



TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology



Diploma Thesis

Development of a Contact Model for Magnetic Track Brake

carried out for the purpose of obtaining the academic degree of a
Diplom-Ingenieur

submitted to the Vienna University of Technology

Faculty of Mechanical and Industrial Engineering

by

Alois Steininger BSc

11713107

under the supervision of

Privatdoz. Dr. Eray Arslan

Institute of Mechanics and Mechatronics

and

Univ.Prof. Dipl.-Ing. Dr.techn. Johannes Edelmann

Institute of Mechanics and Mechatronics

Vienna 08.05.2025

Alois Steininger

This work has been supported by the Christian Doppler Association and the company Knorr Bremse GmbH within the framework of the Christian Doppler Laboratory for Enhanced Braking Behaviour of Railway Vehicles. I confirm, that the printing of this thesis requires the approval of the examination board.

Affidavit

I declare in lieu of oath, that I wrote this thesis and carried out the associated research myself, using only the literature cited in this volume. If text passages from sources are used literally, they are marked as such. I confirm that this work is original and has not been submitted for examination elsewhere, nor is it currently under consideration for a thesis elsewhere. I acknowledge that the submitted work will be checked electronically-technically using suitable and state-of-the-art means (plagiarism detection software). On the one hand, this ensures that the submitted work was prepared according to the high-quality standards within the applicable rules to ensure good scientific practice "Code of Conduct" at the TU Wien. On the other hand, a comparison with other student thesis avoids violations of my personal copyright.

Vienna, 08.05.2025

Alois Steininger

Acknowledgments

First and foremost I want to thank my supervisors Privatdoz. Dr. Eray Arslan, Univ.Prof. Dipl.-Ing. Dr.techn. Johannes Edelmann, and also Projektass. Dipl.-Ing. Dr.techn. Emin Kocbay for their support throughout the whole thesis. Your contribution was both insightful and motivating, and when needed constructively critical, whilst always having an open ear for any problems or questions.

I would also like to express my gratitude to the entire research group of the Christian Doppler Laboratory for enhanced braking behaviour of railway vehicles and the Christian Doppler Forschungsgesellschaft itself, for the opportunity and support. This applies, but is not limited to, Projektass. Dipl.-Ing. Bernhard Ebner, Projektass. Dipl.-Ing. Lorenz Klimon and my predecessor on the contact mechanical part Dipl.-Ing. Andreas Pavicsics. Thanks for the helpful advice and constructive dialogue.

In addition, I wish to express my profound gratitude to Knorr-Bremse, without whom the realisation of the whole project, including this thesis, would not have been possible. In this regard, particular gratitude is extended to Dipl.-Ing. Dr.techn. Daniel Tippelt, for the invaluable practical insights you have provided at Knorr-Bremse and for facilitating a generally very good cooperation between the research group and your company.

Finally, I would like to express my deepest gratitude to my girlfriend, my friends and family for their continuous support throughout my entire academic journey. Your unwavering encouragement and emotional support have been crucial for the completion of this journey. In challenging situations and when I was struggling, your motivation kept me going.

Thank you all.

Abstract

A magnetic track brake is an emergency brake for railway vehicles, where braking magnets are pressed against the rail by magnetic attraction. Through this mechanical and frictional sliding contact, the vehicle experiences an additional braking force.

The aim of this study is to analyse the mechanical contact interaction between the pole shoe and the rail within a simplified and numerically efficient methodology. This is done to gain a more profound knowledge regarding the occurring contact forces and elastic displacements. To facilitate this, a numerical contact model between the two bodies in contact is developed and verified. The basic requirements of the derived model are short computation times while maintaining a decent accuracy for the results, especially concerning the local contact forces e.g. normal pressure and shear traction. A short computation time is critical, because the model is developed with the ultimate aim of utilizing it within a multibody dynamics software at a latter stage of the project.

The physical foundation of the implemented model rests on the elastic half-space theory, which is utilised to calculate the displacement for a given distributed load. The aforementioned loads follow by resolving the occurring contact interaction in the form of a set of equations and inequalities, also known as the Signorini problem. These two basic ideas are combined within a discretised setting, leading to the numerical solution of the normal contact problem. Subsequently, the displacements of the surfaces and shear tractions are calculated, using the Coulomb friction idealisation.

By incorporating some assumptions, a basic algorithm is developed to resolve the normal and frictional sliding contact. The thus derived model is first tested on an academic problem for which an analytical solution exists. The algorithm is then specified for the contact between the pole shoe and the rail. The hence conducted comparisons between the different models reveal that the basic contact model needs to account for overall elastic deformations to be suitable for the here considered application of the magnetic track brake. Therefore, a strategy is devised and implemented to extend the model such, that its applicability is augmented, without compromising on the numerical efficiency. The code and algorithm developed, are implemented within MATLAB, due to its simplicity and hands-on style.

In summary, with the newly introduced model, it is possible to calculate the contact pressure distribution and the elastic displacements of the contacting surfaces very efficiently and quickly while maintaining a high accuracy. This is demonstrated by several benchmark and application-oriented problems within this thesis.

Kurzfassung

Die Magnetschienenbremse stellt eine Notbremse für Schienenfahrzeuge dar, deren Funktionsprinzip auf der Verwendung von Bremsmagneten, im englischen als pole shoe bezeichnet, basiert. Diese werden durch eine magnetische Anzugskraft gegen die Schiene gepresst. Durch diesen mechanischen, reibungsbehafteten Gleitkontakt kann eine zusätzliche Bremskraft erzeugt werden.

Ziel dieser Arbeit ist es, diesen Kontakt zwischen dem Bremsmagneten und der Schiene unter Zuhilfenahme eines selbst entwickelten vereinfachten, numerisch effizienten Modells zu analysieren. Hierdurch sollen weiterführende Erkenntnisse über die auftretenden Kontaktkräfte und elastischen Verschiebungen gewonnen werden. Das Kontaktmodell bildet den Kern dieser Arbeit und die Entwicklung und Verifikation dieses wird im Weiteren thematisiert. Die grundlegenden Anforderungen an das Modell sind schnelle Berechnungszeiten bei gleichzeitig hoher Genauigkeit der Ergebnisse, insbesondere bezüglich der lokalen Kontaktkräfte. Die Simulationsdauer ist dabei von Relevanz, da das Modell mit dem Ziel entwickelt wird, es in einer späteren Phase des Projekts in eine Software für Mehrkörpersystemdynamik zu implementieren.

Die physikalische Grundlage des Modells bildet der elastische Halbraum. Dieser wird zur Berechnung der Verschiebungen bei einer gegebenen verteilten Last verwendet. Die zu berücksichtigenden Lasten ergeben sich aus dem Lösen eines Satzes von Ungleichungen, die unter dem Begriff Signorini-Problem bekannt sind und die Kontaktinteraktion beschreiben. Die zuvor genannten grundlegenden Konzepte werden in einer diskretisierten Umgebung kombiniert, um eine numerische Lösung des Normalkontaktproblems zu erhalten. Im Anschluss werden die Verschiebungen der diskretisierten Oberflächen und die Scherspannungen unter der Annahme des Coulomb'schen Reibgesetzes berechnet.

Die Entwicklung eines grundlegenden Algorithmus zur Lösung des Kontaktes erfolgt unter Einbeziehung einiger Vereinfachungen. Das so abgeleitete Modell wird zunächst an einem akademischen Problem getestet, für das eine analytische Lösung existiert. Im Weiteren wird der Algorithmus für den Kontakt zwischen dem Bremsmagneten und einer Schiene spezifiziert. Die Bewertung des Algorithmus erfolgt mittels eines ersten Benchmark-Vergleichs. Dieser zeigt, dass das Kontaktmodell den Kontakt nur bis zu einer gewissen Schichtdicke gut simulieren kann. Um alle auftretenden Phänomene beim Kontakt der Magnetschienenbremse zu berücksichtigen, wird das Kontaktmodell so erweitert, dass es elastische Deformationen des Körpers mit einbezieht. Das erweiterte Modell wird ebenfalls verifiziert und erste Ergebnisse einer Kontaktsimulationen werden präsentiert und thematisiert. Die Implementierung des Algorithmus erfolgt vorerst in MATLAB, da sich diese Software durch einen einfachen und praxisnahen Umgang auszeichnet.

Zusammenfassend lässt sich sagen, dass das neu eingeführte Modell eine effiziente und schnelle Berechnung der Kontaktdruckverteilung und der elastischen Verschiebungen der Kontaktflächen ermöglicht.

Table of contents

Acknowledgments	ii
Abstract	iii
Kurzfassung	iv
1 Introduction	1
1.1 State of the art	2
1.1.1 The magnetic track brake	2
1.1.2 Contact mechanics	2
1.2 Aims and scopes of this thesis	3
2 The contact model	5
2.1 Physical background for the modelling	5
2.1.1 Analytical solution for the elastic half-plane	5
2.1.2 Scattered data approximation	6
2.1.3 Formulating and solving the contact problem	7
2.1.4 Considerations and assumptions	8
2.2 The basic algorithm	9
2.2.1 Structure of the basic algorithm	9
2.2.2 Calculating the \mathbf{R} matrices	12
2.2.3 Resolving the contact in the normal direction of the surface	13
2.2.4 Additional displacements and tractions due to contact	14
2.2.5 Enabling back-deformation	15
2.2.6 Updating the normal directions	15
2.3 Extension of the contact model	15
2.3.1 The simple FE model	16
2.4 Time complexity of the algorithm	19
3 Verification of the algorithm	20
3.1 Verification of the basic algorithm	20
3.1.1 Comparison to an academic problem	20
3.1.2 Benchmarks comparison for the pole shoe rail contact	24
3.2 Verification of the extension	29
3.2.1 Simple contact problem of a block against a flat plane	29
3.2.2 Extension compared to the benchmark results	31
4 Results for the MTB	34
4.1 Results for a constant magnetic load	34
4.2 Results for different loadings	36

5 Conclusion and Outlook	39
References	40
Appendix	43
A Derivation of the time complexity	43
B Additional plots for the basic contact model	45
C Convergence analysis of different parts	47
C.0.1 Convergence of the basic contact algorithm	47
C.0.2 Convergence of the FE model utilised within the extension	48
D Additional plots for the extended contact model	51

List of Figures

1.1	Electromagnet with three pole shoes, see [3]	2
1.2	Sketch of the contact problem between pole shoe and rail, see [21]	4
2.1	Visualization of the penetrating interaction of two objects and their resolved state, see [21]	8
2.2	Schematic overview of the basic contact algorithm	11
2.3	Overview of the iteration process to resolve the contact	12
2.4	Depiction of the \mathbf{R}_z values for a parabola, with a discretisation of the grid of six by six	13
2.5	Overview of the iteration process within the extension to adapt the load increment/ position of the extension within the algorithm	17
2.6	Different meshed pole shoe geometries utilised in the extension, with $\delta_{\text{FE model}} = 0.004\text{m}$	17
3.1	Sketch of the analytical problem of a paraboloid on a half-plane, see [21]	21
3.2	Results of (a) the vertical traction $p(x, y = 0)$, and (b) displacements $u_z(x, y = 0)$ for the elastic plane	22
3.3	Results of (a) the horizontal traction $q_x(x, y = 0)$, and (b) displacements $u_x(x, y = 0)$ for the elastic plane	22
3.4	Cut through the 3-D objects (a) before the deformation, (b) after the deformation, and (c) after the deformation with a scaling factor of 2500 for the horizontal displacement	23
3.5	Sketch of the contact problem used as a benchmark, see [21]	24
3.6	Contact pressure distribution, basic contact algorithm compared to the FE benchmark results	26
3.7	Contact pressure distribution, basic contact algorithm compared to the FE benchmark results at four more cross-sections	26
3.8	Vertical displacement, contact algorithm compared to the FE benchmark results	27
3.9	Vertical displacement, comparison with benchmark results at four more cross-sections	27
3.10	Contact pressure distribution, benchmark comparison for $\xi = 28.2\text{mm}$	28
3.11	Vertical displacement, benchmark comparison for $\xi = 28.2\text{mm}$	28
3.12	Contour of a relative error of 10% for the contact pressure distribution, for the comparison between the results from the contact algorithm with the FE benchmark for different values of ξ	29
3.13	Illustration of the contact problem for a block on a flat plane	30
3.14	Contact pressure distribution of the block, comparison Ansys against extended model	31
3.15	Vertical displacement of the block, comparison Ansys against extended model	31
3.16	Deformation of the block in the vertical direction, side view for the two different simulations	32
3.17	Contact pressure distribution, benchmark comparison at four cross-sections for $\xi = 28.2\text{mm}$	33

3.18	Vertical displacement, benchmark comparison at four cross-sections for $\xi = 28.2\text{mm}$	33
4.1	Contact status for different friction coefficients, results for the full pole shoe simulation	34
4.2	Contact pressure distribution, result from the extended model for $\mu = 0.5$	35
4.3	Vertical displacement, result from the extended model for $\mu = 0.5$	35
4.4	Depiction of the five discrete points at the top, with a prescribed movement of zero for y and z direction, and the four points at the back side, with a prescribed movement of zero for the x direction.	36
4.5	Contact pressure distribution, result from the extended model for a Gaussian-distributed magnetic loading	38
4.6	Vertical displacement, result from the extended model for a Gaussian-distributed magnetic loading	38
A.1	Sketch on how the time complexity is derived	44
B.1	Relative error of the contact pressure distribution, benchmark comparison for $\xi = 1.15\text{mm}$	45
B.2	Relative error of the vertical displacement, benchmark comparison for $\xi = 1.15\text{mm}$	45
B.3	Contact pressure distribution, basic contact algorithm benchmark comparison for $\xi = 7.45\text{mm}$	46
B.4	Vertical displacement, basic contact algorithm benchmark for $\xi = 7.45\text{mm}$	46
C.1	Contact pressure distribution for different grids, cross-section at $y = 0.012\text{m}$ and $y = 0.006\text{m}$	47
C.2	Vertical displacement for different grids, cross-section at $y = 0.012\text{m}$ and $y = 0.006\text{m}$	48
C.3	Contact pressure distribution, convergence of the FE model in the extension	49
C.4	Vertical displacement, convergence of the FE model in the extension	49
D.1	Contact status, Ansys results and extended contact algorithm	51
D.2	Contact pressure distribution, result from the extended model for $\mu = 0.25$	52
D.3	Vertical displacement, result from the extended model for $\mu = 0.25$	52
D.4	Contact pressure distribution, result from the extended model for an arbitrary non-constant magnetic loading	53
D.5	Vertical displacement, result from the extended model for an arbitrary nonconstant magnetic loading	53

List of Tables

2.1	Summary of all simulation parameters within the contact algorithm	18
3.1	Simulation parameters for the academic problem, paraboloid on a flat plane	21
3.2	Calculation times for the academic problem	23
3.3	Simulation parameters for the benchmark problem, pole shoe on rail	25
3.4	Error values for the contact pressure distribution and the vertical displacement, and computation times for the FE benchmark compared to the basic contact algorithm	28
3.5	Simulation parameters for the simple extension, block on plane	30
3.6	Simulation parameters for the extended contact algorithm, benchmark problem pole shoe on rail	32
3.7	Error values for the contact pressure distribution and the vertical displacement, and computation times for the FE benchmark and the extended contact algorithm (and the basic model)	32
4.1	Calculation times for the extended model, for varying friction coefficients	36
4.2	Simulation parameters for simulation with non-constant magnetic loading, pole shoe on rail	37
C.1	Calculation times for benchmark problem, pole shoe on rail	48
C.2	Calculation times for different FE meshes, within the extended contact algorithm .	50
C.3	Qualitative comparison of the vertical deformation of the pole shoe, for a converging and diverging extension	50

1. Introduction

In recent years the emerging climate crisis has made sustainable means of transport more important and popular. The transportation sector accounts for approximately a quarter of global energy consumption, as described in [1]. The various ways to have a positive impact on the climate and the state of this transition towards more ecologically friendly transportation methods are discussed in greater detail there. One possible contribution towards a "greener" future is to increase the volume of railway transportation because it is more energy efficient compared to conventional automotive transportation methods. This is attributed, among others, to the reduced rolling resistance for railway vehicles, which is e.g. discussed in [2]. The electrification of railway mobility leads to a significant reduction in direct carbon dioxide emissions, with the potential for zero emissions in the near future, provided that renewable sources of electric energy are used. In order to achieve this, many infrastructure projects are implemented across the globe with the objective of further improving the railway network, particularly in light of the extension of high-speed tracks.

To enhance safety and allow for higher velocities and hence shorter travelling times, it is of substantial importance to continue improving the braking systems of trains, trams and metros. In this light, it is imperative to explore and implement additional braking mechanisms, particularly in railway applications. One suitable option for generating additional brake forces is the magnetic track brake (hereafter referred to as the MTB). It serves as an emergency brake, facilitating an additional contact area between the rail and a braking magnet mounted under the bogie.

To improve the understanding of the braking behaviour of railway vehicles in general and the MTB in particular, the "Christian Doppler Laboratory for Enhanced Braking Behaviour of Railway Vehicles"¹ was brought to life. There, the three research partners, the Christian Doppler Association, the Vienna University of Technology, and the commercial partner Knorr Bremse GmbH collaborate to form a dedicated research group. The main topics of interest are the magnetic and mechanical aspects of the magnetic track brake, the dynamic behaviour of the brake frame structure, and the system integration and interaction with other braking systems, including sanding and disk brake equipment. Furthermore, this research unit investigates the impact of external factors such as an ill-conditioned track surface (e.g., water, leaves) and the passage of switches or crossings. As part of this research, this work was initiated with the objective of investigating the mechanical contact within a simplified modelling approach.

To gain insight into the task addressed in this thesis, it is necessary first to provide a brief overview of the fundamental braking system and operational methodology of the MTB. Multiple electromagnets are linked by longitudinal and lateral rods to create a brake frame, which is subsequently mounted under multiple bogies via pneumatic cylinders. Typically, one of these electromagnets is situated between the two wheelsets of a bogie, consisting of pole shoes, endpieces and a coil encased in a coil body. This is shown in Fig. 1.1. As a consequence of the magnetic attraction, the electromagnets are pressed against the rail, thereby generating supplementary braking forces through the facilitation of a distinct contact patch.

¹The Christian Doppler Research Association (CDG) is a non-profit association promoting the cooperation between science and businesses through laboratories for application-orientated basic research projects. The specific webpage for the research project mentioned here, can be found at <https://www.cdg.ac.at/en/research-units/labor/enhanced-braking-behaviour-of-railway-vehicles>.

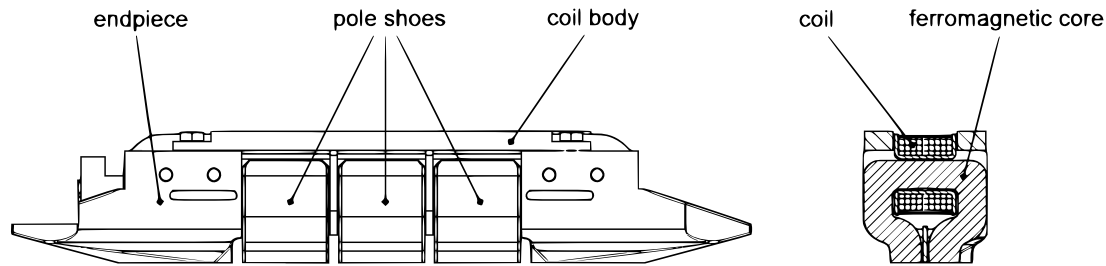


Figure 1.1: Electromagnet with three pole shoes, see [3]

1.1 State of the art

1.1.1 The magnetic track brake

The general effectiveness of the magnetic track brake and its capability of impacting the rail condition, are subjected to detailed analysis and discussion, for example in [4] and [5]. The dynamic behaviour of the brake frame was investigated in e.g. [3]. There, a particular focus was laid on the effects of decreasing speeds during the braking manoeuvre. Self-excited vibrations, how they arise and how to actively control them, is investigated in [6]. The basic idea of integrating multiple components of the MTB into a unified simulation, containing both an electromagnetic and a mechanical aspect was done in [7]. However, the findings demonstrated a potential avenue for enhanced modelling of individual components, thereby facilitating a more profound comprehension of the underlying interactions.

1.1.2 Contact mechanics

In 1882, Hertz conducted an investigation into the interaction between two elastic objects in contact. His findings, which are relevant and essential even to this day, are documented in [8]. They revealed a solution for the contact pressure under the assumption of frictionless contact for bodies of revolution and for bodies in which the contacting surfaces may be approximated well by quadratic functions. Since then, a variety of analytical and numerical methodologies have been elaborated and are currently used in contact simulations.

The most common approach, within solid mechanics, is the finite element method (FEM) that allows for the implementation of various methods to enforce contact constraints. These include among others the mixed method, the direct constrained method, and the regularisation method. The first method explicitly introduces the Lagrange multipliers and necessitates spatial as well as force-like degrees of freedom, hence the name mixed method. It is utilised and discussed e.g. in [9], [10]. For the second method the constraints are not enforced via the Lagrangian multipliers but by compatibility of the displacement fields, which has to be done beforehand, see e.g. [11], [12]. The third method neither uses Lagrangian multipliers nor assumes compatible displacement fields [13], [14]. Therefore the results allow for a slight infringement of the constraints. The traction is assumed to be a function of the normal distance where large penetration depths lead to larger required normal forces. This approach gives rise to a number of notable methodologies, including the penalty method [13] and the augmented Lagrangian method [14] (depending on the chosen penalty function). A more detailed account of the various methodologies and mathematical formulations employed in the implementation of contact for the FEM software can be found in [15]. The aforementioned types of the regularisation method are employed for example in the

commercial finite element software Abaqus/CAE ² for the simulation of contact interactions. In order to enforce friction and thereby calculate the frictional traction, a predictor-corrector scheme is applied. For further details, refer to the Abaqus manual [16].

The most significant drawback of simulating contact within a finite element environment is the generally high computational cost, which is the price to pay for the obtained excellent accuracy. This is shown e.g. in [17] for a simple contact problem. There, it is compared with an alternative approach for dealing with contact, namely the boundary element method (BEM). The BEM is a numerical tool that uses influence-coefficients derived from analytical solutions to determine the relationship between force and displacement at any discrete points of the surfaces involved. Although it comes with strict restrictions or requirements, the main advantage lies in the fact that only the surfaces of the contacting bodies need to be discretized, with disregard to the interior. It is used e.g. in the computer software CONTACT, developed by Vollebregt, see its user guide [18]. In fact, most of the simulations of contact problems for railway vehicles are performed using half-space methodologies and, frequently, the BEM, as mentioned in [19].

Another, historically significant, contribution was made by Kalker et al. [20]. They investigated the rolling contact and, among other things, developed an algorithm called FASTSIM, which combines multiple different numerical and analytical approaches.

Within this here presented approach, only the surfaces are discretized and modelled for simulating the contact interaction. The formulation of the contact problem is similar to the BEM, while certain assumptions and simplifications are used, to specify the developed model for the particular application of contact between MTB (pole shoe) and rail. Through that, a reduced computation is achievable, which highlights the practical relevance of proposed methodology. Moreover, with the employed two-step solving approach, even overall elastic deformations within the respective contacting bodies, may be computed to some extent. In further studies this newly developed model is going to be integrated in a comprehensive simulation of the MTB. This unified simulation will include both, contact interaction as well as magnetic attraction, modelled in an accurate way, within a multibody dynamics environment.

1.2 Aims and scopes of this thesis

The main objective of this work is to develop an application-oriented algorithm capable of modelling the mechanical contact between the pole shoe and rail, see Fig. 1.2, in a simple and computationally fast way.

The present study is solely concerned with the mechanical aspects, neglecting the magnetic attraction and considering it only as a mechanical loading. As part of the research project, this should establish a good understanding of the contact interaction between the MTB and the rail. The fundamental methodology is similar to the boundary element method, in a sense that only the surfaces of the involved contacting bodies are discretised and with the underlying physical principles established through the elastic half-space approach. For a detailed derivation of the contact model, see section 2.1. Provided that the assumptions for its applicability are met, the algorithm is capable of resolving contact between arbitrarily shaped bodies accurately and efficiently. After testing the basic principle and functionality on a general contact problem, it is specified and extended to be applicable to the contact between the pole shoe and rail.

²Abaqus <https://www.3ds.com/products/simulia/abaqus>

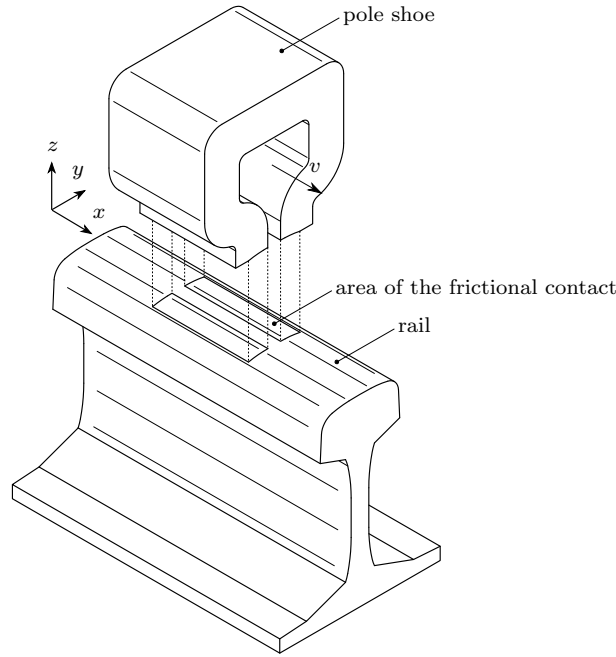


Figure 1.2: Sketch of the contact problem between pole shoe and rail, see [21]

The model is established within the MATLAB³ software package. Post-processing for visualisations and additional calculations are also done there via subroutines, for the sake of convenience. The model is validated with different reference solutions, one of them being an FE model set up at an earlier stage of the research project, see [22]. Through the newly developed model, influences of e.g. velocity and magnetic loading can be analysed efficiently and quickly.

Subsequently, at a later stage of the project it is planned that the developed algorithm will be implemented as a subroutine for the multibody dynamic software SIMPACK⁴ (possibly via Fortran). There the results of the equations of motion may be incorporated and a transient braking manoeuvre could be simulated. Hence, the time efficiency of the algorithm is of high importance. As this realisation exceeds the scope of this thesis, it is therefore not further discussed here.

³MATLAB <https://de.mathworks.com/products/matlab.html>

⁴SIMPACK <https://www.3ds.com/de/produkte-und-services/simulia/produkte/simpack/>

2. The contact model

In this chapter the fundamentals, on which the contact model is built upon, are explained in more detail. The basic equations, which are used for the contact model, and some sketches for visualisation are presented herewith. Additionally, the different parts of the algorithm and the overall workflow are described. The contact model outlined in this work is also the subject of a publication authored by the research group [21]. The notation used in the following section is equivalent to the one used in the prior mentioned publication.

2.1 Physical background for the modelling

2.1.1 Analytical solution for the elastic half-plane

The physical background is set by Boussinesq and Cerruti [23]. The Boussinesq problem analytically describes the deformation of an infinite elastic half-space due to a concentrated point load in the normal direction. Within the basic model, the fundamental assumption is made that only elastic deformations and stresses on the surfaces are non-trivial, with no consideration given to the interior of the bodies. Consequently, the term "half-plane" shall be employed in subsequent discussions instead of "half-space". This simplification is justified as the field variables are reasonably small, and therefore negligible, in the interior of the bodies, within this particular application of the MTB. The equation for the displacement field $\mathbf{u}^P(\mathbf{x}, \mathbf{y})$ of the half-plane is therefore given by:

$$\mathbf{u}^P(\mathbf{x}, \mathbf{y}) = u_x^P(\mathbf{x}, \mathbf{y})\mathbf{i} + u_y^P(\mathbf{x}, \mathbf{y})\mathbf{j} + u_z^P(\mathbf{x}, \mathbf{y})\mathbf{k} = \frac{P(\mathbf{x})}{2\pi G} \left(-\frac{1-2\nu}{2} \frac{(x'-x)}{\|\mathbf{x}-\mathbf{y}\|^2} \mathbf{i} - \frac{1-2\nu}{2} \frac{(y'-y)}{\|\mathbf{x}-\mathbf{y}\|^2} \mathbf{j} + \frac{1-\nu}{\|\mathbf{x}-\mathbf{y}\|} \mathbf{k} \right). \quad (2.1)$$

Here, G and ν denote the shear modulus and the Poisson's ratio of the plane, respectively, and \mathbf{i} , \mathbf{j} and \mathbf{k} define the base vectors of the Cartesian coordinate system. Further, \mathbf{x} and \mathbf{y} define the in-plane position vector for the source point, on which the point load $P(\mathbf{x})$ is acting, and the field point \mathbf{y} for which the displacement is calculated. They are defined through their components with $\mathbf{x} = x\mathbf{i} + y\mathbf{j}$ and $\mathbf{y} = x'\mathbf{i} + y'\mathbf{j}$, respectively.

Cerruti formulated an analytical solution for the displacement field $u_x^{Q_x}(\mathbf{x}, \mathbf{y})$ due to a shear force $\mathbf{Q}_x(\mathbf{x})$ acting onto the half-plane (in the tangential direction). It is described here for one direction of the shear force but due to symmetry the other in-plane directions follow by a rotation of the obtained displacement field and the force. This allows for the account of friction later in the model and the fundamental formula for this can be expressed as follows:

$$\mathbf{u}^{Q_x}(\mathbf{x}, \mathbf{y}) = u_x^{Q_x}(\mathbf{x}, \mathbf{y})\mathbf{i} + u_y^{Q_x}(\mathbf{x}, \mathbf{y})\mathbf{j} + u_z^{Q_x}(\mathbf{x}, \mathbf{y})\mathbf{k} = \frac{Q_x(\mathbf{x})}{2\pi G} \left(\left(\frac{1-\nu}{\|\mathbf{x}-\mathbf{y}\|} + \frac{\nu(x'-x)^2}{\|\mathbf{x}-\mathbf{y}\|^3} \right) \mathbf{i} + \frac{\nu(x'-x)(y'-y)}{\|\mathbf{x}-\mathbf{y}\|^3} \mathbf{j} + \frac{(1-2\nu)(x'-x)}{2\|\mathbf{x}-\mathbf{y}\|^2} \mathbf{k} \right). \quad (2.2)$$

In general, the normal and tangential problems are fully coupled and can not be considered independently. The basic analytic derivation of the Boussinesq and Cerruti solutions, and the assumptions made, can be found in [23]. A more in depth explanation on its utilisation in a numerical context can be found in [24]. There one can also find different ways of implementing contact, and a comparison of BEM and FEM with suitable illustrations and examples.

When distributed loads are considered, the solution is derived as the superposition of individual point loads. Subsequently, the sum of all individual solutions is determined as the surface integral

over the half-plane surface Ω . For the displacement field $u_z^p(\mathbf{y})$ in normal direction due to a distributed normal load $p(\mathbf{x})$ the equation yields:

$$u_z^p(\mathbf{y}) = \frac{1-\nu}{2\pi G} \int_{\Omega} \frac{p(\mathbf{x})}{\|\mathbf{x} - \mathbf{y}\|} d\mathbf{x}. \quad (2.3)$$

For the other direction of loading and displacement, the adaption for distributed loads follows analogously.

The so-obtained, analytical solutions can be employed in a discrete setting and even for geometries other than ideal half-planes, albeit certain assumptions must be made. Nonetheless, the physical groundwork remains intact, and the solution can be calculated efficiently. For more details on the necessary assumptions, see [21] and [25].

2.1.2 Scattered data approximation

Using the method of scattered data approximation, the distributed loads are discretised through special functions. This leads to the distributed loads $p(\mathbf{x})$, $q_x(\mathbf{x})$ and $q_y(\mathbf{x})$ being defined as

$$p(\mathbf{x}) \approx \sum_{j=1}^N p^j \Phi^j(\mathbf{x}), \quad (2.4.1)$$

$$q_x(\mathbf{x}) \approx \sum_{j=1}^N q_x^j \Phi^j(\mathbf{x}), \quad (2.4.2)$$

$$q_y(\mathbf{x}) \approx \sum_{j=1}^N q_y^j \Phi^j(\mathbf{x}), \quad (2.4.3)$$

where

$$\Phi^j(\mathbf{x}) = \frac{\omega^j(\|\mathbf{x} - \mathbf{y}^j\|)}{\sum_{k=1}^N \omega^k(\|\mathbf{x} - \mathbf{y}^k\|)}. \quad (2.5)$$

$\Phi^j(\mathbf{x})$ is defined as the shape function for the data approximation, with j denoting the respective discrete point, where a traction is applied, and N the maximal number of discrete points, see [25]. The functions $\omega^j(\|\mathbf{x} - \mathbf{y}^j\|)$ used within this shape function are truncated Gaussians, which are chosen to cover the surface area, and through that guarantee a hole-free coverage. An approximation based on radial basis functions, such as is the case here, is often convenient for scattered data, see [26]. However, such an approach would not be necessary if the grid and the geometry are well-defined.

With these definitions, it is possible to set up so-called \mathbf{R} matrices, linking the displacements and tractions for each direction. For this, the equations of the elastic half-plane theory for distributed loads are reformulated. The surface integrals over the contact area are calculated beforehand, and the sum over the individual discrete loads is tackled afterwards. This leads to the following definitions for all different matrices:

$$R_{xx}^{ij} = \frac{1}{2\pi G} \int_{\Omega} \left(\frac{1-\nu}{\|\mathbf{x} - \mathbf{y}^i\|} + \nu \frac{(x'^i - x)^2}{\|\mathbf{x} - \mathbf{y}^i\|^3} \right) \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.1)$$

$$R_{yx}^{ij} = \frac{1}{2\pi G} \int_{\Omega} \nu \frac{(x'^i - x)(y'^i - y)}{\|\mathbf{x} - \mathbf{y}^i\|^3} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.2)$$

$$R_{zx}^{ij} = \frac{1-2\nu}{4\pi G} \int_{\Omega} \frac{(x'^i - x)}{\|\mathbf{x} - \mathbf{y}^i\|^2} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.3)$$

$$R_{xy}^{ij} = \frac{1}{2\pi G} \int_{\Omega} \nu \frac{(x'^i - x)(y'^i - y)}{\|\mathbf{x} - \mathbf{y}^i\|^3} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.4)$$

$$R_{yy}^{ij} = \frac{1}{2\pi G} \int_{\Omega} \left(\frac{1 - \nu}{\|\mathbf{x} - \mathbf{y}^i\|} + \nu \frac{(y'^i - y)^2}{\|\mathbf{x} - \mathbf{y}^i\|^3} \right) \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.5)$$

$$R_{zy}^{ij} = \frac{1 - 2\nu}{4\pi G} \int_{\Omega} \frac{(y'^i - y)}{\|\mathbf{x} - \mathbf{y}^i\|^2} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.6)$$

$$R_{xz}^{ij} = -\frac{1 - 2\nu}{4\pi G} \int_{\Omega} \frac{(x'^i - x)}{\|\mathbf{x} - \mathbf{y}^i\|^2} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.7)$$

$$R_{yz}^{ij} = -\frac{1 - 2\nu}{4\pi G} \int_{\Omega} \frac{(y'^i - y)}{\|\mathbf{x} - \mathbf{y}^i\|^2} \Phi^j(\mathbf{x}) d\mathbf{x}, \quad (2.6.8)$$

$$R_{zz}^{ij} = \frac{1 - \nu}{2\pi G} \int_{\Omega} \frac{\Phi^j(\mathbf{x})}{\|\mathbf{x} - \mathbf{y}^i\|} d\mathbf{x}. \quad (2.6.9)$$

Regarding the definition of the subscript, the primary indicates the direction of the displacement and the secondary is the direction of the traction. The superscript i refers to the discrete grid point of the displacement and j to the respective grid point of the applied traction. Due to the definition of the \mathbf{R} matrices it follows that

$$R_{xy}^{ij} = R_{yx}^{ij}, \quad -R_{xz}^{ij} = R_{zx}^{ij}, \quad -R_{yz}^{ij} = R_{zy}^{ij}. \quad (2.7)$$

The final equation links the traction at the point j and the displacement at the point i through a process of a matrix multiplication, displayed below

$$\begin{pmatrix} u_x^i \\ u_y^i \\ u_z^i \end{pmatrix} = \begin{pmatrix} R_x^{ij} & R_{xy}^{ij} & R_{xz}^{ij} \\ R_{yx}^{ij} & R_y^{ij} & R_{yz}^{ij} \\ R_{zx}^{ij} & R_{zy}^{ij} & R_z^{ij} \end{pmatrix} \cdot \begin{pmatrix} q_x^j \\ q_y^j \\ p^j \end{pmatrix}. \quad (2.8)$$

It can be condensed to

$$\mathbf{u}^i = \mathbf{R}^{ij} \cdot \mathbf{f}^j, \quad (2.9)$$

with the traction vector \mathbf{f} at the point j and the displacement vector \mathbf{u} at the point i .

In other derivations, the relation between traction and displacement is also referred to as the flexibility and a comparison with the Green's function can be drawn. Under the assumption of small deformations, the \mathbf{R} matrices are assumed to be constant. Consequently, these matrices will not be updated during the simulation and can be pre-calculated for the geometries in contact, resulting in a substantial increase in computation time. Within this work, the matrices are computed according to the elastic half-space theory. However, for more complex cases, the identification could be accompanied by an FE simulation, which would further enhance practicality.

2.1.3 Formulating and solving the contact problem

Up to now, contact has not been tackled, and merely the connection between displacement and traction was derived for a discrete setting. Thus far, the derived formulations provide the displacement response for certain tractions in a straightforward manner. However, having a contact problem at hand, the tractions are not known beforehand and followed by first resolving the contact interactions, which may be formulated as a set of equations and inequalities, also referred to as Signorini problem. The bodies are permitted to penetrate each other at first. Subsequently, this intersection is resolved, applying the constraints through these equations and inequalities.

In the main equation (2.10) the separation s (sometimes also denoted as normal gap function), is calculated as the sum of the initial penetration depth h and the normal displacement due to traction \mathbf{u}_z , see (2.8).

$$\mathbf{s} = \mathbf{R}_z \cdot \mathbf{p} + \mathbf{h}. \quad (2.10)$$

The constraints are such that the separation and the normal traction must be positive (defining the distance and normal traction outwards of the objects as positive). Moreover, it is imperative that either the separation (2.11.1) or the normal traction (2.11.2) is equal to zero at each individual point i . If there is no contact, and therefore a gap between the bodies, the traction is non-existent, and vice versa (2.11.3). These constraints can be listed in terms of mathematical descriptions:

$$s^i \geq 0, \quad (2.11.1)$$

$$p^i \geq 0, \quad (2.11.2)$$

$$\mathbf{s} \cdot \mathbf{p} = 0. \quad (2.11.3)$$

This set of equations and inequalities form a problem, in the literature also referred as a linear complementarity problem (LCP) see [27]. In Fig. 2.1 this procedure is visualised schematically.

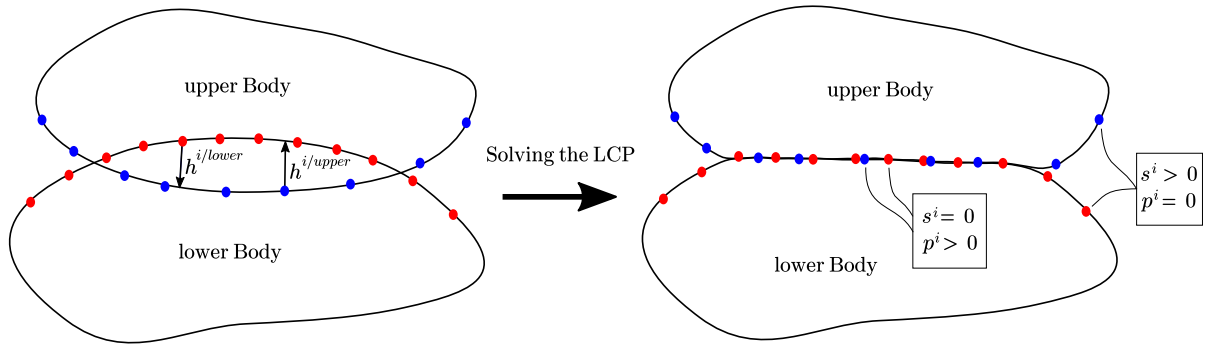


Figure 2.1: Visualization of the penetrating interaction of two objects and their resolved state, see [21]

It is noted that this formulation of the contact constraints is a specific case of the Karush-Kuhn-Tucker condition for optimality of the complementary energy, e.g. described in chapter 12 of [28].

The discrete points have, in general, no direct counterpoint on the surface of the opposing object, and the normal penetration depth is determined through approximating the other surface. Because the exact penetration depth is not known beforehand, it is approached through the minimal distance to the surface. The LCPs are solved for each individual object separately and the procedure is done iteratively, to ensure no penetration remains in the final resolved state.

Within this iteration, not only the normal displacement but also the additional deformations in the lateral direction due to traction in the normal direction (i.e. lateral contraction) are considered through the equation (2.8). After the normal traction is calculated, a friction law can be applied to calculate the shear traction. With this, the displacement due to friction is calculated, again utilising the relation between the displacement and the traction, defined before in (2.8).

2.1.4 Considerations and assumptions

Before introducing the algorithm in detail, the general assumptions and remarks for constructing the algorithm are listed here.

- Linear elastic material behaviour in the vicinity of the contact surfaces is assumed.
- The deformations are expected to be small compared to the geometry and therefore the \mathbf{R} matrices are not updated but computed beforehand. This leads to an immense increase in computational efficiency.
- The normal and tangential problems are not considered completely coupled, as the physical derivation would demand, but are solved one after the other. Through the iteration when resolving the indentation, and also because of the small penetration depth, it is assumed that the contact model nevertheless resembles a physically valid simulation.
- For the tangential problem, it is assumed that for all discrete points in contact, sliding can be assumed. This is equivalent to assuming a quasi-steady gliding condition. If needed, a condition for checking the contact status of each point (sticking or sliding) could be implemented, although this would lead to a higher computational effort.
- For tangential traction, the Coulomb friction law is assumed due to its simplicity, but if necessary, a different constitutive law that connects normal and tangential traction could be implemented.
- To calculate a deformed state of the objects in contact, multiple small load increments are applied and resolved one after the other. This approach is chosen to approach the nonlinearity of the contact problem by splitting it into multiple small linear problems.
- The normal directions of the objects are initially established for the undeformed state, but updated after each small load increment. Only small deformations and small changes of the normal directions of the surfaces are expected, within a single load increment. Therefore, the error due to not knowing the exact normal direction within each increment is expected to be very small and can be neglected.

2.2 The basic algorithm

Following the presentation of the underlying principles in the previous section, in this section some more details on the implementation of the subroutines and the algorithm itself are given. In addition to the main algorithm, pre- and post-processing routines are established, however they are not further described.

2.2.1 Structure of the basic algorithm

The basic way in which the algorithm works is described here. For an illustration in the form of a basic flowchart, see Fig. 2.2.

- The simulation is initialised, defining the objects in contact, the applied external loadings and other modelling and simulation parameters. For an extended description of the input, see below 2.2.1.
- Subsequently, the \mathbf{R} matrices are loaded into the simulation. If the geometry or discretisation is newly established, the matrices have to be computed and saved, resulting in a larger computation time.

- Following the definition and establishment of all variables, the load is applied incrementally through intersections and the resolving of those. After every load increment, the global forces are calculated and compared to the external loading applied.

The resolving of each individual load increment has to be done iteratively. This is necessary, because the resolving procedure uses the minimal distance and not the exact distance of indentation. A flowchart of this detail is provided in Fig. 2.3. The main steps of this iteration are the following:

- First, it is checked if the objects intersect. If no contact is detected, the iteration halts and the next load increment is applied.
 - When contact is detected, the minimal distance to the opposing surface is calculated.
 - This, so estimated penetration depth, is then resolved in the normal direction of the surface (for each discrete point). This is done via the in 2.1.3 established equations.
 - The resulting displacements are calculated for all direction due to the normal traction.
 - Through friction, and hence the friction factor, the tangential traction, and with that the displacements due to this traction are calculated.
- Subsequently, the process of elastic back deformation is implemented for the grid points that have experienced loss of contact. The normal tractions, and with them the tangential tractions, of each discrete point, where no contact occurs after the current increment but contact existed after the latest increment, are reduced or removed. Whilst removing the tractions, the surfaces deform back into their original geometry or regain contact.
 - Following the complete resolution of one load increment, the discrete points describing the object are updated. The normal vectors are updated according to the newly computed surface using this geometry.
 - When the external loading and the forces, resulting from the deformation match (up to a certain predefined tolerance), the incrementation procedure comes to an end.
 - The so-received results are polished and illustrated further, in the post-processing.

It is noteworthy that the grids do not have to be conforming for this method to work, which further increases the practicality. The "resolved" state of the objects after one single iteration may still intersect, albeit to a lesser extent, due to the minimal indentation depth used. This results in the necessity of an iteration. In practice, about three iterations have been found to be sufficient in most cases.

In the initialisation phase, all user inputs have to be defined and a more detailed definition of those is given here. This input comprises geometric parameters, including the geometry of the objects in contact, as well as the discretisation of those objects. The discretisation is defined by specifying an in-plane grid in the x-y plane of the form $z(x, y) = z^i(x^i, y^i)$. For the use cases facilitated here and to enclose the contact patch, a border is introduced, surrounding the grid. This is determined by manually estimating, which parts of the objects may come into contact. These border points are set to never get into contact and are later excluded from the actual results. For the magnetic track brake, the objects in contact are the pole shoe and the rail, although only the geometry of the top surface of the rail and the bottom surface of the pole shoe are needed for the

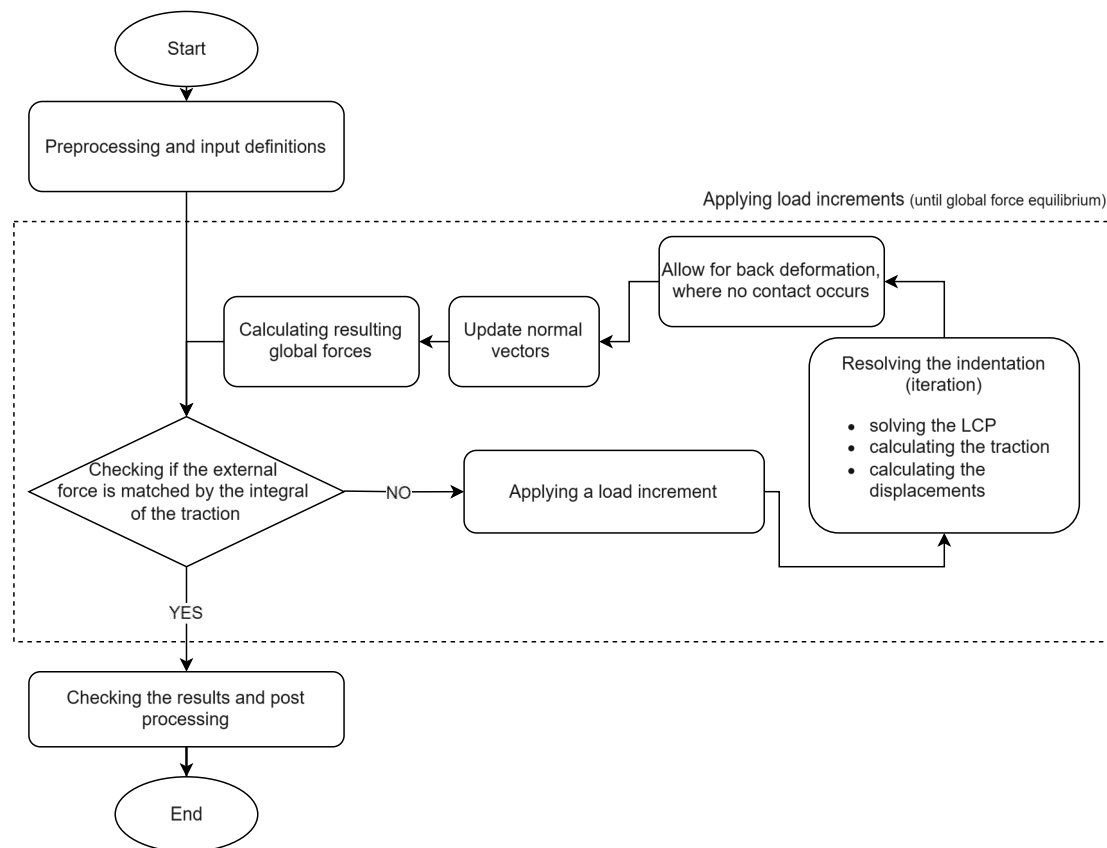


Figure 2.2: Schematic overview of the basic contact algorithm

basic contact simulation. This is because the overall motion of the bodies is expected to be rigid body motions and elasticity is only permitted at the contact surfaces.

In addition, the basic material parameters of both objects, in the form of the poisson ratio ν and shear modulus G have to be defined. With those inputs, the already mentioned \mathbf{R} matrices can be calculated. For more details see 2.2.2.

For simulations with different loads, some of the simulation parameters introduced here may have to be adapted for the algorithm to run smoothly. However, it is noteworthy that in the majority of use cases, it is possible to leave a set of values unchanged. These parameters are the critical distance for contact detection, regarding the resolving as well as the back deformation processes, and the tolerated discrepancy between global forces. Furthermore, it is necessary to specify the number of steps, into which the penetration is subdivided, even though this number is only used as an estimation.

The external loading can be given as a single force or a distributed load. The original concept for this external loading involves pressing one object into the other. Additionally, in a more advanced simulation the attraction of the rail up towards the pole shoe could be simulated by a movement of the rail, however, within this thesis this possibility is not further pursued.

To consider the frictional case and therefore include shear traction, a friction coefficient and the direction of the sliding movement are needed. For simulating the contact of a magnetic track brake with the rail, the algorithm can be fed with the vehicle velocity. This can then determine a predefined friction coefficient (as the friction coefficient is known to be velocity dependent, see e.g. [29]) and magnetic loading (and its distribution), from a provided lookup table. Those

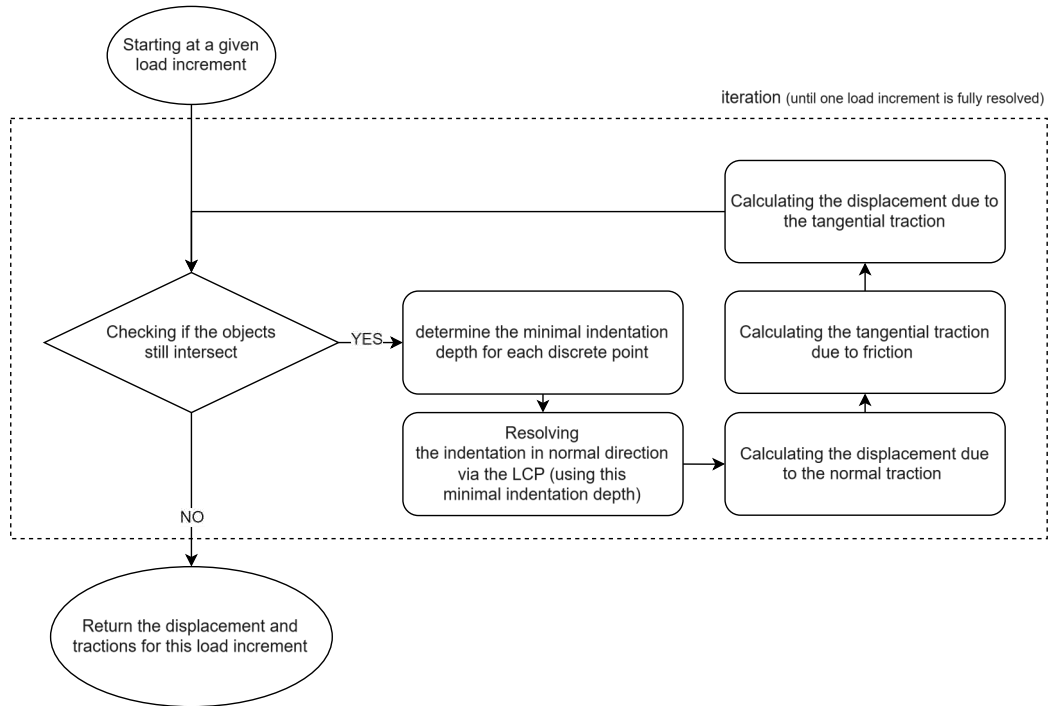


Figure 2.3: Overview of the iteration process to resolve the contact

aforementioned dependencies of the friction coefficient and magnetic loading of the velocities can be obtained through experiments and/or the utilisation of different simulation approaches throughout the research project. The full data set has not been obtained and implemented within the scope of this study, only the principle architecture for the implementation has been outlined.

After establishing the basic structure of the algorithm and its input parameters, in the following subsections some important subroutines, already mentioned above, are described in more detail.

2.2.2 Calculating the \mathbf{R} matrices

The \mathbf{R} matrices are established through surface integrals, see (2.6.1) - (2.6.9). To calculate the entries for these matrices, a subroutine is defined. The numerical integration of these surface integrals is performed with a finer grid, encompassing multiple points between the predefined discrete points (not equidistant). In the shape function $\Phi(\mathbf{x})$, the truncated Gaussians ω are used to establish the scattered data approximation. For those, only the norm of the difference $\|\mathbf{x} - \mathbf{y}\|$ is used, and therefore the mean value μ is set equal to zero. For the standard deviation, a fixed parameter is established, which is adapted according to the mesh density. After some iterations and adjustments to the standard deviation, it became clear that it has no influence on the stimulation outcome, once it is chosen to guarantee a hole-free coverage.

Through the matrices every point is assigned a value to account for the interaction with every other point in every direction, and itself. Consequently, for k discrete points, every \mathbf{R} matrix consists of k columns and k rows. In the subroutine this leads to two nested loops, one within the other, evaluating k by k numerical integrals and filling up the matrices. When accounting for all symmetries of the matrices, a total of six surface integrations per object must be computed. An example of such a matrix, from a later introduced contact problem of a paraboloid, is illustrated in Fig. 2.4. The underlying discretisation is chosen as a grid of six points in each horizontal direction, leading to $k = 36$. Typically, the matrices used are larger, but to keep the figure simple and clear,

a smaller version is depicted here. From Fig. 2.4 it is clearly visible that the highest influence that a point faces is at the point itself. With an increase in distance the influence of the traction on the displacements decreases. Additionally, it is possible to see a certain pattern that is caused by how the values are distributed, according to an underlying grid for the discretisation. The material parameters influence the matrices as factors, and enter the calculation of \mathbf{R} multiplicatively after computing the numerical surface integrals. Within Fig. 2.4 the \mathbf{R}_z matrix is being depicted before including these material parameters, its value has therefore the dimension of a length.

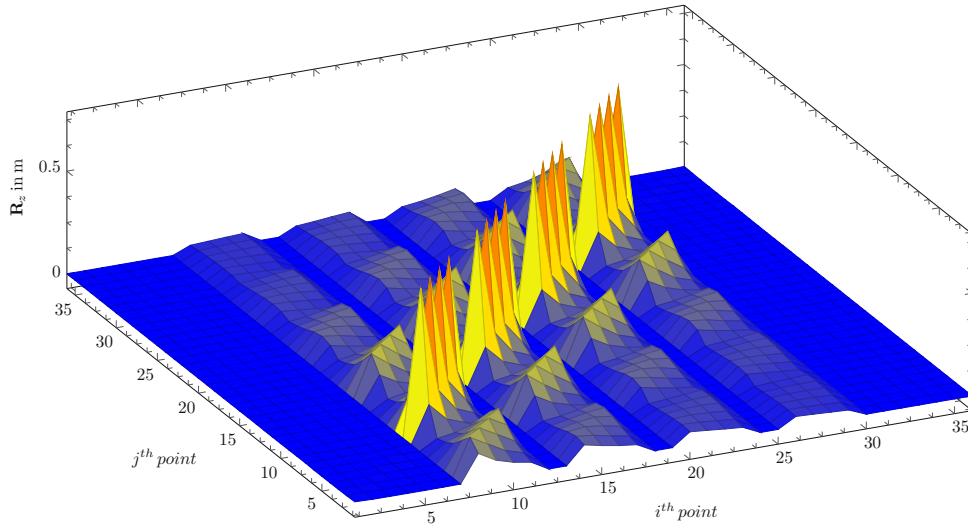


Figure 2.4: Depiction of the \mathbf{R}_z values for a parabola, with a discretisation of the grid of six by six

When the variable point \mathbf{x} overlaps with the point \mathbf{y} , where traction is initiated, the integrand is unbound. Hence, special care needs to be taken when computing the numerical surface integrals. This is bypassed in the subroutine through evaluating the function at points very close to the exact point, as well as to assign the exact unbound integrand a very high but finite value.

2.2.3 Resolving the contact in the normal direction of the surface

In order to resolve the applied load increment in the normal direction, a subroutine has been established. Within this subroutine, the penetration depth is calculated, the input data for the LCP solver is assembled, the solver itself is integrated, and the results are post-processed for further use.

The calculation of the minimal penetration depth is externalised to another subroutine. For the objects and their discretisations, a point correspondence is established when defining them. This does not hinder the points from moving independently but results in a simpler and more efficient way of calculating the penetration depth. The pole shoe is not actually pulled alongside the rail, due to the fact that a quasi-static use-case without actual rigid body motion is constructed. The expected displacements are way smaller than the distances between the discrete points, therefore this correspondence can be justified. Alternatively, an interpolation method could be used to calculate the minimal penetration depth. However, for small penetration depth, this leads to inaccuracy problems when dealing with curved objects and requires a greater computational effort.

Following the computation of the minimal distance to the opposing surface for both objects, the necessary input data is conveyed to the LCP solver. The points, which will not be in contact,

are excluded to reduce the size of the matrices and therefore the computational effort. The solver tackles both objects individually but quasi-simultaneously, meaning the results of one do not influence the other. When individually solving the objects, utilising the full penetration depth, the resulting deformation and traction would be equivalent to the ones, facing an undeformable object. Consequently, to model both objects as deformable, it is necessary to couple the stiffness matrices, to accurately resolve the penetration in normal direction. The input for the solver is the \mathbf{R}_z matrices and associated penetration depth in the normal direction, defined negative if it is intersecting the opposing surface. The results are the normal traction and the separation between the surfaces.

The tractions have to be aligned using the normal vectors, given that the results obtained from the solver are inherently positive. Subsequently, the displacements in the normal direction are calculated using the part linking u_z^i and p^j from the equation (2.8). With that, the discrete points are displaced, giving the newly resolved geometry, and the necessary normal traction to achieve these displacements.

2.2.4 Additional displacements and tractions due to contact

Two further subroutines have been developed, to include additional displacements and tractions into the simulation. The first enables the calculation of shear displacement due to normal traction, whilst the second calculates tangential traction due to friction, and with that an additional displacement due to friction.

In order to model shear displacement due to the normal traction, the connections between u_x^i and p^j , and u_y^i and p^j , from the equation (2.8) are utilised. The values for the displacements are obtained by means of matrix multiplication. These are then aligned along the normal direction, and rotated about the respective other tangential direction, to end up with the shear displacement (for the respective other direction). This procedure is repeated for both tangential directions, thus providing all remaining displacements due to normal traction.

The tangential traction resulting from friction is calculated using the Coulomb friction law with a predefined friction factor. In addition to multiplying this factor with the normal traction, the direction of this frictional traction is computed. In the input file, the user specifies a direction of movement for the upper object, while the lower one is assumed to be resting.

For each individual point, a set of two tangential vectors is calculated, under the premise that they are perpendicular to the normal vector and to each other. The general movement of the object is then projected onto the tangential plane, spanned by the two tangential vectors, to end up with a unit vector for the direction of the frictional traction at the lower object. For the upper body, the direction for this projection onto the tangential plane is in the opposite direction of the given movement. From this traction, represented as q_x^j and q_y^j , the relations from the equation (2.8) are used to calculate the displacements. It is noted that no further orientation changes are required for the displacements in the direction of the frictional traction. However, for the components computed from the out-of-diagonal elements, an alignment of the displacements is necessary. This is achieved by utilising a vector that is perpendicular to the direction of the frictional traction, or the respective normal vector for the normal displacements due to tangential tractions. Upon completion of these steps, all displacements and tractions are computed within one load increment of the contact simulation.

2.2.5 Enabling back-deformation

Within the algorithm, due to tangential displacements it is possible that a point had contact after resolving the latest load increment, but has no contact after the current increment. For the steps, where it had contact, the point already accumulated traction values in multiple directions. However, it is evident and already defined in (2.11.3), that at points where the surfaces are separated, no traction can occur. Therefore, the traction at those discrete points has to be released completely or released to a certain extent so that the points regain contact again.

This is done by a subroutine that performs the so called back-deformation. Within this routine, the points without contact, to which a traction value is assigned, are identified. For these points, the distance to the opposing surface and the maximum possible back-deformation, using the normal traction values and the \mathbf{R}_z matrix, are computed. This is put into relation, and this fraction of the traction is released through computing the corresponding normal displacement. Subsequently, these displacements are added to move the discrete points back into contact, or as far as the completely released traction value is able to move them. The tangential tractions and all possible directions of displacements are reduced and moved back according to this ratio. At the end of this subroutine, an iteration of the contact resolution is performed to compensate for any newly occurring intersection of the objects due to the back-deformation routine.

2.2.6 Updating the normal directions

After one load increment is successfully resolved, and possible back deformations are accounted for, the algorithm updates the geometry, and with this, the direction of its normal vectors. The geometry update is simply overwriting the position of the discrete points with the newly computed ones, although the old positions are stored. Those two positions for all discrete points are then used in this subroutine to update the normal directions.

The first step is to calculate all distance vectors from one point to its neighbouring points on the grid, for both the old and the new versions of the discrete surfaces. From those, the confined angle between the new and the old vectors to one neighbouring point within the horizontal grid is calculated. This is done for all four neighbouring points and the mean value of the two angles on the opposing sides of the individual point is calculated. This angle is regarded as estimation of the change in tangential direction at this discrete point. This is done for all points, for points at the edge of the grid, only the existing neighbours are utilised, and no calculation of the mean value is required. The two angles are employed to compute two rotation matrices around the respective other directions, which are subsequently combined into one single rotation matrix. This matrix for each individual point is then applied to the respective normal vector to update it, according to the change of the tangential plane at that point.

2.3 Extension of the contact model

During the application of the algorithm and when comparing it with benchmark results, obtained from an FE model, it became clear that the algorithm is not able to simulate all the phenomena that occur during the contact of an MTB with the rail. Within the basic contact algorithm only the deformations of the surfaces are considered, which turns out to be inadequate when simulating this conforming type of contact. The main concern lies in the inability of the algorithm to accommodate the structural elastic deformations of the object itself. Therefore, the possibility and feasibility to extend the model, to account for elastic deformations for the simulation

of a pole shoe, is examined.

In order to capture all emerging phenomena with sufficient accuracy, a two-step solution approach incorporating an FE model was selected to establish an extension of the basic contact algorithm. A simple finite element model, set up with the "Partial Differential Equation Toolbox", found within the MATLAB environment, is established for the extension, and described further in this section. It is important to note that this extension is specific to the pole-shoe geometry. Whilst it is possible to adapt this to other geometries, this would require several changes to the inputs, including a different 3-D geometry file and a redefinition of the boundary conditions of the FE model.

2.3.1 The simple FE model

Within the extension including the FE model, the load increment is adapted. For the basic algorithm, this increment is solely defined through the external loading, while in the extension it is changed, to account for the structural deformations. The main idea is that one basic load increment is applied, and the traction results are then applied as an external distributed load onto the FE model. Through the FE simulation, an additional displacement due to the structural deformation of the pole shoe is obtained. These displacements are superimposed to the surfaces, therefore new penetrations occur at some points, while other points may lose contact, leading to a necessary back-deformations. The application of the "contact resolving" and "back-deformation" subroutines results in a change of the traction. This modified traction is in turn again applied to the FE model, which internally calculates a new set of displacements for the discrete points of the pole shoe surface. The aforementioned procedure is iterated until the change in displacements resulting from the FE model lies within a certain tolerance. In order to achieve this convergence of this two-step routine, not the complete displacements, that arise within the FE simulation, but only an increment of them, are applied. For that a new simulation specific parameter is introduced, defining the respective increment of the full FE solution utilised. An illustration of how the FE model works and where it is implemented to extend the algorithm, see Fig. 2.5.

The determination of the simulation parameter to achieve convergence is not an entirely straightforward process, and a certain degree of trial and error is involved. Therefore, it is identified for some basic use cases and a multidimensional fit is applied to allow for a user-friendly handling. For some visualisations and remarks on the convergence of the FE model, including the impact of this parameter, see C.0.2.

As mentioned above, additional inputs are needed for the extension of the algorithm. Within the iteration, a tolerance for the difference between the displacements of the FE model from one iteration to the next, and a maximum number of iterations, have to be chosen. A range of pole shoe geometries, varying in complexity, have been defined and can be selected according to the intended use, see Fig. 2.6. Additionally, the user is able to select a specific discretisation for the surface at the bottom of the pole shoe. The MATLAB algorithm meshes the geometry internally, utilising a mesh composed of quadratic tetrahedral elements. The analysis type is set as a structural analysis for a static solid problem, because the whole algorithm is designed to find the quasi-static solution of this problem. For a more comprehensive account of the MATLAB intern FE simulation, please refer to their website [30].

In conclusion, all introduced simulation specific parameters are given in Table 2.1.

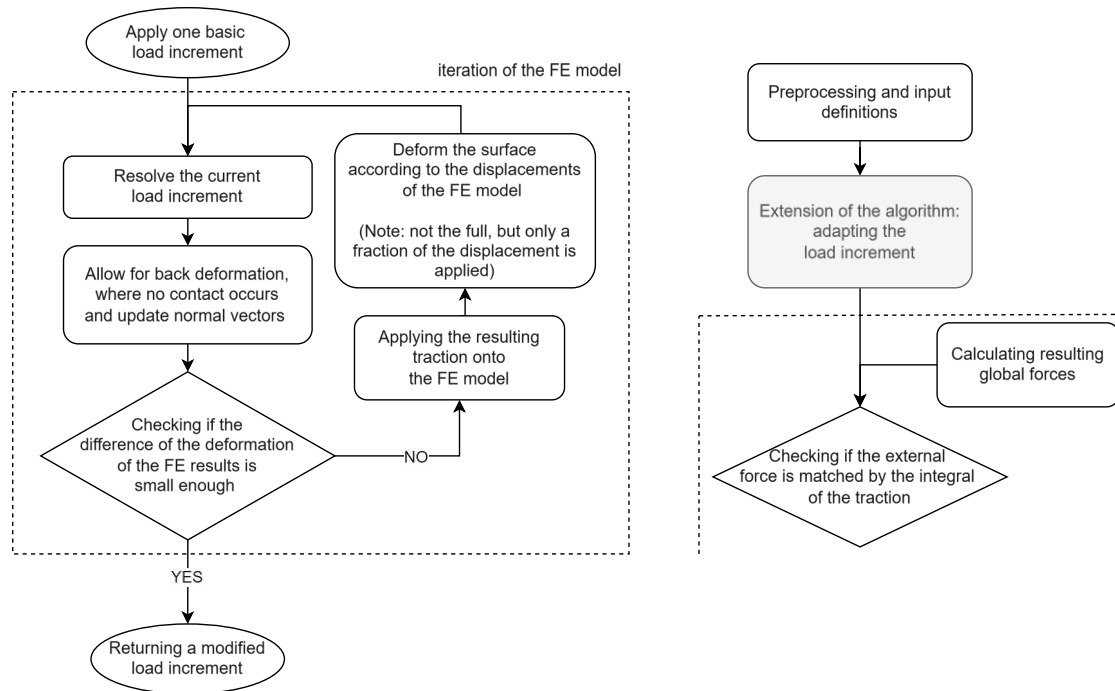


Figure 2.5: Overview of the iteration process within the extension to adapt the load increment/ position of the extension within the algorithm

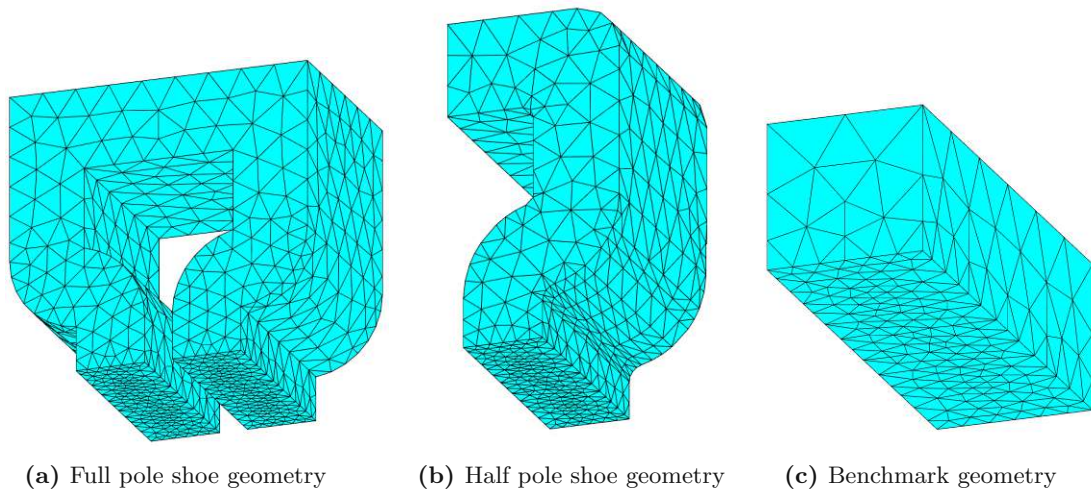


Figure 2.6: Different meshed pole shoe geometries utilised in the extension, with $\delta_{\text{FE model}} = 0.004\text{m}$

Formula symbol		Description of the purpose of the simulation parameter
$i_{\text{estimated steps}}$		Estimation of the number of load increments for the simulation of the contact
$\epsilon_{\text{force tolerance}}$	[N]	Tolerance for the global force
$\epsilon_{\text{distance tolerance}}$	[m]	Tolerance for the contact detection
$\delta_{\text{grid } x/y}$	[m]	Distance between the grid points for a given direction
$\delta_{\text{FE model}}$	[m]	Maximal distance between the nodes on the surface in contact for the FE model
$i_{\text{FE model}}$		Maximal iterations to find a convergent solution for the FE model in the extension
v	km/h	Velocity of the moving pole shoe, (optional, for assigning a stored linkage between velocity, frictional coefficient and magnetic load distribution)
$type_{\text{FE model}}$		Definition of which FE model is used for the simulation (see Fig. 2.6)
$f_{\text{FE model}}$		Increment of the deformations of the FE results, used in the FE iteration (can be set to "automatic", for a fit from already working values of this parameter)

Table 2.1: Summary of all simulation parameters within the contact algorithm

2.4 Time complexity of the algorithm

To further describe the algorithm, an estimation of its computational complexity, or more precisely, its time complexity, is briefly described here. This is done through an upper bound with O , as described in Chapter 3 of the book Foundations of Computer Science, see [31]. For another comprehensive book on the subject of the configuration and evaluation of algorithms, see also [32]. More detailed derivations and a figure in "tree form", describing the time complexity of the different subroutines, is given in the appendix 5.

The time complexity of the complete contact algorithm is composed of

$$\begin{aligned}
 O_{\text{contact algorithm}} = & O(n^2 \cdot i_{\text{LCP}} \cdot i_{\text{friction}} \cdot i_{\text{backdef}} \cdot i_{\text{FE model}}) \\
 & + O(n^2 \cdot i_{\text{LCP}} \cdot i_{\text{friction}} \cdot i_{\text{backdef}} \cdot i_{\text{load increment}}) \\
 & + O(n^2 \cdot i_{\text{LCP}} \cdot i_{\text{friction}} \cdot i_{\text{backdef}} \cdot i_{\text{tolerance}}).
 \end{aligned} \tag{2.12}$$

Here, n denotes the number of discrete points in contact on the contact surface, with a higher number of contact points. The variable i is employed to define the count for the respective loop. More specifically i_{LCP} is for the LCP solver, i_{friction} is the number of iterations within the resolving of one load increment, i_{backdef} is for the back-deformation subroutine, $i_{\text{FE model}}$ is for finding a convergent FE solution, $i_{\text{load increment}}$ is the actual number of increments applied, until the global force is overshoot and $i_{\text{tolerance}}$ is the number of increments applied until the error tolerance of the global force is reached. They are in general not known beforehand, but some estimations are made here. If the description of O is simplified, to contain one term, and it is assumed that $i_{\text{tolerance}} < i_{\text{FE model}} < i_{\text{load increment}}$ we end up with

$$O_{\text{contact algorithm}} = O(n^2 \cdot i_{\text{LCP}} \cdot i_{\text{friction}} \cdot i_{\text{backdef}} \cdot i_{\text{load increment}}). \tag{2.13}$$

In general $i_{\text{load increment}}$ is dependent on the input $i_{\text{estimated steps}}$, while $i_{\text{tolerance}}$ is dependent on the loading, in relation to $\epsilon_{\text{force tolerance}}$. For $i_{\text{FE model}}$, i_{backdef} and i_{friction} an upper limit is set, with typically values ranging from 5 to 20. However, the upper limit for i_{LCP} is at least three orders of magnitude higher (~ 10000). If the loop count is assumed to be independent on the number of discrete points in contact, one could argue that the whole algorithm has a time complexity of $O(n^2)$. In realised examples, a small dependency of i_{LCP} on n is observed for a very large number n .

As a side note, the time complexity of the calculation of the \mathbf{R} matrices is not taken into consideration here, and can be estimated with $O(n^4)$. This explains the much higher computation times for this subroutine, especially for an increasing number of discrete points. Therefore, it is very important to pre-calculate these matrices.

3. Verification of the algorithm

The algorithm, derived and discussed in chapter 2, is verified with simple problems and benchmark results within this chapter. Further, the extension of the model, which is discussed in section 2.3, is also verified to end up with an accurate, and physically plausible simulation. The convergence of the basic algorithm and the finite element model introduced in the extension is conducted, a brief summary can be found in C.

3.1 Verification of the basic algorithm

Initially, the basic algorithm is checked against an academic problem. For the second stage, the results of the basic algorithm are compared with benchmark results, for the contact between the pole shoe and the rail. The issue that gives rise to the extension is outlined and becomes clear at this point.

3.1.1 Comparison to an academic problem

The academic problem, used for the comparison, is a 3-D parabola, hence a paraboloid, indenting into, and afterwards sliding on a flat elastic finite plane. A sketch illustrating the problem is depicted in Fig. 3.1, the different simulation and material parameters are introduced below. The analytical solution of this problem is given e.g. by Popov et al., see [33]. The frictionless normal contact problem is rotationally symmetrical while the frictional problem is only symmetrical along the plane of the horizontal movement. The problem is equivalent to the contact between a rigid paraboloid and an elastic half-plane if both bodies are assumed to satisfy the half-plane approximation. The effective modulus E^* , which is necessary for the analytical solution, is defined through

$$\frac{1}{E^*} = \frac{1 - \nu_1^2}{E_1} + \frac{1 - \nu_2^2}{E_2}, \quad (3.1)$$

where $E_{1,2}$ are the Young's modulus and $\nu_{1,2}$ are the Poisson's ratio of the respective bodies.

The stress field in the vertical direction, resulting from a normal point load F_N , can be calculated with

$$\sigma_z(r) = \begin{cases} -p(r) = -\frac{2E^*}{\pi R} \sqrt{a^2 - r^2}, & r \leq a \\ 0, & r \geq a \end{cases} \quad (3.2)$$

R denotes the radius of curvature of the geometry, $a = \sqrt[3]{\frac{3}{4} \frac{F_N R}{E^*}}$ is defined as the contact radius and $d = \frac{a^2}{R}$ as the indentation depth. The analytical solution for the traction, resulting from a tangential force $F_X = \mu F_N$, is described by

$$\sigma_{zx}(r) = \begin{cases} -q_x(r) = -\frac{2\mu E^*}{\pi R} \sqrt{a^2 - r^2}, & r \leq a \\ 0, & r \geq a \end{cases} \quad (3.3)$$

Here, a Coulomb friction model is assumed, using the well-known friction factor μ . The reference solution for the displacements is derived through a numerical integration of the analytical Boussinesq and Cerruti solutions with the given analytical tractions. The simple contact algorithm assumes full slip, so the terms in the analytical solution that account for sticking are omitted from the equations.

For the comparison between the basic contact model and the analytical solution, a set of

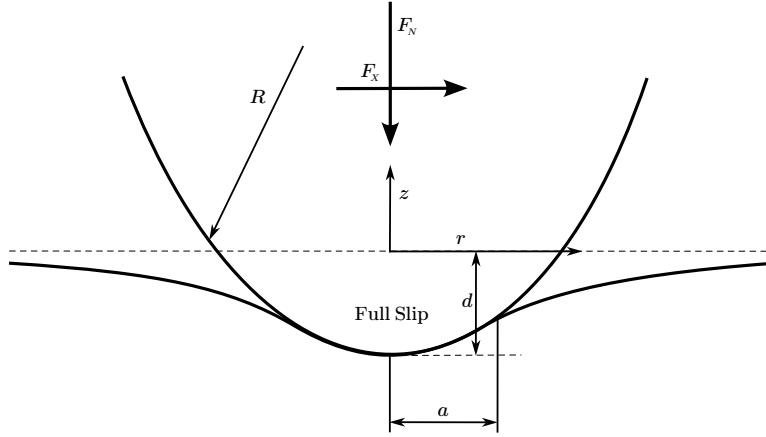


Figure 3.1: Sketch of the analytical problem of a paraboloid on a half-plane, see [21]

parameters has been chosen, see Table 3.1. The material parameters correspond to a steel-on-steel contact, while the applied force is selected to induce significant deformation in both objects while remaining in the elastic range for the steel.

$G_{\text{paraboloid}}$ [Pa]	$\nu_{\text{paraboloid}}$	G_{plane} [Pa]	ν_{plane}	F_N [kN]	R [m]	μ
$80 \cdot 10^9$	0.3	$80 \cdot 10^9$	0.3	5000	1000	0.5
$i_{\text{estimated steps}}$	$\epsilon_{\text{force tolerance}}$ [N]	$\epsilon_{\text{distance tolerance}}$ [m]				
20	10^3	10^{-5}				

Table 3.1: Simulation parameters for the academic problem, paraboloid on a flat plane

The simulation is done for different grids, spanning from quite coarse to relatively fine. In Fig. 3.2, the vertical displacement u_z and traction p of the flat elastic plane are depicted for these different grids compared to the analytical solution. The results are displayed for a general x while y is set to zero, or expressed in the mathematical notation $u_z(x, y = 0)$ and $p(x, y = 0)$. For this, a conversion from the radial coordinate r to the horizontal coordinate x is performed for the analytical solution. The direction of the steady-state movement is the x axis. Analogously, in Fig. 3.3 the horizontal traction $q_x(x, y = 0)$ and displacement $u_x(x, y = 0)$ are depicted.

From Fig. 3.2 and Fig. 3.3 it is easy to conclude, what was already expected, namely for a finer grid, the results are getting closer to the analytical solution. It is demonstrated that, for a sufficiently fine grid, the results obtained by the algorithm are in good agreement with the analytically derived results. However, a slight asymmetry in the contact pressure distribution and hence in the shear traction, which is absent in the analytical solution, is observed in Fig. 3.2 and Fig. 3.3. This is owed to the different loading history within the two approaches. In the present contact algorithm the normal and tangential problems are solved consecutively for each individual load increment. Therefore, the obtained state is a result of sequential solutions of incremental normal and tangent problems. For the analytical solution, the contact pressure distribution is derived completely independently from the tangential traction, and hence is symmetrical.

For a more vivid representation of the problem, a cut through the 3-D objects, before and after the deformation, including a scaled version for better visualisation, is in given in Fig. 3.4.

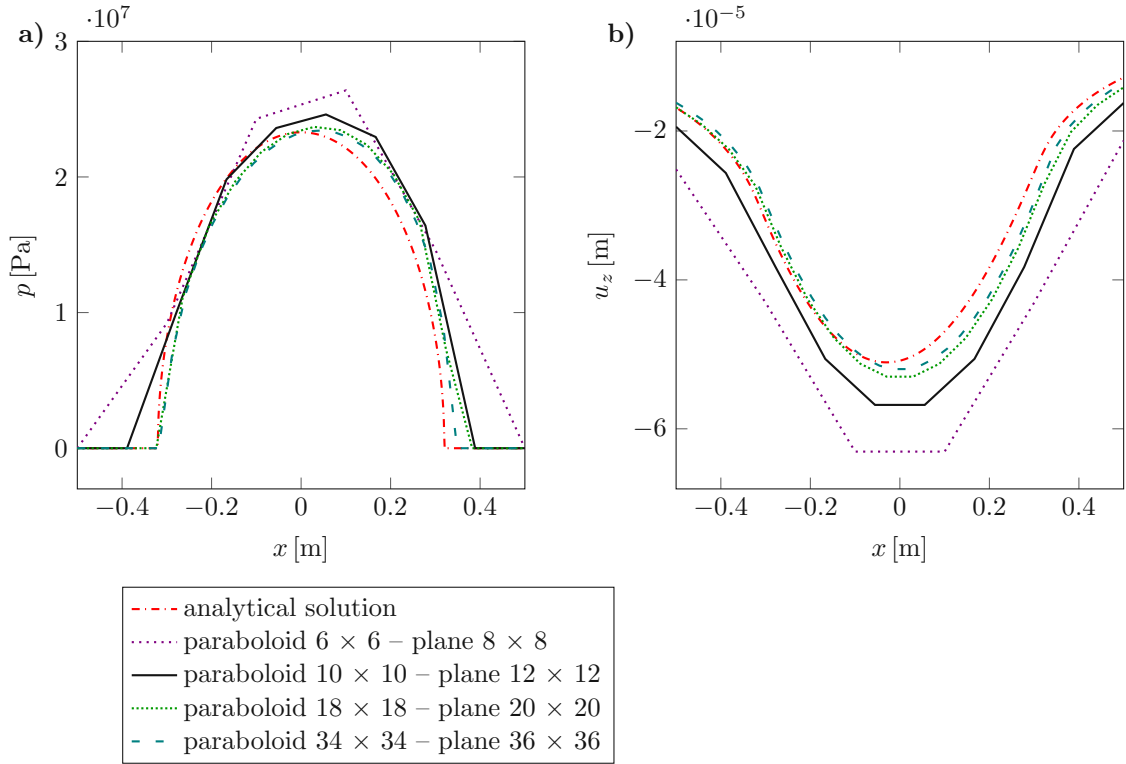


Figure 3.2: Results of (a) the vertical traction $p(x, y = 0)$, and (b) displacements $u_z(x, y = 0)$ for the elastic plane

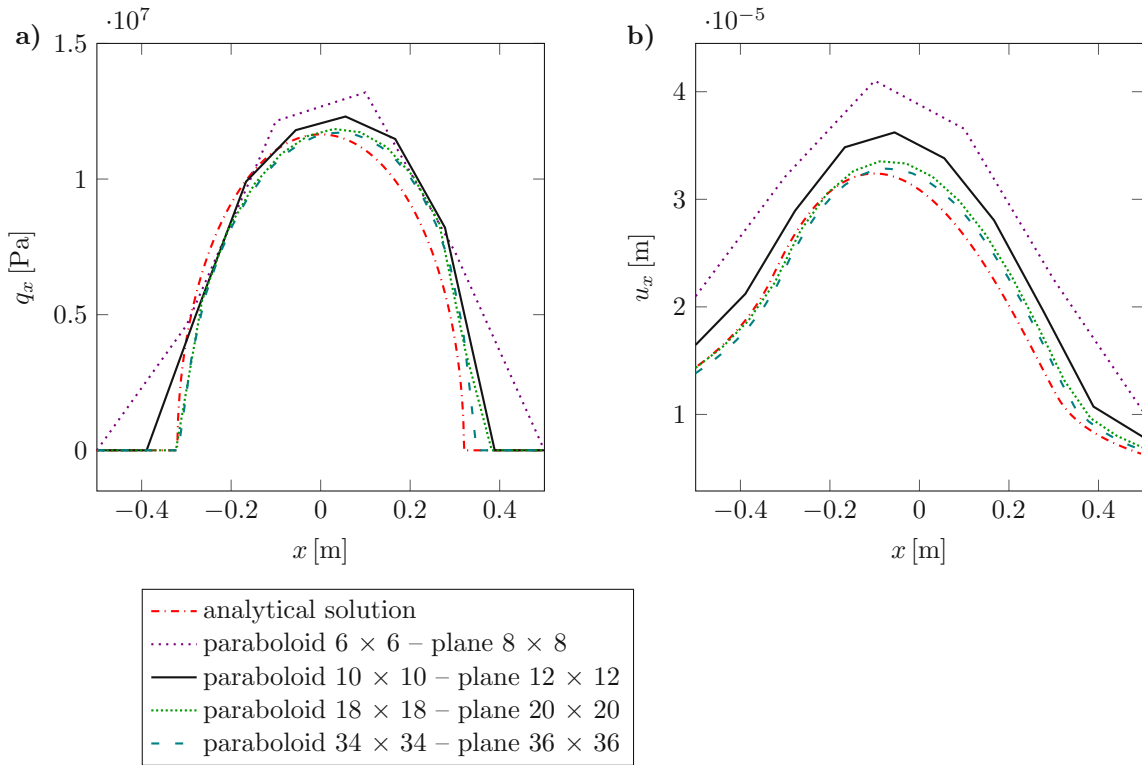


Figure 3.3: Results of (a) the horizontal traction $q_x(x, y = 0)$, and (b) displacements $u_x(x, y = 0)$ for the elastic plane

The calculation times, when using the contact algorithm that is developed in the present study, are listed below in Table 3.2 for the various grids¹. From them, it is apparent that the calculation times scale strongly with finer grids (see section 2.4), albeit the method still remains relatively efficient and fast.

number of grid points, paraboloid	6×6	10×10	18×18	34×34
number of grid points, half-plane	8×8	12×12	20×20	36×36
$n_{\text{half-plane}}$	64	144	400	1296
$\delta_{\text{grid}} x/y[\text{m}]$	0.2	0.11	0.059	0.030
calculation time t [s]	1.9	2.1	14.5	163
Factor to the next column for n	2.25	2.78	3.24	
Factor to the next column for t	$1.1 = 1.05^2$	$6.93 = 2.63^2$	$11.24 = 3.35^2$	

Table 3.2: Calculation times for the academical problem, including an estimation of the time complexity²

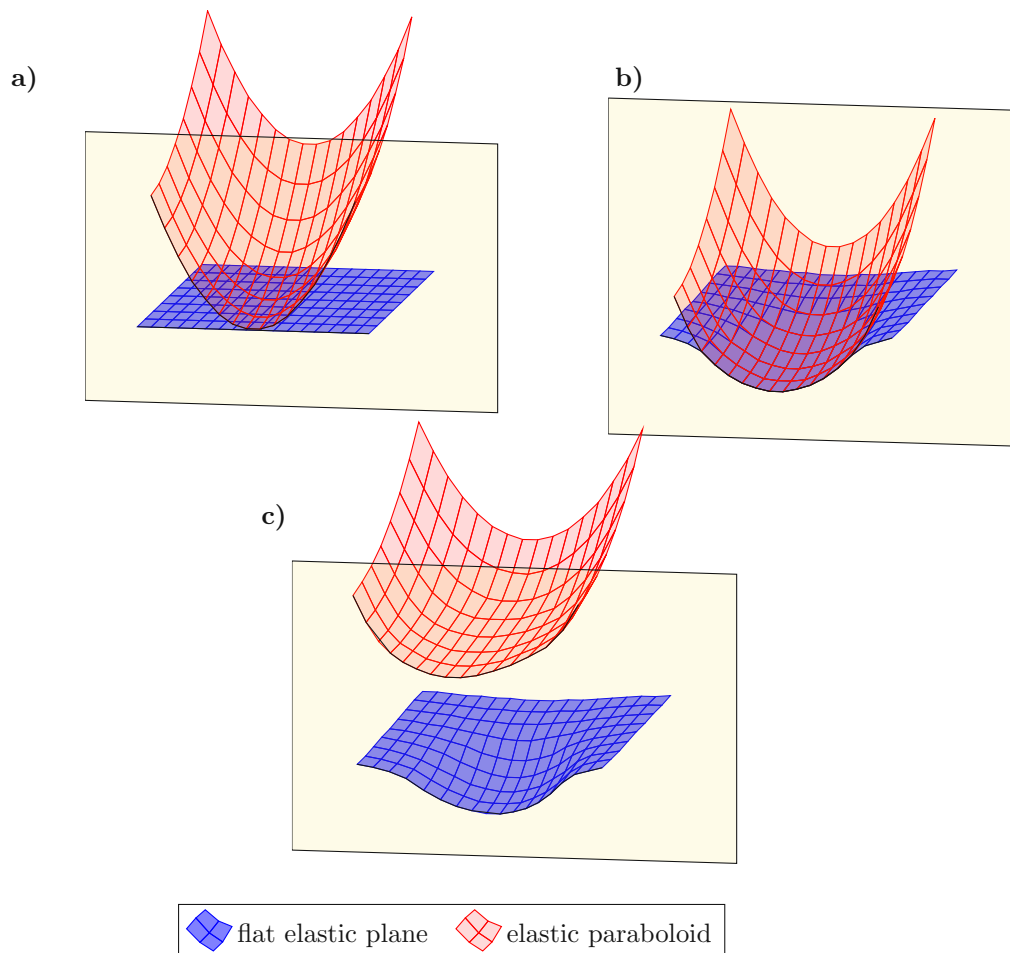


Figure 3.4: Cut through the 3-D objects (a) before the deformation, (b) after the deformation, and (c) after the deformation with a scaling factor of 2500 for the horizontal displacement

¹The computation is done on an Intel Xeon CPU E5-1620 v3 @ 3.50GHz, with 32 GB of RAM.

²As described in 2.4, the upper bound for the time complexity is estimated with $O(n^2)$. This can be observed in a quite accurate way from those calculation times. To make this easier to recognise, the factor between two discretisations (from one column to the next) is given in the two last rows, for $n_{\text{half-plane}}$ and t . The factor t is close to the square of the factor $n_{\text{half-plane}}$, verifying the estimation for the time complexity (with the half-plane being the surface in contact with more discrete points and therefore the dominating parameter for the time complexity). An exception to this are the factors between the two coarsest grids in the left two columns, where the calculation times are in general very low.

3.1.2 Benchmarks comparison for the pole shoe rail contact

For a further verification, the contact problem between the pole shoe and rail is considered. As a benchmark, the results of a finite element model, which was the topic of a previous master's thesis within the research group [22], are utilised. As rail geometry the UIC60 profile is used, see [34]. It is assumed that between the rail and pole shoe a wear profile evolves. Therefore, the bottom of the pole shoe geometry is assumed to be the same as the UIC profile for 60% of its lateral direction, as described in the norm EN 16207 [35].

In the following, the tractions and displacements gained from the contact algorithm and different FE benchmarks are compared. Because the problem is symmetric with respect to the x - z plane (see Fig. 3.5), only the left contact patch is displayed. The discrete points have three spatial coordinates. Hence, for the visualisation with 3-D figures, only the two-dimensional grid position is displayed, while along the vertical axis the respective result is plotted.

In the finite element model, quadratic 3D continuum elements are used with an edge length of 0.5 mm at the surfaces in contact. The rail is rigidly fixed at the bottom and defined as free surfaces at the ends. The pole shoe is pulled over the rail with a constant velocity. A height parameter ξ is introduced, up to where the pole shoe is modelled, see Fig. 3.5. At this height, the longitudinal motion is applied. The utilised benchmark results are obtained at the point of reaching a certain stopping distance, to rule out any influences of the boundaries. The value for ξ is varied to analyse the influence of the thickness of the elastic layer of the pole shoe.

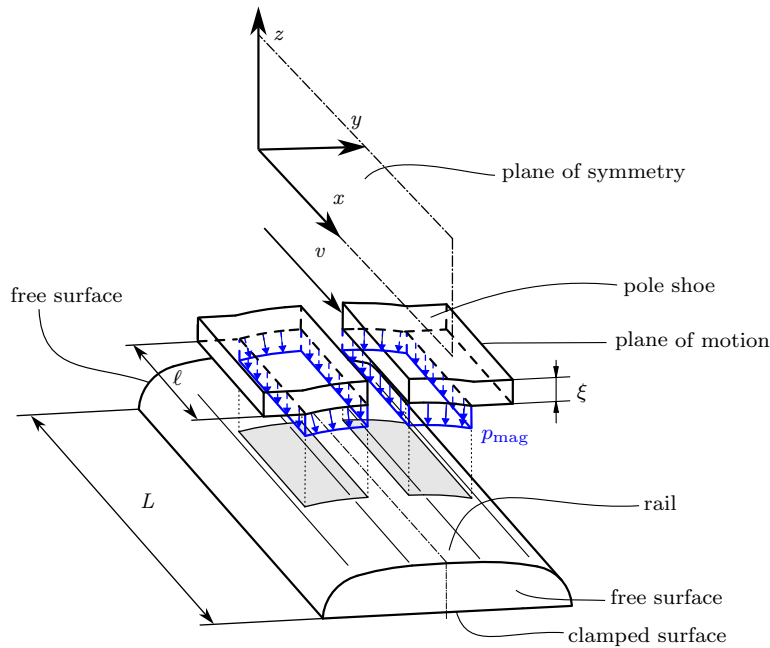


Figure 3.5: Sketch of the contact problem used as a benchmark, see [21]

The magnetic loading p_{mag} is assumed constant and evenly distributed on the contact surface for this benchmark, while it is varied later on for the results in chapter 4. The material parameters of steel are utilised for both objects, and the friction factor is established at the upper limit of what is reasonable with $\mu = 0.5$. It is anticipated that the deformations will be even smaller than those for the paraboloid, so the tolerance for distances and forces in the algorithm is lowered by several orders of magnitude. For a full list of all parameters used, see Table 3.3.

At first, the contact pressure distribution in the normal direction for the left side of the pole shoe

$G_{\text{pole shoe}} [\text{Pa}]$	$\nu_{\text{pole shoe}}$	$G_{\text{rail}} [\text{Pa}]$	ν_{rail}	μ	$p_{\text{mag}} [\text{N/mm}^2]$
$80 \cdot 10^9$	0.3	$80 \cdot 10^9$	0.3	0.5	3
$\delta_{\text{grid } x/y} [\text{m}]$	$i_{\text{estimated steps}}$	$\epsilon_{\text{force tolerance}} [\text{N}]$	$\epsilon_{\text{distance tolerance}} [\text{m}]$		
0.003	8	1	10^{-12}		

Table 3.3: Simulation parameters for the benchmark problem, pole shoe on rail

from the contact algorithm is compared to the FE results for $\xi = 1.15\text{mm}$ in Fig. 3.6 and Fig. 3.7. Within this comparison, a good agreement is observed at highly enhanced computational efficiency. The pressure peaks at the border do not match exactly, due to a coarser grid used in the contact algorithm, and the fact that analytically the gradient would approach infinity there. Further, the displacement in vertical direction is compared to the benchmark, see Fig. 3.8 and Fig. 3.9. The benchmark solution depicts a longitudinal asymmetry, for the contact pressure distribution as well as the vertical displacement. They are both lower at the rear of the pole shoe compared to the front. This leads to the conclusion that an elastic body deformation might influence the FE results, although the results match pretty well. More specifically, this could be described as a lean towards the frontside and a lift off at the backside of the pole shoe.

The height of the elastic body is described by the parameter ξ within the FE model. To further investigate the possibilities and limits of the contact algorithm, its value is varied. The resulting contact pressure distribution and vertical displacement of the left side of the pole shoe for $\xi = 28.2\text{mm}$ at two longitudinal cross-sections are displayed in Fig. 3.10 and Fig. 3.11. Additional plots with an additional value for ξ can be found in the appendix B. In comparison with the unaltered results from the contact algorithm developed, the discrepancies become more apparent. The Fig. 3.11 demonstrates that an increase in height of the elastic layer is accompanied by a higher penetration at the front and a more apparent lift off at the back of the pole shoe. This can also be observed in Fig. 3.10 due to a higher pressure peak at the front and the complete absence of any contact pressure at the back. These observations provide further evidence for an ongoing tilting motion, through an elastic body deformation within the FE solution, which the basic contact algorithm is not able to account for.

To conclude this initial comparative analysis, the relative error between the basic contact model and the benchmark results are discussed. In Fig. 3.12 contour lines of the 10% relative error of the contact pressure distribution for the left contact patch of the pole shoe is depicted, for varying values of the parameter ξ . On the inside of those lines, the relative error is lower, on the outside it is higher. It is clearly visible that with an increase in the height, the area with a relative error below 10% decreases. This is particularly evident at the front and rear of the pole shoe, indicating again an elastic body deformation resulting in a tilting towards the front. It can thus be argued that the contact algorithm can only simulate accurately up to a certain elastic height. Upon reaching this height, it is imperative to consider the effect of the elasticity of the pole shoe. An illustration of the relative error of the contact pressure distribution as well as the vertical displacement over the contact area for $\xi = 1.15\text{mm}$ can be found in B.

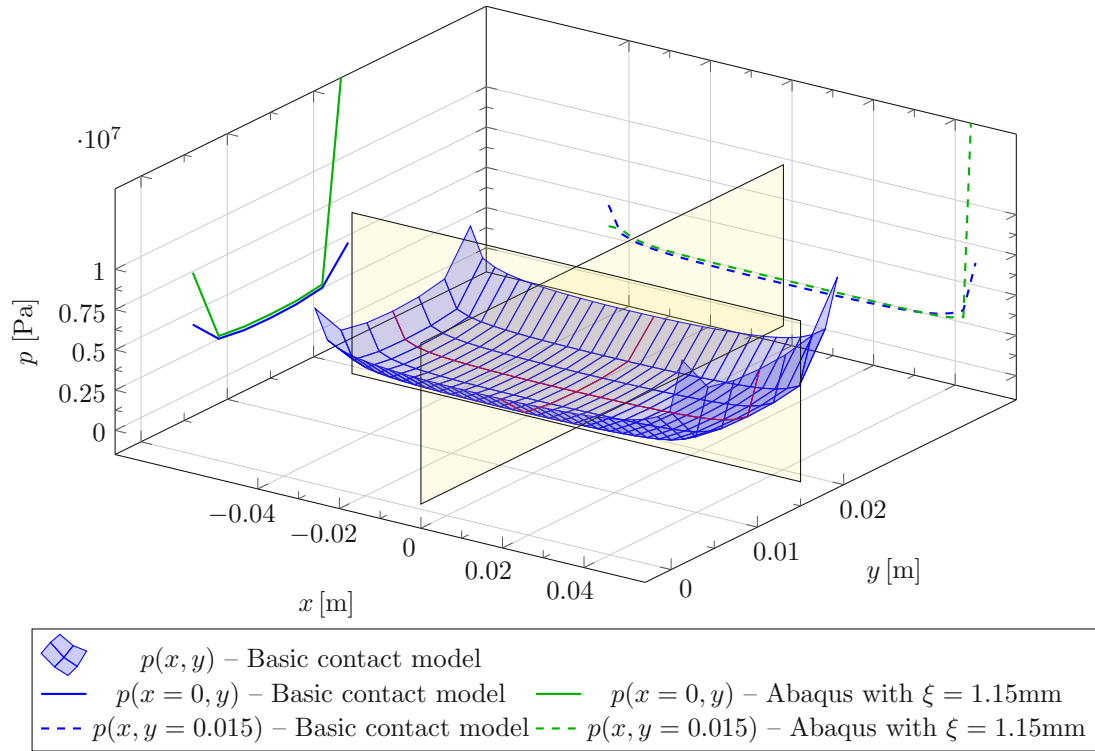


Figure 3.6: Contact pressure distribution, basic contact algorithm compared to the FE benchmark results

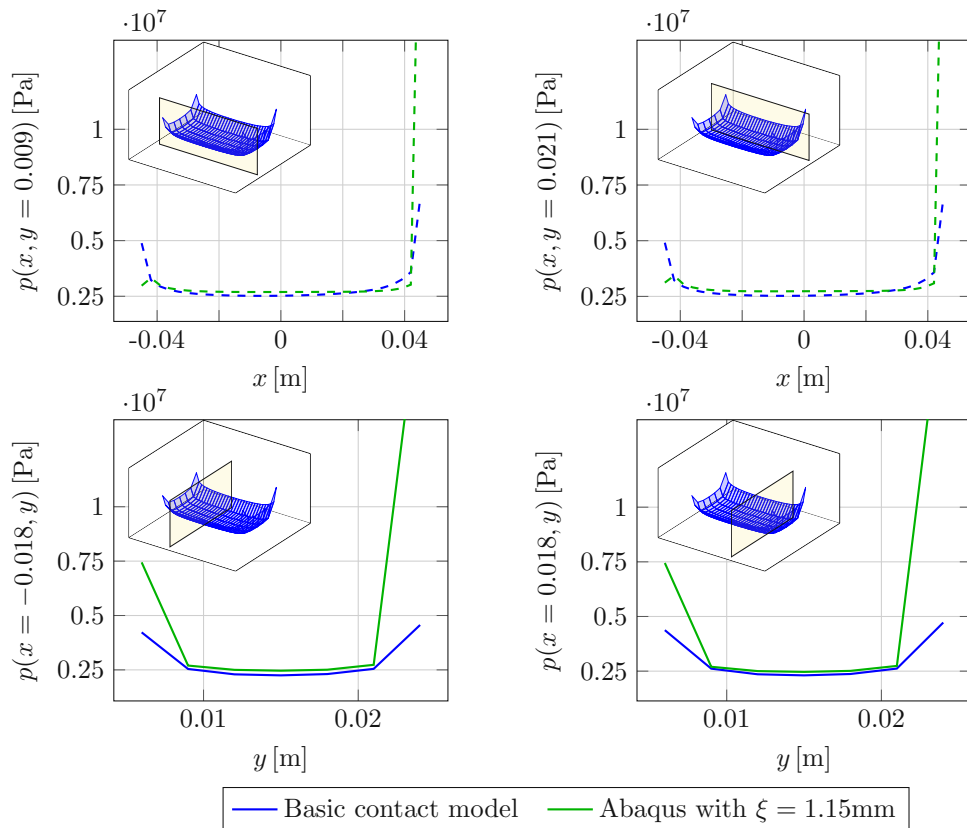


Figure 3.7: Contact pressure distribution, basic contact algorithm compared to the FE benchmark results at four more cross-sections

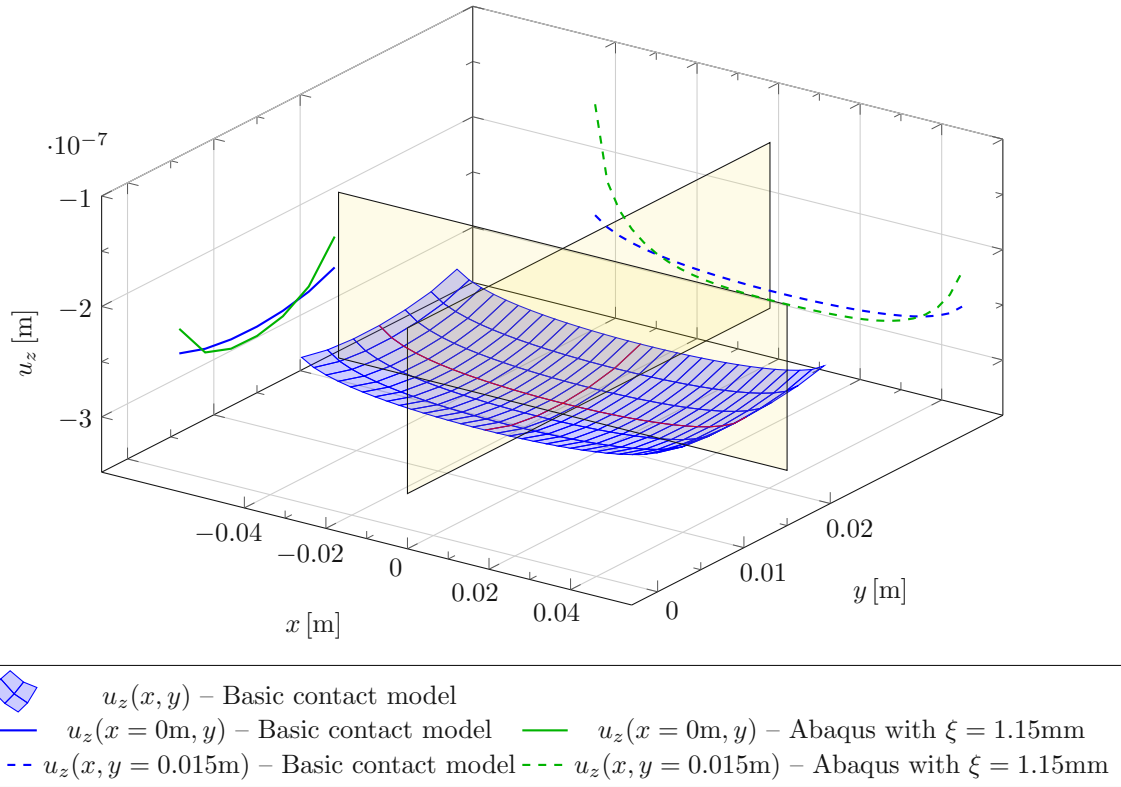


Figure 3.8: Vertical displacement, contact algorithm compared to the FE benchmark results

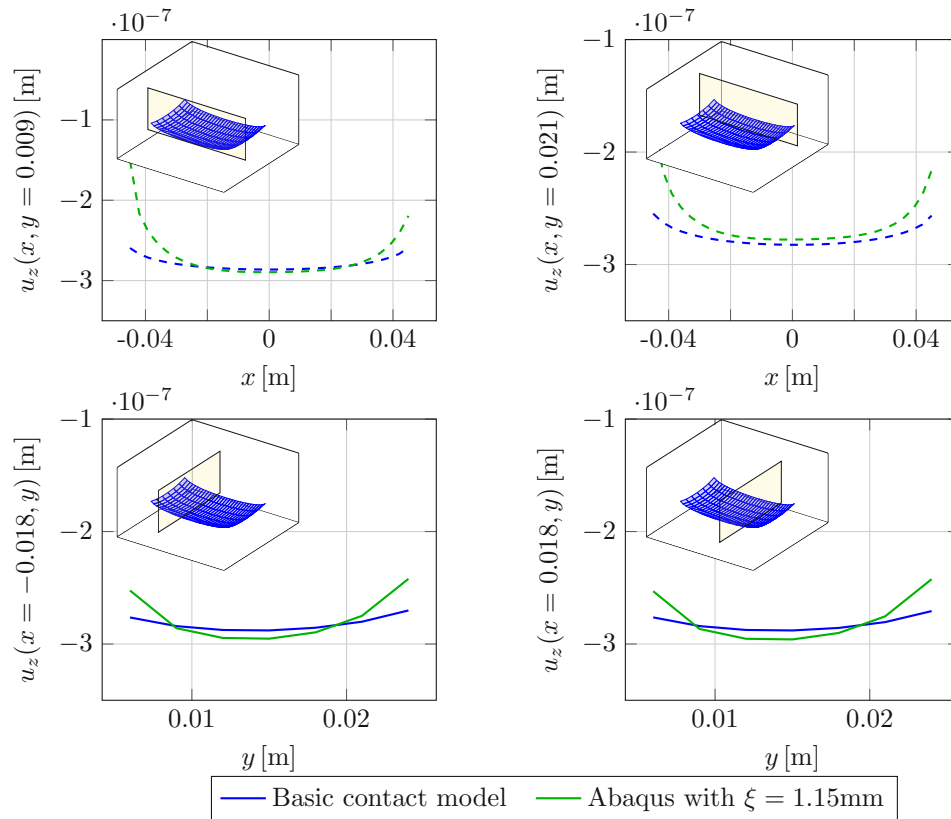


Figure 3.9: Vertical displacement, comparison with benchmark results at four more cross-sections

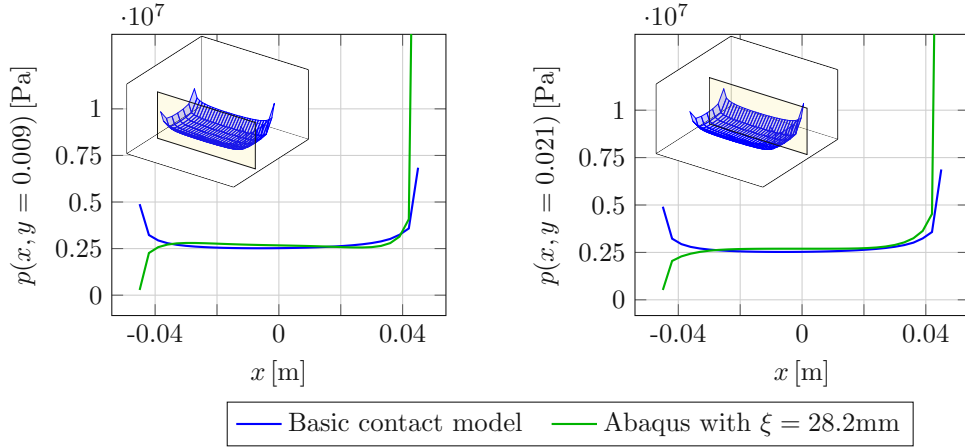


Figure 3.10: Contact pressure distribution, benchmark comparison for $\xi = 28.2\text{mm}$

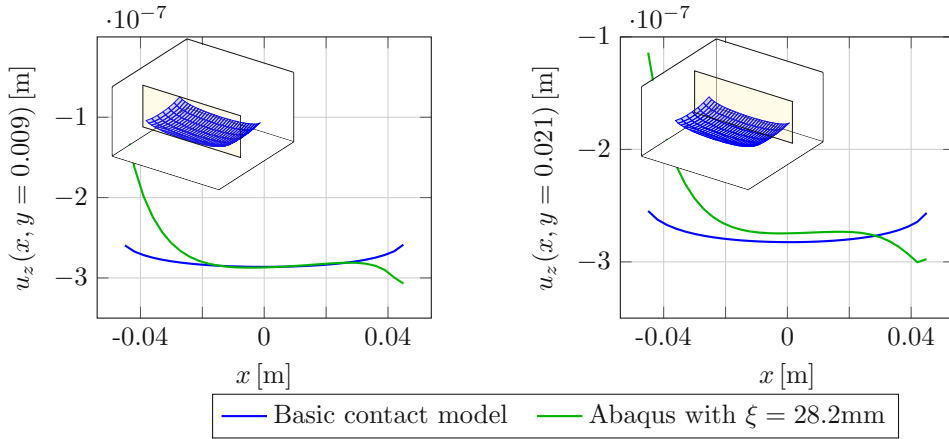


Figure 3.11: Vertical displacement, benchmark comparison for $\xi = 28.2\text{mm}$

In order to quantify the aforementioned statement, an error measure is defined for the contact pressure distribution as $\epsilon_p = \frac{\sum_{i=1}^N \Delta p_i}{N}$ and for the vertical displacement as $\epsilon_{u_z} = \frac{\sum_{i=1}^N \Delta u_{z,i}}{N}$. It utilises the relative errors of the contact pressure Δp_i , and respectively the relative errors of the vertical displacement $\Delta u_{z,i}$ at the i^{th} discrete grid point. The relative error is defined as the difference between the benchmark result and result from the basic contact model, divided by the benchmark result. It should be noted that, due to the fact of the non-matching of contact pressure distribution at the boundary points, these outlier values are not taken into account for the calculations. The error values and the calculation times for FE benchmark and the contact algorithm can be found in Table 3.4. It is clearly visible, that for an increase in ξ , the relative errors ϵ_p and ϵ_{u_z} increase. Additionally a huge difference in calculation time can be observed.

ξ [m]	ϵ_p	ϵ_{u_z}	t_{FE} [s]	$t_{\text{contact algorithm}}$ [s]
$0.50 \cdot 10^{-3}$	8.09%	5.59%	13400	252
$1.15 \cdot 10^{-3}$	9.98%	5.77%	17700	
$7.45 \cdot 10^{-3}$	22.29%	7.58%	16800	
$28.2 \cdot 10^{-3}$	45.89%	11.69%	16800	

Table 3.4: Error values for the contact pressure distribution and the vertical displacement, and computation times for the FE benchmark compared to the basic contact algorithm

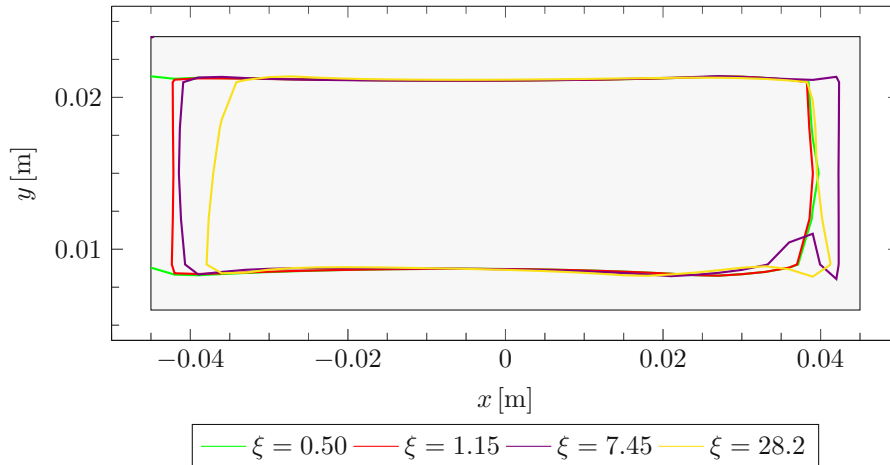


Figure 3.12: Contour of a relative error of 10% for the contact pressure distribution, for the comparison between the results from the contact algorithm with the FE benchmark for different values of ξ

3.2 Verification of the extension

The extension of the contact algorithm, motivated by the observed differences in section 3.1.2, incorporating a simple FE model for the elastic deformations of the body, is validated through a simple contact problem. The reference solution is derived via the utilisation of the commercial FE software Ansys³. Additionally, the results from the extended contact algorithm are compared to the benchmark results, see 3.1.2. There, the geometry of the benchmark (see Fig. 2.6c) is introduced into the extension and the achieved adaptation of the contact algorithm is discussed.

3.2.1 Simple contact problem of a block against a flat plane

The contact problem for the verification of the extension, comprises a block sliding over a flat plane. This facilitates a straightforward comparison of the elastic deformations, modelled by the extended contact algorithm, avoiding the need to deal with a large number of interfering phenomena. A sketch of the contact problem is depicted in Fig. 3.13. The flat plane is modelled with a certain depth in the FE software to ensure a half-plane like behaviour. A defined movement along one horizontal axis is applied at the top of the block, while the displacement in the other two directions is prohibited there. At the bottom a constant pressure p_{mag} is applied and a friction factor μ for the interaction between block and plane is assumed. The magnetic load and the constant horizontal displacement at the top of the block are applied simultaneously, to best fit the procedure from the extended contact algorithm. Nonetheless, no substantial change is evident in a scenario, where the load is applied initially and the movement is separately later on. The values of the parameters are listed in Table 3.5.

This finite element model consists of 15.520 elements of the type HEX20 with 70.130 nodes. Of those, 6.664 elements are used for modelling the block and 8.856 elements are used for modelling the flat plane within the FE simulation. The convergence of the FE simulations with Ansys is checked but not discussed further here. The extension of the contact algorithm employs an STL (Standard Template Library) file to define the geometry. Consequently, within the finite element model, the same file is utilised to define the geometry. The contact interaction between the block

³Ansys Mechanical <https://www.ansys.com/products/structures/ansys-mechanical>

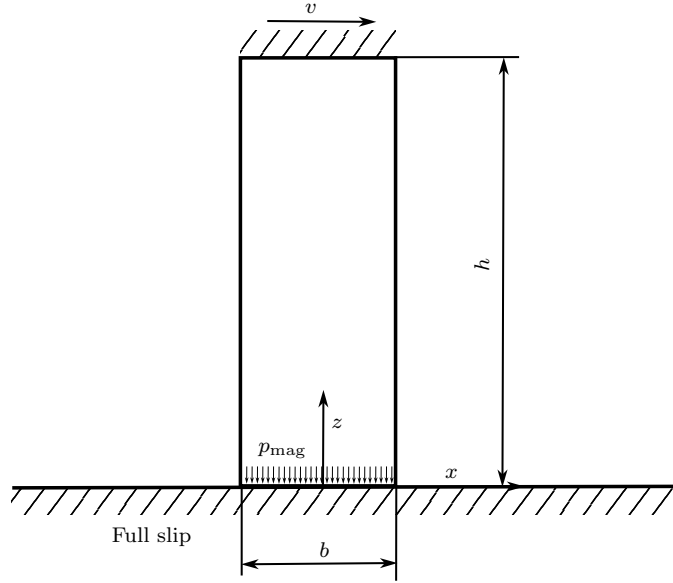


Figure 3.13: Illustration of the contact problem for a block on a flat plane

G_{block} [Pa]	ν_{block}	G_{plane} [Pa]	ν_{plane}	μ	p_{mag} [N/mm ²]	b [m]	h [m]
$80 \cdot 10^9$	0.3	$80 \cdot 10^9$	0.3	0.5	3	0.1	0.5
$\delta_{\text{grid } x/y}$ [m]	$i_{\text{estimated steps}}$	$\epsilon_{\text{force tolerance}}$ [N]		$\epsilon_{\text{distance tolerance}}$ [m]			
0.005	8	1		10^{-12}			
$\delta_{\text{FE model}}$ [m]	$type_{\text{FE model}}$	$i_{\text{FE model}}$	$f_{\text{FE model}}$				
0.006	block	20	20				

Table 3.5: Simulation parameters for the simple extension, block on plane

and the plane is defined through the option entitled "frictional", within Ansys. The calculation time required for the reference solution is approximately 10 minutes, and the same time is required for the extended contact algorithm.

For the verification, the contact pressure distribution is depicted in Fig. 3.14, for two cross-sections along the direction of the applied movement. The contact pressure distribution from the extended contact algorithm matches the one calculated from the FE software pretty well. It can be observed, that the pressure peaks at the border are higher, due to a slightly finer grid in the reference solution. Further, the vertical displacement of the block within the contact area is depicted in Fig. 3.15. As is evident from Fig. 3.15, the influence of elastic body deformation can be adequately reproduced by the extension of the contact algorithm. A depiction of the contact status can be found in the appendix D. This further proves the validity of the extension of the contact algorithm, which albeit in this simple example does not provide any computational advantage, in terms of computation time. However, generally speaking in more application-oriented and complicated simulations, the difference in computation time will be much more compelling.

Additionally, the vertical deformation of the block is plotted, as a side view, see Fig. 3.16. It demonstrates that the extended contact algorithm, with its contact interaction, can simulate the elastic body deformations of the block sufficiently accurate. For this representation the applied load is lower by a factor of eight in Ansys, to match one specific load increment from the extended contact algorithm.

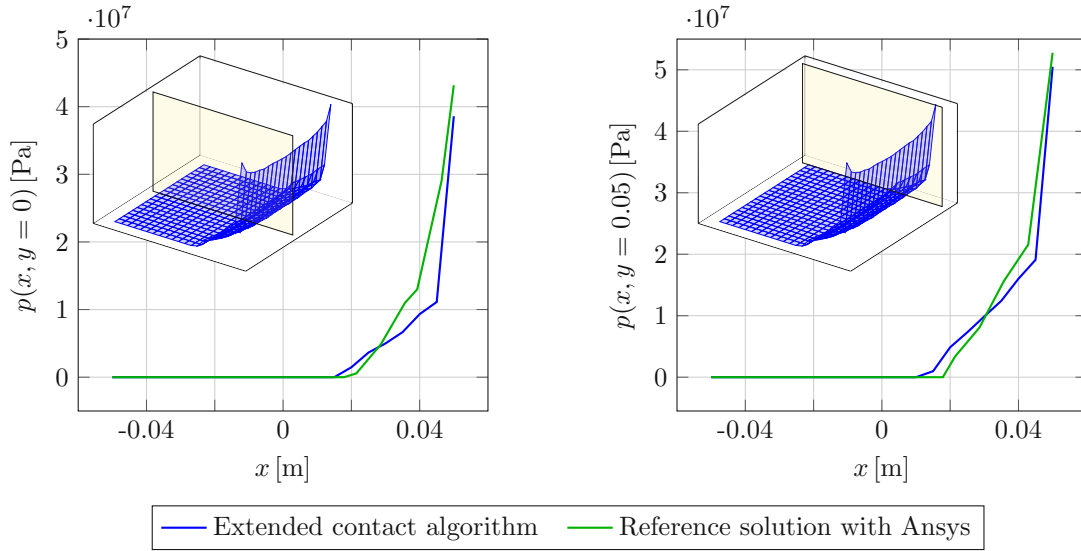


Figure 3.14: Contact pressure distribution of the block, comparison Ansys against extended model

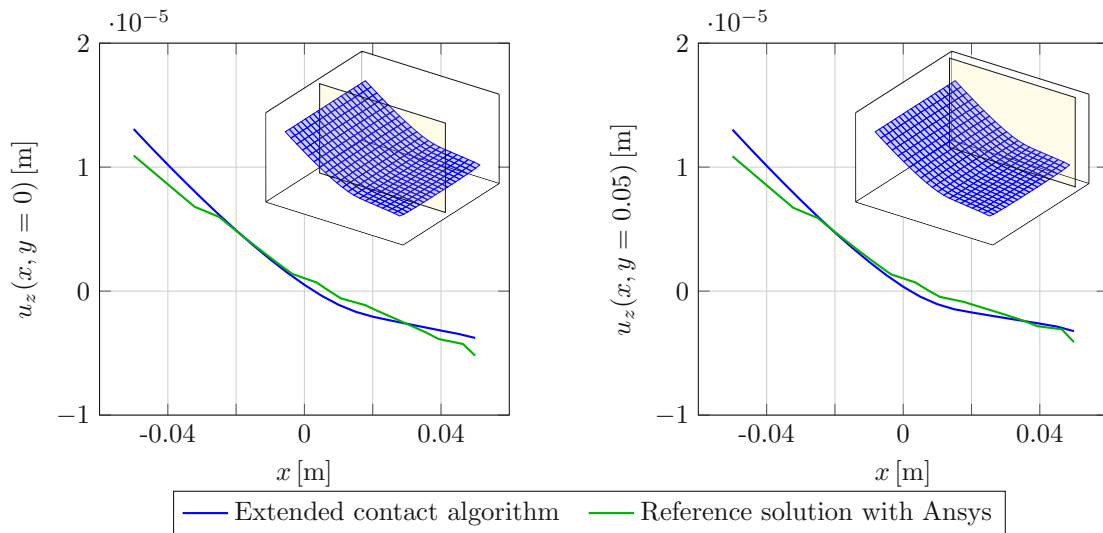


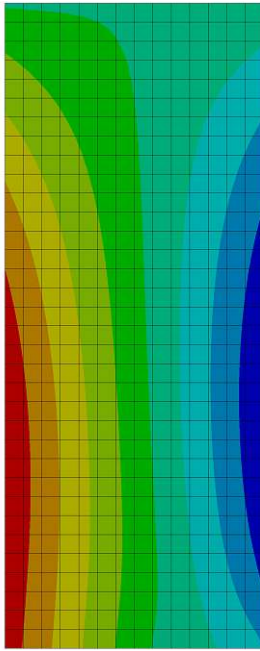
Figure 3.15: Vertical displacement of the block, comparison Ansys against extended model

3.2.2 Extension compared to the benchmark results

Finally, a comparison to the benchmark from section 3.1.2 is made once again. The parameters utilised in the simulations are identical to those used in the convergence analysis of the FE model and can be found in the appendix, see Table 3.6.

Fig. 3.17 and Fig. 3.18 show that through the extension, the contact algorithm matches the benchmark results almost perfectly. The only discrepancy is that the pressure peaks at the lateral borders computed by the extended contact algorithm can not match the ones from the benchmark. This can be attributed once again to a difference in the refinement of the discrete points.

Vertical deformation, Ansys



Vertical deformation, extended contact model

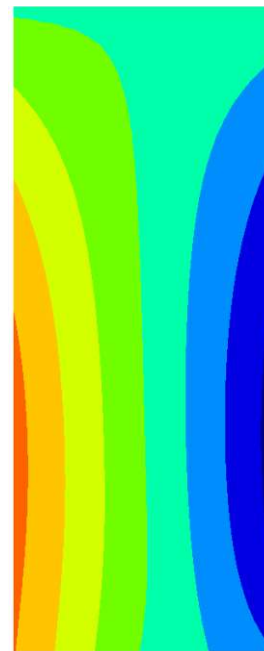


Figure 3.16: Deformation of the block in the vertical direction, side view for the two different simulations

$G_{\text{pole shoe}} [\text{Pa}]$	$\nu_{\text{pole shoe}}$	$G_{\text{rail}} [\text{Pa}]$	ν_{rail}	μ	$p_{\text{mag}} [\text{N/mm}^2]$
$80 \cdot 10^9$	0.3	$80 \cdot 10^9$	0.3	0.5	3
$\delta_{\text{grid } x/y} [\text{m}]$	$i_{\text{estimated steps}}$	$\epsilon_{\text{force tolerance}} [\text{N}]$	$\epsilon_{\text{distance tolerance}} [\text{m}]$		
0.003	8	1	10^{-12}		
$\delta_{\text{FE model}} [\text{m}]$	$type_{\text{FE model}}$			$i_{\text{FE model}}$	$f_{\text{FE model}}$
0.003	benchmark, with $\xi = 28.2\text{mm}$			20	25

Table 3.6: Simulation parameters for the extended contact algorithm, benchmark problem pole shoe on rail

Further, the relative error values, computed in the same way as in 3.1.2 are given here for the extension in Table 3.7. They show the improvement, achieved through the extension of the contact algorithm.

	$\xi [\text{m}]$	ϵ_p	ϵ_{u_z}	$t_{\text{FE}} [\text{s}]$	$t_{\text{contact algorithm}} [\text{s}]$
extended model	$28.2 \cdot 10^{-3}$	16.73%	2.37%	16800	1200
basic model	$28.2 \cdot 10^{-3}$	45.89%	11.69%	16800	252

Table 3.7: Error values for the contact pressure distribution and the vertical displacement, and computation times for the FE benchmark and the extended contact algorithm (and the basic model)

The verification of the contact model is thus finalised through these comparisons. It has been demonstrated that the extended model possesses the capacity to simulate a variety of contact problems in a sufficiently accurate manner, while being relatively fast.

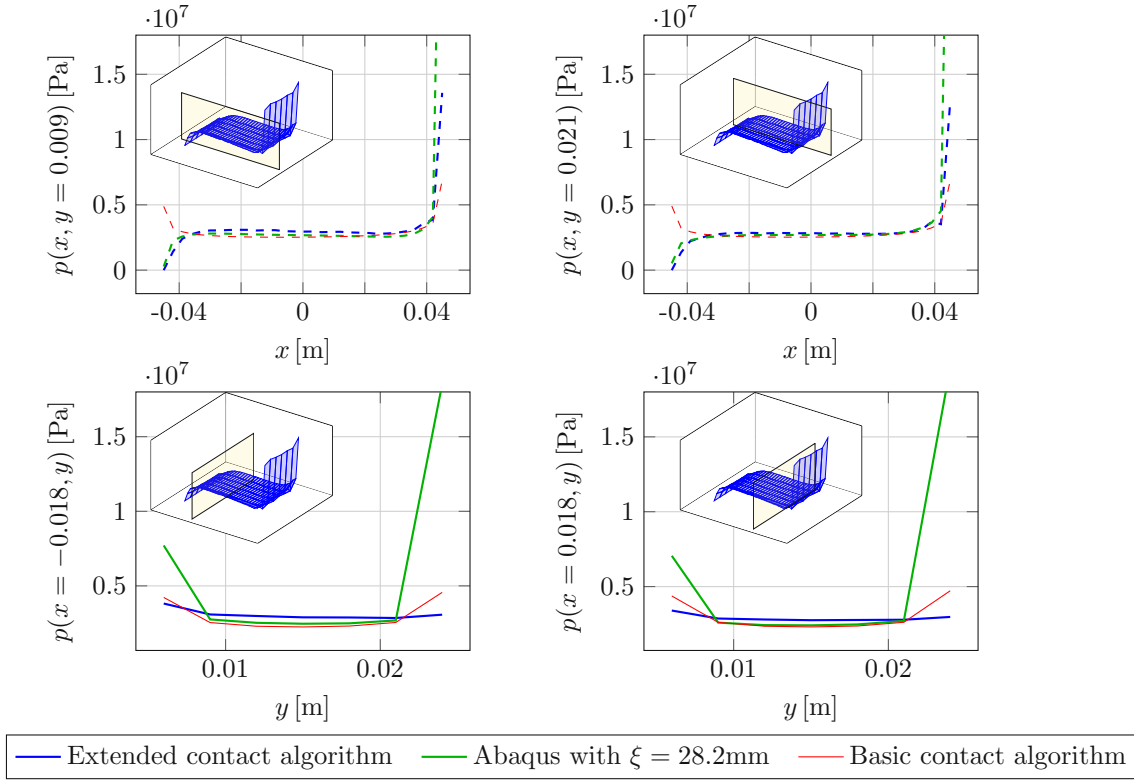


Figure 3.17: Contact pressure distribution, benchmark comparison at four cross-sections for $\xi = 28.2\text{mm}$

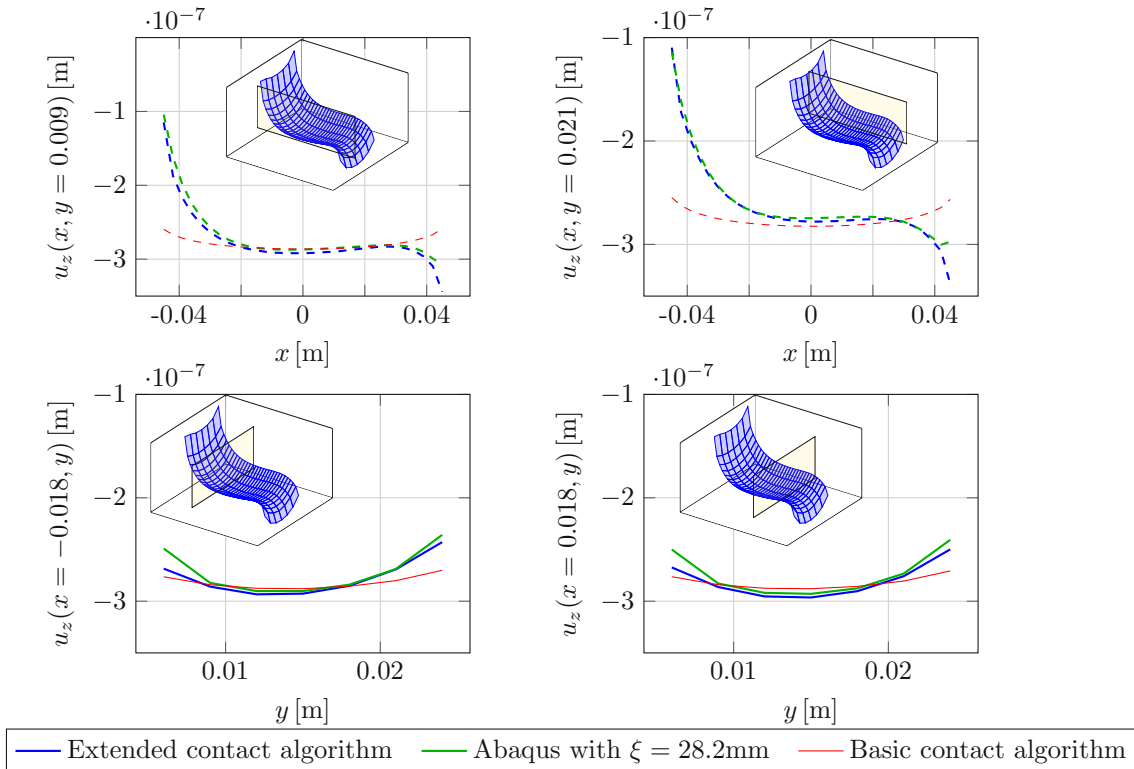


Figure 3.18: Vertical displacement, benchmark comparison at four cross-sections for $\xi = 28.2\text{mm}$

4. Results for the MTB

Following the establishment and verification of the contact algorithm, some results are presented here for some more general use cases of the contact between the pole shoe and the rail. Some parameters are varied for this, in particular the friction coefficient and the magnetic load.

4.1 Results for a constant magnetic load

In this section, the magnetic loading p_{mag} is kept constant, whilst the friction coefficient is varied throughout the section. To begin with, the parameters are all set identical to those in Table 3.6, except for $\delta_{\text{FE model}} = 0.003\text{m}$ and the utilisation of the full pole shoe geometry for the extension. The results of the contact pressure distribution and vertical displacement are depicted in Fig. 4.2 and Fig. 4.3, once more for the left contact area of the pole shoe. In contrast to the basic contact model in 3.1.2, the results obtained here are significantly different. A tilting to the front and a lift-off at the back are clearly visible in the plot for the vertical displacement. This adaptation is achieved through the inclusion of the elastic body deformations of the pole shoe. Consequently, a markedly differing contact pressure distribution and contact area is attained. The pole shoe experiences a loss of contact at the rear, thus resulting in an absence of pressure there. Conversely, an increase in penetration depth and contact pressure is observed at the front.

Moreover, the friction coefficient is varied to demonstrate the system behaviour and the general capability of the algorithm. The parameter $f_{\text{FE model}}$ varies in order to achieve a convergent solution, although this can be achieved automatically. The calculation times for the simulations with different friction factors are listed in Table 4.1. In Fig. 4.1 the lines, separating the transition from 'no contact' on the left of it, to 'sliding contact' on the right of said line, are depicted. A comparable figure can be found in a previous master's thesis for the pole shoe rail contact, see [22], where it was obtained from a full finite element analysis. The discrepancy in calculation times can be attributed to the computational complexity being dependent on the number of discrete points in contact. It is evident from the depiction of the contact area that, for a higher friction factor, the number of points in contact is reduced. Consequently, the calculation time is also lower as the number of the unknowns in the LCP decreases.

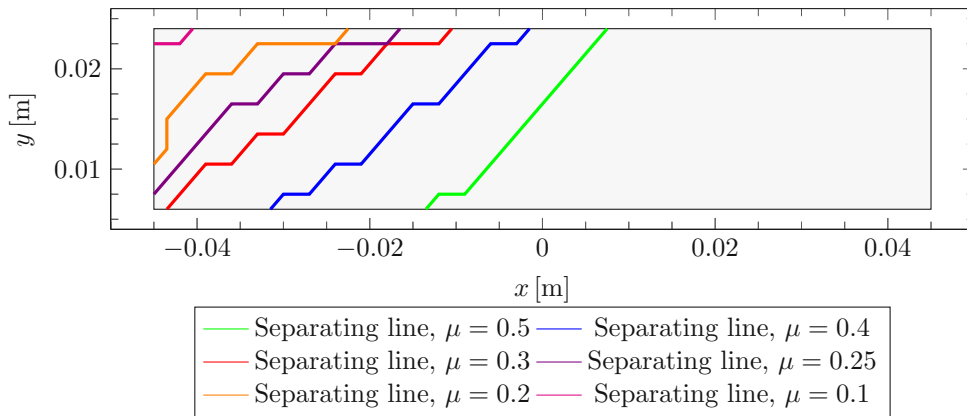


Figure 4.1: Contact status for different friction coefficients, results for the full pole shoe simulation

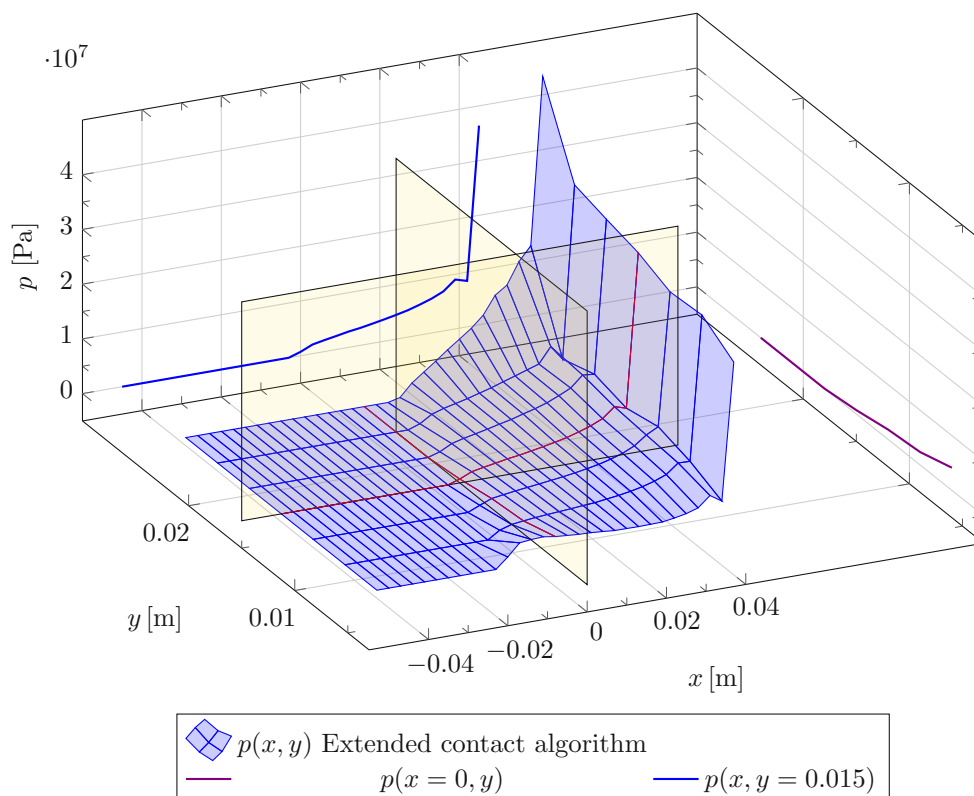


Figure 4.2: Contact pressure distribution, result from the extended model for $\mu = 0.5$

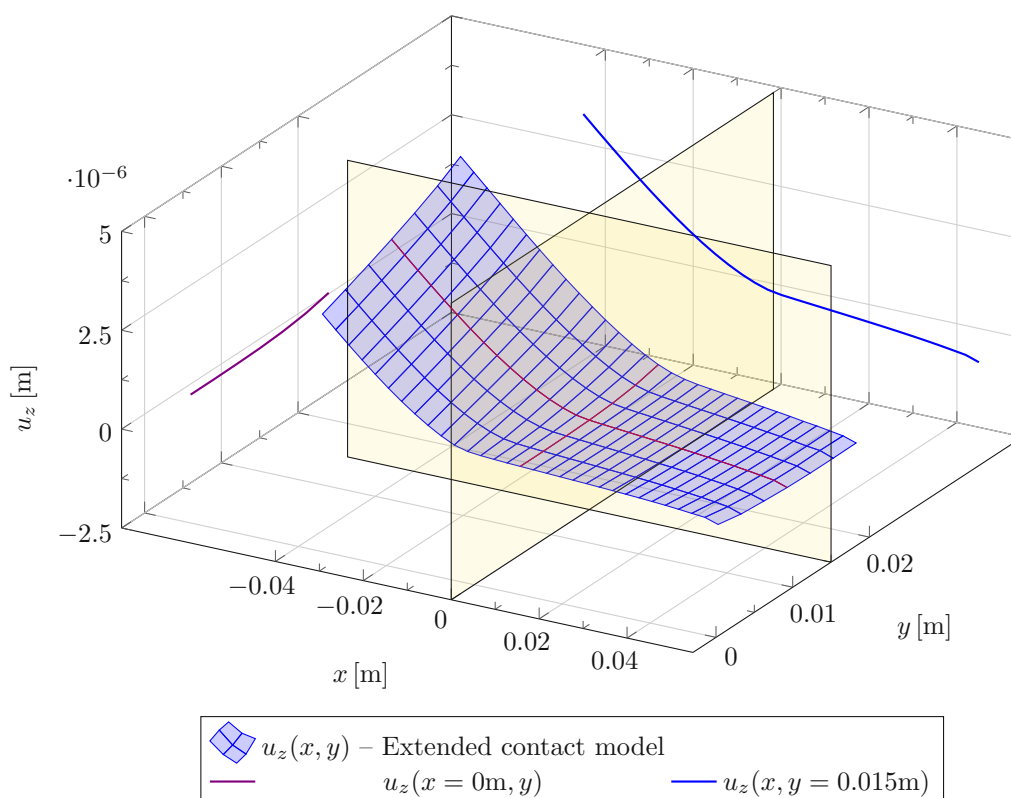


Figure 4.3: Vertical displacement, result from the extended model for $\mu = 0.5$

μ	0.5	0.4	0.3	0.25	0.2	0.1
calculation time t [s]	494	648	816	940	998	1057

Table 4.1: Calculation times for the extended model, for varying friction coefficients

Further plots of the results for a constant magnetic load with a different friction coefficient can be found in the appendix D.

Here a brief discussion on the boundary conditions used in the extension of the contact algorithm is held, however a more extensive examination is beyond the scope of this work. It is generally possible to adapt them to the specific use case within the extension. The results displayed in this chapter have been obtained by fixing the pole shoe on top at five discrete points in the lateral and vertical directions, see Fig. 4.4. The displacement in the longitudinal direction is prohibited at four discrete points at a certain height on the back side of the pole shoe. These points are modelled as being connected to the main body of the MTB through an intermediate plate. Therefore, when comparing the results obtained in this study to those from other sources, such as those from previously done FE simulations, it is imperative to consider the potential discrepancy in boundary conditions across different models.

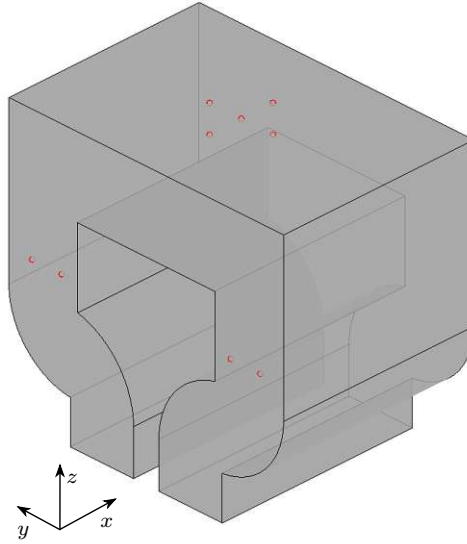


Figure 4.4: Depiction of the five discrete points at the top, with a prescribed movement of zero for y and z direction, and the four points at the back side, with a prescribed movement of zero for the x direction.

4.2 Results for different loadings

As within the application of the MTB a constant magnetic load is not really realistic. Hence, to further demonstrate the applicability of the model, the results for a different non-constant magnetic load are presented. The parameters used for these simulations are listed in Table 4.2. The applied magnetic load is arbitrarily chosen as a Gauss function along the longitudinal and lateral directions, with the maximum value set at the back-side of the pole shoe. It is described by the equation

$$p_{\text{mag Gaussian}} = p_0 \left(\frac{1}{4} + \frac{3}{4} \left(e^{-\frac{(x-c_x)^2}{2\sigma_x^2}} e^{-\frac{(y-c_y)^2}{2\sigma_y^2}} \right) \right), \quad (4.1)$$

with x and y as the horizontal coordinates of the discrete points describing the surfaces, c_x being half the length of the pole shoe, and the other parameters chosen to end up with an appropriate function.

$G_{\text{pole shoe}}$ [Pa]	$\nu_{\text{pole shoe}}$	G_{rail} [Pa]	ν_{rail}	μ	p_0 [N/mm ²]
$80 \cdot 10^9$	0.3	$80 \cdot 10^9$	0.3	0.2	3
$\delta_{\text{grid } x/y}$ [m]	$i_{\text{estimated steps}}$	$\epsilon_{\text{force tolerance}}$ [N]	$\epsilon_{\text{distance tolerance}}$ [m]		
0.003	8	1	10^{-12}		
$\delta_{\text{FE model}}$ [m]	$type_{\text{FE model}}$	$i_{\text{FE model}}$	$f_{\text{FE model}}$		
0.003	full pole shoe	20	85		

Table 4.2: Simulation parameters for simulation with non-constant magnetic loading, pole shoe on rail

The contact pressure distribution and vertical displacement are illustrated in Fig. 4.5 and Fig. 4.6. A clear discrepancy emerges in the comparison with the simulation that utilised a constant magnetic load. A pressure peak is still observed at the leading edge of the poles, although its magnitude is rather small in comparison to the rest of the contact pressure distribution. In the context of this particular loading, no lifting of the contact occurs at any point and a higher displacement at the back of the pole shoe can be identified. The calculation time for this simulation is 942s. The results of a further simulation, incorporating an alternative arbitrary magnetic load, can be found in the appendix D.

Overall, it is evident that the contact algorithm in question facilitates the basic quasi-static simulation of an MTB with an arbitrary magnetic load in a satisfactory manner. The versatility and computational efficiency that can be achieved are demonstrated. However, it should be noted that further adaptations and implementations will be necessary to ensure the effectiveness of the algorithm. A discourse on the potential of the algorithm is provided in the conclusion and outlook of this work, as outlined in chapter 5.

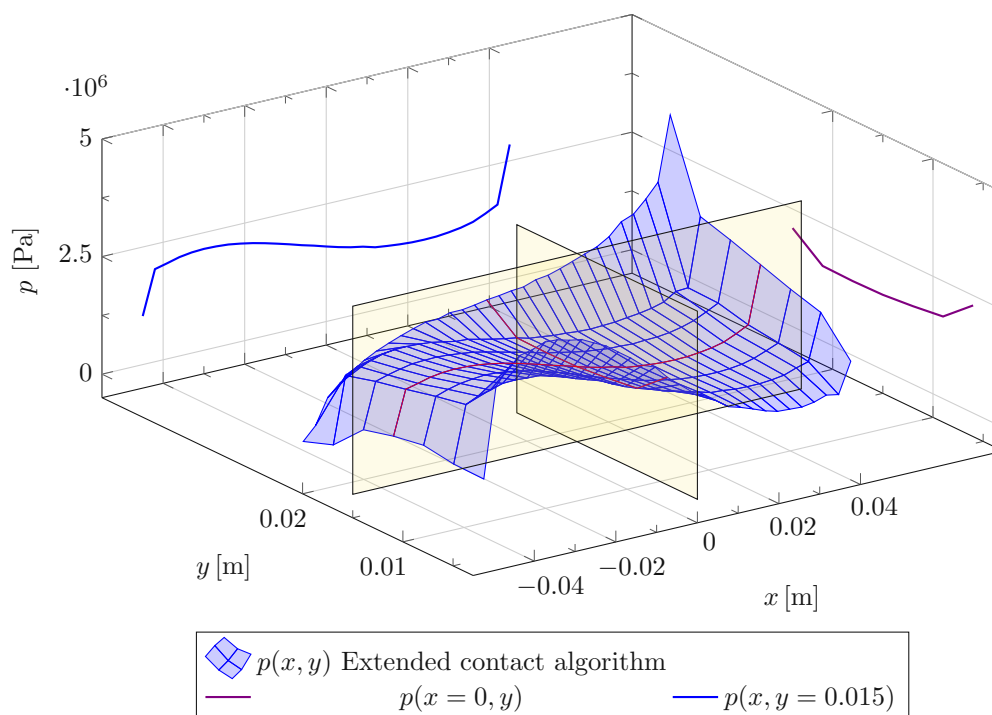


Figure 4.5: Contact pressure distribution, result from the extended model for a Gaussian-distributed magnetic loading

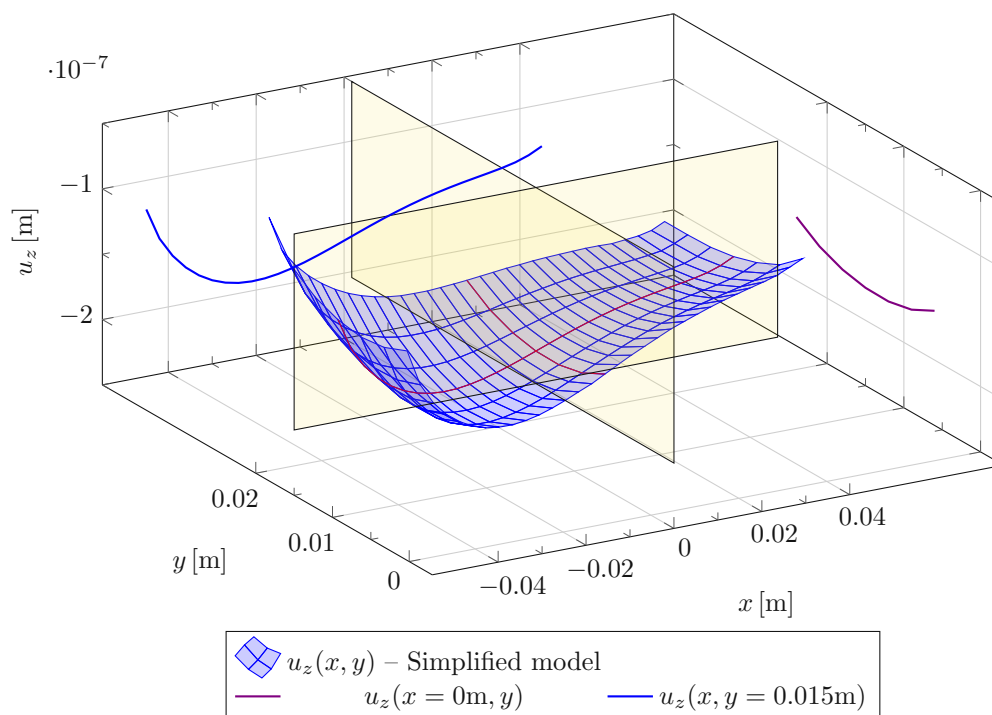


Figure 4.6: Vertical displacement, result from the extended model for a Gaussian-distributed magnetic loading

5. Conclusion and Outlook

The present work derives and establishes a simplified numeric contact model, that is specifically designed for the application of the magnetic track brake, where a frictional sliding contact arises. The approach based on a discrete setting of the elastic half-space, in combination with the formulation of the linear complementarity problem, appears suitable for these kinds of contact problems, and hence is employed here.

Based on these formulations, a basic contact algorithm was established within MATLAB. It is observed that this algorithm can resolve contact problems sufficiently accurate, for objects undergoing no or very little elastic body deformations. However, for contact problems where the objects experience higher elastic deformations, discrepancies occur. Therefore, an extension, including those deformations is incorporated. It is coupled such that the kinematic task of contact resolution is independent thereof and thus the computational efficiency remains high. The verification is split into one section for the basic and one for the extended contact model. It comprises the following contact problems:

- A paraboloid on a flat plane, with an analytical solution, to verify the basic algorithm.
- The contact between the pole shoe and the rail, modelled up to a certain height to assess the impact of the elastic body deformations. The reference solution is derived through an FE simulation and they are compared against the basic as well as the extended contact algorithm.
- A block on a flat plane, through which the FE model of the extension is further verified. The compared results are derived utilising a commercial FE simulation.

The accuracy of the contact algorithm and its extension are proven through those substantially different contact problems.

With the derived contact algorithm, it is already possible to obtain highly individualised results for different input parameters. The boundary conditions can be set to mirror reality further and different loadings can be applied to the left and right sides of the pole shoe in order to take into account asymmetries. It is principally possible to vary the geometry and move the pole shoe out of its symmetry plane, although no such studies have been conducted so far. The computation times for the newly developed simulations are equal to or faster than those for classical FE models. However, the pre-computation of the matrix of influence coefficients necessary for the algorithm, which exceeds the basic computation time by multiple orders of magnitude, has to be considered.

For further development, an important step would be to transfer the algorithm from MATLAB to a software capable of facilitating more rapid simulations. The intention is to utilise a programming language that can be implemented into the multi-body dynamics software SIMPACK. This would also enable the simulation of not only a single quasi-static pole shoe in contact, but a transient braking manoeuvre, or even multiple pole shoes mounted on an MTB connected to a railway vehicle. The potential for the incorporation of a third body layer represents a further developmental opportunity that is of interest for the ongoing research project. Whilst there is still no definitive implementation in place, it appears that facilitating a factor, determined by experimental data, or simulating an additional body in contact are both viable options to consider.

References

- [1] IEA Paris. ‘Tracking clean energy progress 2023’. (2023), [Online]. Available: <https://www.iea.org/reports/tracking-clean-energy-progress-2023> (visited on 25/11/2024).
- [2] J. Ihme, *Schienefahrzeugtechnik*. Springer, 2016.
- [3] D. Tippelt, *Self-excited vibrations of magnetic track brakes: Modelling, analysis and mitigation*. PhD thesis, TU Wien, 2022.
- [4] O. Arias-Cuevas and Z. Li, ‘Field investigations into the performance of magnetic track brakes of an electrical multiple unit against slippery tracks. part 1: Adhesion improvement’, *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 225, no. 6, pp. 613–636, 2011.
- [5] O. Arias-Cuevas and Z. Li, ‘Field investigations into the performance of magnetic track brakes of an electrical multiple unit against slippery tracks. part 2: Braking force and side effects’, *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 226, no. 1, pp. 72–94, 2012.
- [6] B. Ebner, D. Tippelt, J. Edelmann and M. Plöchl, ‘Active mitigation of self-excited vibrations of a magnetic track brake’, *Journal of Physics: Conference Series*, vol. 2647, no. 15, p. 152007, 2024. DOI: 10.1088/1742-6596/2647/15/152007. [Online]. Available: <https://dx.doi.org/10.1088/1742-6596/2647/15/152007>.
- [7] E. Galardi, E. Meli, D. Nocciolini, L. Pugi and A. Rindi, ‘Development of efficient models of magnetic braking systems of railway vehicles’, *International Journal of Rail Transportation*, vol. 3, no. 2, pp. 97–118, 2015.
- [8] H. Hertz, ‘Ueber die berührung fester elastischer körper.’, *Journal für die reine und angewandte Mathematik*, 1882.
- [9] Q. Wang, J. Hu and G. Zhou, ‘The mixed method with two lagrange multiplier formulations for the signorini problem’, *Journal of Computational and Applied Mathematics*, vol. 452, p. 116115, 2024, ISSN: 0377-0427. DOI: <https://doi.org/10.1016/j.cam.2024.116115>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377042724003649>.
- [10] L. Baillet and T. Sassi, ‘Mixed finite element methods for the signorini problem with friction’, *Numerical Methods for Partial Differential Equations: An International Journal*, vol. 22, no. 6, pp. 1489–1508, 2006.
- [11] K. Wang, Q. H. Qin, Y. L. Kang, J.-S. Wang and C.-Y. Qu, ‘A direct constraint-trefftz fem for analysing elastic contact problems’, *International journal for numerical methods in engineering*, vol. 63, no. 12, pp. 1694–1718, 2005.
- [12] A. Eterovic and K. Bathe, ‘On the treatment of inequality constraints arising from contact conditions in finite element analysis’, *Computers & Structures*, vol. 40, no. 2, pp. 203–209, 1991.
- [13] A. S. Lone, G. Harmain and A. Jameel, ‘Modeling of contact interfaces by penalty based enriched finite element method’, *Mechanics of Advanced Materials and Structures*, vol. 30, no. 7, pp. 1485–1503, 2023.
- [14] E. Burman, P. Hansbo and M. G. Larson, ‘The augmented lagrangian method as a framework for stabilised methods in computational mechanics’, *Archives of Computational Methods in Engineering*, vol. 30, no. 4, pp. 2579–2604, 2023.

- [15] G. Kloosterman, 'Contact methods in finite element simulations', Ph.D. dissertation, Universiteit Twente, 2002.
- [16] K. Hibbitt and I. Sorensen, *Abaqus/cae user's manual*.
- [17] C. Renaud and Z.-Q. Feng, 'Bem and fem analysis of signorini contact problems with friction', *Computational mechanics*, vol. 31, pp. 390–399, 2003.
- [18] E. Vollebregt, *User guide for contact, rolling and sliding contact with friction, v24.1, june 2024*.
- [19] E. Vollebregt and G. Segal, 'Solving conformal wheel–rail rolling contact problems', *Vehicle System Dynamics*, vol. 52, no. sup1, pp. 455–468, 2014.
- [20] J. J. Kalker, *Three-dimensional elastic bodies in rolling contact*. Springer Science & Business Media, 2013, vol. 2.
- [21] E. Kocbay, A. Steininger, A. Pavicsics, E. Arslan and J. Edelmann, 'Efficient and simplified numerical contact model for the braking simulation of a magnetic track brake', *Meccanica*, Jan. 2025. DOI: 10.1007/s11012-024-01926-8.
- [22] A. Pavicsics, 'Finite element modelling of the frictional sliding contact within the application of the magnetic track brake', M.S. thesis, Technische Universität Wien, Wien, 2024. DOI: 10.34726/hss.2024.110750.
- [23] K. L. Johnson, 'Point loading of an elastic half-space', in *Contact Mechanics*. Cambridge University Press, 1985, pp. 45–83.
- [24] M. Paggi, A. Bemporad and J. Reinoso, 'Computational methods for contact problems with roughness', in Jan. 2020, pp. 131–178, ISBN: 978-3-030-20376-4. DOI: 10.1007/978-3-030-20377-1_4.
- [25] M. Pauly, D. K. Pai and L. J. Guibas, 'Quasi-rigid objects in contact', in *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '04, Grenoble, France: Eurographics Association, 2004, pp. 109–119, ISBN: 3905673142. DOI: 10.1145/1028523.1028539. [Online]. Available: <https://doi.org/10.1145/1028523.1028539>.
- [26] H. Wendland, *Scattered data approximation*. Cambridge university press, 2004, vol. 17.
- [27] R. W. Cottle, J.-S. Pang and R. E. Stone, *The linear complementarity problem*. SIAM, 2009.
- [28] S. J. Wright, *Numerical optimization*, 2006.
- [29] G. Ostermeyer, 'On the dynamics of the friction coefficient', *Wear*, vol. 254, no. 9, pp. 852–858, 2003, Papers presented at the 280th We-Hereaus Seminar Integrating Friction and Wear Research, ISSN: 0043-1648. DOI: [https://doi.org/10.1016/S0043-1648\(03\)00235-7](https://doi.org/10.1016/S0043-1648(03)00235-7). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0043164803002357>.
- [30] 'The mathworks inc, matlab, partial differential equation toolbox'. (), [Online]. Available: https://de.mathworks.com/help/pde/index.html?s_tid=CRUX_lftnav. (accessed: 17.02.2025).
- [31] A. Aho and J. Ullman, *Foundations of Computer Science: C Edition* (Principles of computer science series). W. H. Freeman, 1994, ISBN: 9780716782841.

- [32] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to algorithms*. MIT press, 2022.
- [33] V. L. Popov, M. Heß and E. Willert, *Handbook of contact mechanics: exact solutions of axisymmetric contact problems*. Springer Nature, 2019.
- [34] D. D. I. für Normung, *Bahnanwendungen - oberbau - schienen - teil 1, din en 13674-1*, 2017.
- [35] T. K. CEN/TC256, *Bahnanwendungen – bremse – anforderungen an die funktion und leistungsfähigkeit von magnetschienenbremssystemen für schienenfahrzeuge, en 16207*, 2024.
- [36] I. Farmaga, P. Shmigelskyi, P. Spiewak and L. Ciupinski, ‘Evaluation of computational complexity of finite element analysis’, in *2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*, 2011, pp. 213–214.
- [37] J. H. Friedman, J. L. Bentley and R. A. Finkel, ‘An algorithm for finding best matches in logarithmic expected time’, *ACM Trans. Math. Softw.*, vol. 3, no. 3, pp. 209–226, Sep. 1977, ISSN: 0098-3500. DOI: 10.1145/355744.355745. [Online]. Available: <https://doi.org/10.1145/355744.355745>.

Appendix

A. Derivation of the time complexity

To get a better understanding of how the upper bound of the time complexity is derived, a sketch in form of a tree chart is plotted in Fig. A.1. Within it, x is the number of nodes of the FE model and n denotes the number of discrete points in contact on the contact surface, with a higher number of contact points. The counts for the respective loops are described by i .

The time complexity of on FEM model is described in [36] and the time complexity of the "knnsearch" algorithm is discussed in great depth in [37]. The LCP Solver uses basic operations and a loop with the maximum loopcount of i_{LCP} . The other operations used in the different subroutines are basic operations and there respective time complexity is described in multiple sources, e.g. [32].

If neglecting all loop counts, one could argue that the FE simulation would define the upper bound of the time complexity with $O(x^2)$. This can be argued, because in general $n^2 < x^2$ holds, due to the fact that the FE model has more discrete points than the surfaces in the contact algorithm. However, the number of nodes in the FE model is usually not that high, because it is assumed to be a simple and easily solvable FE model, and the loop count of the LCP solver is quite high. Therefore, the complexity of the FE solver can be neglected for $O(n^2 \cdot i_{LCP})$.

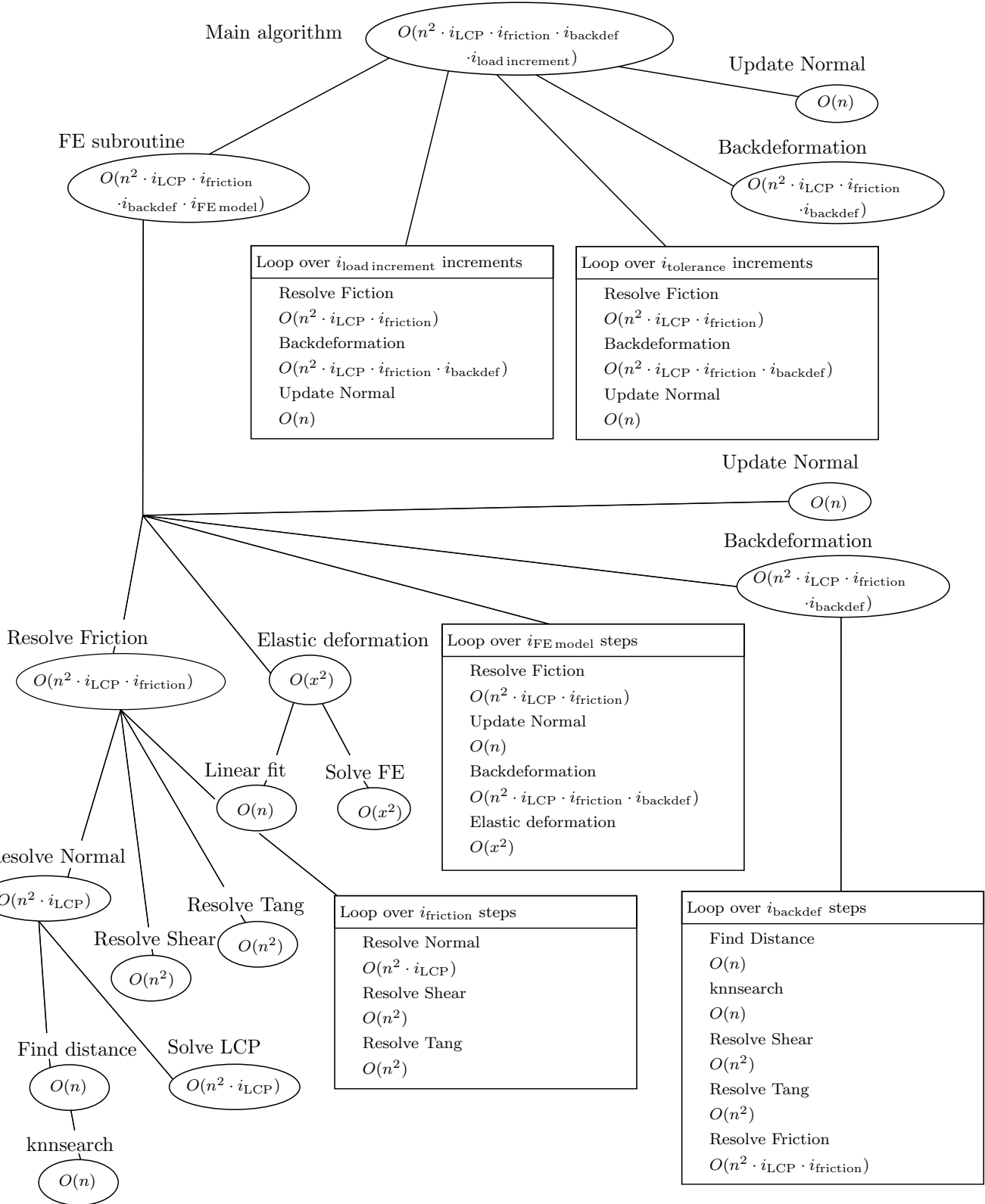


Figure A.1: Sketch on how the time complexity is derived

B. Additional plots for the basic contact model

In Fig. B.3 and Fig. B.4 the contact pressure and displacement in vertical direction simulated from the contact algorithm is compared to the result from the benchmark with the height parameter set to 7.45mm. The already discussed longitudinal asymmetry, in the form of an increase in displacement and pressure at the front and decrease in the back can be observed once more.

As an addition, the relative error between the basic contact algorithm and the benchmark results for the contact pressure and vertical displacement is displayed in Fig. B.1 and Fig. B.2. As previously mentioned, the discrepancies in the pressure peaks can be observed once more, while the error of the vertical displacements achieves a fairly good fit.

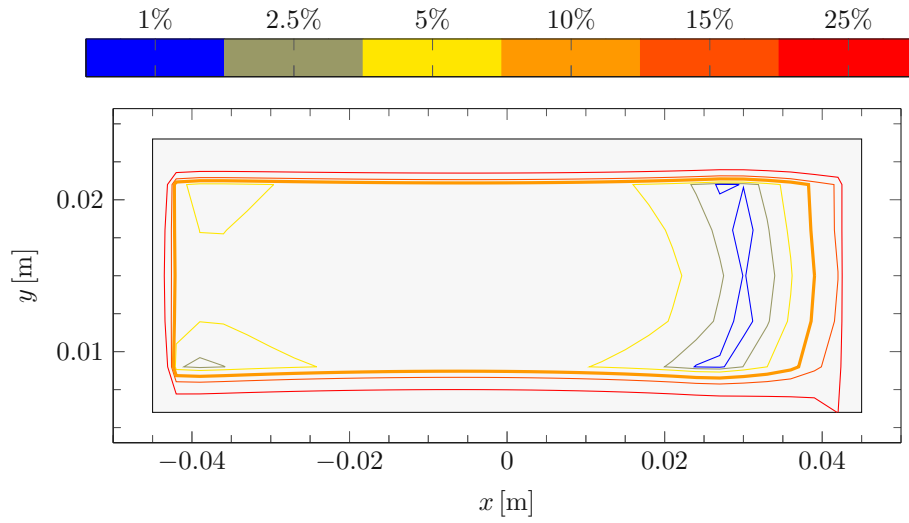


Figure B.1: Relative error of the contact pressure distribution, benchmark comparison for $\xi = 1.15\text{mm}$

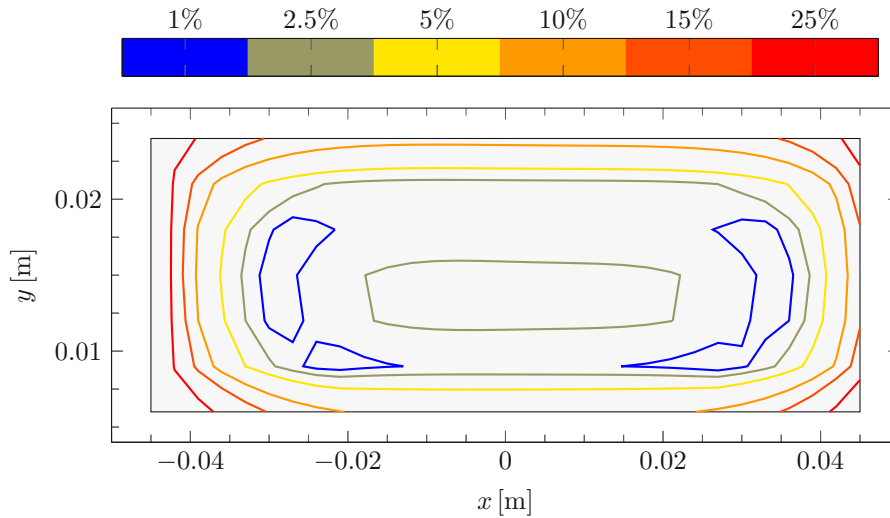


Figure B.2: Relative error of the vertical displacement, benchmark comparison for $\xi = 1.15\text{mm}$

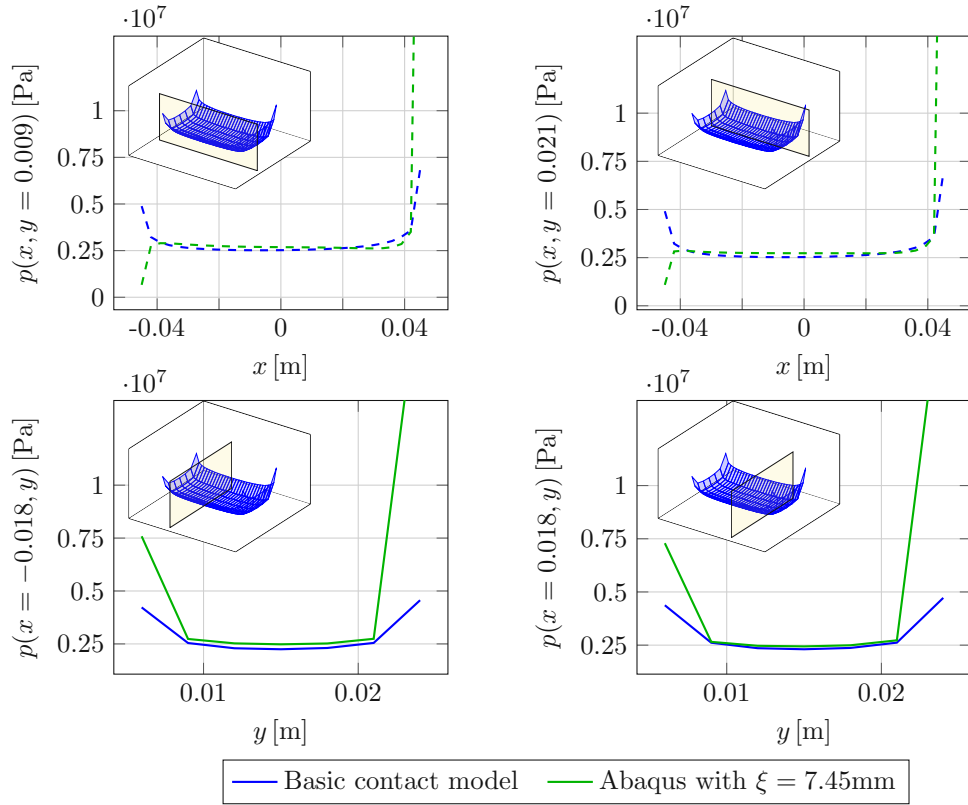


Figure B.3: Contact pressure distribution, basic contact algorithm benchmark comparison for $\xi = 7.45\text{mm}$

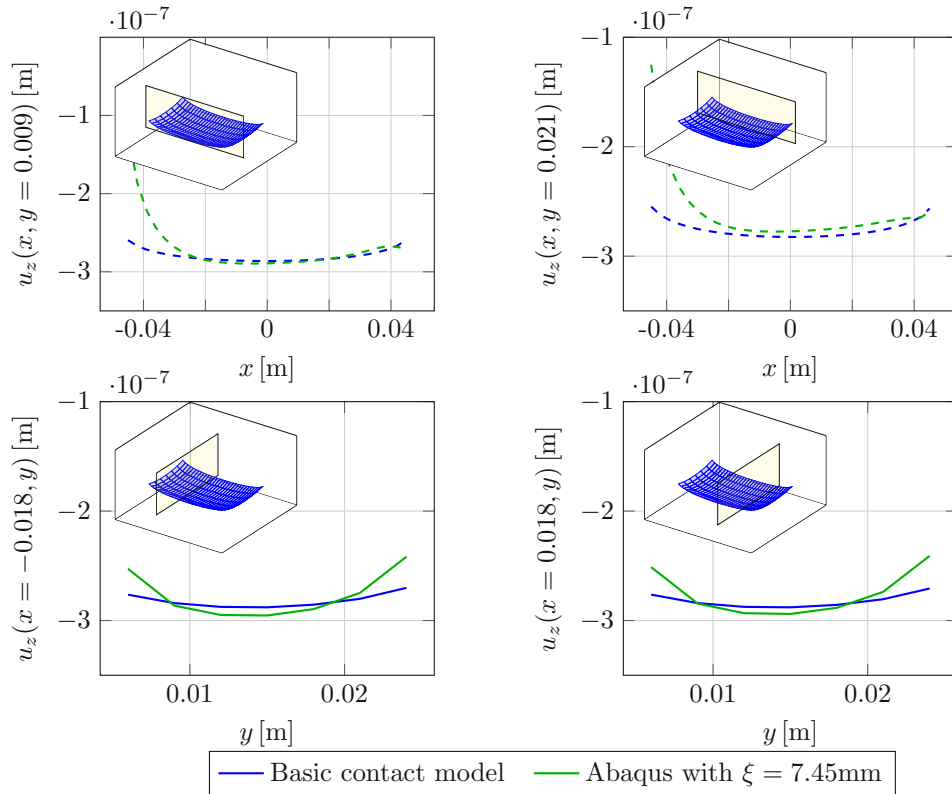


Figure B.4: Vertical displacement, basic contact algorithm benchmark for $\xi = 7.45\text{mm}$

C. Convergence analysis of different parts

C.0.1 Convergence of the basic contact algorithm

In the context of the benchmark comparison for the basic contact algorithm 3.1.2, a distance of 3mm between the discrete points was chosen for the contact algorithm. Here, an argument is presented that demonstrates the sufficiency of this discretisation for the contact simulation between the pole shoe and rail. The parameters employed in these simulations are the same as in Table 3.3, except for $\delta_{\text{grid } x/y}$ which is varied. In order to show convergence, a comparison is made between the results obtained with three distinct grids in the following figures. A cross-sectional depiction along the longitudinal direction is chosen for this, see Fig. C.1 and Fig. C.2.

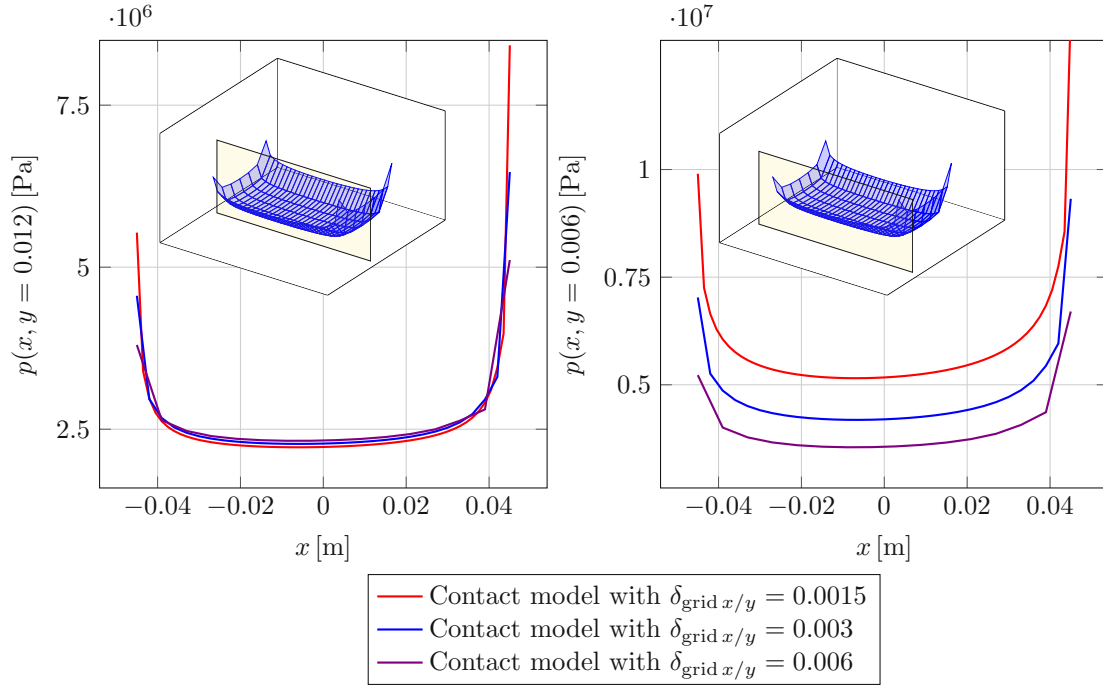


Figure C.1: Contact pressure distribution for different grids, cross-section at $y = 0.012\text{m}$ and $y = 0.006\text{m}$

As can be seen from Fig. C.1, the contact pressure becomes more accurate with finer discretisation, especially at the border, where theoretically an infinite stress gradient occurs. In order to match the same global forces, the inner points experience a slightly higher contact pressure, due to the values at the borders being lower. For the cross-section at the border, the superelevation for the contact pressure shows a greater impact. On initial observation, the discrepancies for the displacements, depicted in Fig. C.2, seem large, but when considering that they are of the order of tenths of a micrometre, they become negligible. The behaviour of increasing pressure peaks for a finer discretisation is e.g. discussed in [22] in the context of the convergence of the finite element model, and is an expected result of the contact pressure at sharp edges. In conclusion, the level of accuracy obtained for $\delta_{\text{grid } x/y} = 3\text{mm}$ is sufficient and leads to fast calculation times.

The calculation times for the different discretisations are presented alongside data regarding the number of discrete points on the rail and a factor comparing the estimated computational complexity, see Table C.1. Here, the estimated complexity of $O(n^2)$ is supported by the factor between the 3mm and 6mm grids, while the factor between the finer two discretisations can not confirm the assessed time complexity. This is most likely attributed to the loop count of the LCP

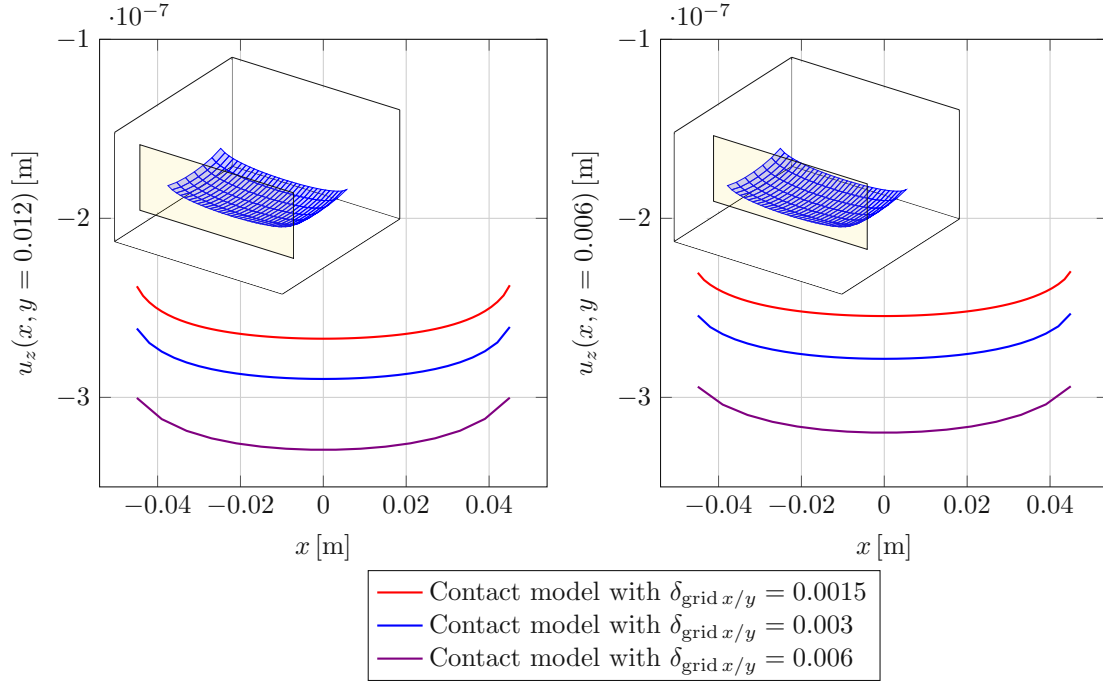


Figure C.2: Vertical displacement for different grids, cross-section at $y = 0.012\text{m}$ and $y = 0.006\text{m}$

solver i_{LCP} exhibiting a certain dependency on the number of points in contact, and consequently increasing the order of the time complexity.

grid of the pole shoe rail	$2 \times 18 \times 7$ $2 \times 31 \times 7$	$2 \times 33 \times 12$ $2 \times 61 \times 13$	$2 \times 63 \times 22$ $2 \times 121 \times 25$
$\delta_{\text{grid } x/y} [\text{m}]$	0.006	0.003	0.0015
n_{rail}	434	1586	6050
calculation time t [s]	20	252	11632
Factor to the next column for n_{rail}	3.65	3.81	
Factor to the next column for t	12.6 = 3.55^2	46.16 = 6.79^2	

Table C.1: Calculation times for benchmark problem, pole shoe on rail

C.0.2 Convergence of the FE model utilised within the extension

The following section presents a brief convergence analysis for the FE model, utilised in the extension of the contact algorithm. The two relevant parameters influencing the convergence are the mesh, especially in the contact area, and the implemented factor $f_{\text{FE model}}$. The full set of parameters used for the simulations is the same as in Table 3.6. The only discrepancies are the geometry used for the extension, which has been adapted to align with the full geometry of the poleshoe (see Fig. 2.6a), and the parameter for the FE mesh chosen for the extension, which is varied, $\delta_{\text{FE model}} = 0.002/0.003/0.004/0.006\text{m}$.

In order to verify the convergence of the FE model within the extension, the final results are analysed. Should these results converge towards a solution, it is assumed that the FE model is also converging. In Fig. C.3 and Fig. C.4, the contact pressure and the displacement in the vertical direction are plotted at two cross-section along the longitudinal axis. Concerning the contact pressure, it is clearly visible that with a finer grid in the contact area, the solution becomes smoother. The appearing ripples for a coarser mesh indicate, that the chosen grid is not sufficiently

fine. In contrast to this, the results for the displacement show no major discrepancies between the different grids used in the extension.

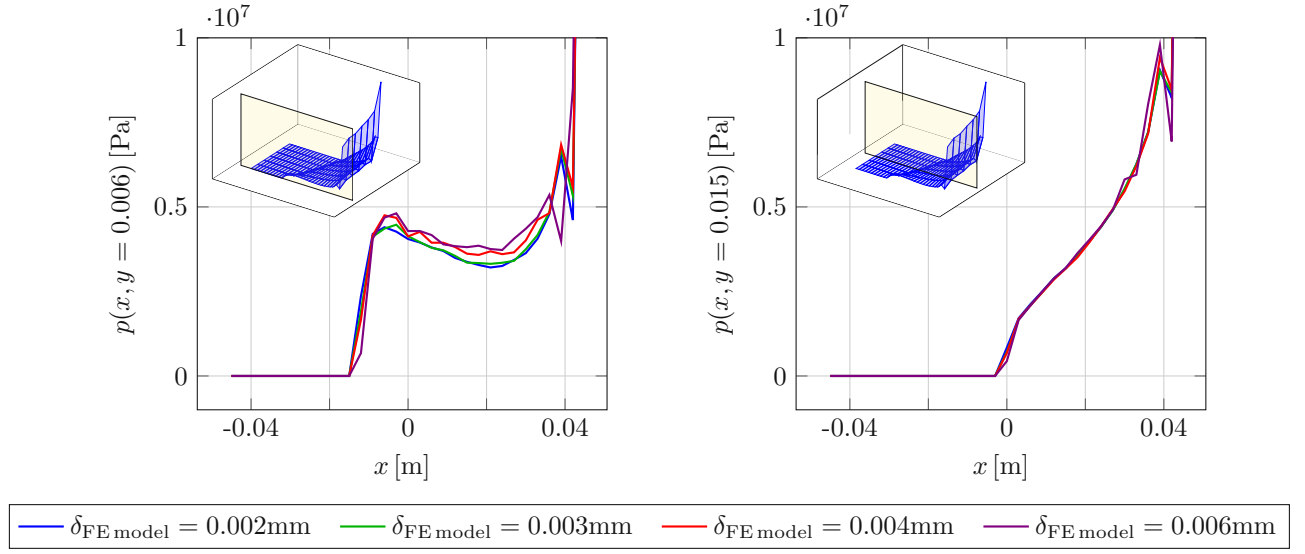


Figure C.3: Contact pressure distribution, convergence of the FE model in the extension

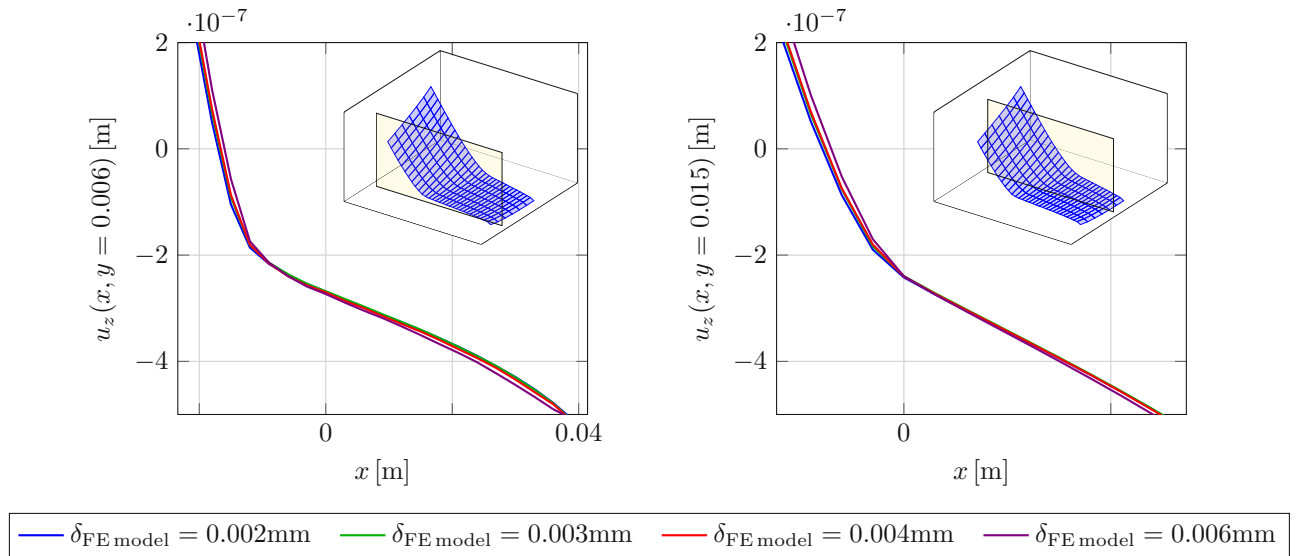


Figure C.4: Vertical displacement, convergence of the FE model in the extension

The calculation times are listed in Table C.2, to facilitate an understanding of the potential trade-off between calculation accuracy and time. With these results, the selection of the grid with 3mm is made on the basis of the satisfactory quality of the results obtained, in combination with the lower computational times. It should be noted that this is the same distance as for the discrete points in the basic contact model. For the extension the values of the tractions are interpolated linearly, therefore the mesh of the extension has to at least match the discretisation of the basic algorithm, to assure a converging simulation. This is shown clearly with these results.

For a working implementation of the FE model, the factor $f_{\text{FE model}}$ has to be set correctly. This can be done by hand or through an automatic fit with already existing data. To show the

$\delta_{\text{FE model}} [\text{m}]$	0.002	0.003	0.004	0.006
calculation time t [s]	764	494	383	312

Table C.2: Calculation times for different FE meshes, within the extended contact algorithm

dependency of the convergence of the FE model on this factor, in the Table C.3 a converging and a diverging simulation is depicted. This is illustrated here through the vertical displacement of the full pole shoe for the last three iterations of the FE model in the extension, for a fixed iteration maximum of $i_{\max} = 20$ iterations. It is clearly visible that the three results on the left side converge towards a certain deformation. The three on the right alternate between two non-physical solutions, and therefore no stable solution can be calculated through the extension.

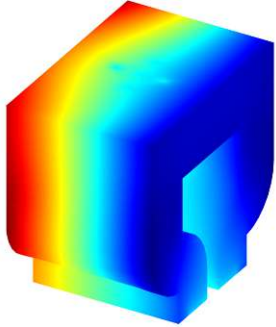
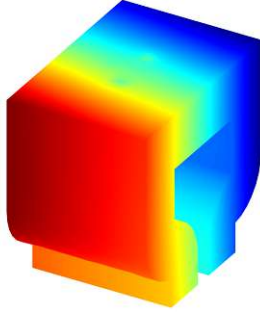
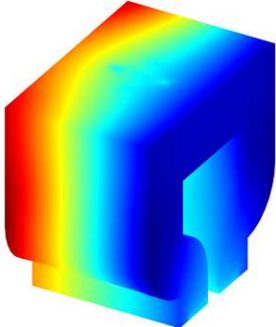
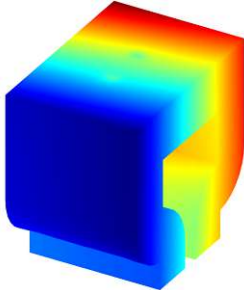
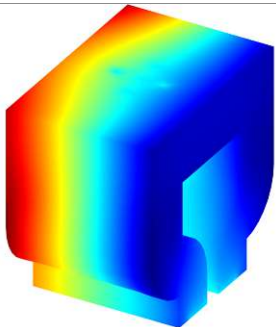
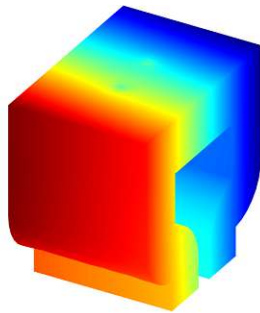
	Converging FE solution with $f_{\text{FE model}} = 25$	Diverging FE solution with $f_{\text{FE model}} = 10$
$i_{\max} - 2$		
$i_{\max} - 1$		
i_{\max}		

Table C.3: Qualitative comparison of the vertical deformation of the pole shoe, for a converging and diverging extension

D. Additional plots for the extended contact model

Additional plots from the block against the flat plane

Fig. D.1 shows the difference of the reference solution and the extended contact algorithm, concerning the contact status. On the left side of the line, the objects are not in contact, while on the right side of the respective lines the objects have sliding contact.

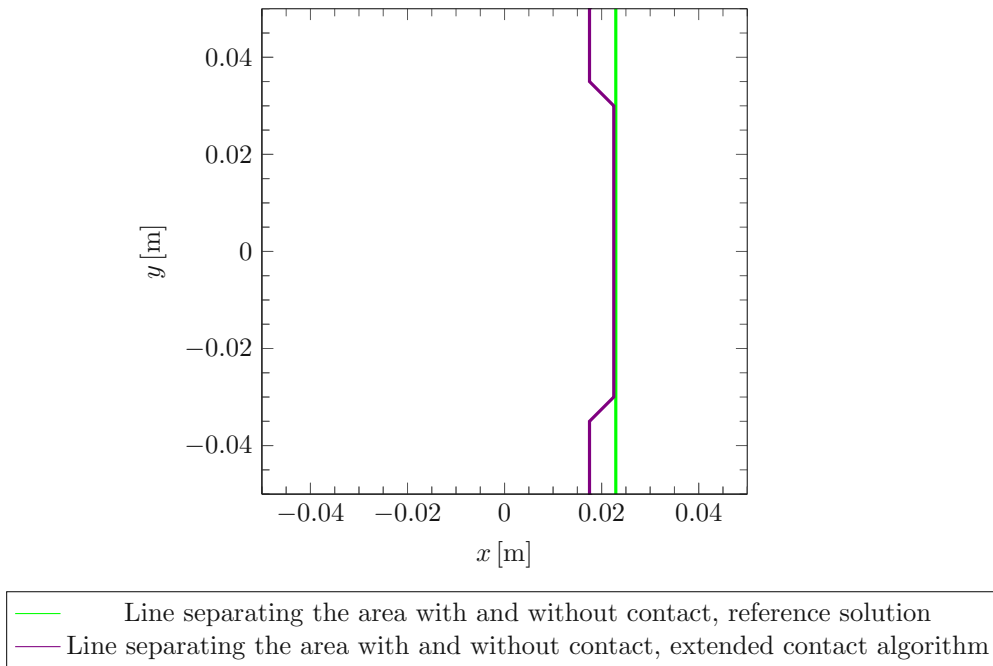


Figure D.1: Contact status, Ansys results and extended contact algorithm

Additional plots for the final results

In Fig. D.2 and Fig. D.3 the contact pressure and vertical displacement is displayed for an additional friction coefficient of $\mu = 0.25$. Further, in Fig. D.4 and Fig. D.5 the contact pressure and vertical displacement for an other arbitrary input of the magnetic load is depicted, to show the versatility of the extended contact algorithm.

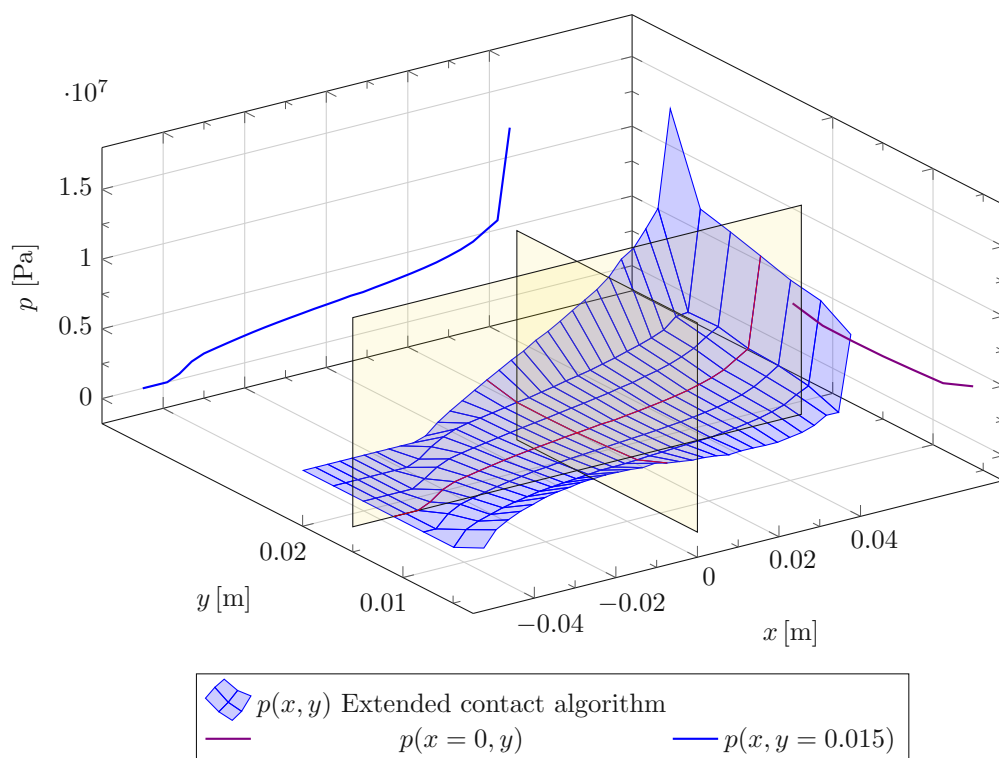


Figure D.2: Contact pressure distribution, result from the extended model for $\mu = 0.25$

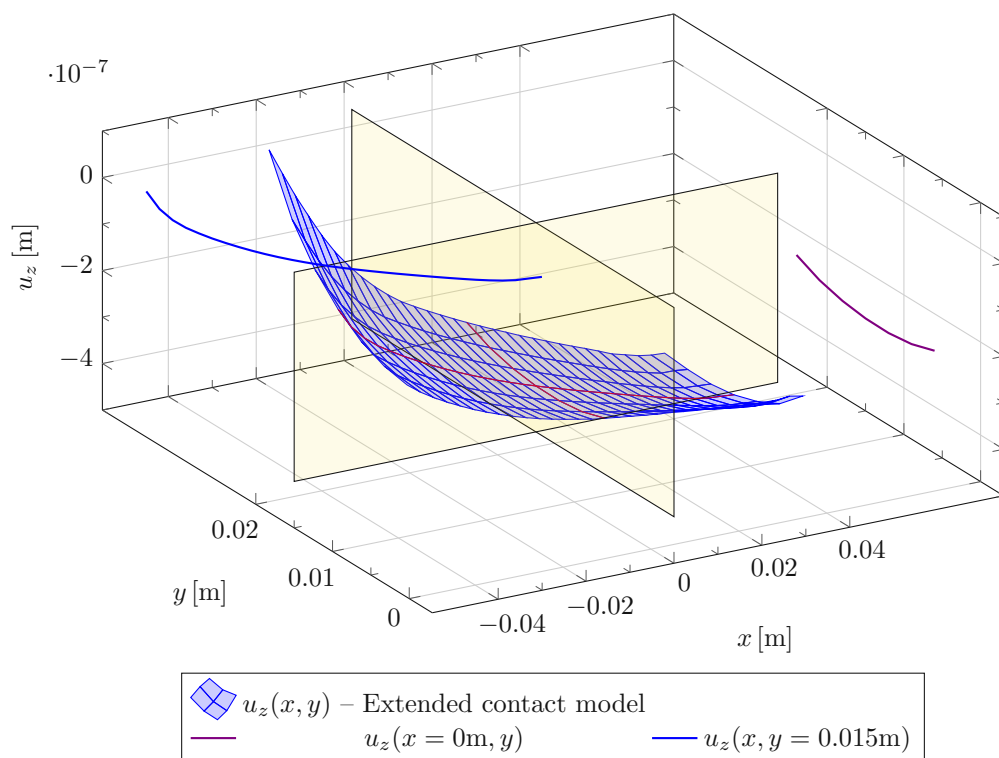


Figure D.3: Vertical displacement, result from the extended model for $\mu = 0.25$

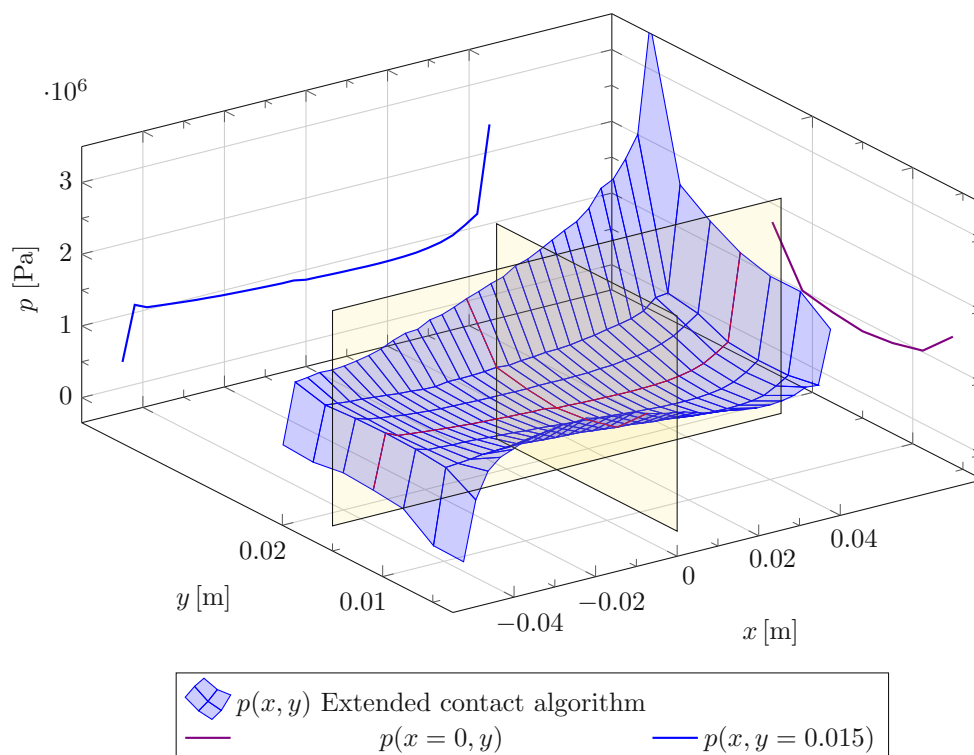


Figure D.4: Contact pressure distribution, result from the extended model for an arbitrary nonconstant magnetic loading

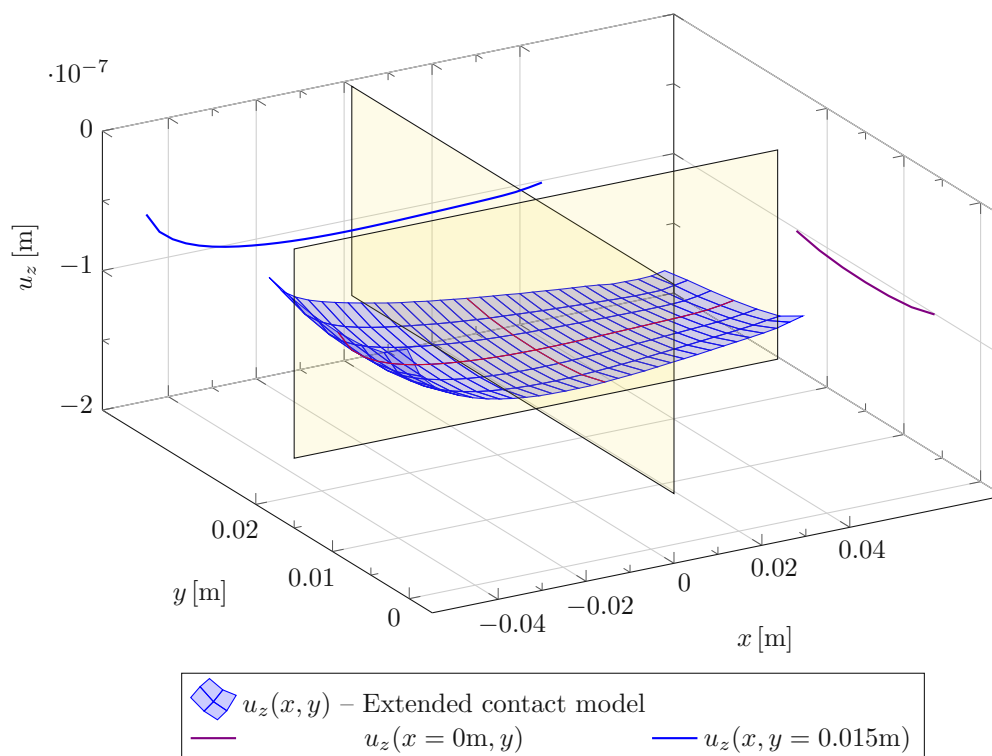


Figure D.5: Vertical displacement, result from the extended model for an arbitrary nonconstant magnetic loading