COMPUTING SCIENCE

Utilisation of Video Game Physics Techniques in Real Time Simulation of the Wheel Rail Interface for Predicted Derailment of Rail Vehicles

Michael Simpson, William Blewitt, Gary Ushaw and Graham Morgan

TECHNICAL REPORT SERIES

No. CS-TR-1438 October 2014

TECHNICAL REPORT SERIES

No. CS-TR-1438 October, 2014

Utilisation of Video Game Physics Techniques in Real Time Simulation of the Wheel Rail Interface for Predicted Derailment of Rail Vehicles

M. Simpson, W. Blewitt, G. Ushaw and G. Morgan

Abstract

We apply real time physics techniques from video game development to the simulation of rail vehicles for rapid analysis of the stability of the wheel rail interface. We introduce a fast simulation algorithm for the wheel rail interface, using a spline-based approach to approximate the gravitational stiffness force. We compare our simulation to results achieved from solving the Nadal equation for a range of rail vehicle speeds. We extend the technique to multiple railway bogies, and use the simulation to gather data on flange collisions. As we use a physics library and techniques designed for real time gaming, our results are achieved considerably more quickly than rail industry standard simulations, allowing for multiple scenarios to be modelled and analysed.

Printed and published by Newcastle University,
Computing Science, Claremont Tower, Claremont Road,
Newcastle upon Tyne, NE1 7RU, England.

Bibliographical details

SIMPSON, M., BLEWITT, W., USHAW, G., MORGAN, G.

Utilisation of Video Game Physics Techniques in Real Time Simulation of the Wheel Rail Interface for Predicted Derailment of Rail Vehicles
[By] M. Simpson, W. Blewitt, G. Ushaw and G. Morgan

Newcastle upon Tyne: Newcastle University: Computing Science, 2014.

(Newcastle University, Computing Science, Technical Report Series, No. CS-TR-1438)

Added entries

NEWCASTLE UNIVERSITY
Computing Science. Technical Report Series. CS-TR-1438

Abstract

We apply real time physics techniques from video game development to the simulation of rail vehicles for rapid analysis of the stability of the wheel rail interface. We introduce a fast simulation algorithm for the wheel rail interface, using a spline-based approach to approximate the gravitational stiffness force. We compare our simulation to results achieved from solving the Nadal equation for a range of rail vehicle speeds. We extend the technique to multiple railway bogies, and use the simulation to gather data on flange collisions. As we use a physics library and techniques designed for real time gaming, our results are achieved considerably more quickly than rail industry standard simulations, allowing for multiple scenarios to be modelled and analysed.

About the authors

Mike Simpson is a PhD student in the School of Computing Science at Newcastle University studying real time simulation techniques. He is a graduate from the MSc in Computer Games Engineering and has worked in the video games industry as a physics programmer.

William Blewitt is a teaching fellow in the School of Computing Science at Newcastle University. He achieved his PhD at Leicester de Montford University, and has multiple masters degrees, including the Computer Games Engineering MSc.

Dr Gary Ushaw is a lecturer in the School of Computing Science at Newcastle University. Having worked for 15 years in the video games industry as Engineering Director and Project Manager, following PhD attained from the University of Edinburgh's Signal Processing Group.

Graham Morgan is a senior lecturer in the School of Computing Science at the University of Newcastle. Graham leads the Game Engineering MSc at Newcastle and carries out research in the areas of systems and simulation. His interests are in the area of distributed applications, including web services, networked virtual environments, fault tolerance and group communications.

Suggested keywords

REAL TIME SIMULATION
WHEEL RAIL INTERFACE

UTILISATION OF VIDEO GAME PHYSICS TECHNIQUES IN REAL TIME SIMULATION OF THE WHEEL RAIL INTERFACE FOR PREDICTED DERAILMENT OF RAIL VEHICLES

Michael Simpson, William Blewitt, Gary Ushaw, Graham Morgan
School of Computing Science, Newcastle University,
Newcastle-upon-Tyne, United Kingdom

KEYWORDS
Real time simulation, Wheel rail interface

ABSTRACT

We apply real time physics techniques from video game development to the simulation of rail vehicles for rapid analysis of the stability of the wheel rail interface. We introduce a fast simulation algorithm for the wheel rail interface, using a spline-based approach to approximate the gravitational stiffness force. We compare our simulation to results achieved from solving the Nadal equation for a range of rail vehicle speeds. We extend the technique to multiple railway bogies, and use the simulation to gather data on flange collisions. As we use a physics library and techniques designed for real time gaming, our results are achieved considerably more quickly than rail industry standard simulations, allowing for multiple scenarios to be modelled and analysed.

INTRODUCTION

The simulation of rolling stock on railway tracks is vitally important to the rail industry in terms of both safety and cost. New rail vehicle designs, and new track sections, undergo extensive simulation before being green-lit for construction. Traditional simulation techniques are slow and expensive (although still considerably cheaper than building rail vehicle prototypes). The time consuming nature of these simulations prohibits the number of different scenarios which can be considered, leading to conservative options being modelled from the outset.

The video game industry employs physics simulation techniques which operate in real time for multiple agents on relatively lightweight computational platforms. Consequently simulation is both cheaper and faster than more traditional techniques. However this speed comes at the cost of the accuracy of the simulation results. It is our contention that real time video game simulation techniques can be used to model multiple scenarios quickly and cheaply in order to identify the parameters with the most promising results. These chosen scenarios can then be targeted for higher fidelity simulation. In this manner a much wider range of parameters can be included in the simulations at an early stage, with the high cost time-consuming simulations only targeting the most promising combinations.

A key aspect of rail vehicle simulation is the wheel rail interface. Analysing the contact point between each wheel and the rail leads to prediction techniques for when a piece of rolling stock will derail. Wheel climb derailment occurs when the gravitational stiffness force is unable to keep the rails on the track. This is usually caused by a rail vehicle travelling at too high a velocity around a curve in the track. Rail climb derailment is believed to be the cause of sixteen derailments in the USA between 1998 and 2000 [Iwnicki, 2003]. The Nadal formula [Nadal, 1896] describes wheel-climb behaviour of a flanged pair of wheels on a set of rails, calculating the minimum conditions at which derailment will occur. In this paper we utilise NVidia's PhysX libraries to simulate railway bogies on tracks of different curvature, and employ a spline-based algorithm to simulate the gravitational stiffness force. We measure the speed at which the simulated wheel rail interface fails due to rail climb derailment, and compare the results to those calculated from the Nadal formula.

As our simulation runs in real time, it can be used to analyse many different scenarios in order to identify the parameters most worthy of high-fidelity simulation. Further to this, we show that the work can be extended to more complex scenarios involving multiple bogies and carriages. We also demonstrate that the simulation can be used to gather data on crucial indicators of rail degradation such as flange collisions. It is anticipated that this work will lead to a much wider range of simulation scenarios being considered when designing rolling stock and rail sections.

BACKGROUND

The wheel rail interface is described, and the Nadal formula for calculating wheel-climb behaviour is presented. Current techniques for simulating rail vehicle dynamics are then reviewed, which are not real time solutions. This leads to the contribution of the paper: utilising a commercial physics package from game development for real time simulation of the wheel rail interface.

Wheel Rail Interface

A wheel-set comprises wheels, flanges and axle, as shown in Figure 1. Flanges only prevent the derailment during normal operation and do not guide the wheels around bends. Instead, the conical shape of the wheels produces a self-centring effect. As the wheel drifts away from the centre of the track, the rolling radius becomes larger on the outer wheel than on the inner wheel, changing the effective size of the wheels and causing the wheel-set to yaw about the vertical axis and centre itself between the rails. This phenomenon is sometimes known as the Gravitational Stiffness Force [Iwnicki, 2003] and allows the wheel-set to corner without wear and tear on the flanges or rails, which would be potentially dangerous and expensive.

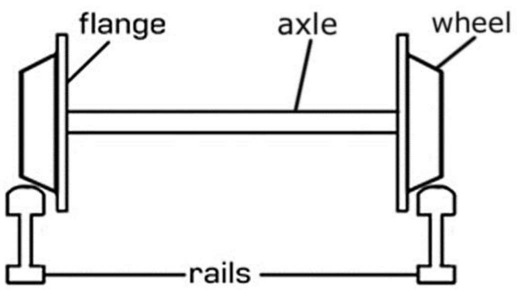

Figure 1: The Wheel Rail Interface.

When the gravitational stiffness force is not sufficient to keep the wheels on the rail, wheel climb will occur, leading to derailment if the wheel rises above the extent of the rail. The most common causes of wheel climb derailment are when the rail vehicle is travelling at too high a speed around a curve in the track, or as a result of hunting oscillation (which is a side effect of the gravitational stiffness force itself) [Marquis and Greif, 2011].
The Nadal formula describes wheel climb behaviour and represents the minimum conditions at which wheel climb derailment will occur [Nadal, 1896].

$$\frac{L}{V} = \frac{\tan\delta - \mu}{1 + \mu\tan\delta}$$

L and V represent the lateral and vertical forces acting on the wheelset, μ represents the coefficient of friction and δ represents the wheel's maximum angle of attack. The angle of attack is the angle (from horizontal) between the wheel and the rail at the contact point. Figure 2 shows the angle of attack δ between the wheel and rail during normal operation.

Rail Vehicle Simulation Techniques

Little work has been carried out to date on developing real time solutions for rail simulation. The challenges in-

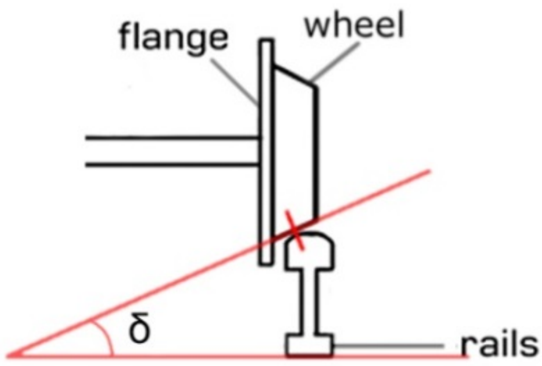

Figure 2: The Angle of Attack.

volved in simulating rail vehicle dynamics are described in [Evans and Berg, 2009]. The work does not address real-time simulation directly, but rail vehicle dynamic simulation in general is discussed. The key challenges are identified as modelling the wheel rail contact surfaces, suspension components, car body flexibility, inter-vehicle connections and track models. It is also pointed out that validation of a model is important, but problematic due to the difficulties measuring real world force data at the required fidelity.
Existing work has focused on the contact surface between the rail and wheel, for application within a wider simulation for high fidelity results. [Polach, 2000] models the contact surface as an ellipsoid with normal stress distribution. This gives shorter computing time than previous approaches, and has been applied to multiple simulations successfully, but is not a real-time solution. A multi-layer spline function algorithm is described in [Shabana et al., 2001], for calculation of the third derivatives with respect to the contact surface parameters. A numerical integration method is employed in [Anyakwo et al., 2012] to solve first order differential equations representing a two degree of freedom model, taking into account the lateral displacement of the wheel-set and the yaw angle. The contact point between the rail and wheel is also the focus of [Wang and Li, 2012] which uses a technique to transform the three dimensional space of the wheel and rail surfaces into a curve of possible contact points for less complex computation. These are all computationally intense algorithms which are unlikely to be suited to real time simulation.
A method for designing and implementing real time simulation algorithms, based on the use of field-programmable gate array technology is presented in [Monga et al., 2012]. A high-performance reconfigurable platform was developed to run various simulations. Results from the simulation environment are successfully compared to a mathematical approach and the system meets the real-time constraints. This, however,

involved the use of specialised and expensive hardware; we hope that the use of PhysX will allow our simulations to run on a standard desktop PC.
Some studies have considered other applications of physics libraries for video games. In [Luo et al., 2009] a commercial physics middleware solution is employed to build a simulation and debugging environment for small, robotic cars. Two vehicles were successfully built with software that was developed in the environment, showing that the physics simulation worked well on this scale. Rail vehicles are much larger and heavier, but this work implies that video games physics software can produce meaningful results.

IMPLEMENTATION

We have produced a bespoke simulation tool called "Locomotion" for this work. The tool has been developed in C++, using the OpenGL and PhysX libraries. It simulates the progress of one or more train carriages along tracks of varying curvature at gradually increasing speed. The tool logs the number of flange collisions for each wheel, and the speed at which wheel climb derailment occurs. Figure 3 shows a screen-shot from Locomotion. The train model is based on the parameters of the Metro de Madrid 5000 Series, supplied to us by NewRail.

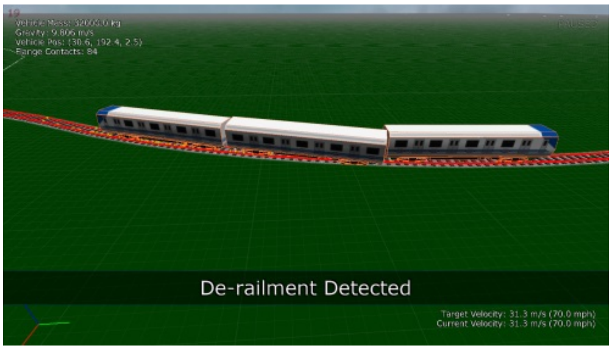

Figure 3: Screen Shot of Locomotion Simulation Tool.

To replicate the gravitational stiffness effect, we use a spline-based approach. Each section of track is represented by a spline which runs between the two rails. Each spline is defined by a Bezier curve. Every wheel/rail collision is registered, and then the nearest point on the spline to each wheel-set is calculated using an iterative search of the Bezier curve. A force is applied to the wheel-set, replicating the effect of the gravitational stiffness force. This is shown in Figure 4. The central line is the spline and the dot on the axle is the wheels global position. As the centrifugal force x causes the wheel to drift, a force y is applied to push it back towards the spline.
This force is a function of the distance to the central point (a vector), multiplied by the mass of the vehicle (divided by the number of wheel-sets) and the simulation time-step. This calculation is carried out for each wheel-set. The force is only applied while the wheels are in contact with the rails. The train will derail when the applied force is unable to counteract vehicle instability/centrifugal forces.

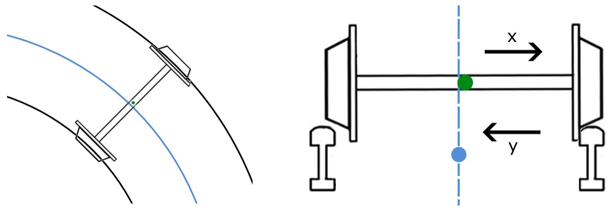

Figure 4: The Spline Based Approach.

The main body of the vehicle is constructed from a single rigid body. The mass and centre of gravity can be altered to simulate different load distributions. Each wheel is a rigid body to which the flanges are attached. The flanges are modelled as distinct rigid bodies to allow detection of flange collisions with the rails (a key signifier in rolling stock derailment and track degradation). The wheels are attached to the bogies (and the bogies to the rest of the vehicle) via revolute joints. Suspension is not currently modelled, but is approximated by an inherent, configurable flexibility in the joints. Derailment is detected by checking whether any wheel has left contact with the rail for more than a definable amount of time (in this case 0.5s), or if any component of the vehicle has come into contact with the ground. If derailment has occurred then the speed at which the vehicle was moving when last in contact with the rails is recorded as the derailment velocity.

SIMULATION RESULTS AND EVALUATION

The Nadal formula was used to predict the speed at which a single bogie is derailed, as a benchmark for comparison of our simulation results. Both the simulation and formula use the following values. Wheels have an angle of attack of 69.5°. We assume the wheel and rails are made of steel, with a coefficient of friction of 0.8. This results in a Nadal value of 0.597.
The vertical force (V) is calculated using acceleration due to gravity (9.806m/s^2) and the mass of the bogie, which is 7,000kg (including wheel-sets). Our calculations also assume even distribution of forces between the bogie's two wheel-sets. We approximate the lateral force (L) using centrifugal force and calculate this for a range of speeds and curve radii.
The graph in Figure 5 shows how speed affects the L/V value. The Nadal value predicts the minimum derailment conditions (i.e. the points where the horizontal Nadal line crosses the other graphs). Solving these equations gives us the conditions for derailment shown in the table below.

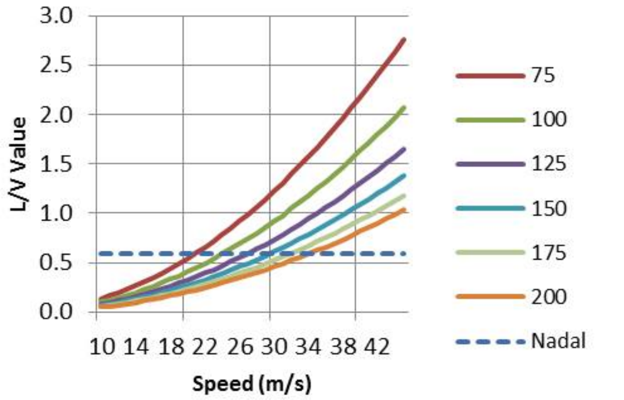

Figure 5: L/V Ratios for Varying Speed and Curve Radii.

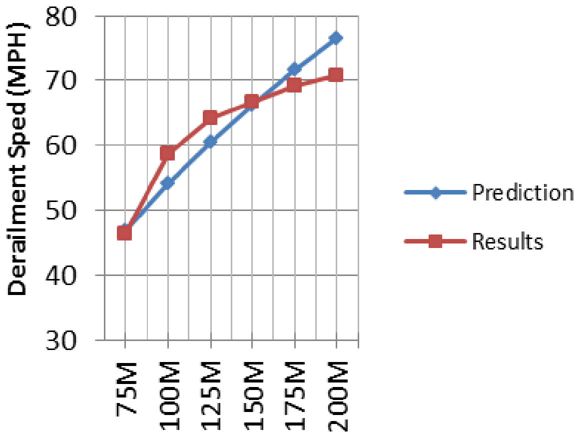

Figure 6: Predicted Speed Versus Simulated Results.

Curve radius (m)	Predicted min derailment speed
75	20.95 m/s (46.87 mph)
100	24.19 m/s (54.12 mph)
125	27.05 m/s (60.52 mph)
150	29.63 m/s (66.29 mph)
175	32.01 m/s (71.6 mph)
200	34.22 m/s (76.55 mph)

The Locomotion software was used to construct a looped track consisting of two horizontal straight sections, and two uniformly curved sections joining them at each end. A bogie was simulated traversing this looped track, with the speed of the bogie increased by 0.5mph on each successful loop of the track until derailment occurred. The experiment was carried out with the curvature of the curved sections ranging from 75 metres to 200 metres. Each simulation was repeated 100 times, resulting in the average speed at which derailment was detected for each curvature of track. Figure 6 shows the results of these simulations, and the corresponding predicted result from solving the Nadal equation.

It can be seen from the graph that the simulated vehicle is capable of achieving higher speeds on wider curves, as expected. All simulated results are within 6mph of the predicted value from the Nadal equation. However at higher speeds, the simulation becomes less capable of successfully traversing curved tracks than the formula predicts. The range of results and standard deviations from running 100 tests of each simulation are presented in the table below.

	75m	100m	125m	150m	175m	200m
Mean	46.31	58.88	64.27	66.72	69.12	70.78
Nadal	46.87	54.12	60.52	66.29	71.6	76.55
Diff	-0.57	4.56	3.75	0.43	-2.48	-5.77
Range	5.0	16.0	11.0	11.0	8.0	4.0
Std Dv	1.09	4.44	2.11	2.27	1.76	0.83

Simulating Multiple Carriages

The Locomotion tool was further used to simulate scenarios with multiple train carriages. The results in Figure 7 combine the results already considered for a single bogie with those achieved from simulating a carriage with two bogies, two carriages and three carriages (with two bogies each). In the multiple carriage scenarios, the lead carriage was pulling the rest of the train (this parameter is configurable in our software). The prediction line is from the Nadal equation for a single bogie, and is included for completeness.

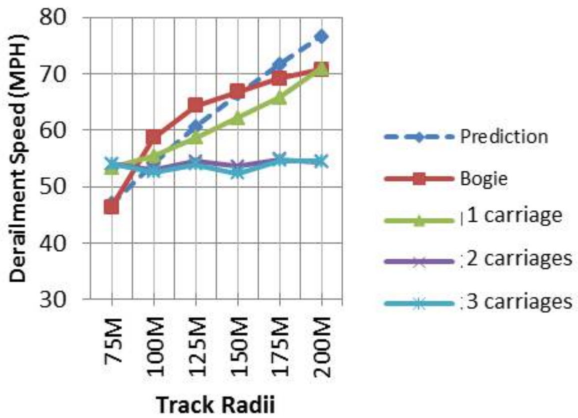

Figure 7: Derailment Speeds Compared to Bogie Predictions.

A single vehicle shows some correlation to the simulated results for a single bogie, but is less stable (and the range of derailment speeds was generally higher). Multiple vehicles are less stable due to the connection between the vehicles affecting each other's stability.

The simulation of a single bogie ran at 60 frames per second (FPS) and was therefore running in real-time. A single carriage averaged 50 FPS, two carriages 40 FPS and three carriages 30 FPS. As we maintain a fixed time step for stability of the physics simulation, this means that the three carriage simulation is running at half speed. However, this is still significantly faster than the high fidelity simulation tools used conventionally.

Flange Collisions

As the Locomotion tool is a physical simulation of the bogie, wheels and rails, it can be utlised to collect additional data on the system. Our results for speed of derailment are sufficiently consistent with the predictions from the Nadal equation to consider the simulation as meaningful. Of particular interest to the rail industry is the number and frequency of flange collisions which occur with the track. When the flange hits the track, there is a risk of degradation to the track.

From our simulation we found that when the bogie is following a curved track to the right, 92.3% of flange collisions occur on the left side track. This is to be expected as the bogie is resting more heavily on the outside (left) wheel, creating greater likelihood of flange collisions on that side. In the case of the simulation with three carriages, it was found that the leading carriage accounted for 26.32% of flange collisions, the middle carriage 44.73%, and the trailing carriage 28.95%. This is consistent with real world expectations, as the central carriage has forces acted upon it from the two surrounding carriages.

CONCLUSIONS AND FURTHER WORK

We have successfully developed a rail dynamics simulation tool using NVidia's PhysX library and techniques from the video games industry. Our tests have shown that our spline-based method of approximating the wheel-rail interface produces results that are close to those predicted using the Nadal formula; the averages over 100 tests were within ±6mph of the predictions. While we carried out 100 tests on each curvature of track to achieve confidence in the average values, this batch testing for the bogie simulations took less than 6 hours to run, which is a considerably faster turnaround than an individual test in the more sophisticated simulation tools used in the rail industry.

Further validation is needed from real train data to determine whether our bogie simulation is sufficiently accurate. However, we believe that our results show that physics libraries for video game simulation have the potential to produce real-time engineering simulations.

Of particular interest is the "fast prototyping" aspect of the work. Utilising video game physics libraries and techniques allows us to simulate multiple scenarios in real time. This allows the rail industry to run simulations based on changing many different parameters in the design of the rolling stock and the track layout. The most successful simulations can then be earmarked for high fidelity simulation. We do not envision our approach as a replacement for existing time-consuming techniques but as an initial step in targeting those techniques. Our partners at NewRail have identified a number of fields where these techniques could be applied, including accident investigation, noise abatement, effects of load distribution and explosion modelling.

REFERENCES

[Anyakwo et al., 2012] Anyakwo, A., Pislaru, C., and Ball, A. (2012). A new method for modelling and simulation of the dynamic behaviour of the wheel-rail contact. *International Journal of Automation and Computing*, 9(3):237–247.

[Evans and Berg, 2009] Evans, J. and Berg, M. (2009). Challenges in simulation of rail vehicle dynamics. *Vehicle System Dynamics*, 47(8):1023–1048.

[Iwnicki, 2003] Iwnicki, S. (2003). Simulation of wheel–rail contact forces. *Fatigue & Fracture of Engineering Materials & Structures*, 26(10):887–900.

[Luo et al., 2009] Luo, F., Liu, C., and Sun, Z. (2009). Intelligent vehicle simulation and debugging environment based on physics engine. In *Informatics in Control, Automation and Robotics, 2009. CAR'09. International Asia Conference on*, pages 329–333. IEEE.

[Marquis and Greif, 2011] Marquis, B. and Greif, R. (2011). Application of nadal limit for the prediction of wheel climb derailment. In *2011 Joint Rail Conference*, pages 273–280. American Society of Mechanical Engineers.

[Monga et al., 2012] Monga, M., Karkee, M., Sun, S., KiranTondehal, L., Steward, B., Zambreno, J., et al. (2012). Real-time simulation of dynamic vehicle models using a high-performance reconfigurable platform. *Procedia Computer Science*, 9:338–347.

[Nadal, 1896] Nadal, J. (1896). *Theorie de la Stabilite des Locomotives*. Dunod et Vicq.

[Polach, 2000] Polach, O. (2000). A fast wheel-rail forces calculation computer code. *Vehicle System Dynamics*, 33:728–739.

[Shabana et al., 2001] Shabana, A. A., Sany, J. R., and Berzeri, M. (2001). Numerical procedure for the simulation of wheel/rail contact dynamics. *Journal of Dynamic Systems, Measurement, and Control*, 123(2):168–178.

[Wang and Li, 2012] Wang, W. and Li, G.-x. (2012). Development of high-speed railway vehicle derailment simulation–part i: A new wheel/rail contact method using the vehicle/rail coupled model. *Engineering Failure Analysis*, 24:77–92.