     else

```

```

214         if AccX(((1/fn)/dt) + k-1) < 0
215             tX(k) = tX(k-1) + dt;
216         else
217             tX(k) = (1/fn);
218         end
219     end
220     AdmAccX2(k) = AdmissibleX(tX(k),AccX(((1/fn)/dt) + k));
221     % Determine the admissible accelerations for positive y-accelerations:
222     if AccY(((1/fn)/dt) + k) ≥ 0
223         if AccY(((1/fn)/dt) + k-1) ≥ 0
224             tY(k) = tY(k-1) + dt;
225         else
226             tY(k) = (1/fn);
227         end
228     % Determine the admissible accelerations for negative y-accelerations:
229     else
230         if AccY(((1/fn)/dt) + k-1) < 0
231             tY(k) = tY(k-1) + dt;
232         else
233             tY(k) = (1/fn);
234         end
235     end
236     AdmAccY2(k) = AdmissibleY(tY(k),AccY(((1/fn)/dt) + k));
237 end
238
239 % Determine the admissible x-accelerations corresponding to the non-sustained events:
240 if iX > 0
241     for k = 1:iX
242         t_vec1{k} = lownonsusX(k):dt:highnonsusX(k);
243         for i = 1:(length(t_vec1{k}(:)))
244             idxnonsusX1{k}(i) = ceil((t_vec1{k}(i) - (1/fn))/dt);
245         end
246         AdmAccX3{k} = AdmAccX2(idxnonsusX1{k});
247     end
248 end
249 % Determine the admissible y-accelerations corresponding to the non-sustained events:
250 if iY > 0
251     for k = 1:iY
252         ...
253     end
254 end
255
256 % Determine the peak-to-peak transition times for the x-acceleration data:
257 [pk svalX,pkslocX] = findpeaks(AccX);
258 minlogX = islocalmin(AccX);
259 minlocX = find(minlogX ≠ 0);
260 minvalX = AccX(minlocX);
261 % Determine the peak-to-peak transition times for the y-acceleration data:
262 [pk svalY,pkslocY] = findpeaks(AccY);
263 minlogY = islocalmin(AccY);
264 minlocY = find(minlogY ≠ 0);
265 minvalY = AccY(minlocY);
266
267 % Determine the critical transition peaks for the x-acceleration data:
268 for k = 1:length(zcridxX)
269     lowpksX1{k} = max(pkslocX(find(pkslocX < zcridxX(k))));
270     if isempty(lowpksX1{k}) == 1
271         lowpksX2(k) = 0;
272     else
273         lowpksX2(k) = lowpksX1{k};
274     end
275
276     lowminX1{k} = max(minlocX(find(minlocX < zcridxX(k))));

```

```

277     if isempty(lowminX1{k}) == 1
278         lowminX2(k) = 0;
279     else
280         lowminX2(k) = lowminX1{k};
281     end
282
283     if lowpksX2(k) > lowminX2(k)
284         lowidxX(k) = lowpksX2(k);
285     else
286         lowidxX(k) = lowminX2(k);
287     end
288
289     highpksX1{k} = min(pkslocX(find(pkslocX > zcridxX(k))));
290     if isempty(highpksX1{k}) == 1
291         highpksX2(k) = inf;
292     else
293         highpksX2(k) = highpksX1{k};
294     end
295
296     highminX1{k} = min(minlocX(find(minlocX > zcridxX(k))));
297     if isempty(highminX1{k}) == 1
298         highminX2(k) = inf;
299     else
300         highminX2(k) = highminX1{k};
301     end
302
303     if highpksX2(k) < highminX2(k)
304         highidxX(k) = highpksX2(k);
305     else
306         highidxX(k) = highminX2(k);
307     end
308     difX(k) = (highidxX(k) - lowidxX(k))*dt;
309 end
310 lowrevidxX = lowidxX(find(difX < (1/fn)));
311 lowrevtimeX = (lowrevidxX-1)*dt;
312 lowrevvalX = AccX(lowrevidxX);
313 highrevidxX = highidxX(find(difX < (1/fn)));
314 highrevtimeX = (highrevidxX-1)*dt;
315 highrevvalX = AccX(highrevidxX);
316 % Determine the critical transition peaks for the y-acceleration data:
317 for k = 1:length(zcridxY)
318     ...
319 end
320 lowrevidxY = lowidxY(find(difY < (1/fn)));
321 lowrevtimeY = (lowrevidxY-1)*dt;
322 lowrevvalY = AccY(lowrevidxY);
323 highrevidxY = highidxY(find(difY < (1/fn)));
324 highrevtimeY = (highrevidxY-1)*dt;
325 highrevvalY = AccY(highrevidxY);
326
327 % Rewrite indices of non-sustained events to array for x-acceleration data:
328 j = 0;
329 if iX > 0
330     for i = 1:length(idxnonsusX1)
331         j(i) = j(i)+1;
332         j(i+1) = j(i)+length(idxnonsusX1{i})-1;
333         IdxNonSusX(j(i):j(i+1)) = idxnonsusX1{i};
334     end
335 end
336 % Rewrite indices of non-sustained events to array for y-acceleration data:
337 j = 0;
338 if iY > 0
339     for i = 1:length(idxnonsusY1)

```

```

340     ...
341     end
342 end
343
344 % Reduction of the allowable limit for x-acceleration data by fifty percent
345 % for transition times lower than 200 ms:
346 n = 0;
347 t_vec = (1/fn):dt:t_end;
348 if length(find(difX < (1/fn))) > 0
349     for k = 1:length(find(difX < (1/fn)))
350         n = n+2;
351         if lowrevidxX(k) < (((1/fn)/dt)+1)
352             lowsusX(n-1) = (1/fn)+dt;
353             highsusX(n-1) = (1/fn)+dt;
354         elseif isempty(find(zcridxX < lowrevidxX(k))) == 1
355             lowsusX(n-1) = (1/fn)+dt;
356             highsusX(n-1) = (min(zcridxX(find(zcridxX > lowrevidxX(k)))))*dt;
357         else
358             lowsusX(n-1) = (max(zcridxX(find(zcridxX < lowrevidxX(k)))))*dt;
359             highsusX(n-1) = (min(zcridxX(find(zcridxX > lowrevidxX(k)))))*dt;
360         end
361         if highrevidxX(k) < (((1/fn)/dt)+1)
362             lowsusX(n) = (1/fn)+dt;
363             highsusX(n) = (1/fn)+dt;
364         elseif isempty(find(zcridxX > highrevidxX(k))) == 1
365             lowsusX(n) = (max(zcridxX(find(zcridxX < highrevidxX(k)))))*dt;
366             highsusX(n) = t_end;
367         else
368             lowsusX(n) = (max(zcridxX(find(zcridxX < highrevidxX(k)))))*dt;
369             highsusX(n) = (min(zcridxX(find(zcridxX > highrevidxX(k)))))*dt;
370         end
371     end
372
373     for m = 1:n
374         t_vec3{m} = lowsusX(m):dt:highsusX(m);
375         for i = 1:(length(t_vec3{m}(:)))
376             idxsusX{m}(i) = round((t_vec3{m}(i) - (1/fn))/dt,-0)-1;
377             if (iX > 0) && (isempty(find(idxsusX{m}(i) == IdxNonSusX)) == 0)
378                 idxsusX{m}(i) = 0;
379             end
380         end
381         idxsusX{m}(idxsusX{m} == 0) = [];
382         AdmSusX{m} = t_vec(idxsusX{m});
383         AdmAccX4{m} = AdmAccX2(idxsusX{m})/2;
384     end
385 end
386 % Reduction of the allowable limit for y-acceleration data by fifty percent
387 % for transition times lower than 200 ms:
388 n = 0;
389 if length(find(difX < (1/fn))) > 0
390     for k = 1:length(find(difY < (1/fn)))
391         ...
392     end
393
394     for m = 1:n
395         ...
396     end
397 end
398
399 % Conversion to array for sustained events in x-acceleration data:
400 j = 0;
401 if isempty(find(difX < (1/fn))) == 0
402     for i = 1:length(idxsusX)

```

```

403         j(i) = j(i)+1;
404         j(i+1) = j(i)+length(idxsusX{i})-1;
405         IdxSusX(j(i):j(i+1)) = idxsusX{i};
406     end
407 end
408 % Conversion to array for sustained events in y-acceleration data:
409 j = 0;
410 if isempty(find(difX < (1/fn))) == 0
411     for i = 1:length(idxsusY)
412         ...
413     end
414 end
415
416 % Create an array with indices corresponding to x-acceleration transition regions:
417 m = 1;
418 n(m) = 0;
419 if isempty(find(difX < (1/fn))) == 0
420     for k = 1:length(find(difX < (1/fn)))
421         m = m+1;
422         n(m) = n(m-1) + length(lowrevidxX(k):highrevidxX(k));
423         IdxTransX((n(m-1)+1):n(m)) = lowrevidxX(k):highrevidxX(k);
424     end
425 end
426 % Create an array with indices corresponding to y-acceleration transition regions:
427 if isempty(find(difY < (1/fn))) == 0
428     m = 1;
429     n(m) = 0;
430     for k = 1:length(find(difY < (1/fn)))
431         ...
432     end
433 end
434
435 % Detect limit violations for the x-acceleration data:
436 for k = 1:length(AdmAccX2)
437     if (iX > 0) && (isempty(find(difX < (1/fn))) == 0)
438         if (isempty(find(k == IdxNonSusX)) ≠ 0) && (isempty(find(k == IdxSusX)) ≠ 0)
439             VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
440         elseif isempty(find(k == IdxSusX)) == 0
441             if isempty(find((k + ((1/fn)/dt)) == IdxTransX)) == 0
442                 VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > ...
443                     abs(AdmAccX2(k)/2)));
444             else
445                 VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
446             end
447         else
448             VioChkX(k) = 0;
449         end
450     elseif (iX > 0) && (isempty(find(difX < (1/fn))) == 1)
451         if isempty(find(k == IdxNonSusX)) ≠ 0
452             VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
453         else
454             VioChkX(k) = 0;
455         end
456     elseif (iX == 0) && (isempty(find(difX < (1/fn))) == 0)
457         if isempty(find(k == IdxSusX)) ≠ 0
458             VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
459         else
460             if isempty(find((k + ((1/fn)/dt)) == IdxTransX)) == 0
461                 VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > ...
462                     abs(AdmAccX2(k)/2)));
463             else
464                 VioChkX(k) = ¬isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
465             end
466         end
467     end
468 end

```

```

464     end
465     else
466         VioChkX(k) = ~isempty(find(abs(AccX(k + ((1/fn)/dt))) > abs(AdmAccX2(k))));
467     end
468 end
469
470 % Detect limit violations for the y-acceleration data:
471 for k = 1:length(AdmAccY2)
472     ...
473 end
474
475 %% Transitions in Z
476
477 % Initialization
478 k = 1;
479 tZ(k) = 1/fn;
480 AdmAccZ2(k) = AdmissibleZ(t(k),AccZ((1/fn)/dt));
481 t_vec = (1/fn):dt:t_end;
482
483 for k = 2:length(t_vec)
484     % Determine the admissible accelerations for positive z-accelerations:
485     if AccZ(((1/fn)/dt) + k) ≥ 0
486         if AccZ(((1/fn)/dt) + k-1) ≥ 0
487             tZ(k) = tZ(k-1) + dt;
488         else
489             tZ(k) = (1/fn);
490         end
491     % Determine the admissible accelerations for negative z-accelerations:
492     else
493         if AccZ(((1/fn)/dt) + k-1) < 0
494             tZ(k) = tZ(k-1) + dt;
495         else
496             tZ(k) = (1/fn);
497         end
498     end
499     AdmAccZ2(k) = AdmissibleZ(tZ(k),AccZ(((1/fn)/dt) + k));
500 end
501
502 % Determine if additional criteria apply to the z-acceleration transitions:
503 if isempty(find(AccZ ≤ 0)) == 1
504     disp('The criteria for transitions in a.z do not apply')
505
506     for k = 1:length(AdmAccZ2)
507         VioChkZ(k) = ~isempty(find(abs(AccZ(k + ((1/fn)/dt))) > abs(AdmAccZ2(k))));
508     end
509
510     if isempty(find(VioChkZ == 1)) == 0
511         VioIdxZ = find(VioChkZ == 1);
512         for k = 1:length(VioIdxZ)
513             VioAccZ(k) = AccZ(VioIdxZ(k) + ((1/fn)/dt));
514             VioTimeZ(k) = t_vec(VioIdxZ(k));
515         end
516     end
517 else
518     disp('Please investigate if the criteria for transitions in a.z are met')
519 end

```

# J. Model Validation

The reaction forces between the wheels and track will be validated by evaluating the equilibrium of forces at all four bogies. Since only reaction forces were used to determine the rolling and bearing frictional forces, the upcoming sections will focus on a validation of forces rather than torques. Before this validation can be performed, the equilibrium of forces at respectively the front and rear axle needs to be evaluated first though, as shown in the first section of this appendix.

## J.1 Forces at Front and Rear Hinge Joints

The bogies are connected to the front and rear axle by revolute joints, so the bogies can hinge around the axle. The forces at the spherical and revolute joint between the chassis beam and respectively the front and rear axle have already been validated. Additionally, also the masses of the axles and the linear accelerations at their centres of mass can be regarded as known quantities. The location of the centre of mass and the points of application of the hinge forces have been indicated in Figure J.1.

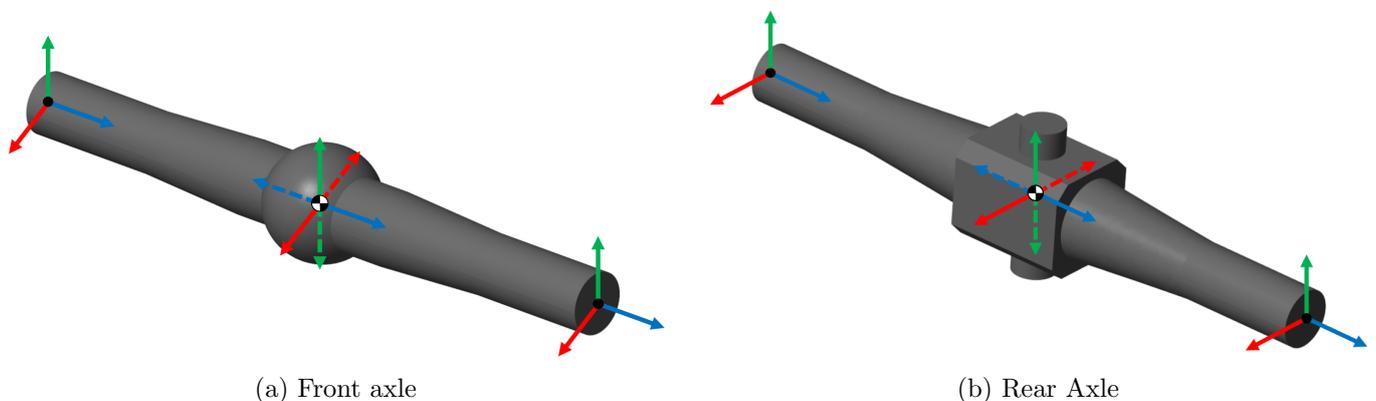


Figure J.1: The forces acting on an axle have been decomposed in  $x$  (red),  $y$  (green), and  $z$ -direction (blue) and they are represented by a solid line, while the accelerations at the centre of mass are represented by a dashed line

Calculating the forces at each individual hinge joint is unfeasible due to the dynamic nature of the model. The presumption of an equal distribution of the load over the two hinge joints is namely flawed for most track sections. Hence, the decision is taken to validate the sum of forces at the two hinge joints, as shown in Equation J.1.

$$F_{x,hinges} = -ma_x - F_{x,axle} \quad F_{y,hinges} = -ma_y - F_{y,axle} \quad F_{z,hinges} = -ma_z - F_{z,axle} \quad (J.1)$$

where  $F_{hinges}$  represents the sum of forces at the hinges,  $F_{axle}$  symbolizes the force at the joint,  $m$  denotes the mass of the axle and  $a$  represents the acceleration at its centre of mass. Calculation of the sum of hinge reaction forces according to Equation J.1 and a subsequent comparison with the measured hinge forces yields the results presented in Figures J.7 and J.8 (Section J.3) for respectively the front and rear axle. The alignment of the calculated and measured hinge forces in these graphs proves that the expressions in Equation J.1 are correct and that the motion of the axles is described by a logical relation between mass, accelerations, and forces. Hence, the validated hinge forces can be used in the validation analysis of the forces between the track and wheels, as presented in the next section.

## J.2 Forces between Track and Wheels

Two local frames have been defined at a specified distance from the running wheels, so their origins are exactly aligned with the heartline of the track. The relative motion of the bogies with respect to the track is constrained by means of point-on-curve constraints, which constrain the origins of the local frames to the track splines. The normal forces exerted by the track on the wheels at the point-on-curve constraints can be measured, and a validation of the measured force values is presented in this section. Hinge joints connect the bogies to the axles, and the forces sensed at these joints have already been validated in the previous section. Hence, these forces can be used for the validation of the reaction forces between the track and wheels. Reversing their direction and reformulating these forces in terms of the prevalent reference coordinate system yields force values that can be used as input for the current validation analysis. The mass of the bogies and the linear accelerations measured at the centre of mass are also required for an appropriate evaluation of the forces exerted by the track on the wheels. These normal forces are decomposed in  $x$ ,  $y$ , and  $z$ -direction according to the convention specified by the local frames, as shown in Figure J.2. The normal forces are subsequently redefined in terms of forces sensed at respectively the running, guide, and upstop wheels. The forces in  $y$ -direction are assigned to either the running or upstop wheels, while the guide-wheels account for the forces in the  $z$ -direction. Please note that the reaction forces in  $z$ -direction are only sensed at the front-left and rear-right bogies. The general expression presented in Equation J.2 applies to all wheelcarriers.

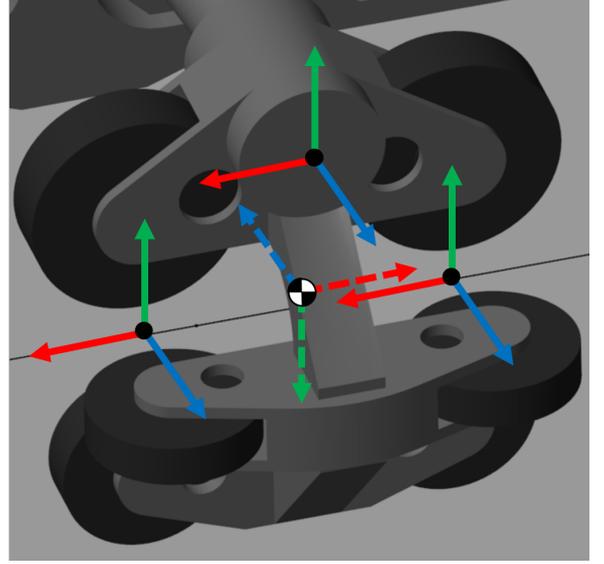


Figure J.2: The forces (solid line) and accelerations (dashed line) acting on a bogie have been decomposed in  $x$  (red),  $y$  (green), and  $z$ -direction (blue)

$$ma = -N_{\text{sum}} - F_{\text{hinge}} \quad \rightarrow \quad N_{\text{sum}} = -ma - F_{\text{hinge}} \quad (\text{J.2})$$

where  $m$  denotes the mass of the bogie including wheels,  $a$  represents the accelerations at the centre of mass,  $N_{\text{sum}}$  symbolizes the sum of normal forces measured at the two point-on-curve constraints for a certain wheel-type, and  $F_{\text{hinge}}$  denotes the forces exerted by the axle on the bogie at the hinge joint. As shown in Figure J.2, the normal forces in  $y$ -direction are assigned to the running wheels if  $N_{y,\text{sum}}$  acts in the negative direction with respect to the local reference frame. On the other hand, these normal forces are assigned to the upstop wheels if  $N_{y,\text{sum}}$  is positive, as shown in Equation J.3.

$$\begin{aligned} N_{y,\text{sum}} < 0 : N_{\text{running}} &= N_{y,\text{sum}} \\ &N_{\text{upstop}} = 0 \\ N_{y,\text{sum}} = 0 : N_{\text{running}} &= 0 \\ &N_{\text{upstop}} = 0 \\ N_{y,\text{sum}} > 0 : N_{\text{running}} &= 0 \\ &N_{\text{upstop}} = N_{y,\text{sum}} \end{aligned} \quad (\text{J.3})$$

Similar conditions can be formulated for the forces measured in the  $z$ -direction at either the front-left or rear-right bogie. The lateral normal forces measured at the front-left bogie can be assigned to the guide wheels on either the front-left or the front-right bogie, depending on the lateral acceleration of the bogie. The conditions formulated in Equation J.4 apply to the normal forces in  $z$ -direction measured at the front-left bogie.

$$\begin{aligned}
N_{z,\text{sum}} > 0 : N_{\text{FL,guide}} &= N_{z,\text{sum}} \\
N_{\text{FR,guide}} &= 0 \\
N_{z,\text{sum}} = 0 : N_{\text{FL,guide}} &= 0 \\
N_{\text{FR,guide}} &= 0 \\
N_{z,\text{sum}} < 0 : N_{\text{FL,guide}} &= 0 \\
N_{\text{FR,guide}} &= N_{z,\text{sum}}
\end{aligned} \tag{J.4}$$

where *FL* and *FR* denote respectively the front-left and front-right bogie. Similar conditions can be formulated for the forces sensed at the rear-right bogie, as shown in Equation J.5.

$$\begin{aligned}
N_{z,\text{sum}} > 0 : N_{\text{FL,guide}} &= -N_{z,\text{sum}} \\
N_{\text{FR,guide}} &= 0 \\
N_{z,\text{sum}} = 0 : N_{\text{FL,guide}} &= 0 \\
N_{\text{FR,guide}} &= 0 \\
N_{z,\text{sum}} < 0 : N_{\text{FL,guide}} &= 0 \\
N_{\text{FR,guide}} &= -N_{z,\text{sum}}
\end{aligned} \tag{J.5}$$

The exchange of forces between the wheels and the track is successfully validated if the calculated normal forces (Equation J.2) equal their measured equivalents (Equation J.3 till J.5) for the running, guide, and upstop wheels. The graphical results of the validation are presented in the next section for all four bogies and passenger occupancy configurations.

## J.3 Model Validation Graphs

### J.3.1 Forces at Pivot Joint

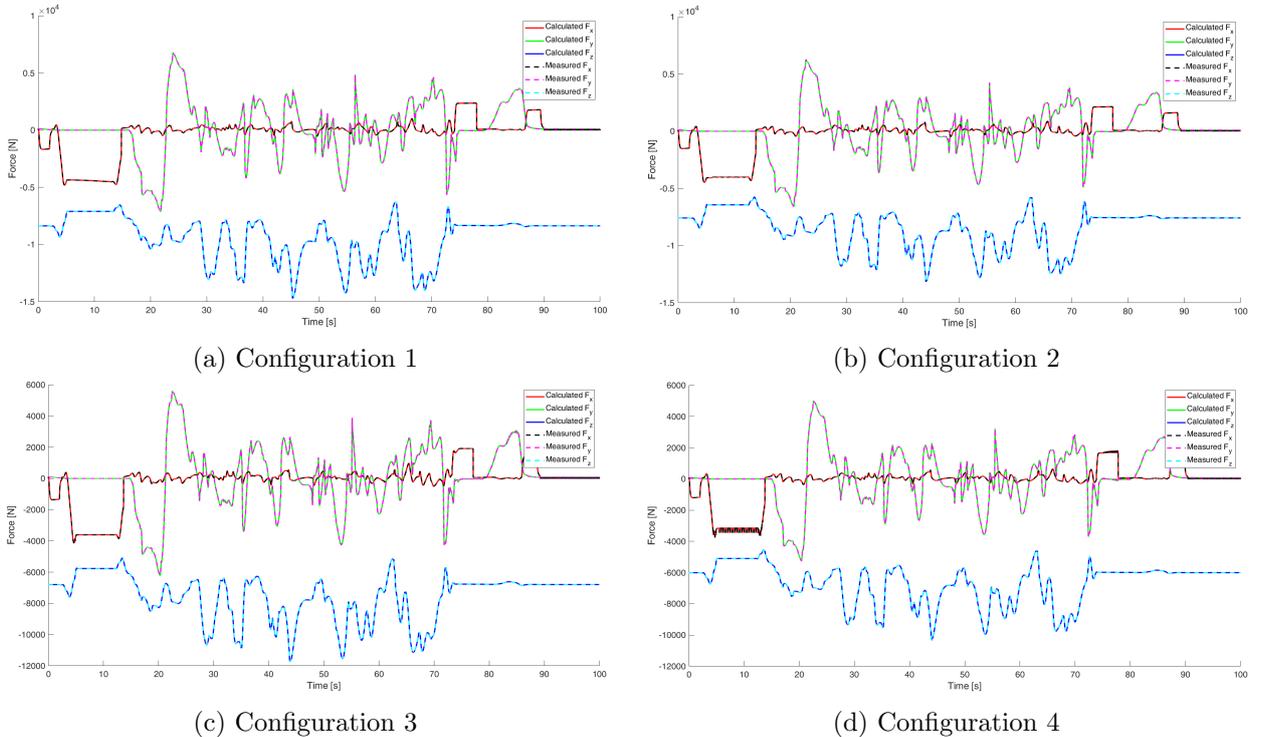
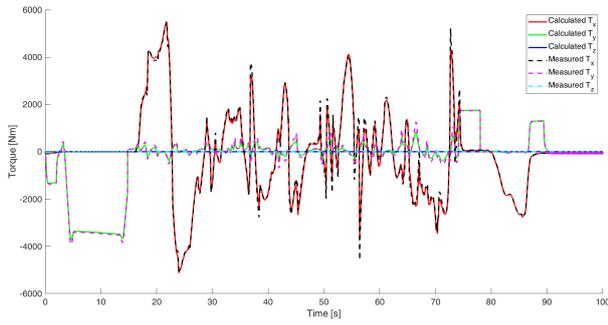
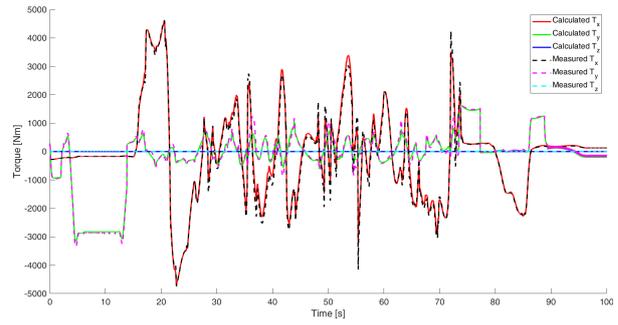


Figure J.3: Comparison between the measured and calculated forces at the pivot joint

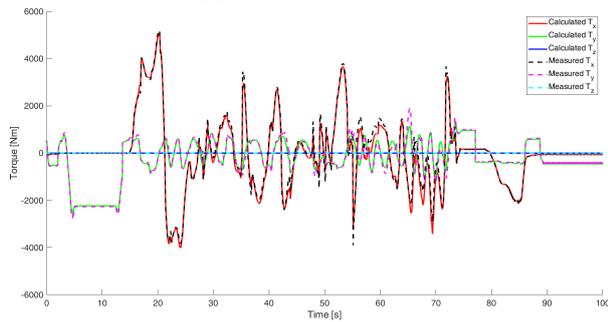
### J.3.2 Torques at Pivot Joint



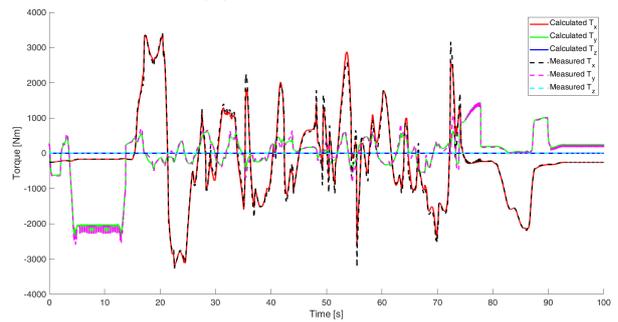
(a) Configuration 1



(b) Configuration 2



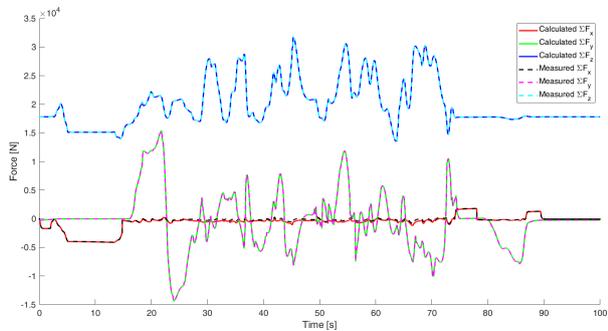
(c) Configuration 3



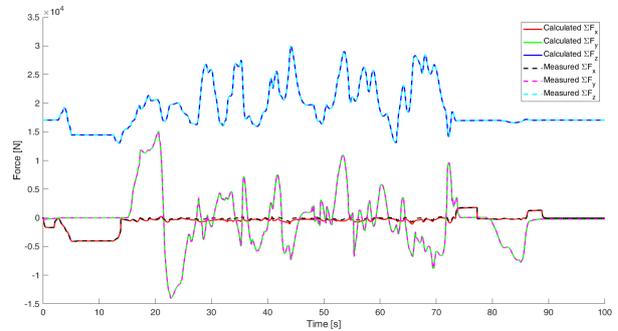
(d) Configuration 4

Figure J.4: Comparison between the measured and calculated torques at the pivot joint

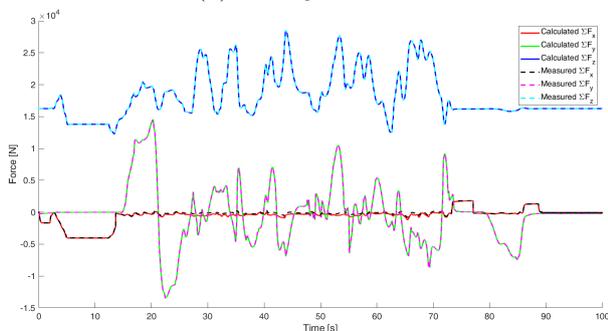
### J.3.3 Forces at Front and Rear Axle Joints



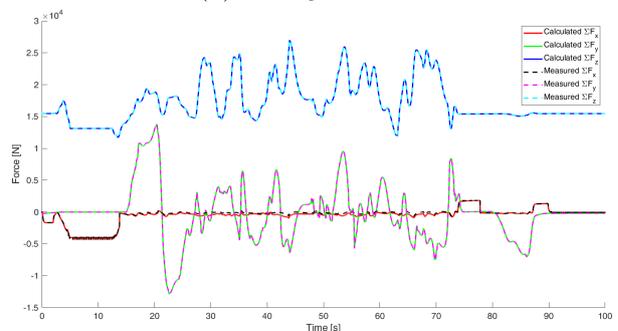
(a) Configuration 1



(b) Configuration 2



(c) Configuration 3



(d) Configuration 4

Figure J.5: Comparison between the measured and calculated forces at the front and rear axle joints

### J.3.4 Torques at Rear Axle Joint

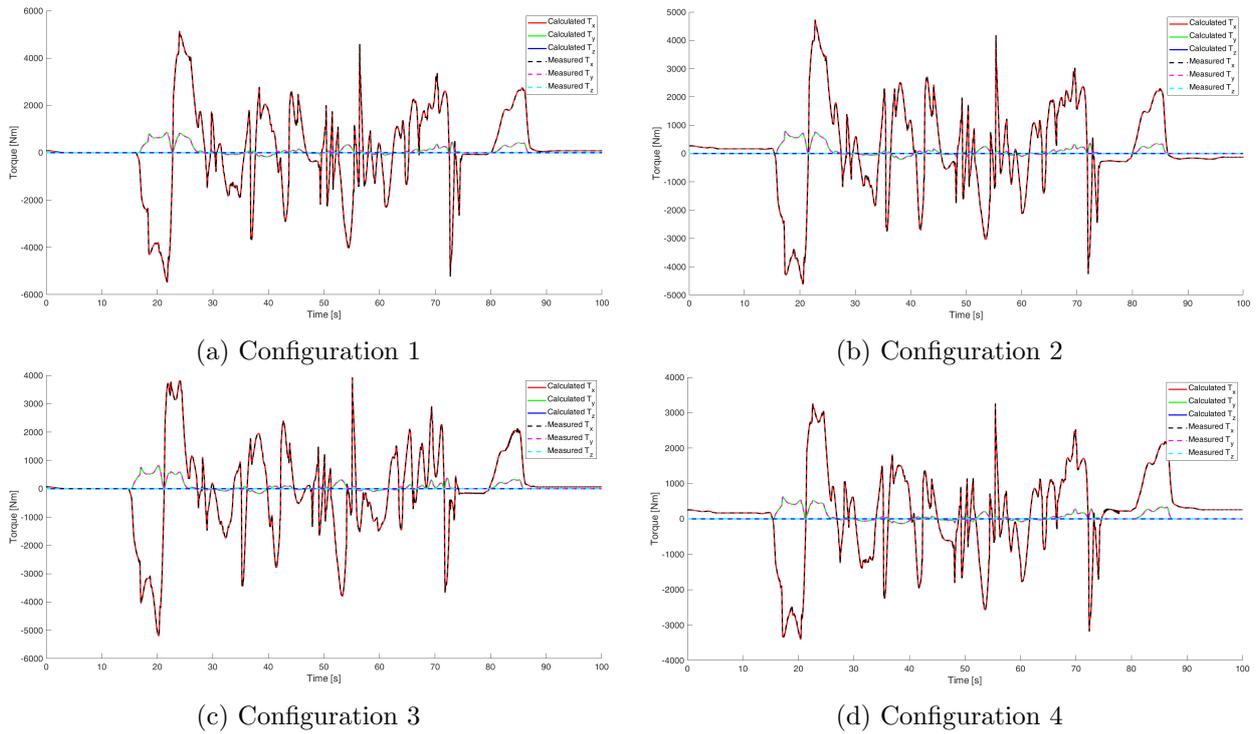


Figure J.6: Comparison between the measured and calculated torques at the rear axle joint

### J.3.5 Forces at Front Hinge Joints

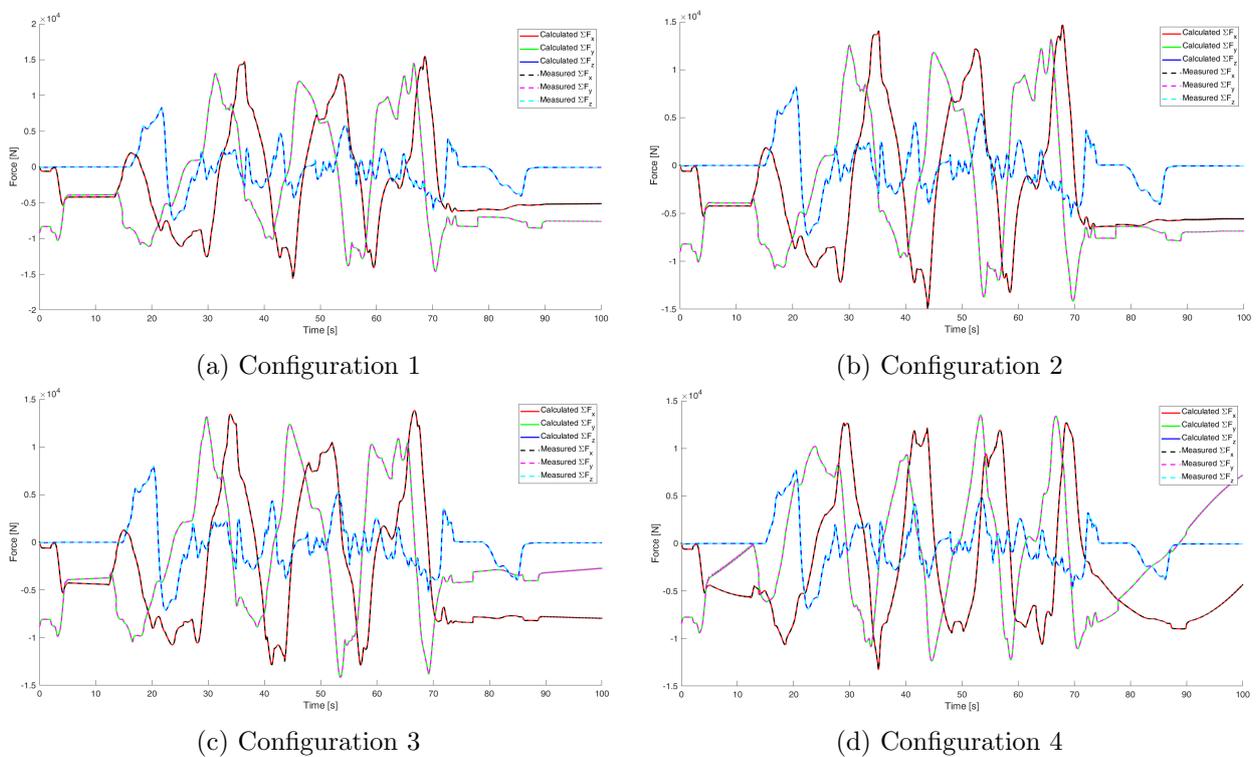


Figure J.7: Comparison between the measured and calculated hinge forces at the front bogies

### J.3.6 Forces at Rear Hinge Joints

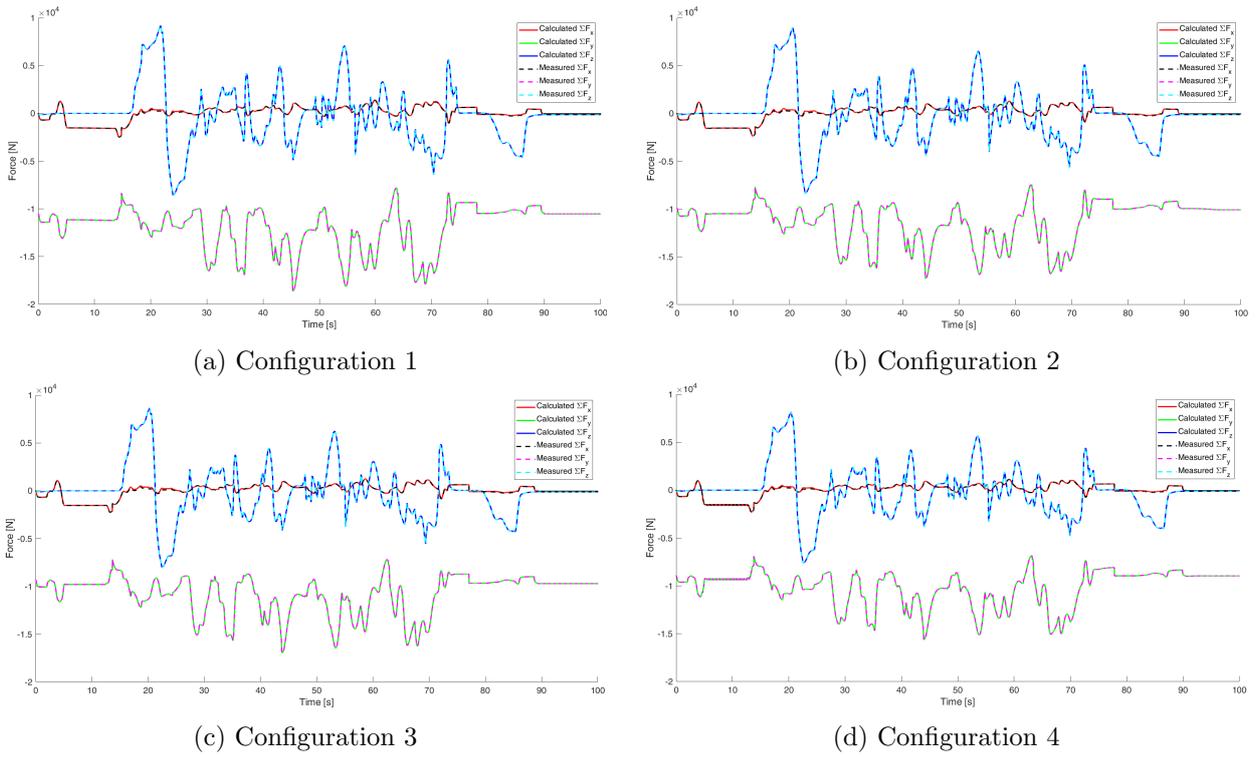


Figure J.8: Comparison between the measured and calculated hinge forces at the front bogies

### J.3.7 Forces between Track and Wheels at Front-Left Bogie

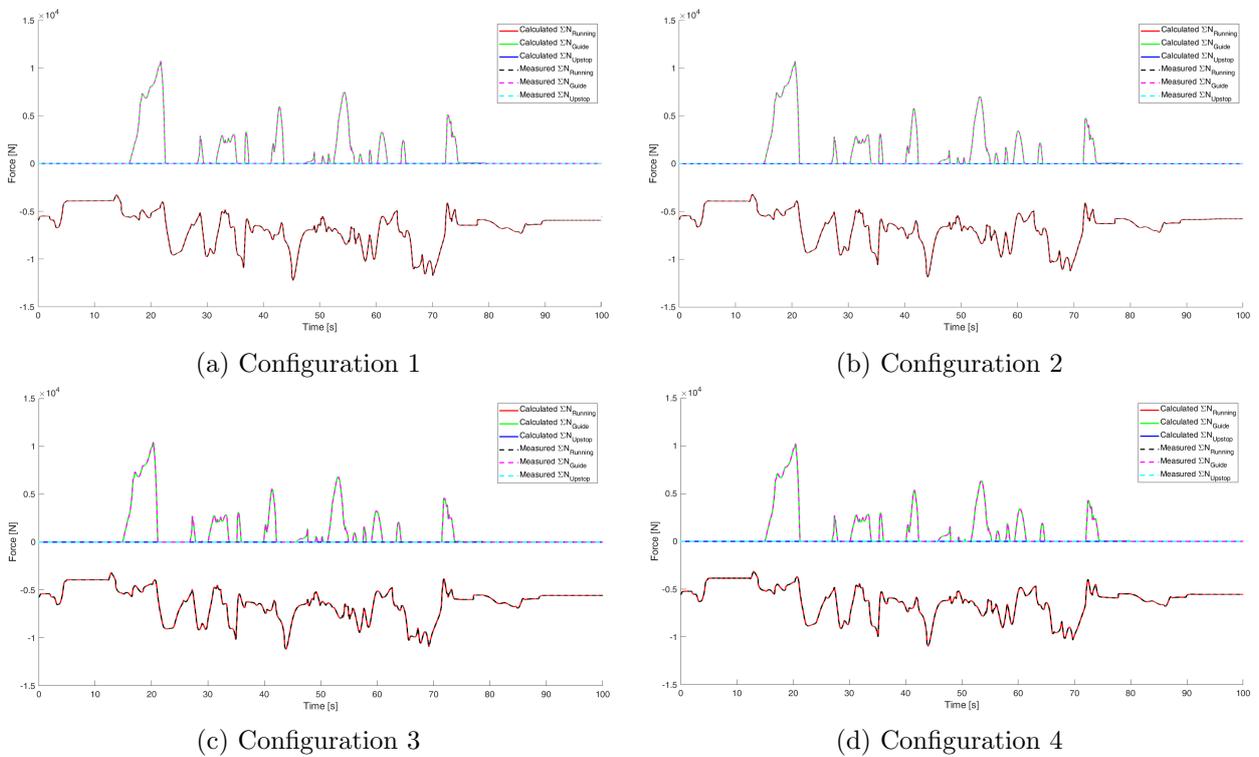


Figure J.9: Comparison between the measured and calculated track-wheel forces at the front-left bogie

### J.3.8 Forces between Track and Wheels at Front-Right Bogie

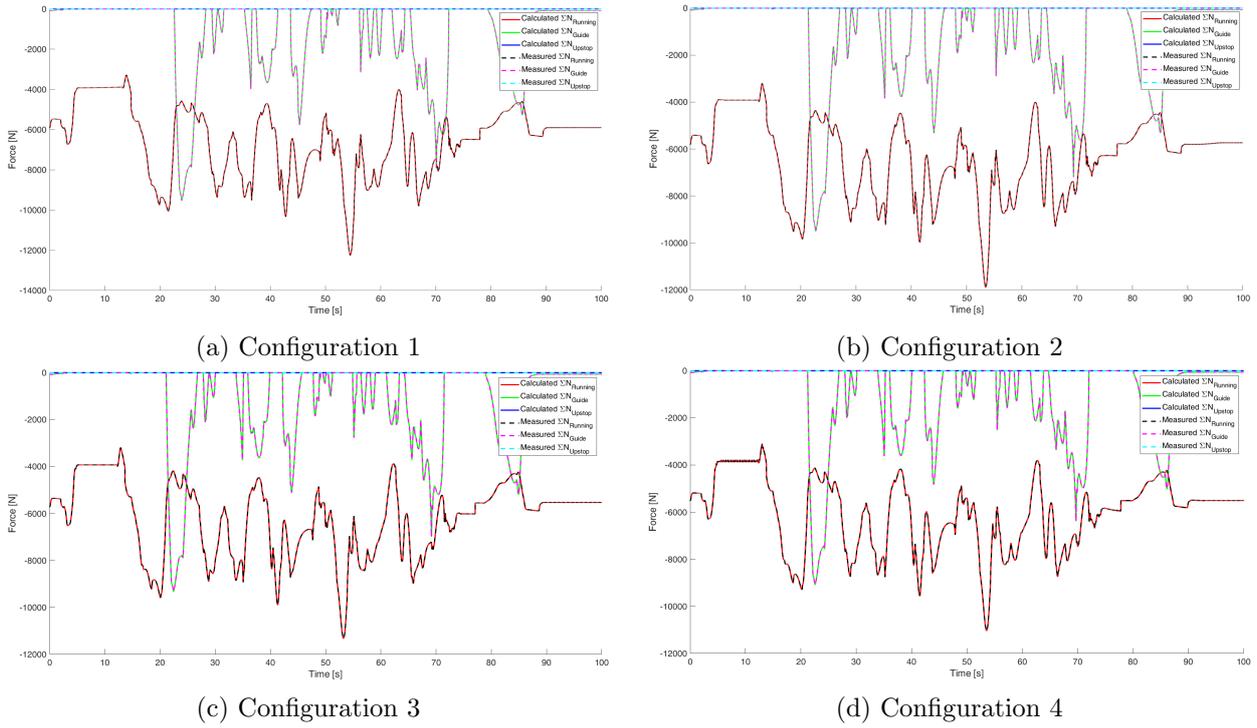


Figure J.10: Comparison between the measured and calculated track-wheel forces at the front-right bogie

### J.3.9 Forces between Track and Wheels at Rear-Left Bogie

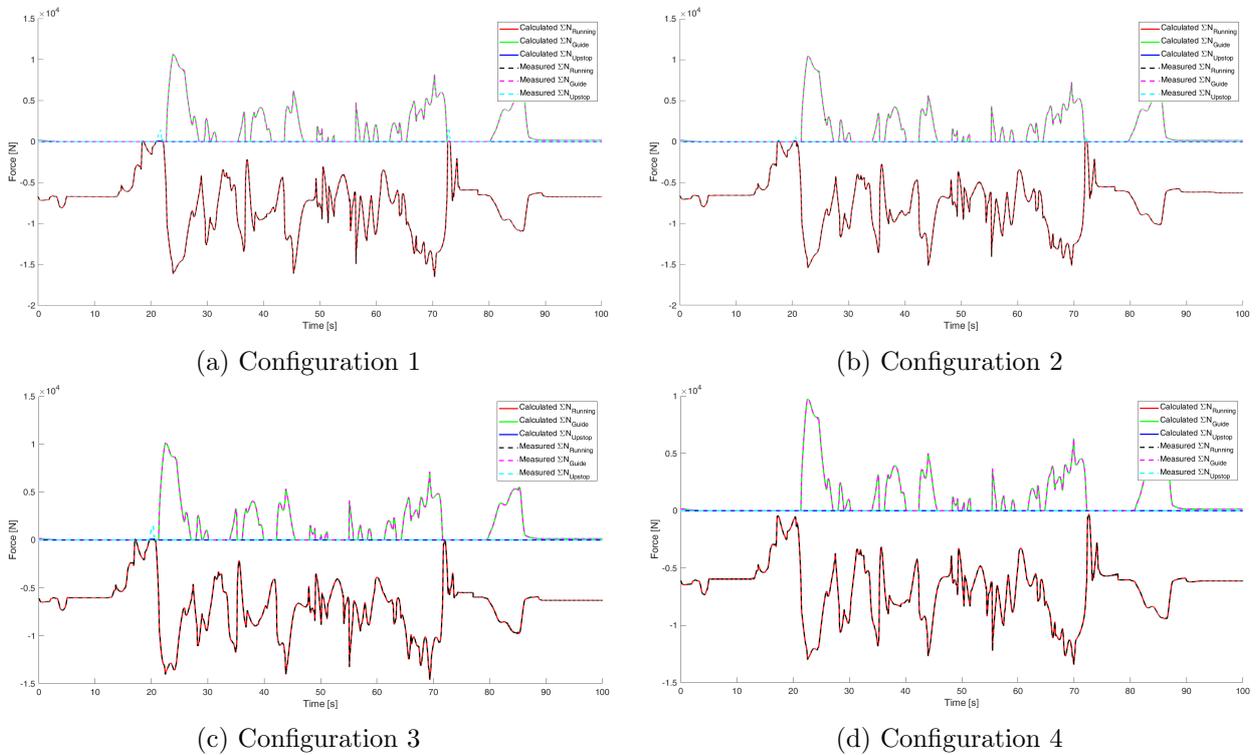


Figure J.11: Comparison between the measured and calculated track-wheel forces at the rear-left bogie

### J.3.10 Forces between Track and Wheels at Rear-Right Bogie

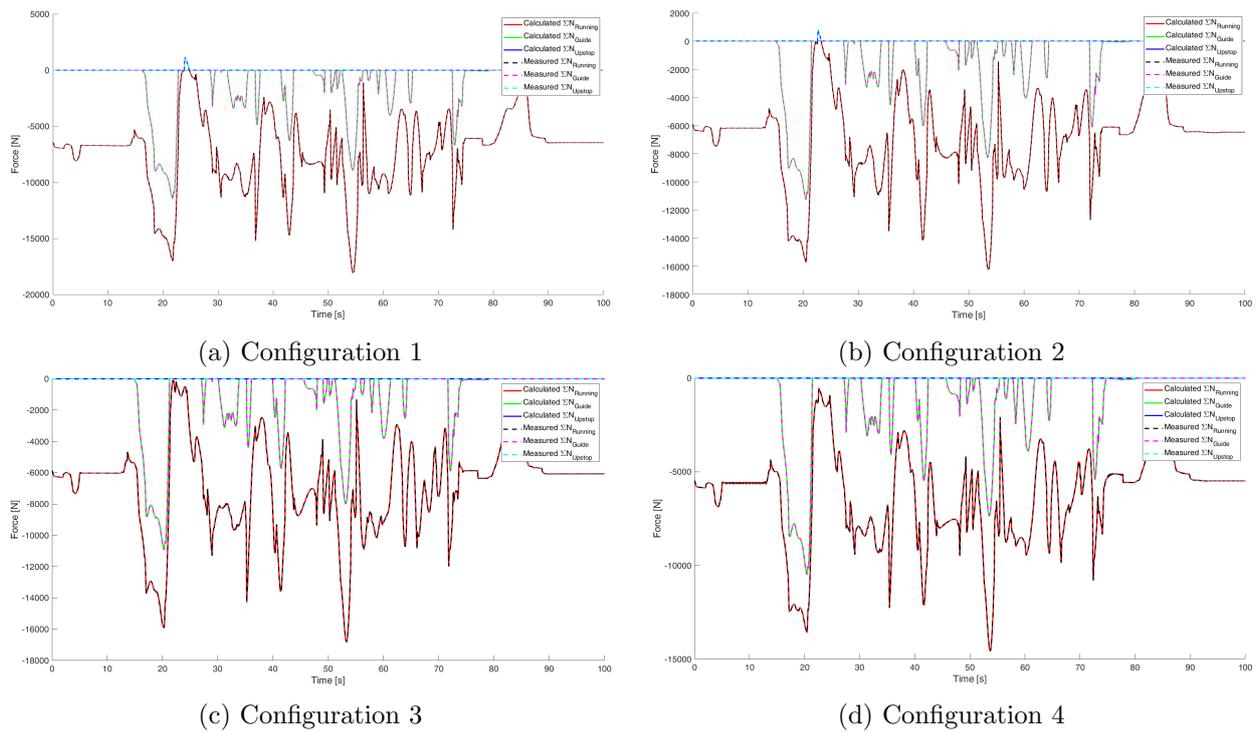


Figure J.12: Comparison between the measured and calculated track-wheel forces at the rear-left bogie

# K. Loadcases and Reaction Forces

## K.1 Loadcases Optimized Method

### K.1.1 Forces and Torques

| Loadcase | $F_x$ [N] | $F_y$ [N] | $F_z$ [N] | $T_x$ [Nm] | $T_z$ [Nm] |
|----------|-----------|-----------|-----------|------------|------------|
| 1        | -4607.46  | -8260.40  | -0.77     | -0.78      | 3777.74    |
| 2        | -4786.08  | -7949.48  | 0.47      | 0.67       | 3886.92    |
| 3        | -4731.13  | -6824.44  | -1.04     | -0.78      | 3775.43    |
| 4        | -78.00    | -10098.87 | 6871.55   | 5272.15    | -9.22      |
| 5        | -200.97   | -9989.89  | 7131.68   | 5486.47    | 156.24     |
| 6        | 396.73    | -8986.49  | -5522.78  | -4066.52   | -410.70    |
| 7        | -85.93    | -9676.18  | -6782.97  | -5136.38   | 29.71      |
| 8        | 207.81    | -14602.00 | -2733.30  | -1596.01   | -90.80     |
| 9        | -6.99     | -14518.65 | -3178.50  | -2145.92   | 118.68     |
| 10       | 93.22     | -14114.23 | 5328.74   | 3984.29    | 29.82      |
| 11       | 3.21      | -14241.42 | 4890.17   | 3632.41    | 149.99     |
| 12       | 1010.33   | -11608.46 | -2961.39  | -2154.59   | -1241.69   |
| 13       | 624.17    | -13463.67 | -2506.57  | -1647.40   | -803.77    |
| 14       | 695.85    | -6714.95  | 5028.64   | 4135.62    | -798.23    |
| 15       | 2324.48   | -8320.94  | 762.96    | 675.84     | -1721.81   |
| 16       | 2363.39   | -8333.26  | 38.73     | 72.62      | -1739.83   |
| 17       | -4319.25  | -6073.08  | -0.13     | -163.73    | 3243.01    |
| 18       | 366.97    | -8084.37  | -5074.93  | -3926.52   | -205.65    |
| 19       | -95.03    | -8794.89  | -6269.27  | -4731.46   | 377.35     |
| 20       | 439.65    | -6136.12  | 4540.96   | 3596.17    | -701.95    |
| 21       | 2035.85   | -7560.96  | 1742.80   | 1823.71    | -1589.58   |
| 22       | -3846.99  | -5423.25  | -1.39     | -1.05      | 2571.56    |
| 23       | -100.82   | -8184.03  | 6189.84   | 5202.74    | -3.93      |
| 24       | 221.42    | -7241.05  | -4781.31  | -3723.13   | -38.61     |
| 25       | -107.98   | -7900.55  | -5587.93  | -3738.40   | 484.69     |
| 26       | 981.40    | -9846.10  | -2205.05  | -1260.25   | -2105.14   |
| 27       | 312.47    | -5492.20  | 3857.39   | 2677.59    | -24.02     |
| 28       | 1986.15   | -6760.53  | 76.10     | 466.64     | -1621.55   |
| 29       | 1979.48   | -6764.14  | 39.13     | 364.03     | -1736.39   |
| 30       | -3393.39  | -4816.26  | 0.16      | -154.81    | 2376.16    |
| 31       | -2267.91  | -4517.69  | -1.73     | -147.98    | 1487.25    |
| 32       | 294.87    | -6261.93  | -3802.59  | -2635.62   | -68.07     |
| 33       | -57.03    | -6978.86  | -4914.48  | -3137.12   | 97.70      |
| 34       | 84.41     | -4702.45  | -889.64   | -820.65    | -120.57    |
| 35       | 174.42    | -4613.70  | -698.86   | -688.17    | -283.55    |
| 36       | 94.45     | -5167.34  | 3485.59   | 2834.88    | -368.83    |
| 37       | 530.39    | -5063.95  | 3199.96   | 2485.31    | -970.04    |
| 38       | 1646.04   | -5975.11  | 618.07    | 567.14     | -900.59    |
| 39       | 1755.16   | -5982.89  | -80.35    | -296.12    | -1094.37   |

Table K.1: The forces and torques exerted at the pivot joint, which belong to the loadcases of the optimized method with respect to the reference frame used in Ansys

## K.1.2 Accelerations

| Loadcase | $a_x$ [m/s <sup>2</sup> ] | $a_y$ [m/s <sup>2</sup> ] | $a_z$ [m/s <sup>2</sup> ] | $\omega_x$ [rad/s <sup>2</sup> ] | $\omega_y$ [rad/s <sup>2</sup> ] | $\omega_z$ [rad/s <sup>2</sup> ] |
|----------|---------------------------|---------------------------|---------------------------|----------------------------------|----------------------------------|----------------------------------|
| 1        | -4.80                     | -9.45                     | 0.00                      | 0.00                             | 0.00                             | 0.74                             |
| 2        | -5.12                     | -9.07                     | 0.00                      | 0.00                             | 0.00                             | 0.63                             |
| 3        | -5.26                     | -7.85                     | 0.00                      | 0.00                             | 0.00                             | 0.37                             |
| 4        | 0.46                      | -11.91                    | 8.30                      | 0.03                             | -0.10                            | -0.03                            |
| 5        | 0.48                      | -11.69                    | 8.58                      | 0.03                             | -0.06                            | 0.19                             |
| 6        | 0.40                      | -10.64                    | -7.11                     | 0.26                             | 0.87                             | -0.36                            |
| 7        | 0.44                      | -11.36                    | -7.89                     | -0.33                            | 0.02                             | 0.02                             |
| 8        | 0.52                      | -17.63                    | -4.73                     | 1.28                             | 0.16                             | -0.25                            |
| 9        | 0.53                      | -17.48                    | -4.31                     | 0.19                             | -0.07                            | 0.02                             |
| 10       | 0.45                      | -16.84                    | 6.72                      | -0.08                            | 0.18                             | -0.02                            |
| 11       | 0.45                      | -16.91                    | 6.13                      | -0.13                            | 0.36                             | 0.15                             |
| 12       | 0.18                      | -14.53                    | -4.57                     | 0.58                             | 0.80                             | -1.04                            |
| 13       | 0.33                      | -16.44                    | -4.43                     | 1.17                             | 0.18                             | -0.52                            |
| 14       | 0.29                      | -8.42                     | 5.55                      | 0.81                             | -1.08                            | -0.35                            |
| 15       | 2.72                      | -9.76                     | 0.64                      | 0.16                             | 0.32                             | -0.01                            |
| 16       | 2.77                      | -9.77                     | 0.05                      | 0.00                             | 0.00                             | 0.00                             |
| 17       | -5.07                     | -7.62                     | 0.00                      | 0.00                             | 0.00                             | 0.68                             |
| 18       | 0.42                      | -10.60                    | -7.19                     | 0.22                             | 0.92                             | -0.37                            |
| 19       | 0.44                      | -11.40                    | -8.08                     | -0.34                            | 0.02                             | 0.02                             |
| 20       | 0.33                      | -8.31                     | 5.31                      | 1.27                             | -1.20                            | -0.06                            |
| 21       | 2.65                      | -9.75                     | 1.06                      | 1.34                             | 0.44                             | 0.00                             |
| 22       | -5.14                     | -7.65                     | 0.00                      | 0.00                             | 0.00                             | 0.57                             |
| 23       | 0.43                      | -11.73                    | 8.64                      | 0.02                             | -0.05                            | 0.17                             |
| 24       | 0.51                      | -10.67                    | -7.40                     | 0.27                             | 0.88                             | -0.37                            |
| 25       | 0.43                      | -11.41                    | -8.16                     | -0.32                            | 0.02                             | 0.01                             |
| 26       | 0.13                      | -14.73                    | -4.31                     | 0.68                             | 0.71                             | -1.02                            |
| 27       | 0.34                      | -8.27                     | 5.17                      | 1.57                             | -1.31                            | 0.15                             |
| 28       | 2.76                      | -9.76                     | 0.02                      | -0.01                            | 0.00                             | 0.00                             |
| 29       | 2.76                      | -9.77                     | 0.03                      | -0.01                            | 0.00                             | 0.00                             |
| 30       | -5.00                     | -7.61                     | 0.00                      | 0.00                             | 0.00                             | 0.75                             |
| 31       | -3.54                     | -7.45                     | 0.00                      | 0.00                             | 0.00                             | 0.15                             |
| 32       | 0.43                      | -10.37                    | -6.74                     | 0.04                             | 1.03                             | -0.34                            |
| 33       | 0.41                      | -11.40                    | -8.11                     | -0.14                            | -0.01                            | -0.01                            |
| 34       | 0.17                      | -7.84                     | -1.15                     | -0.09                            | -0.53                            | 0.03                             |
| 35       | 0.14                      | -7.71                     | -0.85                     | -0.12                            | -0.46                            | -0.16                            |
| 36       | 0.38                      | -8.38                     | 4.75                      | 2.09                             | -1.42                            | 0.49                             |
| 37       | 0.24                      | -8.67                     | 5.15                      | 0.44                             | -0.89                            | -0.46                            |
| 38       | 2.73                      | -9.76                     | 0.66                      | 0.32                             | 0.31                             | -0.01                            |
| 39       | 2.71                      | -9.77                     | 0.03                      | -0.01                            | 0.00                             | 0.00                             |

Table K.2: The linear and angular accelerations at the centre of mass of the main chassis beam, which belong to the loadcases of the optimized method with respect to the reference frame used in Ansys

## K.2 Torques at Rear Revolute Joint

| Loadcase | $T_z$ [Nm] |
|----------|------------|
| 1        | -974.89    |
| 2        | -974.89    |
| 3        | 1041.33    |
| 4        | 1041.33    |
| 5        | -974.89    |
| 6        | -974.89    |
| 7        | -974.89    |
| 8        | -974.89    |
| 9        | 1041.33    |
| 10       | 1041.33    |
| 11       | 1041.33    |
| 12       | 1041.33    |

| Loadcase | $T_z$ [Nm] |
|----------|------------|
| 1        | 0.00       |
| 2        | 0.00       |
| 3        | 0.00       |
| 4        | -817.48    |
| 5        | -861.61    |
| 6        | -578.56    |
| 7        | -832.60    |
| 8        | -112.48    |
| 9        | -151.19    |
| 10       | -325.87    |
| 11       | -267.54    |
| 12       | -149.95    |
| 13       | -129.82    |
| 14       | -446.04    |
| 15       | -8.74      |
| 16       | 0.01       |
| 17       | 0.02       |
| 18       | -548.04    |
| 19       | -765.59    |
| 20       | -335.95    |
| 21       | -29.63     |
| 22       | 0.00       |
| 23       | -817.31    |
| 24       | -530.34    |
| 25       | -606.53    |
| 26       | -90.40     |
| 27       | -218.14    |
| 28       | 0.04       |
| 29       | 0.03       |
| 30       | 0.01       |
| 31       | 0.00       |
| 32       | -334.94    |
| 33       | -513.14    |
| 34       | 10.23      |
| 35       | 18.29      |
| 36       | -164.97    |
| 37       | -286.01    |
| 38       | -7.63      |
| 39       | -0.03      |

Table K.3: Torques exerted at the rear pilot node for the conventional (left) and optimized (right) method

### K.3 Comparison between Reaction Forces

| Loadcase | $F_{x,total}$ [N] |         |          | $F_{y,front}$ [N] |         |         |
|----------|-------------------|---------|----------|-------------------|---------|---------|
|          | Simulink          | Ansys   | RPD      | Simulink          | Ansys   | RPD     |
| 1        | -2520.2           | -2663.6 | 5.53 %   | 5528.5            | 4142.1  | 28.67 % |
| 2        | -3215.9           | -3368.1 | 4.62 %   | 5139.1            | 5628.3  | 9.09 %  |
| 3        | -3889.2           | -4044.6 | 3.92 %   | 4172.9            | 4729.5  | 12.51 % |
| 4        | -217.1            | -345.9  | 45.76 %  | 10450.8           | 10080.0 | 3.61 %  |
| 5        | -102.9            | -246.4  | 82.15 %  | 10157.5           | 9806.4  | 3.52 %  |
| 6        | -649.9            | -768.1  | 16.67 %  | 9461.2            | 9207.7  | 2.72 %  |
| 7        | -184.3            | -318.4  | 53.37 %  | 9960.9            | 9624.7  | 3.43 %  |
| 8        | -384.6            | -691.0  | 56.97 %  | 14963.2           | 14422.0 | 3.68 %  |
| 9        | -159.6            | -490.0  | 101.74 % | 14705.8           | 14155.0 | 3.82 %  |
| 10       | -284.6            | -512.6  | 57.22 %  | 14273.1           | 13814.0 | 3.27 %  |
| 11       | -182.5            | -416.5  | 78.15 %  | 14231.0           | 13765.0 | 3.33 %  |
| 12       | -875.0            | -1177.1 | 29.45 %  | 12810.8           | 12519.0 | 2.30 %  |
| 13       | -589.1            | -927.3  | 44.60 %  | 14143.2           | 13800.0 | 2.46 %  |
| 14       | -821.4            | -967.0  | 16.29 %  | 7431.2            | 7355.5  | 1.02 %  |
| 15       | 1751.8            | 1693.4  | 3.39 %   | 9402.5            | 8725.1  | 7.47 %  |
| 16       | 1823.6            | 1892.9  | 3.73 %   | 9431.1            | 8737.2  | 7.64 %  |
| 17       | -3731.2           | -3884.4 | 4.02 %   | 3920.1            | 4455.1  | 12.78 % |
| 18       | -641.9            | -757.4  | 16.49 %  | 8998.5            | 8747.8  | 2.83 %  |
| 19       | -181.8            | -309.4  | 51.96 %  | 9467.0            | 9127.9  | 3.65 %  |
| 20       | -651.9            | -747.4  | 13.65 %  | 7003.6            | 6897.7  | 1.52 %  |
| 21       | 1726.0            | 1648.3  | 4.61 %   | 9035.6            | 8378.2  | 7.55 %  |
| 22       | -3746.0           | -3902.5 | 4.09 %   | 4042.7            | 4549.5  | 11.80 % |
| 23       | -167.5            | -300.4  | 56.80 %  | 9550.4            | 9196.6  | 3.77 %  |
| 24       | -570.6            | -694.3  | 19.56 %  | 8637.4            | 8385.3  | 2.96 %  |
| 25       | -186.0            | -293.8  | 44.94 %  | 9018.8            | 8683.1  | 3.79 %  |
| 26       | -871.9            | -1105.1 | 23.60 %  | 12588.2           | 12287.0 | 2.42 %  |
| 27       | -493.3            | -632.2  | 24.69 %  | 6350.0            | 6223.7  | 2.01 %  |
| 28       | 1809.0            | 1779.5  | 1.64 %   | 8750.1            | 8085.1  | 7.90 %  |
| 29       | 1811.7            | 1801.9  | 0.54 %   | 8805.5            | 8139.3  | 7.86 %  |
| 30       | -3713.8           | -3865.3 | 4.00 %   | 3844.9            | 4325.5  | 11.76 % |
| 31       | -3030.0           | -3144.2 | 3.70 %   | 4436.2            | 4708.4  | 5.95 %  |
| 32       | -571.3            | -689.6  | 18.76 %  | 8028.9            | 7782.4  | 3.12 %  |
| 33       | -216.5            | -320.4  | 38.68 %  | 8786.5            | 8460.7  | 3.78 %  |
| 34       | -65.0             | -244.1  | 115.90 % | 5936.4            | 5696.8  | 4.12 %  |
| 35       | -138.6            | -301.1  | 73.94 %  | 5953.5            | 5727.4  | 3.87 %  |
| 36       | -327.2            | -449.7  | 31.53 %  | 6338.8            | 6171.0  | 2.68 %  |
| 37       | -622.2            | -759.2  | 19.83 %  | 6979.0            | 6894.9  | 1.21 %  |
| 38       | 1793.4            | 1761.4  | 1.80 %   | 8124.3            | 7480.9  | 8.25 %  |
| 39       | 1792.9            | 1805.1  | 0.68 %   | 8218.5            | 7567.3  | 8.25 %  |

Table K.4: Comparison between the reaction forces from the kinematic model in Simulink and the finite element model in Ansys for  $F_{x,total}$  and  $F_{y,front}$

| Loadcase | $F_{z,front}$ [N] |         |          | $F_{y,rear}$ [N] |         |        |
|----------|-------------------|---------|----------|------------------|---------|--------|
|          | Simulink          | Ansys   | RPD      | Simulink         | Ansys   | RPD    |
| 1        | -1.1              | 0.4     | 200.00 % | 11763.2          | 12932.0 | 9.47 % |
| 2        | 2.2               | 0.6     | 113.56 % | 11479.9          | 10779.0 | 6.30 % |
| 3        | 2.1               | 0.8     | 90.32 %  | 10153.0          | 9406.3  | 7.63 % |
| 4        | -7004.8           | -6742.2 | 3.82 %   | 11025.6          | 11085.0 | 0.54 % |
| 5        | -7234.4           | -6964.6 | 3.80 %   | 10998.7          | 11053.0 | 0.49 % |
| 6        | 6161.5            | 5889.5  | 4.51 %   | 9685.7           | 9647.7  | 0.39 % |
| 7        | 6726.7            | 6477.8  | 3.77 %   | 10568.2          | 10610.0 | 0.39 % |
| 8        | 3594.1            | 3426.3  | 4.78 %   | 16482.1          | 16553.0 | 0.43 % |
| 9        | 3429.3            | 3310.3  | 3.53 %   | 16510.7          | 16606.0 | 0.58 % |
| 10       | -5530.9           | -5285.5 | 4.54 %   | 15933.1          | 15953.0 | 0.12 % |
| 11       | -4979.2           | -4767.4 | 4.35 %   | 16167.9          | 16200.0 | 0.20 % |
| 12       | 3952.5            | 3637.4  | 8.30 %   | 12678.2          | 12550.0 | 1.02 % |
| 13       | 3498.7            | 3197.3  | 9.00 %   | 15023.0          | 14916.0 | 0.71 % |
| 14       | -5168.0           | -5001.0 | 3.28 %   | 7331.5           | 7173.2  | 2.18 % |
| 15       | -511.8            | -505.1  | 1.31 %   | 8240.6           | 8663.2  | 5.00 % |
| 16       | -37.1             | -37.2   | 0.09 %   | 8236.0           | 8675.5  | 5.20 % |
| 17       | 1.9               | 0.7     | 96.55 %  | 9430.2           | 8723.4  | 7.79 % |
| 18       | 6052.3            | 5776.0  | 4.67 %   | 9209.8           | 9169.8  | 0.43 % |
| 19       | 6632.8            | 6378.5  | 3.91 %   | 10214.6          | 10258.0 | 0.42 % |
| 20       | -4901.8           | -4734.4 | 3.47 %   | 7070.9           | 6956.3  | 1.63 % |
| 21       | -1054.9           | -1041.4 | 1.29 %   | 7836.2           | 8238.1  | 5.00 % |
| 22       | 2.3               | 1.2     | 62.69 %  | 8687.2           | 8003.3  | 8.20 % |
| 23       | -6899.3           | -6631.8 | 3.95 %   | 9836.2           | 9891.3  | 0.56 % |
| 24       | 6039.9            | 5758.0  | 4.78 %   | 8797.2           | 8756.5  | 0.46 % |
| 25       | 6417.9            | 6160.4  | 4.09 %   | 9780.6           | 9820.3  | 0.40 % |
| 26       | 3504.2            | 3196.9  | 9.17 %   | 11331.7          | 11208.0 | 1.10 % |
| 27       | -4614.7           | -4443.4 | 3.78 %   | 7042.3           | 6957.1  | 1.22 % |
| 28       | -39.1             | -39.7   | 1.57 %   | 7337.0           | 7747.8  | 5.45 % |
| 29       | -28.4             | -28.8   | 1.33 %   | 7290.6           | 7702.4  | 5.49 % |
| 30       | 1.3               | 0.3     | 117.75 % | 8244.2           | 7594.5  | 8.20 % |
| 31       | 0.9               | 1.7     | 60.75 %  | 7197.3           | 6736.7  | 6.61 % |
| 32       | 5371.4            | 5106.9  | 5.05 %   | 8136.8           | 8099.9  | 0.45 % |
| 33       | 6125.7            | 5869.2  | 4.28 %   | 9081.2           | 9110.1  | 0.32 % |
| 34       | 774.2             | 734.6   | 5.25 %   | 6255.4           | 6292.0  | 0.58 % |
| 35       | 567.4             | 538.3   | 5.27 %   | 6027.0           | 6046.1  | 0.32 % |
| 36       | -4309.9           | -4127.8 | 4.32 %   | 6838.8           | 6804.5  | 0.50 % |
| 37       | -4199.4           | -4045.6 | 3.73 %   | 6369.9           | 6209.5  | 2.55 % |
| 38       | -464.1            | -457.3  | 1.48 %   | 7172.5           | 7561.0  | 5.27 % |
| 39       | 13.0              | 15.0    | 14.80 %  | 7098.0           | 7494.8  | 5.44 % |

Table K.5: Comparison between the reaction forces from the kinematic model in Simulink and the finite element model in Ansys for  $F_{z,front}$  and  $F_{y,rear}$

| Loadcase | $F_{z, \text{rear}}$ [N] |         |          | $T_{x, \text{rear}}$ [Nm] |         |        |
|----------|--------------------------|---------|----------|---------------------------|---------|--------|
|          | Simulink                 | Ansys   | RPD      | Simulink                  | Ansys   | RPD    |
| 1        | 2.3                      | 0.8     | 99.53 %  | 0.8                       | 0.8     | 0.69 % |
| 2        | -1.6                     | -0.1    | 167.12 % | -0.7                      | -0.7    | 2.06 % |
| 3        | 0.1                      | 1.4     | 179.27 % | 0.8                       | 0.8     | 1.64 % |
| 4        | -7797.2                  | -7839.9 | 0.55 %   | -5236.3                   | -5235.0 | 0.03 % |
| 5        | -8094.4                  | -8138.7 | 0.55 %   | -5449.5                   | -5448.0 | 0.03 % |
| 6        | 6153.4                   | 6206.2  | 0.85 %   | 4031.8                    | 4030.1  | 0.04 % |
| 7        | 7593.2                   | 7636.4  | 0.57 %   | 5105.5                    | 5104.1  | 0.03 % |
| 8        | 3659.6                   | 3696.8  | 1.01 %   | 1562.0                    | 1562.4  | 0.03 % |
| 9        | 3863.9                   | 3873.4  | 0.24 %   | 2125.2                    | 2125.6  | 0.02 % |
| 10       | -6222.0                  | -6299.4 | 1.24 %   | -3954.3                   | -3954.3 | 0.00 % |
| 11       | -5765.4                  | -5831.4 | 1.14 %   | -3604.8                   | -3604.8 | 0.00 % |
| 12       | 3371.9                   | 3536.6  | 4.77 %   | 2127.0                    | 2127.1  | 0.00 % |
| 13       | 3241.2                   | 3419.3  | 5.35 %   | 1615.3                    | 1616.4  | 0.07 % |
| 14       | -5166.5                  | -5147.9 | 0.36 %   | -4118.4                   | -4116.0 | 0.06 % |
| 15       | -867.1                   | -869.6  | 0.28 %   | -675.2                    | -675.2  | 0.01 % |
| 16       | -44.9                    | -43.7   | 2.76 %   | -72.4                     | -72.4   | 0.03 % |
| 17       | -1.2                     | 0.0     | 200.00 % | 163.7                     | 163.7   | 0.01 % |
| 18       | 5892.8                   | 5945.9  | 0.90 %   | 3893.1                    | 3891.3  | 0.05 % |
| 19       | 7358.4                   | 7402.0  | 0.59 %   | 4701.6                    | 4700.2  | 0.03 % |
| 20       | -4710.8                  | -4694.5 | 0.35 %   | -3585.8                   | -3583.4 | 0.07 % |
| 21       | -1697.3                  | -1701.7 | 0.26 %   | -1833.9                   | -1833.7 | 0.01 % |
| 22       | 0.5                      | 1.6     | 100.72 % | 1.0                       | 1.0     | 1.03 % |
| 23       | -7541.1                  | -7581.8 | 0.54 %   | -5168.0                   | -5166.5 | 0.03 % |
| 24       | 5814.2                   | 5868.7  | 0.93 %   | 3689.4                    | 3687.7  | 0.05 % |
| 25       | 6969.6                   | 7014.2  | 0.64 %   | 3709.8                    | 3708.4  | 0.04 % |
| 26       | 2820.7                   | 2987.4  | 5.74 %   | 1234.0                    | 1234.4  | 0.03 % |
| 27       | -4184.2                  | -4171.4 | 0.31 %   | -2672.4                   | -2670.0 | 0.09 % |
| 28       | -58.7                    | -57.5   | 2.01 %   | -466.3                    | -466.3  | 0.00 % |
| 29       | -37.7                    | -36.6   | 3.03 %   | -363.8                    | -363.8  | 0.01 % |
| 30       | -1.6                     | -0.6    | 86.59 %  | 154.8                     | 154.8   | 0.01 % |
| 31       | 2.7                      | 1.9     | 36.46 %  | 148.0                     | 148.0   | 0.00 % |
| 32       | 4872.4                   | 4921.2  | 1.00 %   | 2608.3                    | 2606.5  | 0.07 % |
| 33       | 6535.7                   | 6581.4  | 0.70 %   | 3108.4                    | 3107.0  | 0.04 % |
| 34       | 1211.6                   | 1241.9  | 2.47 %   | 818.1                     | 818.2   | 0.01 % |
| 35       | 947.5                    | 972.4   | 2.60 %   | 686.9                     | 687.0   | 0.02 % |
| 36       | -3711.5                  | -3716.8 | 0.14 %   | -2837.3                   | -2835.1 | 0.08 % |
| 37       | -3917.0                  | -3902.5 | 0.37 %   | -2469.7                   | -2467.6 | 0.09 % |
| 38       | -781.0                   | -783.3  | 0.29 %   | -568.7                    | -568.7  | 0.01 % |
| 39       | 37.2                     | 36.0    | 3.44 %   | 296.1                     | 296.1   | 0.00 % |

Table K.6: Comparison between the reaction forces from the kinematic model in Simulink and the finite element model in Ansys for  $F_{z, \text{rear}}$  and  $T_{x, \text{rear}}$

# L. Ansys Command Lines

```
1 clear all, clc
2
3 % Specify the number of conventional loadcases:
4 n_ConventionalLC = 12;
5 % Specify the number of optimized loadcases:
6 n_OptimizedLC = 39;
7 % Specify the gravitational acceleration constant (in m/s^2):
8 g = 9.80665;
9
10 % Define the files to which the command lines will be written:
11 fileID1 = fopen('AnsysStatic.C.txt','w');
12 fileID2 = fopen('AnsysStatic.O.txt','w');
13
14 % Open the files that list the corresponding nodes for each area:
15 fileID3 = fopen('A1Nodes.txt');
16 A1Nodes = textscan(fileID3,'%f %f %f %f');
17 A1Nodes = A1Nodes{1,1}(:);
18 fclose(fileID3);
19 fileID4 = fopen('A2Nodes.txt');
20 A2Nodes = textscan(fileID4,'%f %f %f %f');
21 A2Nodes = A2Nodes{1,1}(:);
22 fclose(fileID4);
23 fileID5 = fopen('A3Nodes.txt');
24 A3Nodes = textscan(fileID5,'%f %f %f %f');
25 A3Nodes = A3Nodes{1,1}(:);
26 fclose(fileID5);
27 fileID6 = fopen('A4Nodes.txt');
28 A4Nodes = textscan(fileID6,'%f %f %f %f');
29 A4Nodes = A4Nodes{1,1}(:);
30 fclose(fileID6);
31 fileID7 = fopen('A96Nodes.txt');
32 A96Nodes = textscan(fileID7,'%f %f %f %f');
33 A96Nodes = A96Nodes{1,1}(:);
34 fclose(fileID7);
35 fileID8 = fopen('A97Nodes.txt');
36 A97Nodes = textscan(fileID8,'%f %f %f %f');
37 A97Nodes = A97Nodes{1,1}(:);
38 fclose(fileID8);
39 fileID9 = fopen('A98Nodes.txt');
40 A98Nodes = textscan(fileID9,'%f %f %f %f');
41 A98Nodes = A98Nodes{1,1}(:);
42 fclose(fileID9);
43 fileID10 = fopen('A99Nodes.txt');
44 A99Nodes = textscan(fileID10,'%f %f %f %f');
45 A99Nodes = A99Nodes{1,1}(:);
46 fclose(fileID10);
47
48 %% Enter the Structural Pre-Processor Phase
49
50 % Import the geometry:
51 fprintf(fileID1,'FINI\r\n/CLEAR,NOSTART\r\n-PARAIN,MainBeam4,x-t,,SOLIDS,0,0\r\n/);
52 fprintf(fileID1,'FACET,NORML\r\nVPLOT\r\n');
53 fprintf(fileID2,'FINI\r\n/CLEAR,NOSTART\r\n-PARAIN,MainBeam4,x-t,,SOLIDS,0,0\r\n/);
54 fprintf(fileID2,'FACET,NORML\r\nVPLOT\r\n');
55
56 % Enter the structural pre-processor phase:
57 fprintf(fileID1,'/PREP7\r\nSMRT,OFF\r\n/TRIAD,OFF\r\nKEYW,STRUCTRL,1\r\n/);
58 fprintf(fileID1,'UNITS,SI\r\n/VIEW,1,0,0,1\r\n');
```

```

59 fprintf(fileID2, '/PREP7\r\nSMRT,OFF\r\n/TRIAD,OFF\r\nKEYW,STRUCTRL,1\r\n/');
60 fprintf(fileID2, 'UNITS,SI\r\n/VIEW,1,0,0,1\r\n');
61
62 % Define the element types:
63 fprintf(fileID1, 'ET,1,SOLID186\r\nET,2,TARGE170\r\nET,3,CONTA174\r\n');
64 fprintf(fileID2, 'ET,1,SOLID186\r\nET,2,TARGE170\r\nET,3,CONTA174\r\n');
65
66 % Define the element type options:
67 fprintf(fileID1, 'KEYOPT,3,1,0\r\nKEYOPT,3,2,2\r\nKEYOPT,3,4,2\r\nKEYOPT,3,12,5\r\n');
68 fprintf(fileID2, 'KEYOPT,3,1,0\r\nKEYOPT,3,2,2\r\nKEYOPT,3,4,2\r\nKEYOPT,3,12,5\r\n');
69
70 % Specify the material properties:
71 fprintf(fileID1, 'EX,1,200E9\r\nPRXY,1,0.3\r\nDENS,1,7500\r\n');
72 fprintf(fileID2, 'EX,1,200E9\r\nPRXY,1,0.3\r\nDENS,1,7500\r\n');
73
74 % Mesh the model:
75 fprintf(fileID1, 'SMRTSIZE,6\r\nMSHAPE,1,3D\r\nMSHKEY,0\r\nVMESH,1\r\n');
76 fprintf(fileID2, 'SMRTSIZE,6\r\nMSHAPE,1,3D\r\nMSHKEY,0\r\nVMESH,1\r\n');
77
78 %% Specify the Constraints at the Front Axle
79
80 % Determine which nodes correspond to the areas at the front spherical joint:
81 j = 1;
82 FrontVec = unique([A4Nodes;A96Nodes]);
83 FrontMin(j) = min(FrontVec);
84 FrontMax(j) = 0;
85 for i = 2:length(FrontVec)
86     if FrontVec(i) ≠ FrontVec(i-1)+1
87         j = j+1;
88         FrontMin(j) = FrontVec(i);
89         FrontMax(j-1) = FrontVec(i-1);
90     end
91 end
92 FrontMax(length(FrontMax)+1) = max(FrontVec);
93 % Select the nodes at the front spherical joint:
94 formatSpec = 'NSEL,S,NODE,,%6.0f,%6.0f\r\n';
95 fprintf(fileID1, formatSpec, FrontMin(1), FrontMax(1));
96 fprintf(fileID2, formatSpec, FrontMin(1), FrontMax(1));
97 for k = 2:length(FrontMin)
98     formatSpec = 'NSEL,A,NODE,,%5.0f,%5.0f\r\n';
99     fprintf(fileID1, formatSpec, FrontMin(k), FrontMax(k));
100    fprintf(fileID2, formatSpec, FrontMin(k), FrontMax(k));
101 end
102 % Define the geometric surface for the target segment elements:
103 fprintf(fileID1, 'REAL,1\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
104 fprintf(fileID2, 'REAL,1\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
105 % Specify a pilot node at the front spherical joint centre:
106 fprintf(fileID1, 'N,200000,2.74095,0.1395,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200000\r\n');
107 fprintf(fileID2, 'N,200000,2.74095,0.1395,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200000\r\n');
108 % Define the translational constraints at the front axle:
109 fprintf(fileID1, 'F,200000,FX,0\r\nD,200000,UY,0\r\nD,200000,UZ,0\r\n');
110 fprintf(fileID2, 'F,200000,FX,0\r\nD,200000,UY,0\r\nD,200000,UZ,0\r\n');
111 % Define the rotational constraints at the front axle:
112 fprintf(fileID1, 'F,200000,MX,0\r\nF,200000,MY,0\r\nF,200000,MZ,0\r\n');
113 fprintf(fileID2, 'F,200000,MX,0\r\nF,200000,MY,0\r\nF,200000,MZ,0\r\n');
114
115 %% Specify the Constraints at the Rear Axle
116
117 % Determine which nodes correspond to the areas at the rear revolute joint:
118 j = 1;
119 RearVec = unique([A1Nodes;A2Nodes;A98Nodes;A99Nodes]);
120 RearMin(j) = min(RearVec);
121 RearMax(j) = 0;

```

```

122 for i = 2:length(RearVec)
123     if RearVec(i) ≠ RearVec(i-1)+1
124         j = j+1;
125         RearMin(j) = RearVec(i);
126         RearMax(j-1) = RearVec(i-1);
127     end
128 end
129 RearMax(length(RearMax)+1) = max(RearVec);
130 % Select the nodes at the rear revolute joint:
131 formatSpec = 'NSEL,S,NODE, ,%6.0f,%6.0f\r\n';
132 fprintf(fileID1,formatSpec,RearMin(1),RearMax(1));
133 fprintf(fileID2,formatSpec,RearMin(1),RearMax(1));
134 for k = 2:length(RearMin)
135     formatSpec = 'NSEL,A,NODE, ,%5.0f,%5.0f\r\n';
136     fprintf(fileID1,formatSpec,RearMin(k),RearMax(k));
137     fprintf(fileID2,formatSpec,RearMin(k),RearMax(k));
138 end
139 % Define the geometric surface for the target segment elements:
140 fprintf(fileID1,'REAL,2\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
141 fprintf(fileID2,'REAL,2\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
142 % Specify a pilot node at the rear revolute joint centre:
143 fprintf(fileID1,'N,200001,0.40695,0.13275,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200001\r\n');
144 fprintf(fileID2,'N,200001,0.40695,0.13275,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200001\r\n');
145 % Define the translational constraints at the rear axle:
146 fprintf(fileID1,'D,200001,UX,0\r\nD,200001,UY,0\r\nD,200001,UZ,0\r\n');
147 fprintf(fileID2,'D,200001,UX,0\r\nD,200001,UY,0\r\nD,200001,UZ,0\r\n');
148 % Define the rotational constraints at the rear axle:
149 fprintf(fileID1,'D,200001,ROTX,0\r\nF,200001,MY,0\r\n');
150 fprintf(fileID2,'D,200001,ROTX,0\r\nF,200001,MY,0\r\n');
151
152 %% Define a Pilot Node at the Pivot Joint Centre
153
154 % Determine which nodes correspond to the areas at the pivot joint:
155 j = 1;
156 PivotVec = unique([A3Nodes;A97Nodes]);
157 PivotMin(j) = min(PivotVec);
158 PivotMax(j) = 0;
159 for i = 2:length(PivotVec)
160     if PivotVec(i) ≠ PivotVec(i-1)+1
161         j = j+1;
162         PivotMin(j) = PivotVec(i);
163         PivotMax(j-1) = PivotVec(i-1);
164     end
165 end
166 PivotMax(length(PivotMax)+1) = max(PivotVec);
167 % Select the nodes at the pivot joint:
168 formatSpec = 'NSEL,S,NODE, ,%6.0f,%6.0f\r\n';
169 fprintf(fileID1,formatSpec,PivotMin(1),PivotMax(1));
170 fprintf(fileID2,formatSpec,PivotMin(1),PivotMax(1));
171 for k = 2:length(PivotMin)
172     formatSpec = 'NSEL,A,NODE, ,%5.0f,%5.0f\r\n';
173     fprintf(fileID1,formatSpec,PivotMin(k),PivotMax(k));
174     fprintf(fileID2,formatSpec,PivotMin(k),PivotMax(k));
175 end
176 % Define the geometric surface for the target segment elements:
177 fprintf(fileID1,'REAL,3\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
178 fprintf(fileID2,'REAL,3\r\nTYPE,3\r\nTSHAP,QUA8\r\nESURF\r\n');
179 % Specify a pilot node at the pivot joint centre:
180 fprintf(fileID1,'N,200002,1.29375,0.13275,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200002\r\n');
181 fprintf(fileID2,'N,200002,1.29375,0.13275,0.13425\r\nTYPE,2\r\nTSHAP,PILO\r\nE,200002\r\n');
182 % Specify free rotation around the y-axis at the pivot joint:
183 fprintf(fileID1,'F,200002,MY,0\r\n');
184 fprintf(fileID2,'F,200002,MY,0\r\n');

```

```

185
186 %% Solve the Model for the Conventional Loadcases
187
188 for LC = 1:n.ConventionalLC
189 % Specify the reaction torque at the rear revolute joint
190 formatSpec = 'F,200001,MZ,%10.5f\r\n';
191 fprintf(fileID1,formatSpec,Tz_RearAC(LC,1));
192
193 % Specify the force in x-direction at the pivot joint:
194 formatSpec = 'F,200002,FX,%10.5f\r\n';
195 fprintf(fileID1,formatSpec,Fx_PivotSC(LC,1));
196
197 % Specify the force in y-direction at the pivot joint:
198 formatSpec = 'F,200002,FY,%10.5f\r\n';
199 fprintf(fileID1,formatSpec,Fy_PivotSC(LC,1));
200
201 % Specify the force in z-direction at the pivot joint:
202 formatSpec = 'F,200002,FZ,%10.5f\r\n';
203 fprintf(fileID1,formatSpec,Fz_PivotSC(LC,1));
204
205 % Specify the torque around the x-axis at the pivot joint:
206 formatSpec = 'F,200002,MX,%10.5f\r\n';
207 fprintf(fileID1,formatSpec,Tx_PivotSC(LC,1));
208
209 % Specify the torque around the z-axis at the pivot joint:
210 formatSpec = 'F,200002,MZ,%10.5f\r\n';
211 fprintf(fileID1,formatSpec,Tz_PivotSC(LC,1));
212
213 % Specify the linear accelerations:
214 formatSpec = 'ACEL,%10.5f,%10.5f,%10.5f\r\n';
215 fprintf(fileID1,formatSpec,-aX_BeamSC(LC,1),-aY_BeamSC(LC,1),-aZ_BeamSC(LC,1));
216
217 % Solve the model:
218 fprintf(fileID1,'FINISH\r\n/SOLU\r\nANTYPE,0\r\nALLSEL,ALL\r\n');
219 fprintf(fileID1,'SOLVE\r\nFINISH\r\n/POST1\r\n');
220
221 % Export the equivalent stress values:
222 formatSpec = '/OUTPUT,Conventional.LC%1.0f.NodalS, TXT\r\n';
223 fprintf(fileID1,formatSpec,LC);
224 fprintf(fileID1,'PRNSOL,S,PRIN,\r\n');
225 fprintf(fileID1,'/OUT\r\n');
226 % Export the displacement sum values:
227 formatSpec = '/OUTPUT,Conventional.LC%1.0f.NodalU, TXT\r\n';
228 fprintf(fileID1,formatSpec,LC);
229 fprintf(fileID1,'PRNSOL,U,COMP\r\n');
230 fprintf(fileID1,'/OUT\r\n');
231 fprintf(fileID1,'FINISH\r\n!\r\n/PREP7\r\n');
232 end
233 fclose(fileID1);
234
235 %% Solve the Model for the Optimized Loadcases
236
237 for LC = 1:n.OptimizedLC
238 % Specify the reaction torque at the rear revolute joint
239 formatSpec = 'F,200001,MZ,%10.5f\r\n';
240 fprintf(fileID2,formatSpec,Tz_RearAO(LC,1));
241
242 % Specify the force in x-direction at the pivot joint:
243 formatSpec = 'F,200002,FX,%10.5f\r\n';
244 fprintf(fileID2,formatSpec,Fx_PivotSO(LC,1));
245
246 % Specify the force in y-direction at the pivot joint:
247 formatSpec = 'F,200002,FY,%10.5f\r\n';

```

```

248 fprintf(fileID2,formatSpec,Fy_PivotSO(LC,1));
249
250 % Specify the force in z-direction at the pivot joint:
251 formatSpec = 'F,200002,FZ,%10.5f\r\n';
252 fprintf(fileID2,formatSpec,Fz_PivotSO(LC,1));
253
254 % Specify the torque around the x-axis at the pivot joint:
255 formatSpec = 'F,200002,MX,%10.5f\r\n';
256 fprintf(fileID2,formatSpec,Tx_PivotSO(LC,1));
257
258 % Specify the torque around the z-axis at the pivot joint:
259 formatSpec = 'F,200002,MZ,%10.5f\r\n';
260 fprintf(fileID2,formatSpec,Tz_PivotSO(LC,1));
261
262 % Specify the linear accelerations:
263 formatSpec = 'ACEL,%10.5f,%10.5f,%10.5f\r\n';
264 fprintf(fileID2,formatSpec,-aX_BeamSO(LC,1),-aY_BeamSO(LC,1),-aZ_BeamSO(LC,1));
265
266 % Define the location of the beam centre of mass:
267 fprintf(fileID2,'CGLOC,1.68895,0.13275,0.13575\r\n');
268
269 % Specify the angular accelerations:
270 formatSpec = 'DCGOMG,%10.5f,%10.5f,%10.5f\r\n';
271 fprintf(fileID2,formatSpec,-bX_BeamSO(LC,1),-bY_BeamSO(LC,1),-bZ_BeamSO(LC,1));
272
273 % Solve the model:
274 fprintf(fileID2,'FINISH\r\n/SOLU\r\nANTYPE,0\r\nALLSEL,ALL\r\n');
275 fprintf(fileID1,'SOLVE\r\nFINISH\r\n/POST1\r\n');
276
277 % Export the nodal reaction forces:
278 formatSpec = '/OUTPUT,Optimized_LC%1.0f_ReactionF,TXT\r\n';
279 fprintf(fileID2,formatSpec,LC);
280 fprintf(fileID2,'PRRSOL,F\r\nPRRSOL,M\r\n');
281 fprintf(fileID2,'/OUT\r\n');
282 % Export the nodal loads:
283 formatSpec = '/OUTPUT,Optimized_LC%1.0f_AppliedF,TXT\r\n';
284 fprintf(fileID2,formatSpec,LC);
285 fprintf(fileID2,'FLIST,200000\r\nFLIST,200001\r\nFLIST,200002\r\n');
286 fprintf(fileID2,'/OUT\r\n');
287 % Export the equivalent stress values:
288 formatSpec = '/OUTPUT,Optimized_LC%1.0f_NodalS,TXT\r\n';
289 fprintf(fileID2,formatSpec,LC);
290 fprintf(fileID2,'PRNSOL,S,PRIN,\r\n');
291 fprintf(fileID2,'/OUT\r\n');
292 % Export the displacement sum values:
293 formatSpec = '/OUTPUT,Optimized_LC%1.0f_NodalU,TXT\r\n';
294 fprintf(fileID2,formatSpec,LC);
295 fprintf(fileID2,'PRNSOL,U,COMP\r\n');
296 fprintf(fileID2,'/OUT\r\n');
297 fprintf(fileID2,'FINISH\r\n!\r\n/PREP7\r\n');
298 end
299 fclose(fileID2);

```

# M. Resultant Stresses and Displacements

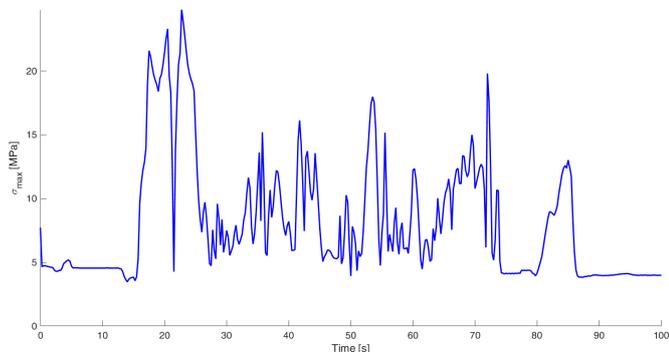
| Loadcase  | $\sigma_{\max}$ [MPa] | $d_{\max}$ [mm] |
|-----------|-----------------------|-----------------|
| <b>1</b>  | 24.78                 | 0.188           |
| <b>2</b>  | 27.50                 | 0.238           |
| <b>3</b>  | 26.89                 | 0.197           |
| <b>4</b>  | 29.05                 | 0.235           |
| <b>5</b>  | 26.40                 | 0.192           |
| <b>6</b>  | 27.61                 | 0.191           |
| <b>7</b>  | 27.68                 | 0.229           |
| <b>8</b>  | 28.89                 | 0.229           |
| <b>9</b>  | 26.72                 | 0.192           |
| <b>10</b> | 25.87                 | 0.191           |
| <b>11</b> | 30.21                 | 0.226           |
| <b>12</b> | 29.72                 | 0.226           |

| Loadcase  | $\sigma_{\max}$ [MPa] | $d_{\max}$ [mm] |
|-----------|-----------------------|-----------------|
| <b>1</b>  | 9.08                  | 0.025           |
| <b>2</b>  | 5.57                  | 0.047           |
| <b>3</b>  | 4.99                  | 0.040           |
| <b>4</b>  | 26.33                 | 0.211           |
| <b>5</b>  | 27.22                 | 0.217           |
| <b>6</b>  | 21.48                 | 0.173           |
| <b>7</b>  | 26.61                 | 0.205           |
| <b>8</b>  | 12.17                 | 0.137           |
| <b>9</b>  | 14.11                 | 0.146           |
| <b>10</b> | 22.57                 | 0.192           |
| <b>11</b> | 21.16                 | 0.182           |
| <b>12</b> | 12.49                 | 0.131           |
| <b>13</b> | 11.68                 | 0.130           |
| <b>14</b> | 20.08                 | 0.154           |
| <b>15</b> | 6.00                  | 0.071           |
| <b>16</b> | 4.14                  | 0.062           |
| <b>17</b> | 4.49                  | 0.038           |
| <b>18</b> | 20.54                 | 0.164           |
| <b>19</b> | 24.81                 | 0.193           |
| <b>20</b> | 17.69                 | 0.139           |
| <b>21</b> | 10.64                 | 0.087           |
| <b>22</b> | 4.03                  | 0.035           |
| <b>23</b> | 25.45                 | 0.199           |
| <b>24</b> | 19.62                 | 0.156           |
| <b>25</b> | 20.93                 | 0.170           |
| <b>26</b> | 8.75                  | 0.108           |
| <b>27</b> | 13.83                 | 0.117           |
| <b>28</b> | 4.63                  | 0.058           |
| <b>29</b> | 4.19                  | 0.058           |
| <b>30</b> | 3.70                  | 0.034           |
| <b>31</b> | 3.16                  | 0.033           |
| <b>32</b> | 14.89                 | 0.126           |
| <b>33</b> | 18.25                 | 0.152           |
| <b>34</b> | 5.30                  | 0.051           |
| <b>35</b> | 4.76                  | 0.047           |
| <b>36</b> | 14.32                 | 0.113           |
| <b>37</b> | 12.66                 | 0.108           |
| <b>38</b> | 5.02                  | 0.056           |
| <b>39</b> | 3.97                  | 0.052           |

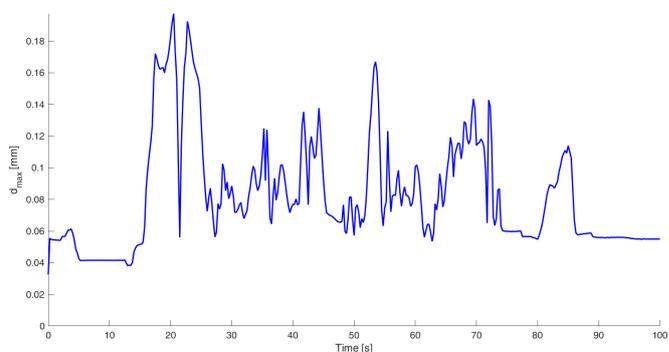
Table M.1: Resultant stresses and displacements for the conventional (left) and optimized (right) method

# N. Stress and Displacement as a function of Time

## N.1 Configuration 2



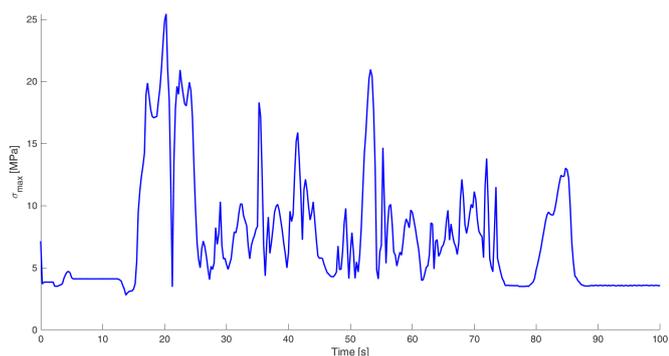
(a) Maximum Von Mises stress



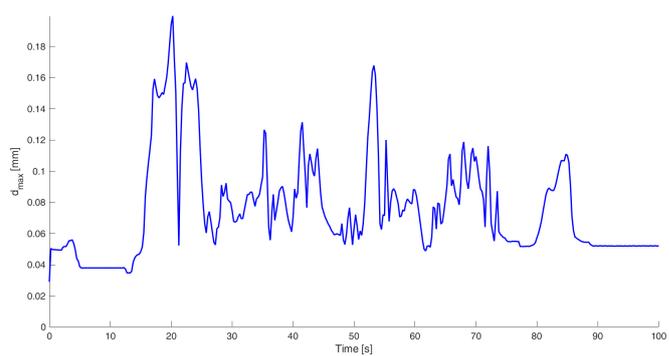
(b) Maximum nodal displacement

Figure N.1: The maximum Von Mises stress and nodal displacement as a function of time

## N.2 Configuration 3



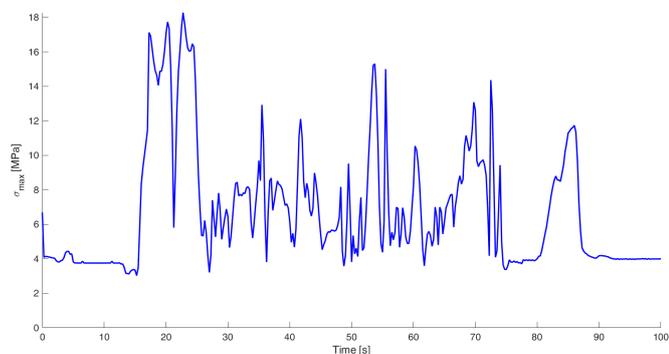
(a) Maximum Von Mises stress



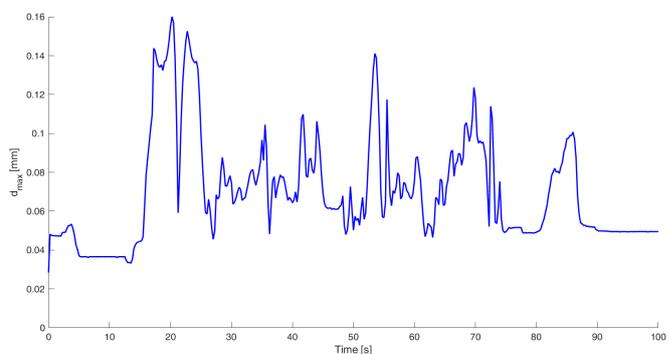
(b) Maximum nodal displacement

Figure N.2: The maximum Von Mises stress and nodal displacement as a function of time

## N.3 Configuration 4



(a) Maximum Von Mises stress

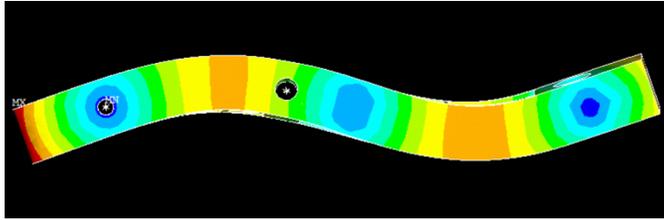


(b) Maximum nodal displacement

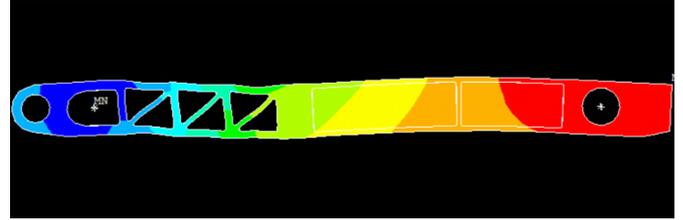
Figure N.3: The maximum Von Mises stress and nodal displacement as a function of time

# O. Mode-Superposition Method

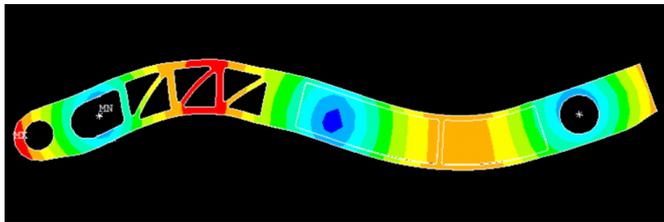
## O.1 Modal Analysis



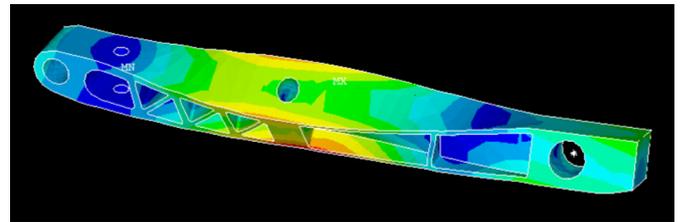
(a) Fourth eigenmode ( $f_4 = 271.7$  Hz)



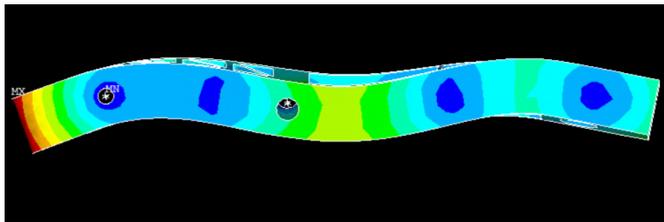
(b) Fifth eigenmode ( $f_5 = 367.7$  Hz)



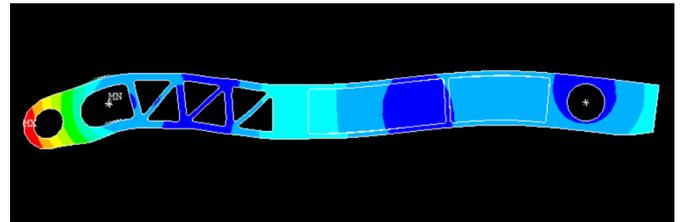
(c) Sixth eigenmode ( $f_6 = 397.0$  Hz)



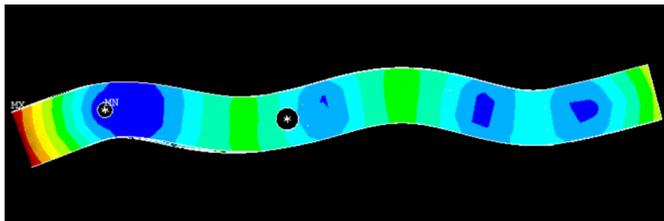
(d) Seventh eigenmode ( $f_7 = 438.2$  Hz)



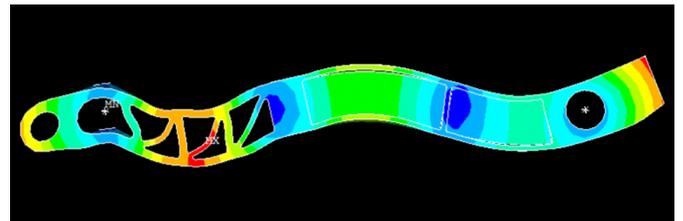
(e) Eighth eigenmode ( $f_8 = 488.1$  Hz)



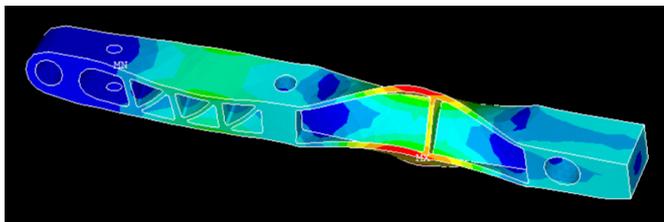
(f) Ninth eigenmode ( $f_9 = 608.5$  Hz)



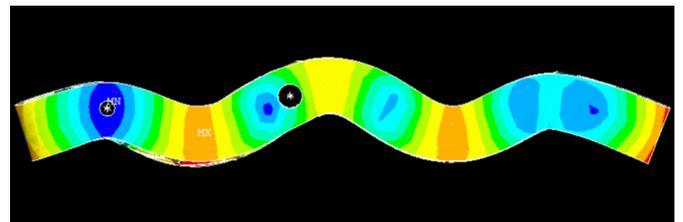
(g) Tenth eigenmode ( $f_{10} = 693.2$  Hz)



(h) Eleventh eigenmode ( $f_{11} = 771.3$  Hz)



(i) Twelfth eigenmode ( $f_{12} = 885.5$  Hz)

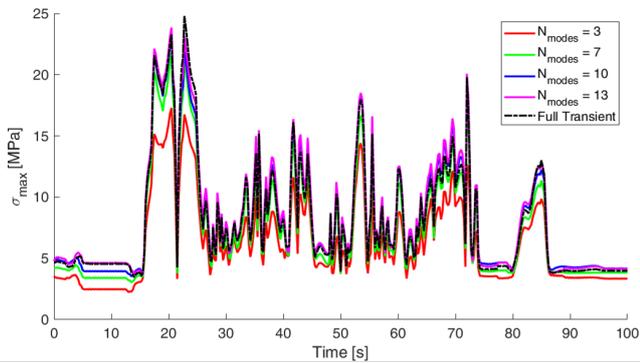


(j) Thirteenth eigenmode ( $f_{13} = 986.7$  Hz)

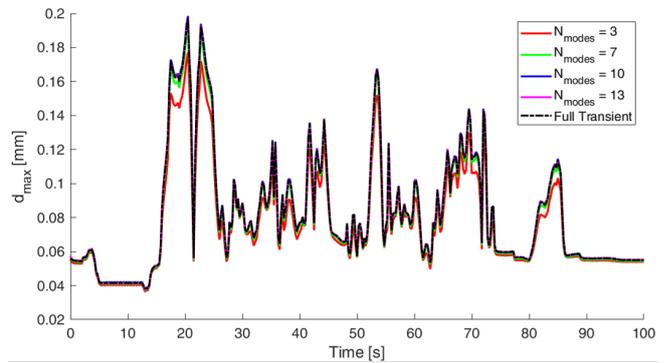
Figure O.1: The eigenmodes of the main chassis beam, up to the thirteenth eigenmode

## O.2 Comparison with Full Transient Analysis

### O.2.1 Configuration 2



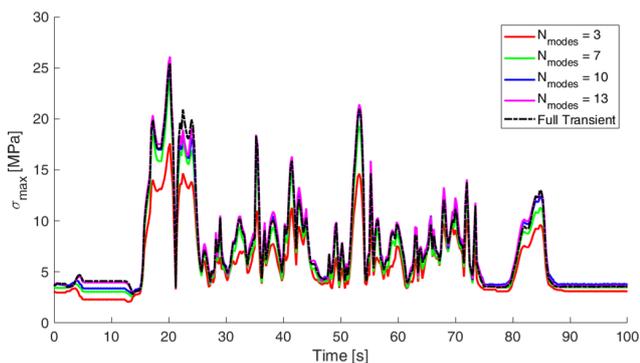
(a) Maximum Von Mises stress



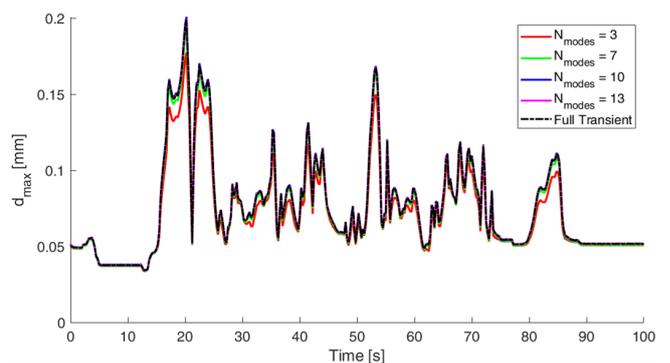
(b) Maximum nodal displacement sum

Figure O.2: Comparison between the full and mode-superposition transient analyses for configuration 2

### O.2.2 Configuration 3



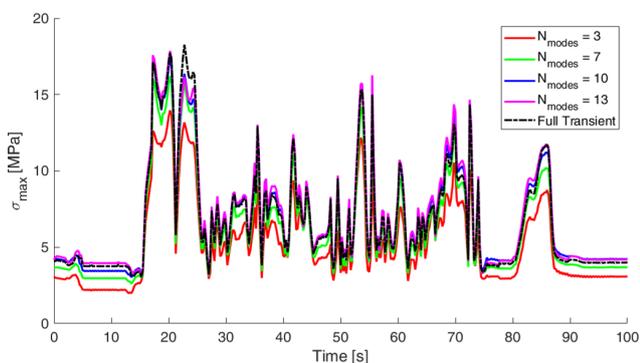
(a) Maximum Von Mises stress



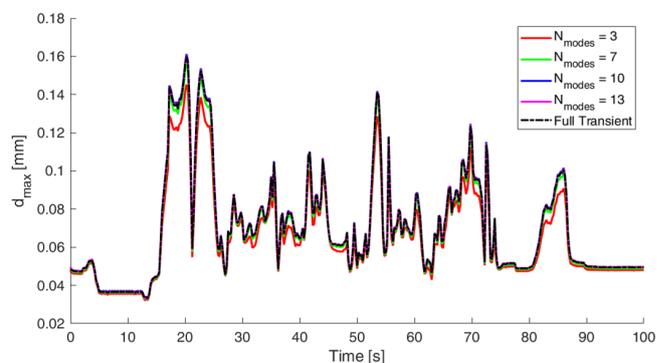
(b) Maximum nodal displacement sum

Figure O.3: Comparison between the full and mode-superposition transient analyses for configuration 3

### O.2.3 Configuration 4



(a) Maximum Von Mises stress



(b) Maximum nodal displacement sum

Figure O.4: Comparison between the full and mode-superposition transient analyses for configuration 4

### O.3 Tabulated Results

| Configuration | N <sub>modes</sub> | $\Delta\sigma_{eq,max}$ [MPa] |         | $\Delta d_{sum,max}$ [mm] |                       | CPU Time [min] |
|---------------|--------------------|-------------------------------|---------|---------------------------|-----------------------|----------------|
|               |                    | Mean                          | Maximum | Mean                      | Maximum               |                |
| 1             | 3                  | 2.18                          | 9.08    | $5.780 \cdot 10^{-3}$     | $23.40 \cdot 10^{-3}$ | 12             |
|               | 7                  | 0.77                          | 4.12    | $1.590 \cdot 10^{-3}$     | $5.270 \cdot 10^{-3}$ | 18             |
|               | 10                 | 0.44                          | 2.97    | $0.382 \cdot 10^{-3}$     | $1.955 \cdot 10^{-3}$ | 19             |
|               | 13                 | 0.42                          | 1.73    | $0.234 \cdot 10^{-3}$     | $2.010 \cdot 10^{-3}$ | 22             |
| 2             | 3                  | 1.90                          | 8.04    | $5.402 \cdot 10^{-3}$     | $20.03 \cdot 10^{-3}$ | 14             |
|               | 7                  | 0.69                          | 3.79    | $1.452 \cdot 10^{-3}$     | $4.850 \cdot 10^{-3}$ | 17             |
|               | 10                 | 0.41                          | 2.90    | $0.361 \cdot 10^{-3}$     | $1.780 \cdot 10^{-3}$ | 19             |
|               | 13                 | 0.42                          | 1.72    | $0.229 \cdot 10^{-3}$     | $1.826 \cdot 10^{-3}$ | 20             |
| 3             | 3                  | 1.72                          | 7.86    | $4.747 \cdot 10^{-3}$     | $21.90 \cdot 10^{-3}$ | 14             |
|               | 7                  | 0.63                          | 3.00    | $1.319 \cdot 10^{-3}$     | $4.820 \cdot 10^{-3}$ | 18             |
|               | 10                 | 0.35                          | 2.28    | $0.344 \cdot 10^{-3}$     | $1.680 \cdot 10^{-3}$ | 18             |
|               | 13                 | 0.31                          | 2.14    | $0.213 \cdot 10^{-3}$     | $1.621 \cdot 10^{-3}$ | 22             |
| 4             | 3                  | 1.54                          | 5.09    | $4.422 \cdot 10^{-3}$     | $14.82 \cdot 10^{-3}$ | 15             |
|               | 7                  | 0.60                          | 2.53    | $1.177 \cdot 10^{-3}$     | $3.640 \cdot 10^{-3}$ | 17             |
|               | 10                 | 0.30                          | 1.91    | $0.320 \cdot 10^{-3}$     | $1.419 \cdot 10^{-3}$ | 18             |
|               | 13                 | 0.35                          | 2.12    | $0.206 \cdot 10^{-3}$     | $1.466 \cdot 10^{-3}$ | 22             |

Table O.1: Deviation of the results obtained with the mode-superposition method with respect to the results of a full-transient analysis for each passenger occupancy configuration, including the computational time

| Configuration | N <sub>modes</sub> | $\sigma_{eq,max}$ [MPa] | $d_{sum,max}$ [mm] | CPU Time [min] |
|---------------|--------------------|-------------------------|--------------------|----------------|
| 1             | Full               | 27.23                   | 0.217              | 110            |
|               | 3                  | 19.15 (-29.67%)         | 0.194 (-10.60%)    | 12 (-89.09%)   |
|               | 7                  | 25.51 (-6.32%)          | 0.212 (-2.30%)     | 18 (-83.64%)   |
|               | 10                 | 27.04 (-0.70%)          | 0.219 (+0.92%)     | 19 (-82.73%)   |
|               | 13                 | 27.90 (+2.46%)          | 0.218 (+0.46%)     | 22 (-80.00%)   |
| 2             | Full               | 24.80                   | 0.197              | 110            |
|               | 3                  | 17.28 (-30.32%)         | 0.177 (-10.15%)    | 14 (-87.27%)   |
|               | 7                  | 21.74 (-12.34%)         | 0.193 (-2.03%)     | 17 (-84.55%)   |
|               | 10                 | 23.13 (-6.73%)          | 0.199 (+1.02%)     | 19 (-82.73%)   |
|               | 13                 | 23.85 (-3.83%)          | 0.198 (+0.51%)     | 20 (-81.82%)   |
| 3             | Full               | 25.45                   | 0.200              | 108            |
|               | 3                  | 17.59 (-30.88%)         | 0.178 (-11.00%)    | 14 (-87.04%)   |
|               | 7                  | 23.89 (-6.13%)          | 0.195 (-2.50%)     | 18 (-83.33%)   |
|               | 10                 | 25.29 (-0.63%)          | 0.201 (+0.50%)     | 18 (-83.33%)   |
|               | 13                 | 26.11 (+2.59%)          | 0.200 (0.00%)      | 22 (-79.63%)   |
| 4             | Full               | 18.27                   | 0.160              | 108            |
|               | 3                  | 13.96 (-23.59%)         | 0.145 (-9.38%)     | 15 (-86.11%)   |
|               | 7                  | 16.21 (-11.28%)         | 0.157 (-1.88%)     | 17 (-84.26%)   |
|               | 10                 | 17.33 (-5.15%)          | 0.161 (+0.63%)     | 18 (-83.33%)   |
|               | 13                 | 17.86 (-2.24%)          | 0.160 (0.00%)      | 22 (-79.63%)   |

Table O.2: Comparison between the mode-superposition methods and a full-transient analysis in terms of the maximum Von Mises stress, maximum displacement sum, and computational time for each passenger occupancy configuration