

10 mm

5 mm

Invitation

You are cordially invited
to the defense of
my PhD dissertation:

Fast Solvers for Concentrated Elastic Contact Problems

The defense will take place on:

Tuesday,
23rd June 2015,
at 15:00,
in the Aula of TU Delft
Mekelweg 5, Delft,
The Netherlands.

At 14:30 a short
presentation will be
given.

Fast Solvers for Concentrated Elastic Contact Problems

Jing Zhao

Fast Solvers for Concentrated Elastic Contact Problems

Jing Zhao

ISBN 978-94-6186-488-8



9 789461 864888 >

The reception will
follow the defense.

Jing Zhao

Fast Solvers for Concentrated Elastic Contact Problems

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K. C. A. M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
dinsdag 23 juni 2015 om 15:00 uur

door

JING ZHAO

Master of Science in Applied Mathematics,
Northwestern Polytechnical University, China,
geboren te Baoji, Shaanxi Province, China.

This dissertation has been approved by the

Promotor: Prof. dr. ir. C.W. Oosterlee

Copromotor: Dr. ir. E.A.H. Vollebregt

Composition of the doctoral committee:

Rector Magnificus,	voorzitter
Prof. dr. ir. C.W. Oosterlee,	Technische Universiteit Delft, promotor
Dr. ir. E.A.H. Vollebregt,	Technische Universiteit Delft and VORtech BV, copromotor

Independent members:

Prof. dr. ir. A.W. Heemink,	Technische Universiteit Delft
Prof. dr. ir. B. Koren,	Technische Universiteit Eindhoven
Prof. dr. ir. C.H. Venner,	Universiteit Twente
Prof. dr. P. Jimack,	University of Leeds, United Kingdom
Prof. dr. F.J. Gaspar,	Universidad Zaragoza, Spain
Prof. dr. ir. C. Vuik,	Technische Universiteit Delft, reservelid



Fast Solvers for Concentrated Elastic Contact Problems.

Dissertation at Delft University of Technology.

ISBN 978-94-6186-488-8

This research was supported by the China Scholarship Council (CSC).

Copyright © 2015 by Jing Zhao

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form of by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without the prior permission of the author.

Published by Gildeprint

Printed in the Netherlands

The cover is designed by Alex Kiricheck and Jing Zhao.

Acknowledgements

This dissertation concludes my PhD research in the Department of Applied Mathematics in Delft University of Technology, The Netherlands. I would like to attribute this success to a number of people for their kind help and support.

First of all, I would like to express my deepest gratitude to my promotor Prof. C.W. Oosterlee, for offering me this PhD position, guiding me with his professional experience in the research and supporting me to attend international conferences. It is him who led me to know how powerful the multigrid method can be. He always provides immediate help although he has a busy schedule. Moreover, he is concerned about every issue arising in my PhD research, and helps to guarantee efficient progress speed.

I would also like to give my sincere gratitude to my co-promotor Dr. E.A.H. Vollebregt, who brought me into the world of wheel-rail contact and inspired me with those excellent numerical solvers and smart ideas. As his first PhD student, I received his greatly patient and detailed help for any problem encountered.

Many thanks to both of my promotors again. I would never accomplish this PhD work without their professional guidance and warm encouragement. I am grateful to them for taking countless time to modify my papers, thesis and progress reports, which helps to improve my skills of writing and presentation. It is great pleasure to work with them in a form of our regular tripartite meeting. I learned a lot from their enthusiastic and strict attitude towards research and life.

Next, I would like to thank committee members for their time with reading and improving this thesis. Special thanks are due to Prof. Jimack for his detailed comments. Moreover, I am grateful to Guus Segal for his help with using finite element package SEPRAN, and to Dennis den Ouden for his help with the Dutch translation of my Propositions and Summary for this thesis. Regarding official affairs, many thanks to Franca Post, Wim van Horssen, Deborah Dongor and Cindy Bosman for their kind help. My appreciation also goes to Carl Schneider and Kees Lemmens for their technical support. Furthermore, I would like to thank Prof. Yufeng Nie and Prof. Shenggui Zhang for encouraging me to study abroad.

The accomplishment of my PhD work also attributes to the friendly and helpful atmosphere, provided by my colleagues in Numerical Analysis group. Many thanks to academic staff including Prof. Vuik, Domenico, Fred, Duncan, Jennifer, Matthias, Martin, Neil, Valia, Peter, Johan and Fons. I would also like to thank former and

current PhD students including Bowen, Paulien, Lech, Fei, Menel, Manuel, Daniël, Reinaldo, Martijn, Gabriela, Luis, Fahim, Joost, Guido, Lisa, Zaza, Pavel, Abdul, Fang, Bin, Elwin, Tijmen, Jok and Behrouz. Moreover, my thanks also go to Qian and Marjon in CWI.

My PhD life has been enriched by the valuable friendship with Peiyao Luo, Xin He, Thea Vuik, Rohit Gupta, Xiaozhou Li, Yue Qiu, Jiao Chen, Fei Xu, Yan Yan, Jie Li, Yang Lei, Rui Liu, Duowa Zhutian, Cong Li, Wenbo Wang, Sha Lu and Guangliang Fu. I would like to thank them for all the unforgettable time that we spent together. Moreover, many thanks to my friends and relatives in China for their constant mental support.

My faithful gratitude goes to my husband Alex. He makes me to believe that life is full of miracles and challenges, and love brings hope and confidence. I am so lucky to have his never-ending support and delightful accompany.

The last but the most important, I would like to give my soulful gratitude to my parents, for their endless and selfless love and support which makes me brave and optimistic in this colorful life.

Jing Zhao
Delft, May 2015

Summary

Fast Solvers for Concentrated Elastic Contact Problems

Jing Zhao

Rail transportation plays an important role in our everyday life, and there is fast development and modernization in the railway industry to meet the growing demand for swifter, safer and more comfortable trains. At the same time, the security of train operation and the maintenance of rails have to be considered. A lot of research on these issues has been carried out, among which the study of the contact between a train's wheel and the rail is particularly significant.

The contact problem considers two elastic bodies. When they are pressed together, a contact area is formed where the two body surfaces coincide with each other. Moreover, an elastic field of stress, strain and displacement arises in each body. These stresses consist of normal stress (pressure) and frictional stress (traction) acting in the tangential direction. When solving the so-called *normal contact problem*, we search for the contact area and the pressure on it. The *tangential contact problem* is studied when the two bodies are brought into relative motion. If the relative velocity of the two surfaces is small, a creeping motion may be observed which is largely caused by the elastic deformation at the contact region. In those parts of the contact area where the tangential stress is small, the surfaces of the two bodies stick to each other. Otherwise, local relative sliding may occur. The research question is to find the adhesion and slip areas, and the tangential tractions.

The solution methods for contact problems have been studied from the late nineteenth century, resulting in a variety of analytic and numerical approaches, w.r.t. their own specific applications. Motivated by the requirement of fast computation for involved applications such as the simulation of railway wheel-rail dynamics, we aim at developing fast numerical solvers for concentrated elastic contact problems in this thesis.

Our work focuses on the contact between bodies of linear homogeneous elastic material. Moreover, it is a concentrated contact, i.e. the contact area is small compared to the dimensions of the contacting bodies. The models in use

are provided by a variational formulation, which is based on a boundary element method (BEM). It gives rise to a convex optimization problem with linear or non-linear constraints. The corresponding Karush-Kuhn-Tucker conditions provide the governing equations and contact conditions, that are numerically solved.

The most time-consuming part attributes to solving a Fredholm integral of the first kind, resulting from the BEM. The corresponding Green's function expresses the relation between tractions and deformation, using a half-space approach. This integral yields linear systems with coefficient matrices that are dense, symmetric and positive definite. Moreover, they are Toeplitz in two-dimensional (2D) problems and block Toeplitz with Toeplitz blocks in three-dimensional (3D) problems. Fast computing techniques such as the fast Fourier transform (FFT) are explored.

We start our work by solving the normal contact problem in Chapter 2. It is modeled by a linear complementarity problem, for which a full multigrid method (FMG) is presented. This method combines a multigrid (MG) method, an active set strategy and a nested iteration technique. It is applied to a Hertzian smooth contact and a rough surface contact. The results show the efficiency and robustness of the FMG method.

Tangential contact is considered in Chapter 3 and Chapter 4. A 2D no-slip tangential problem is first studied in Chapter 3, where we mainly solve the surface integral. A fast MG method is proposed with an FFT smoother, where a Toeplitz preconditioner is constructed to approximate the inverse of the coefficient matrix. This smoother reduces many error components but enlarges some smooth error modes. Techniques such as subdomain deflation and row sum modification (RSM) are incorporated. Numerical experiments indicate rapid convergence and mesh-independence of MG with the FFT+RSM smoother. Moreover, FFT+RSM as a stand-alone solver also shows its efficiency. The complexity of these two methods is $\mathcal{O}(n \log(n))$, with n the number of unknowns.

We work on the 3D tangential contact in Chapter 4, where a nonlinear constrained optimization problem arises. A fast solver, called TangCG, is proposed. It combines an active set strategy and a nonlinear conjugate gradient method. The most pronounced component of this method is that it employs two types of variables in the adhesion and slip areas. Techniques including the FFT and diagonal preconditioning are also incorporated. The TangCG method is tested for Cattaneo shift problems, with different amounts of slip. It dramatically reduces the computational time, compared to the state-of-art ConvexGS method.

The numerical methods presented above are based on the influence coefficients (ICs) that give the relation between tractions and deformation. In Chapter 5, we investigate ICs by computing them numerically. Based on a concentrated contact setting, an elastic model is built for this purpose and a finite element method (FEM) is employed. Suggestions about the FEM meshing and element types are given, considering the accuracy and computational cost. The effects of employing the numerical ICs on contact solutions are examined. The work in this chapter provides a guidance for developing fast solvers for conformal contact

problems, which typically are governed by a larger and curved contact region.

With the research presented in present PhD thesis, and with the resulting improved numerical solution techniques, it becomes one step closer to incorporate detailed contact models in the numerical simulation of rail vehicle dynamics and in the simulation of rail and wheel wear and track deterioration.

Samenvatting (Dutch Summary)

Snelle oplostechnieken voor geconcentreerde elastische contactproblemen

Jing Zhao

Spoorvervoer speelt een belangrijke rol in ons dagelijks leven, en er is een snelle ontwikkeling en modernisering in de spoorwegsector om aan de groeiende vraag naar snellere, veiligere en comfortabelere treinen te voldoen. Tegelijkertijd moeten de veiligheid van de treinexploitatie en het onderhoud van de sporen worden beschouwd. Er is veel onderzoek betreffende deze kwesties uitgevoerd, waarbij de studie van het contact tussen een treinwiel en het spoor bijzonder belangrijk is.

In deze context, is het zogeheten contactprobleem gebaseerd op twee elastische lichamen. Wanneer ze worden samengedrukt, wordt een contactgebied gevormd waar de twee lichaamsoppervlakken met elkaar in contact komen. Bovendien ontstaat een elastisch gebied van spanning, rek en verplaatsing in elk lichaam. Deze spanningen bestaan uit normale spanning (druk) en wrijvingsstress (tractie) in de tangentiële richting. Bij het oplossen van het zogenaamde *wrijvingsloze contactprobleem*, berekenen wij het contactgebied en de druk hier op. Voor het *wrijvingscontactprobleem* wordt ook onderzocht wanneer en hoe de twee lichamen relatief tot elkaar bewegen. Wanneer de relatieve snelheid van de twee oppervlakken klein is, kan een kruipende beweging opgemerkt worden welke grotendeels veroorzaakt wordt door de elastische deformatie van het contactgebied. In die delen van het contactoppervlak waar de tangentiële spanning klein is, hechten de oppervlakken van de beide lichamen aan elkaar. In de andere delen kan plaatselijk een relatieve schuifbeweging optreden. De onderzoeksvraag is de hechtings- en schuifgebieden en de tangentiële spanningen te bepalen.

De oplossingsmethoden voor contactproblemen worden bestudeerd sinds het einde van de negentiende eeuw, resulterend in een verscheidenheid aan analytische en numerieke benaderingen m.b.t. specifieke toepassingen. Gemotiveerd door de eis van snelle berekeningen voor toepassingen, zoals voor de simulatie van de wiel-spoor dynamiek, richten wij ons in dit proefschrift op de ontwikkeling

van snelle numerieke oplosmethoden voor zogenaamde geconcentreerde elastische contactproblemen.

Ons werk focuseert op het contact tussen lichamen van lineair homogeen elastisch materiaal. Bovendien is het een geconcentreerd contact, dat wil zeggen dat het contactgebied klein is in vergelijking met de afmetingen van de in contact staande lichamen. De gebruikte modellen worden gegeven door een variationele formulering, die is gebaseerd op de rand-element methode (BEM). Hieruit ontstaat een convex optimalisatieprobleem met lineaire of niet-lineaire nevenvoorwaarden. De bijbehorende Karush-Kuhn-Tucker voorwaarden leveren de betreffende vergelijkingen en contactvoorwaarden, die numeriek dienen te worden opgelost.

Het meest tijdrovende deel van de berekening wordt veroorzaakt door het oplossen van een Fredholm integraal van de eerste soort volgend uit de BEM formulering. De corresponderende Greense functie drukt de relatie tussen tracties en vervorming uit, met behulp van een halfruimte benadering. Deze integraal levert lineaire systemen met coëfficiëntmatrices die vol, symmetrisch en positief-definitief zijn. Bovendien zijn ze Toeplitz voor tweedimensionale (2D) problemen en blok-Toeplitz met Toeplitz-blokken in driedimensionale (3D) problemen. Snelle berekeningstechnieken zoals de fast Fourier transformatie (FFT) worden onderzocht.

We beginnen ons werk in Hoofdstuk 2 met het oplossen van het wrijvingsloze contactprobleem. Het wordt gemodelleerd door een lineair complementariteit-probleem waarvoor een full multigrid methode (FMG) wordt gepresenteerd. Deze methode combineert een multigrid (MG) methode, een actieve-set strategie met een geneste iteratietechniek. Het wordt toegepast op een Hertziaans glad contactprobleem en ook op een ruw contactoppervlak. De resultaten tonen de efficiëntie en robuustheid van de FMG methode.

Wrijvingscontact wordt beschouwd in Hoofdstuk 3 en Hoofdstuk 4. Een 2D tangentieel probleem wordt eerst onderzocht in Hoofdstuk 3, waar we vooral een oppervlakte-integraal oplossen. Een snelle MG methode wordt gepresenteerd met een FFT smoother, waarbij een Toeplitz structuur wordt opgezet om de inverse van de coëfficiëntmatrix te benaderen. De resulterende smoother reduceert veel foutcomponenten van de oplossing, maar vergroot sommige gladde foutcomponenten. Technieken zoals subdomeindeflatie en rij-som modificatie (RSM) worden toegevoegd en vergeleken. Numerieke experimenten duiden op snelle convergentie en roosteronafhankelijkheid in convergentie van MG met de FFT+RSM smoother. Bovendien toont de FFT+RSM methode als losstaande oplostechniek ook efficiëntie. De complexiteit van deze beide methoden is $\mathcal{O}(n \log(n))$, met n het aantal onbekenden.

We werken vervolgens aan een 3D wrijvingscontactprobleem in Hoofdstuk 4, waarbij een optimalisatieprobleem met niet-lineaire nevenvoorwaarden ontstaat. Een nieuwe snelle oplostechniek, genaamd TangCG, wordt hier voorgesteld. Het combineert een actieve-set strategie en een niet-lineair geconjugeerde-gradiënten methode. Het meest interessante onderdeel van deze methode is dat het gebruik

maakt van twee typen variabelen in de hechtings- en schuifgebieden van het contactprobleem. Technieken zoals de FFT en diagonale preconditionering worden ook gebruikt. De TangCG methode wordt getest op Cattaneo schuifproblemen met verschillende mate van verschuiving. De nieuwe methode vermindert de rekentijd drastisch, in vergelijking met de state-of-the-art ConvexGS methode.

De numerieke rekentechnieken hierboven beschreven zijn gebaseerd op matrices met als matrixelementen de invloedscoëfficiënten (ICs) die de relatie tussen de krachten en vervorming aangeven. In Hoofdstuk 5 onderzoeken we ICs door deze numeriek te berekenen in plaats van de analytische coëfficiënten te gebruiken. Op basis van een geconcentreerd contact wordt een elastisch model gebouwd voor dit doel en een eindige-elementen methode (EEM) toegepast. Suggesties voor het EEM rekenrooster en element types worden gegeven, met als criteria de nauwkeurigheid en kosten van de berekening. De effecten van het gebruik van de numerieke ICs op contactoplossingen worden onderzocht. Het werk in dit hoofdstuk geeft een richtlijn voor de ontwikkeling van snelle oplosmethoden voor conforme contactproblemen, die doorgaans worden beschreven door een groter en gebogen contactoppervlak.

Met het gepresenteerde onderzoek in het huidige proefschrift, en met de resulterende verbeterde numerieke oplossingstechnieken, komen we een stap dichterbij het gebruik van gedetailleerde contactmodellen in de numerieke simulatie van spoor-treindynamica en in de simulatie van spoor- en wielslijtage.

Contents

Acknowledgements	iii
Summary	v
Samenvatting (Dutch Summary)	viii
1 Introduction	1
1.1 Contact Problems	1
1.2 History of Contact Problems	3
1.3 General Summary of Solution Strategies	4
1.4 Target and Scope of Present Work	5
1.5 Thesis Organization	6
2 A Full Multigrid Method for Elastic Normal Contact Problems	9
2.1 Introduction	9
2.2 Problem Formulation	11
2.2.1 Physics of the Normal Contact Problem	11
2.2.2 Contact Conditions	12
2.2.3 The Half-space Approach	12
2.2.4 Discretization	13
2.2.5 Generation of Rough Surface	15
2.3 Original Method: Active Set Algorithm	17
2.4 A Full Multigrid Method (FMG)	18
2.4.1 MG+Active Set Algorithm	18
2.4.2 Multigrid Components	19
2.4.3 The FMG Scheme	21
2.4.4 Analysis on Jacobi and Gauss-Seidel Smoothers	22
2.5 Numerical Experiment and Results	25
2.5.1 Test I: Hertzian Smooth Contact	25
2.5.2 Test II: Rough Surface Contact	29
2.6 Conclusion	33

3	Multigrid with FFT smoother for a 2D Frictional Contact Problem	35
3.1	Introduction	35
3.1.1	Physics	35
3.1.2	Solution Strategies	36
3.1.3	Solution Algorithms	37
3.1.4	Contents of This Chapter	38
3.2	Formulation of the 2D Frictional Contact Problem	38
3.2.1	Physics of the Frictional Contact Problem	39
3.2.2	Mathematical Model of the Frictional Contact	39
3.2.3	The Half-Space Approach	41
3.2.4	Discretization	41
3.2.5	The First Model Problem	42
3.3	FFT Smoother for Multigrid Method	43
3.3.1	Construction of the FFT Smoother	43
3.3.2	Discussion about Smoothing Analysis	45
3.3.3	Multigrid Method Based on FFT Smoother	47
3.4	Remedies for the FFT Smoother	48
3.4.1	Subdomain Deflation	48
3.4.2	Row Sum Modification	51
3.5	Numerical Results	53
3.5.1	Results by MG with the FFT+SD Smoother	54
3.5.2	Results by MG with the FFT+RSM Smoother	55
3.5.3	Results by the FFT+RSM Used as a Solver	56
3.5.4	Model Problem 3	56
3.6	Conclusion	57
4	A Fast Nonlinear Conjugate Gradient-based Method for 3D Concentrated Frictional Contact Problems	59
4.1	Introduction	59
4.2	Formulation of the 3D Frictional Contact Problem	61
4.2.1	Notations	61
4.2.2	Mathematical Model of the 3D Frictional Contact Problem	61
4.2.3	Discretization	62
4.2.4	Existing Algorithms	64
4.3	New Method: TangCG Algorithm	64
4.3.1	Main Components of the TangCG Algorithm	65
4.3.2	Illustration of the TangCG Algorithm	66
4.3.3	The Resulting Algorithm	71
4.3.4	2×2 Test	73
4.4	Numerical Results	74
4.4.1	Optimal K	75
4.4.2	Comparison between TangCG(3) with and without Preconditioning	77

4.4.3	Comparison with ConvexGS Method	77
4.5	Conclusion	79
5	Extending the Range of Applicability of the Contact Approach	81
5.1	Introduction	81
5.1.1	Strategies for Solving Concentrated Contact Problems . . .	81
5.1.2	Strategies for Solving Conformal Contact Problems	82
5.1.3	Content and Structure of This Chapter	83
5.2	An Elastic Model for Computing Influence Coefficients	84
5.2.1	Influence Coefficients on Half-Space	84
5.2.2	Model	86
5.3	Influence Coefficients A^{zz} by a Piecewise Constant Load	89
5.3.1	Mesh Strategies	89
5.3.2	A^{zz} on Hexahedron Mesh1	91
5.3.3	A^{zz} on Prismatic Mesh2	93
5.3.4	Discussion on Mesh Strategies	93
5.4	Error Propagation in a Cattaneo Shift Problem	94
5.4.1	The Normal Problem	94
5.4.2	The Tangential Problem	98
5.5	Conclusion	100
6	Conclusions and Outlook	103
6.1	Introduction	103
6.2	Conclusions	104
6.3	Outlook	107
	Curriculum Vitae	118
	List of Publications	120
	List of Presentations	122

Introduction

1.1 Contact Problems

Every day, people are traveling by train to other cities and abroad, for work or pleasure. At the same time, numerous trains are carrying a variety of goods to different places in the world. Rail transport has been one of the most convenient and efficient forms of transport. And there is still an increasing demand for trains to be faster, safer and more comfortable.

The security of train operation in the railway industry is crucial. In the history, there have been several serious train accidents where a train ran off the tracks. Such derailment causes casualties and property loss. One of the reasons for derailment is that wheels experience so large lateral forces that wheel flanges climb up and over the top of the rail. Another reason is that wheels, due to the large lateral forces, can push rails and wheels fall between rails [108].

Another important problem is the maintenance of rails. It aims at prolonging the rail operation life from both economic and safety point-of-view. The rails can be damaged in the form of fatigue crack, wear, and spalling. This leads to a change of the profile of the wheel and rail contact, and may threaten the performance and security. It becomes a serious issue especially for high speed and heavy haul trains. Resulting from this damage is overloading and excess frictional stress on rails, due to an intensive wheel-rail contact. This issue has been addressed in rolling contact fatigue (RCF) [29], fatigue life of machine elements [89], friction and wear [22, 77, 41]. Among them, the study of wheel-rail contact is particularly significant and inevitable.

Contact problems concern elastic contact between two bodies. When they are pressed together, the forces they obtain from each other result in elastic deformation. This yields a contact area where the surfaces of the two bodies coincide, and exert stresses on each other. These stresses are composed of normal stress (pressure), and frictional stress (traction) acting in the tangential direction. When and where the frictional stress is small, the two bodies stick to each other. However, local sliding occurs where the frictional stress is large enough.

This phenomenon can be illustrated by an example arising from a well-known sport, tug of war, in Fig. 1.1, where two teams of players are pulling the rope to their own sides. Consider the team on the right-hand side. These players can be regarded as one overall body, which is contacting to the ground due to gravity. The formed contact area consists of these players' feet region. This body receives a force of pulling from the other side, and as a result, it may move as a whole to the left. This can be dealt with in two ways. For instance, the two players closest to the right-hand side may straighten their arms and lean forward, the other two players may step with their feet. The former is like elastic deformation, and the latter shows local slip. This illustrates an important aspect of a contact phenomenon, i.e., that within the contact region that there may be local sliding in some area and sticking or adhesion in other places.

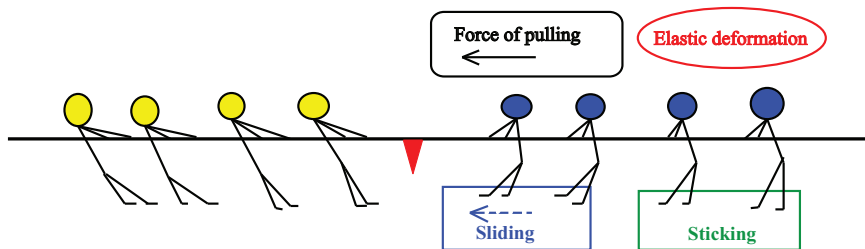


Figure 1.1: Illustration of the frictional contact.

Contact problems can be classified basically into two types. One is the so-called “normal contact problem”. It solves for the contact area and normal pressures that are perpendicular to the contacting surfaces. Two complementary conditions are required for the solutions. One of them states no penetration between the two bodies. This implies that the gap between the two surfaces is zero in the contact area but is positive outside. The other condition points out that the pressure is compressive (≥ 0) in the contact area and vanishes outside.

The second type of contact problems is the so-called “frictional” or “tangential contact problem”, in which we study the effect of friction on the contact region. A challenge is to find out the adhesion region where the two contacting bodies are sticking to each other, and the sliding area where local sliding occurs. Further we need to find the distribution of the frictional stress in these regions.

Studying frictional contact often starts with a known contact area and pressure distribution. A specific frictional law then determines the traction bound. The frictional stress should satisfy two conditions. In the adhesion area, the magnitude of the tractions should not exceed the traction bound, and there is no slip. In the slip area, the traction bound is reached, and the resulting slip points in the opposite direction of the tractions.

1.2 History of Contact Problems

Contact problems have been studied since 1882, when Hertz [36] contributed to the normal contact research. His theory considered the contact between smooth surfaces of quadratic profiles. With assumptions of linearity, absence of friction and a half-space approach, it provided solutions for the resulting contact area and pressures, regarding the applied normal force and the approach of the two surfaces.

Classical solutions for tangential contact problems originated from the independent work by Carter (1926) [12] and Fromm (1927) [27]. They solved two-dimensional (2D) rolling contact problems. The former considered the wheel and rail as a cylinder and a plane, respectively, and adopted a half-space approach. The solution indicated that in the case of small creepage, an adhesion zone could always be found near the leading edge and local sliding occurred near the trailing edge. The same result was found by the latter, who solved the contact between two elastic cylinders without the aid of a half-space approach. They both adopted Coulomb's friction law. Solutions for three-dimensional (3D) frictional contact were presented by Cattaneo (1938) [13] and Mindlin (1949) [64] independently. They considered a sphere pressed and tangentially shifted on a plane where partial slip on the contact area could occur. The corresponding distribution of tangential stress was provided.

Johnson was the first to solve 3D rolling contact problems. In 1958, he showed that the spin caused lateral contact forces in the rolling motion of an elastic sphere on a plane [43, 42]. In 1985, he published a book [44], which contained substantial information on solutions for different aspects of contact problems. Johnson has been regarded as one of the most important contributors in the recent fifty years.

Another key contributor is Kalker from Delft University of Technology. In 1967, he proposed the "linear theory" [46], which stated that creep forces linearly depended on creepage with the assumption of small creepages and zero slip. In 1973, he developed a "simplified theory" [47], which has been widely used in the railway industry, applying a basic approximate relation between tractions and displacements. This method was programmed in the FASTSIM algorithm [48]. Then, in 1979, he presented the full theory of 3D rolling concentrated contact [49], which is a numerical approach and related to the variational theory of Duvaut and Lions [20]. This has been incorporated into the software CONTACT [101], which is now still being developed by Vollebregt. In 1990, Kalker published the book [51] which summarized his work on 3D elastic rolling contact problems. We refer to Popov [74] for a modern introduction of contact problems, and to Knothe [54] for detailed history of wheel-rail contact.

The field of computational contact problems was promoted by the fast development of the finite element method (FEM). It dates back to 1933, when Signorini [83] posed a general equilibrium problem of a linearly elastic body resting on a rigid frictionless foundation due to its mass forces. For this problem,

Fichera (1964) [24] proposed the variational theory and provided the proof for the existence and uniqueness of solutions. The work by Duvaut and Lions (1972) [20] started to investigate the methods for these formulations of variation inequality, particularly solving contact with friction and large deformation. A great deal of relevant research has been done on the FEM and the corresponding numerical solutions, we refer to books by Kikuchi and Oden (1988) [53], Laursen (2002) [58] and Wriggers (2006) [107].

Other computational contact research is attributed to the boundary element method (BEM), e.g. [2, 62]. Fundamental work has been contributed by Kalker [51], in terms of the corresponding variational formulation and numerical algorithms. His theory and methods have been applied to many different aspects of contact problems, for instance, to the contact subject to varying tangential load [17, 68], the effects of non-Hertzian and non-steady states on rail corrugation growth at the contact area [94], and the dynamic contact effect [71].

1.3 General Summary of Solution Strategies

The FEM is particularly suited for contact problems in the cases of general contacting profiles, large deformation, non-homogeneous and nonlinear elastic materials. The formulation of variational inequalities leads to a minimization problem with inequality constraints. Hence, the FEM formulation often incorporates optimization approaches such as the penalty approach, the augmented Lagrangian technique, Lagrange multiplier method and their combinations [111, 58, 107]. The FEM is based on large numbers of elements covering whole contacting bodies. Sparse matrices are solved, but the size of the matrices can be very large. Therefore, this method can be computationally and memory intensive especially for large problems.

The BEM is applicable for concentrated contact problems with assumptions of small deformation and deformation gradients. The 3D boundary value problem is transformed to a 2D boundary integral equation, and then only solves at the discrete boundary. The corresponding variational formulation [51] represents a convex optimization problem, for which the equivalent Karush-Kuhn-Tucker (KKT) conditions provide both sufficient and necessary conditions for the existence of a unique solution [57].

The KKT conditions lead to a linear complementarity problem (LCP) for normal contact problems. Quadratic programming solvers, e.g. the extension of the simplex algorithm, the interior point method or the active set approach can be applied [65]. A frequently used strategy for these contact problems is the active set approach, due to its highly satisfactory performance and direct mechanical explanation. In each active set iteration, a linear system needs to be solved, and the solutions in turn are used to modify the active set. This strategy has been combined with iterative solvers such as the conjugate gradient method in Kalker's NORM algorithm [51] and with a multigrid method in [91].

In 3D frictional contact problems, the minimization problem is formed with

nonlinear constraints. Based on the KKT conditions, the active set strategy is again employed to treat constraints, which yields a nonlinear system to be solved in each active set iteration. Iterative solvers for the nonlinear system, e.g. Newton’s method, are applied. This forms the basis for Kalker’s TANG algorithm [51]. An alternative solution method for this kind of problems is the ConvexGS method [95], which is a block Gauss-Seidel approach.

Comparing with the FEM, the BEM discretizes only the boundary, which leads to smaller-sized problems with dense coefficient matrices. The FEM has given rise to significantly larger problems although they are represented by sparse matrices. Moreover, when focusing only on the contact area, with a BEM, more elements are placed in that region, and hence more detailed information can be found there. Regarding the algorithms, we encounter similar solution methods for the nonlinear equations, like Newton-based methods [107], or a nonlinear Gauss-Seidel method [45], in both FEM and BEM.

When the contacting bodies are of different materials, the tangential tractions and normal displacements interact with each other. In that case, the normal and tangential problems can not be easily separated. A straightforward way to process is to solve a fully coupled formulation [106]. Another popular approach is via the so-called “Panagiotopoulos process” [69, 28, 51]. In each iteration, the normal problem is solved first followed by the tangential problem. When contacting bodies are of the same material, i.e. a so-called quasi-identity case, these two problems can be decoupled, and one iteration is sufficient [51].

1.4 Target and Scope of Present Work

In this thesis we aim at developing fast numerical solvers for concentrated elastic contact problems. This is motivated by the requirement of high computational speed for involved applications, such as for the simulation of railway wheel-rail dynamics.

The present work deals with concentrated contact between linear homogeneous elastic materials. We work on both normal and tangential contact problems. The models for these problems are provided by Kalker’s variational formulation [51], which is based on a BEM formulation, and the corresponding KKT conditions are solved numerically. We start to develop efficient algorithms, on the basis of existing algorithms like NORM for normal problems, and TANG and ConvexGS for tangential problems. Moreover, the fast Fourier transform (FFT) technique is also taken into consideration for accelerating matrix-vector products, since it has been successfully applied in contact problems [84].

Normal contact problems are solved when contact surfaces are smooth and rough. The tangential contact is investigated by first solving a 2D model and then a 3D problem. Moreover, we also investigate numerical influence coefficients and their effect on contact solutions, as a preparation for developing fast solvers for conformal contact (which has a curved and relatively large contact area).

1.5 Thesis Organization

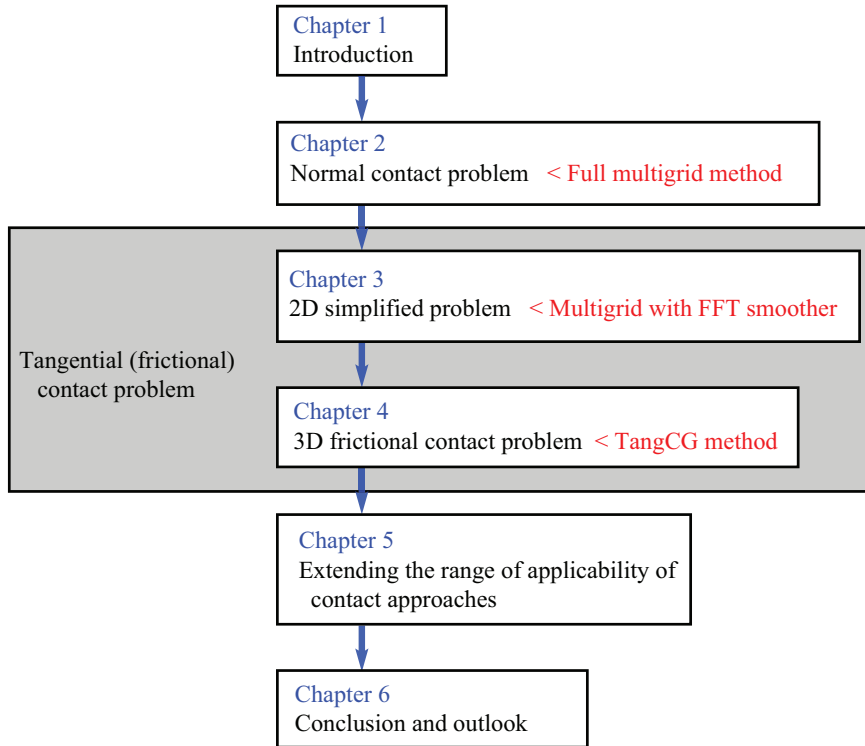


Figure 1.2: Organization of the present thesis

The organization of this thesis is generally shown in Figure 1.2. Six chapters are included, with the first one an introduction.

- In **Chapter 2**, we solve normal contact problems, where a linear complementary problem arises. A full multigrid (FMG) method is proposed, which combines a multigrid (MG) method, an active set strategy and a nested iteration technique. This method is applied to Hertzian smooth contact and rough surface contact problems. The contents in this chapter are published in [112].
- We present the work on tangential contact problems from **Chapter 3** on. Due to the complexity of such problems, in this chapter we consider a simplified 2D problem, which leads to a Fredholm integral equation of the first kind. A fast multigrid solver is proposed with an “FFT smoother”, based on the Toeplitz structure of the coefficient matrix resulting from the BEM discretization. The combination with techniques of deflation and row sum modification is also investigated. The contents of this chapter have been published in [113].
- In **Chapter 4** we consider 3D tangential contact problems, for which a nonlinear constrained optimization problem arises. A fast solver named

“TangCG” is proposed for the corresponding KKT conditions. It incorporates an active set strategy and a nonlinear conjugate gradient method. This method employs two different sets of variables in the adhesion and slip areas [98]. A diagonal preconditioner and the FFT technique are used. The contents in this chapter are published in [115]

- The concentrated contact problems solved in the previous chapters are based on the half-space formulation, where analytic influence coefficients in the traction-deformation relation are known. In **Chapter 5** we extend the range of applicability of the contact approach by computing the influence coefficients numerically. For this purpose, an elastic model is built and solved the FEM. We recommend proper strategies of FEM meshing and element types, considering accuracy and computational cost. The effects of computed influence coefficients to contact solutions are examined. The contents of this chapter are included in [114], as a reference for studying fast solvers for conformal contact problems.
- **Chapter 6** concludes the work in this thesis with main contributions, and presents possible topics for future work.

Chapter 2

A Full Multigrid Method for Elastic Normal Contact Problems

In this chapter we consider efficient solution of the normal contact problem in a frictionless case, where a linear complementarity problem (LCP) arises. A full multigrid (FMG) technique is presented, that combines a multigrid method, an active set algorithm and a nested iteration technique. The governing system in this LCP is derived from a Fredholm integral of the first kind, and its coefficient matrix is dense, symmetric and positive definite. One multigrid cycle is applied to solve this system approximately in each active set iteration. Moreover, this multigrid solver incorporates a special strategy to handle the complementarity conditions, including restricting both the defect and the contact area (active set) to the coarse grid, and setting all quantities outside contact to zero. The smoother is chosen by some analysis based on the eigenvectors of the iteration matrix. This method is applied to a Hertzian smooth contact and a rough surface contact problem.¹

2.1 Introduction

Normal contact problems are governed by the elastic deformations by forces perpendicular to the contact surfaces. We assume that friction is absent. When two elastic bodies are pressed together, a contact area is formed due to their deformations. The challenge is to find this contact region and the pressure (normal tractions) on the interface. Two complementary conditions are required for the solutions. First of all, these two bodies do not penetrate into each other. This indicates that the gap between the two bodies is zero in the contact region but

¹The contents of this chapter have been published in paper [112]: J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee. A full multigrid method for linear complementarity problems arising from elastic normal contact problems. *Mathematical Modelling and Analysis*, 19(2):216-240, 2014. This work received the *Honorable Mention* in the Student Paper Competition at the 12th Copper Mountain Conference, Colorado, the United States, 2012.

positive outside. Secondly, the pressure is compressive (≥ 0) in the contact area and vanishes outside.

Analytic solutions for normal contact problems can be found for simple geometries, such as the Hertz theory [36, 44, 74], where the surfaces of two contacting bodies are twice continuously differentiable, and the radii of curvature are constant. Moreover, solutions for the contact between a rigid cylindrical indenter punch or a cone and an elastic half-space are also available [74].

Numerical solvers in the framework of the boundary element method (BEM) include Kalker's variational method [51], which uses Green's functions for the elastic half-space. Particularly, it applies an active set algorithm, called "NORM", to search for the contact area. In each active set iteration, the governing system is solved in the current contact area, which is then modified by the resulting solution. The iterations terminate when all conditions are satisfied. Subsequent to [9], a multigrid strategy for contact problems was developed by Venner and Lubrecht [91], where the complementarity conditions were handled in a distributive smoother and a nonlinear full approximation scheme (FAS), with a multi-level multi-integration (MLMI) technique for matrix-vector products. Another method which combines the MLMI and the conjugate gradient (CG) algorithm for rough contact problems is proposed in [73]. The fast Fourier transform (FFT) technique is also employed, see [84]. A comparison of different methods can be found in [78].

The variational formulation [51] leads to a linear complementarity problem (LCP) with the use of Karush-Kuhn-Tucker (KKT) conditions. For this LCP, one can apply quadratic programming solvers such as the extension of the simplex algorithm, the interior point method, or the active set algorithm [65]. In contact problems, the active set strategy is frequently used, due to its efficient performance and easy mechanical explanation [51]. It has been combined with the CG method [101] and with a multigrid method [91].

The most time-consuming part of the solution procedure for the normal contact problem is the surface integral, which is a Fredholm integral of the first kind in the prescribed contact area. The discretization of such integral yields a linear system with the coefficient matrix being dense, symmetric and positive definite (SPD). Direct methods can be used but not for large problems. A popular iterative method is preconditioned CG [15, 30, 38, 85]. Multigrid is also studied for integral equations of the first kind, see, e.g. [10, 104, 80, 92].

The current state of the art solver for the concentrated normal contact problem may be the multigrid method in [91]. However, it is quite involved and nontrivial. This motivated our research into multigrid for this problem. Moreover, the solver should be able to handle the constraints in the corresponding LCP.

The outline of this chapter is as follows. In Section 2.2, the normal contact problem is introduced and the LCP is formulated. The approach proposed for generating Gaussian rough surfaces with a given standard deviation, is also described. Section 2.3 presents the original active set algorithm "NORM", by

Kalker [51]. Section 2.4 describes the structure of the full multigrid (FMG) method and each component of the multigrid method, particularly the restriction and interpolation to deal with the constraints in the LCP. Two experiments, a Hertzian smooth contact and a rough surface contact, are discussed and solved numerically in Section 2.5, after which conclusions are summarized.

2.2 Problem Formulation

This section gives a general description of the normal contact problem, and builds the mathematical model, resulting in the LCP which we will solve using a multigrid method. Furthermore, the approach for generating a rough surface is described at the end of this section.

2.2.1 Physics of the Normal Contact Problem

When the surfaces of two bodies roll over each other, a contact area occurs between them, carrying normal and tangential tractions. An elastic field of displacements, strains, and stresses in the bodies appears with these tractions, all of which result in deformations [51]. The tractions and the contact area are the main quantities we solve for. In the *normal contact problem*, the forces of the contacting bodies are perpendicular to the surface [74] and we only consider the normal tractions (pressures) here.

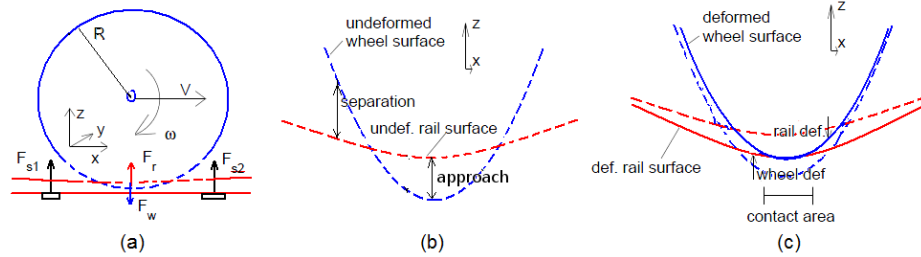


Figure 2.1: *Geometry of contact problems for railway applications [102]: (a) the overall geometry, (b) the undeformed state, (c) the deformed state.*

Fig. 2.1(a) shows the general geometry of the contacting phenomenon in railway applications. It displays a wheel rolling on the rail with a forward velocity V and an angular velocity ω . The rail is bent, caused by the forces from the wheel, F_w , and from the sleepers, F_{s1} , F_{s2} . The wheel receives the force from the rail, F_r , and it approaches to the rail with some distance. We stretch this graph vertically in order to have a detailed look at this phenomenon. Fig. 2.1(b) shows the “undeformed state” with the profiles of the wheel and rail, where the distance that the wheel is shifted down is the maximal penetration and called “approach”. Fig. 2.1(c) presents the “deformed state”, where the surfaces of wheel and rail coincide due to the deformation. The distance between their surfaces is positive outside the contact area without any tractions there.

2.2.2 Contact Conditions

First of all, we identify the geometry of two elastic bodies in undeformed state. A coordinate system $Oxyz$ is used to specify the position (x, y, z) of each particle of the bodies. This coordinate is assumed to move with the contact region, with z pointing normally to the other body and x, y directing tangentially.

Deformation occurs when the two bodies are brought into contact. An elastic field including stress $\sigma^{(i)}$, strain $\epsilon^{(i)}$ and displacements $\mathbf{u}^{(i)}$ arises in bodies $i = 1, 2$ and on their surfaces. The formulation of contact problems particularly focuses on the surface quantities: surface displacements $\mathbf{u}^{(i)}$, and surface tractions $\mathbf{p}^{(i)}$ which is defined by $\mathbf{p}^{(i)} = \sigma^{(i)} \cdot \mathbf{n}^{(i)}$ with $\mathbf{n}^{(i)}$ the outer normal of body i . The deformation at position (x, y, z) is defined by the displacement difference, i.e., $\mathbf{u}(x, y, z) = \mathbf{u}^{(1)}(x, y, z) - \mathbf{u}^{(2)}(x, y, z)$. Moreover, we solve for only tractions $\mathbf{p} := \mathbf{p}^{(1)}$, since the relation $\mathbf{p}^{(1)} = -\mathbf{p}^{(2)}$ is valid for all particles on the surfaces.

Define $\mathbf{u} := [u_x, u_y, u_z]^T$ and traction $\mathbf{p} = [p_x, p_y, p_z]^T$ by components in the x -, y - and z -directions. Since we look at the contact area which is in the plane $z = 0$, we use coordinate system (x, y) instead of (x, y, z) from now on in the chapter.

Let C denote the contact area and E the exterior area. Then, the *contact conditions* for the normal contact problem are:

$$e(x, y) = 0, p_z(x, y) \geq 0, \quad \text{for each position } (x, y) \in C, \quad (2.1)$$

$$e(x, y) > 0, p_z(x, y) = 0, \quad \text{for each position } (x, y) \in E, \quad (2.2)$$

where $p_z(x, y)$ is the pressure (normal traction) at position (x, y) , and $e(x, y)$ is the deformed distance (i.e. gap) between the wheel and rail. Let $h(x, y)$ denote the undeformed distance, then the deformed distance, $e(x, y)$, is given by:

$$e(x, y) = h(x, y) + u_z(x, y), \quad (2.3)$$

where $u_z(x, y)$ is the deformation in normal direction. It is assumed that the bodies can not interpenetrate, that the pressure occurs only where the bodies are in contact and that adhesion effects (attraction) are negligible.

2.2.3 The Half-space Approach

The method to calculate the deformation u_z is based on four simplifying assumptions. First of all, the contact area is very small compared to the dimensions of the two contacting bodies, and hence this area is considered to be *flat*. The second is that the contacting bodies are assumed to be made of homogeneous linear elastic material. Then we assume small displacements and displacement gradients. The last assumption is to ignore inertial effects in the motion.

These assumptions allow the use of the *half-space approach* [51], which approximates the elastic field of two contacting bodies considering each body as a *semi-infinite elastic solid* bounded by a plane surface. Based on the classic solutions by Boussinesq and Cerruti (see Johnson [44]) and considering 3D normal

contact problems, we give the normal traction-deformation relation, as follows:

$$u_z(x, y) = \iint_C A^{zz}(x, y, \xi, \eta) p_z(\xi, \eta) d\xi d\eta. \quad (2.4)$$

Here the superscript z denotes the normal direction, u_z and p_z are the normal deformation and pressure, respectively. $A^{zz}(x, y, \xi, \eta)$ is the influence function for normal deformation at the surface position (x, y) , due to the contribution of a unit pressure at surface position (ξ, η) . This influence function is calculated by:

$$A^{zz}(x, y, \xi, \eta) = \frac{1 - \nu}{\pi G} \cdot \frac{1}{\rho}, \quad (2.5)$$

where ν is the Poisson ratio, ρ is the distance between points (x, y) and (ξ, η) , i.e. $\rho = [(x - \xi)^2 + (y - \eta)^2]^{\frac{1}{2}}$. G is the combined shear modulus of the two contacting bodies, obtained by $\frac{2}{G} = \frac{1}{G_1} + \frac{1}{G_2}$ with G_1 and G_2 being the shear modulus of each body. For two bodies of the same material, we have $G = G_1 = G_2$; in the case of one rigid body, it follows that $G_1 = \infty$, so that $1/G_1 = 0$, and $G = 2G_2$.

Eq. (2.5) indicates that the value of $A^{zz}(x, y, \xi, \eta)$ is determined by the relative distance between points (x, y) and (ξ, η) , rather than their own locations. This means:

$$A^{zz}(x, y, \xi, \eta) = A^{zz}(x - \xi, y - \eta). \quad (2.6)$$

Such property will yields special structure of the corresponding coefficient matrix after discretization in the following.

2.2.4 Discretization

Solving contact problems often starts with the discretization of the 2D potential contact area, which is chosen such that it contains the true contact area. The potential contact area is discretized by $n_x \times n_y$ rectangular elements of size $\delta x \times \delta y$. Each element is numbered by (ix, iy) or (jx, jy) , where $1 \leq ix, jx \leq n_x$ and $1 \leq iy, jy \leq n_y$. *Since all quantities in this chapter refer to the normal direction, as a reminder, subscript z is dropped from quantities u_z, p_z , and superscript zz is dropped from influence function A^{zz} .*

The surface pressure p in the integral (2.4) is approximated by a piecewise constant function, where the pressure at each element center is used. This means that the integral (2.4) is evaluated by a 2D ‘‘mid-point rule’’. Due to boundary effects, the discrete error of the pressure reduces to $\mathcal{O}(h)$ [51]. The discretization of integral (2.4) yields:

$$u_{(ix, iy)} = \sum_{jy=1}^{n_y} \sum_{jx=1}^{n_x} A_{(ix, iy)(jx, jy)} p_{(jx, jy)}, \quad (2.7)$$

where subscripts (ix, iy) and (jx, jy) denote the element. The influence coefficient $A_{(ix, iy)(jx, jy)}$ is calculated by integrating (2.5) for a unit pressure in

a single element (jx, jy) w.r.t. an observation point (x, y) at the center of element (ix, iy) . It has the property that $A_{(ix,iy)(jx,jy)} = A_{(kx,ky)(lx,ly)}$ when $jx - ix = lx - kx, jy - iy = ly - ky$, i.e. the coefficients are identical for pairs of elements that have the same relative distance, as stated by Eq. (2.6). Fig. 2.2 shows an example of the influence coefficients scaled by G (for Test I in Section 2.5).

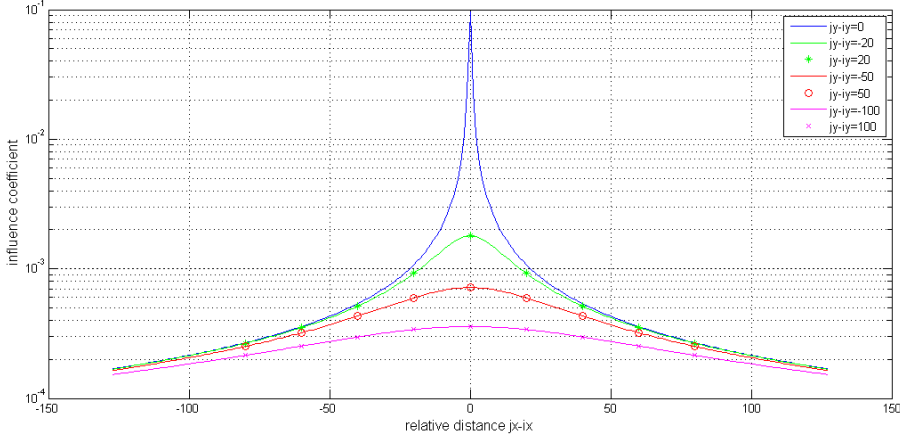


Figure 2.2: Influence coefficient $A_{(ix,iy)(jx,jy)}$ (scaled by G).

We denote the total number of elements by $n := n_x \cdot n_y$. If all n elements of the potential contact area are in contact, equation (2.7) leads to:

$$\mathbf{u} = A\mathbf{p}, \quad \mathbf{p}, \mathbf{u} \in \mathbb{R}^n, A \in \mathbb{R}^{n \times n}, \quad (2.8)$$

where the influence coefficient matrix, A , is a dense, symmetric and BTTB matrix, i.e. block Toeplitz matrix with Toeplitz blocks.² Combining equations (2.3) and (2.8), and considering the discretization of the contact conditions (2.1) and (2.2), we obtain a linear complementarity problem $\text{LCP}(\mathbf{h}, A)$, which aims to find the pressure, \mathbf{p} , and the contact area, C , satisfying

$$\mathbf{e} = \mathbf{h} + A\mathbf{p}, \quad (2.9)$$

$$e_I = 0, \quad p_I \geq 0, \quad \text{for element } I \in C, \quad (2.10)$$

$$e_I > 0, \quad p_I = 0, \quad \text{for element } I \in E, \quad (2.11)$$

where I is the element index with $I = (iy - 1)n_x + ix$, $1 \leq I \leq n$, and E is the exterior area. Note that the constraints on the deformed distance \mathbf{e} are the primary constraints: $e_I = 0$ defines the active set, i.e. the contact area, and $e_I > 0$ are inactive constraints, i.e. the exterior area. This LCP represents a mathematical model for the normal contact problem.

Noticing the nice regular BTTB structure of the coefficient matrix A in (2.8), one may think that fast Toeplitz solvers or solvers based on the FFT can be used.

²A Toeplitz matrix is a matrix with each descending diagonal from left to right being constant.

This is however generally not directly true because of the additional constraints. We do not allow any solution p , but require $p = 0$ in the exterior area E . Setting $p_I = 0$ for $I \in E$ removes the corresponding rows and columns from matrix A and destroys the BTTB structure. Solvers for a BTTB matrix are not considered in this chapter.³ According to condition (2.10), we have $\mathbf{e} = \mathbf{h} + A\mathbf{p} = \mathbf{0}$ in the contact area C , and thus:

$$A\mathbf{p} = -\mathbf{h}, \quad \text{in } C, \quad (2.12)$$

which is the main system to be solved. We will discuss the original method and the FMG method for solving the LCP in the sections to follow.

2.2.5 Generation of Rough Surface

Contact surfaces are not smooth for most cases in real life. Hence, we also study normal contact problems, where rough surfaces are involved. Our interest in rough surfaces stems from railway applications [11, 52]. First of all, the influence of roughness on the creep force curve is desired. Secondly, the real stresses between railway wheels and rails are needed for assessing wear and material damage phenomena. In the numerical experiments, we will focus on frictionless rough elastic contact problems that have been studied by Greenwood, Williamson [31] and Johnson [44].

A rough surface has to be generated when studying rough surface contact. It is regarded here as a random process, characterized by a height distribution and an auto-correlation function (ACF), which defines the correlation between two points at the surface. Patir [70] defined a method to generate a random surface whose height is Gaussian distributed: a linear transformation was made on a random matrix to obtain the surface height where the given ACF was used to determine the amplitudes for each random component. Hu and Tonder [37] used finite impulse response (FIR) filters for the generation of rough surfaces. They regarded this procedure as input Gaussian distributed signals passing through a FIR filter. Their method was considered not highly satisfactory in [109], in which another approach based on the FFT technique for the generation was proposed. Besides Gaussian rough surfaces, the generation of non-Gaussian rough surfaces is also studied, for example by Patir [70], Wu [110], amongst others.

We would like to discuss a method, based on [52], to generate Gaussian rough surfaces, where the standard deviation of the surface heights, $\hat{\sigma}$, is given.

The formula in [52] for the surface height $z(x, y)$ on a rectangle $[0, l_x] \times [0, l_y]$ is a randomized version of the two-dimensional wavy surface given by Johnson [44], which we write as:

$$z(x, y) = \sum_{m=1}^M \sum_{n=1}^N a_{mn} \cdot \cos\left(\frac{2\pi mx}{l_x} + \varphi_{mn}\right) \cdot \cos\left(\frac{2\pi ny}{l_y} + \theta_{mn}\right), \quad (2.13)$$

³An algorithm that does exploit the structure of the BTTB matrix using FFT's is presented in [100]. The combination of this with MG method will be investigated in Chapter 3.

where each frequency pair (m, n) has phases φ_{mn} and θ_{mn} , that are uniformly distributed on $[0, 2\pi]$, amplitude a_{mn} is uniformly distributed on $[0, \bar{a}_{mn}]$ and, here, $\bar{a}_{mn} := \bar{a} \cdot \gamma^{m+n-2}$ with \bar{a} a reference amplitude. The decay factor γ satisfies $0 < \gamma \leq 1$. The choices for M , N and γ in this model are flexible and when $M = N = 1$ this results in a wavy surface.

The expression for the auto-correlation function $R(x, y)$ of the surface height in [52] can be written as:

$$R(x, y) = \frac{1}{4} \sum_{m=1}^M \sum_{n=1}^N a_{mn}^2 \cos\left(\frac{2\pi mx}{l_x}\right) \cos\left(\frac{2\pi ny}{l_y}\right). \quad (2.14)$$

Then, $R(0, 0)$ is equal to $\hat{\sigma}^2$, i.e.:

$$R(0, 0) = \hat{\sigma}^2 = \frac{1}{4} \sum_{m=1}^M \sum_{n=1}^N a_{mn}^2. \quad (2.15)$$

We would like to calculate the surface height z , when $\hat{\sigma}$ is given. A random number generator is used to obtain φ_{mn} and θ_{mn} for each frequency pair (m, n) . The determination of the amplitude a_{mn} requires the value of the reference amplitude \bar{a} , which is chosen to be 1, and then the true amplitudes are scaled in accordance with the desired standard deviation $\hat{\sigma}$. This can be done by the following steps:

1. Obtain random coefficients \tilde{a}_{mn} uniformly distributed on $[0, \gamma^{m+n-2}]$ by a random number generator.
2. Calculate the auto-correlation, which is denoted by $\tilde{R}(0, 0)$, using \tilde{a}_{mn} with equation (2.15).
3. Scale the coefficients \tilde{a}_{mn} by a factor $\frac{\hat{\sigma}}{\sqrt{\tilde{R}(0, 0)}}$, i.e.

$$a_{mn} = \tilde{a}_{mn} \cdot \frac{\hat{\sigma}}{\sqrt{\tilde{R}(0, 0)}} \quad (2.16)$$

After the amplitudes a_{mn} are obtained, formula (2.13) will give the surface heights at the positions (x, y) .

To test this rough surface generation technique, we generate rough surfaces on $[-6, 6] \times [-5, 5]$ mm^2 with $\hat{\sigma} = 0.1$. A 64^2 -grid is used for the discretization and we choose $M = 12$, $N = 10$ in (2.13). Fig. 2.3 displays the resulting rough surfaces with $\gamma = 0.5$, 0.8 and 0.95 in (a) (b) and (c), respectively. We find that the surface heights of these three rough surfaces having zero mean and the desired standard deviation being $\hat{\sigma} = 0.1$. In Fig. 2.3(d), a normal probability plot of the surface heights of Fig. 2.3(b) indicates that the surface heights are indeed Gaussian distributed (since the values are very close to the red straight

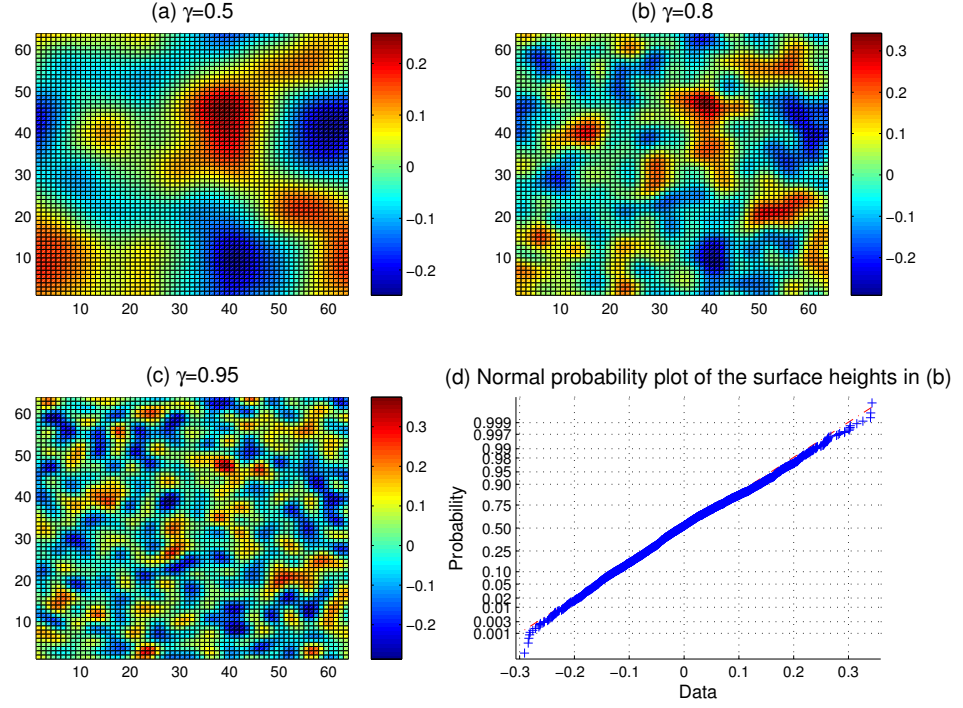


Figure 2.3: *Rough surfaces on $[-6, 6] \times [-5, 5] \text{ mm}^2$ with $\hat{\sigma} = 0.1$: (a) $\gamma = 0.5$; (b) $\gamma = 0.8$; (c) $\gamma = 0.95$; (d) The normal plot of the surface heights in (b).*

line which represents the Gaussian distribution). We also generated surfaces for different values for M and N , and found that large values of M and N result in better Gaussian distributed surface heights. We can always achieve the desired standard deviation $\hat{\sigma}$, independent of the values of M and N .

Comparing the three different rough surfaces in Fig. 2.3, we see that as γ increases, more peaks will be found on the corresponding rough surface. This implies that increasing the value of γ leads to a larger *asperity density*, as is introduced in [44].

2.3 Original Method: Active Set Algorithm

The active set algorithm is well-known for optimization problems, such as the LCP above. The main idea is to partition the computational domain into two groups, i.e. *active and inactive sets*. Those parts that satisfy the constraints are put in the active set and the others are in the inactive set. Only the active set is considered when solving the equality problem. One active set iteration has two steps, i.e. first determine a current active set as the working set, and then solve the equality problem in this active set. These two steps are repeated until all constraints are satisfied [65].

The convergence rate of the active set algorithm depends on the concrete problems to be solved. In contact problems, generally 5 to 15 iterations are required [100]. The contact area is part of the potential contact region and regarded as the active set, and the exterior area is the inactive set. The original algorithm “NORM” by Kalker [51] starts with an initial contact area, in which the undeformed distance is less or equal to zero. Initially, linear system (2.12) was solved in the contact area by Gauss-elimination. Later, instead of this direct solution method, CG has been applied [101]. Results are used to update the contact area, according to conditions (2.10) and (2.11). The process is repeated until the contact area converges.

We have to solve a linear system (2.12) in each active set iteration. This system is large and has a dense system matrix. Several active set iterations are needed before the correct contact area is obtained, which means that we have to solve many such systems.

2.4 A Full Multigrid Method (FMG)

This section presents the FMG method for the LCP (2.9)-(2.11). We start with an “MG+active set” algorithm, which inserts a multigrid solver into an active set iteration. Each component of the multigrid solver is detailed here. The MG+active set algorithm is incorporated in a nested iteration technique, resulting in the FMG method.

2.4.1 MG+Active Set Algorithm

When examining the contact conditions, there is no direct need for exact values of deformed distance \mathbf{e} or pressures \mathbf{p} , since the conditions only depend on whether values are positive, negative or zero.⁴ Therefore, in each active set iteration, we can solve system (2.12) *approximately*, by only one multigrid cycle. This yields an *MG+active set algorithm*, which is described as follows:

1. Start with some grid discretization for the potential contact area. Check undeformed distance \mathbf{h} : put element I satisfying $h_I \leq 0$ in the initial contact area C^0 .
2. Set the pressures in the exterior area E^k to zero, and define the system (2.12) in the current contact area C^k .
3. Apply one multigrid cycle to this system, resulting in the approximate pressures \mathbf{p} , and compute the deformation $\mathbf{u} = A\mathbf{p}$.
4. Check the contact conditions:
 - (4.a) Consider each element I in the contact area C^k : if pressure $p_I < 0$ then put element I into the exterior area, and set $p_I = 0$. This leads

⁴For a conjecture on the convergence of “NORM”, see [99].

to the intermediate contact area \overline{C}^{k+1} and exterior area \overline{E}^{k+1} .

(4.b) Consider each element I in the exterior area \overline{E}^{k+1} : put element I into the contact area C^{k+1} with its deformed distance $e_I = h_I + u_I \leq 0$. Now, a new contact area C^{k+1} , and exterior area E^{k+1} , have been obtained.

5. If this updated contact area C^{k+1} is different from the previous area C^k , then go to step 2. If not, go to step 6.
6. If the solution is not of the required accuracy, then go to step 2. Otherwise, stop since we have found the converged contact area and the pressures on it.

Remark 2.1. *To check whether a solution reaches the required accuracy in Step 6 is equivalent to checking whether a stopping criterion for Eq. (2.12) is satisfied. An example is criterion (2.18) in Section 2.5.*

The multigrid method plays an important role in this algorithm. We will give more details about its components in the following subsection.

2.4.2 Multigrid Components

The multigrid method is one of the most efficient numerical solvers, especially for systems of equations arising from elliptic PDEs. But here we will use the multigrid algorithm to deal with system (2.12), which arises from an integral equation and is governed by a dense coefficient matrix, in an active set iteration. Moreover, the complementarity conditions should be taken into consideration.

A two-grid algorithm is given below. (It is easy to be extended to a multigrid algorithm.) The subscripts h and H are the mesh sizes, denoting the fine and coarse grid, respectively.

1. *Pre- and post- smoothing:* We use the damped Gauss-Seidel method as the smoother. Notice that we only deal with the pressures within the contact area C_h^k , because the pressures in the exterior area E_h^k are fixed to zero.
2. *Calculate defects:* The defects in the contact area, C_h^k , are computed, and the defects in the exterior area E_h^k are set to zero. This gives us the fine grid defects, \mathbf{d}_h .
3. *Restrict* the defects, \mathbf{d}_h , and the contact area, C_h^k , to the coarse grid, resulting in coarse grid defects, \mathbf{d}_H , and a coarse contact area, C_H^k , respectively.
4. *On the coarse grid:* Apply an exact solver, e.g. a Gauss elimination method, for the defect equation, $A_H \overline{\mathbf{v}}_H = \mathbf{d}_H$, defined by the coarse contact area, C_H^k . The coarse grid correction, \mathbf{v}_H , is set as: $\mathbf{v}_H = \overline{\mathbf{v}}_H$ in C_H^k , and $\mathbf{v}_H = 0$ in E_H^k .

5. *Interpolate* the coarse grid correction \mathbf{v}_H to the fine grid, yielding the fine grid correction \mathbf{v}_h , then set $\mathbf{v}_h = 0$ in the exterior area E_h^k .
6. *Correction*: Add correction \mathbf{v}_h to the pressures \mathbf{p}_h for all elements in the potential contact area.

The difference between this algorithm and the standard multigrid method is that not only the defects \mathbf{d}_h but also the contact area C_h^k (active set) are restricted to the coarse grid. The resulting area C_H^k defines the defect equation on the coarse grid. In order to satisfy the condition that $p_I = 0$ in the exterior area, all quantities including the pressure, defects, corrections in the exterior areas, E_h^k and E_H^k , are set equal to zero in this algorithm.

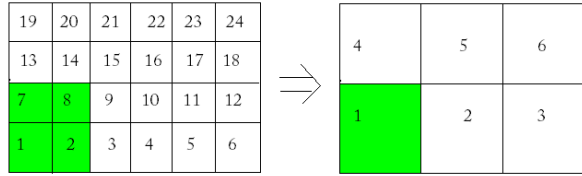
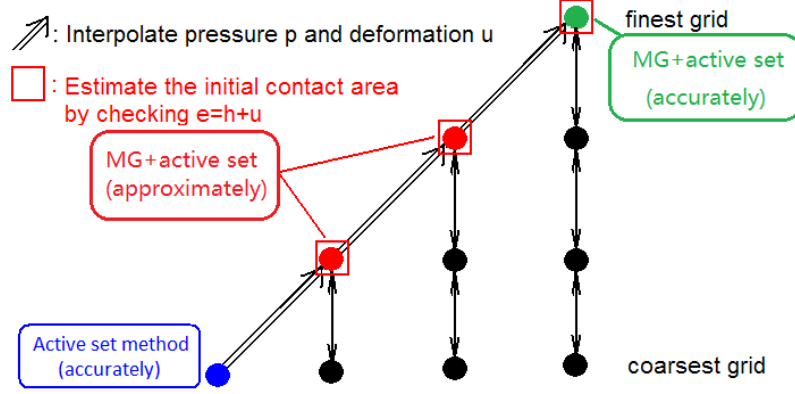


Figure 2.4: *Standard coarsening means combining 2×2 cells into one coarse grid cell. For example, the shadow element 1 on the coarse grid (on the right) is obtained by combining four shadow elements 1,2,7,8 on the fine grid (on the left).*

The idea for coarsening is the simple and the most frequently used choice, *standard coarsening*, i.e. doubling the element size in each direction ($H = 2h$). An example is depicted in Fig. 2.4, where all elements are numbered in lexicographic ordering. The restriction and interpolation operators are described based on the shadow elements in Fig. 2.4. The details are as follows, where the formulas in the first and the third items are used for the elements in both contact and exterior areas.

1. *Restriction of the defects*: we apply the “four-point average”, a frequently used restriction approach on cell-centered meshes. Take the shadow elements in Fig. 2.4 as an example, we have $d_H^1 = \frac{1}{4}(d_h^1 + d_h^2 + d_h^7 + d_h^8)$, where the superscripts denote the element numbers.
2. *Restriction of the contact area*: among the elements 1,2,7,8, if there is at least one element in the contact area C_h , then the coarse grid element 1, which is obtained by combining these four elements, is put in the coarse contact area C_H .
3. *Interpolation of the correction*: we “copy” the correction of coarse element 1, v_H^1 , to the correction of the four corresponding fine grid elements, 1,2,7,8, i.e.: $v_h^1 = v_h^2 = v_h^7 = v_h^8 = v_H^1$. This seemingly very basic interpolation is consistent with the restriction approach of the contact area, and appears to be a robust choice for the irregular and possibly tiny contact regions encountered in the rough contact problem, discussed below.

2.4.3 The FMG Scheme

Figure 2.5: *The FMG scheme.*

It is well known that an accurate initial guess can accelerate the convergence. This is even true for the contact problem where we must also identify the active contact region. Therefore, the MG+active set algorithm is combined with a nested iteration, which aims at providing an improved initial contact area and pressure for the target grid. This leads to the FMG method. Fig. 2.5 illustrates its structure, where each circle denotes a grid level. The method is described as follows:

1. We start on the coarsest grid, where the LCP is solved accurately by an active set algorithm. In other words, we use a direct solver for the governing integral equation.
2. Solving the LCP results in a contact area C , pressures \mathbf{p} , and the deformations \mathbf{u} where $u_I = -h_I$ for $I \in C$. The pressures \mathbf{p} and deformations \mathbf{u} are interpolated to a finer grid. The interpolation strategy of the former is the “copy” strategy discussed before. For the latter, we use bi-linear interpolation.
3. On the finer grid:
 - (3.a) Obtain an initial contact area based on the deformed distance \mathbf{e} , calculated by $\mathbf{e} = \mathbf{h} + \mathbf{u}$, where the undeformed distance \mathbf{h} can be computed on this grid level. All the elements I for which

$$e_I \leq \beta \cdot \max(\mathbf{u}), \quad (2.17)$$

are put into an initial contact area, with β a tolerance parameter.

- (3.b) The LCP on this grid level is solved by several iterations of the MG+active set algorithm, until the contact area does not change anymore.

4. Repeat steps 2 and step 3 until we reach the target grid, where the MG+active set algorithm is applied to solve the final contact area, and the pressures to the required accuracy here.

Remark 2.2. *Setting a proper value for β in Step (3.a) is helpful. A large β -value may result in a large initial contact area, whereas a small β -value typically yields a small initial contact area which is contained in the true contact region.*

Remark 2.3. *In this FMG algorithm, the LCPs on intermediate grids are only solved approximately, which gives a converged contact area, but the pressures do not need to be of required accuracy. The converged contact area is “interpolated” to the finer grid, based on the deformed distance. This appears sufficient for an improved initial contact area and pressures on the target grid in the numerical tests as seen in Section 2.5.*

2.4.4 Analysis on Jacobi and Gauss-Seidel Smoothers

Since Jacobi and Gauss-Seidel (GS) relaxations are often used as smoothers in multigrid for elliptic PDEs, we study them for our integral problem here. For simplicity, a 2D problem is built: we consider a cylinder of infinite length, rolling on a flat surface. Then, the contact area can be defined as $[-\alpha, \alpha]$ in the x -direction and “ $[-\infty, \infty]$ ” in the y -direction. Thus, we focus on the pressure distribution in the x -direction, leading to a 2D problem. We take the contact area to be $[-2.56, 2.56] \times [-500, 500] \text{ mm}^2$, which is discretized by a 32×1 grid. The resulting matrix A is dense, symmetric and Toeplitz.

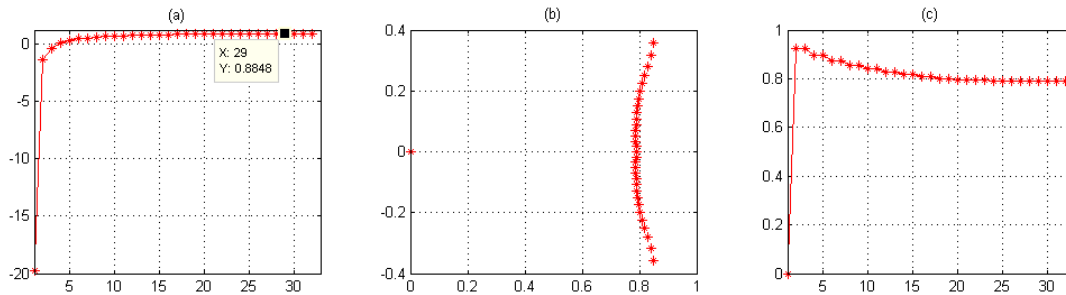


Figure 2.6: (a) The eigenvalues of the Jacobi (no damping) iteration matrix. (b) The eigenvalues of the (undamped) Gauss-Seidel iteration matrix, their absolute values are given in (c).

Fig. 2.6(a) presents the eigenvalues of the Jacobi iteration matrix. As can be seen, there is one eigenvalue far away from 1. This causes multigrid divergence as the coarse grid correction might fail to reduce the component enlarged so much. Damping by a very small parameter may shift it towards 1, but then the correction to the solution in each iteration will be very small, making the convergence slow. On the other hand, most eigenvalues are around 0.9. This

implies inefficiency to eliminate the related eigenvectors. Hence, Jacobi as a smoother for MG in our method is rejected.

Fig. 2.6(b) shows the eigenvalues of the GS iteration matrix in the complex plane. The absolute eigenvalues are given in (c). All eigenvalues are bounded by 1, implying that GS is a convergent solver.

The spectral radii of Jacobi and GS for this integral equation are relatively large. A dense coefficient matrix A for the integral equation [91] implies global (i.e. nonlocal) dependence in the system. During relaxation, updating the pressure at one point results in changes of the deformations at all other points. These deformations accumulate, which influences the efficiency of the Jacobi and GS relaxations.

To understand whether GS can be a smoother in MG for our problem, one may suggest to apply the local Fourier analysis (LFA). This technique is however based on the assumption that the relaxation is a local process, i.e. updating each unknown uses the information from only a few neighbors. However, this is not the case for our problem, where the information on all elements in contact is needed to update one unknown, since each entry of the original coefficient matrix is non-zero. Moreover, Fourier modes fail to resemble the shape of the eigenvectors of the iteration matrix of this integral equation. We will therefore only check with the help of the *eigenvectors* and *eigenvalues* of the iteration matrix for the smoothing efficiency.

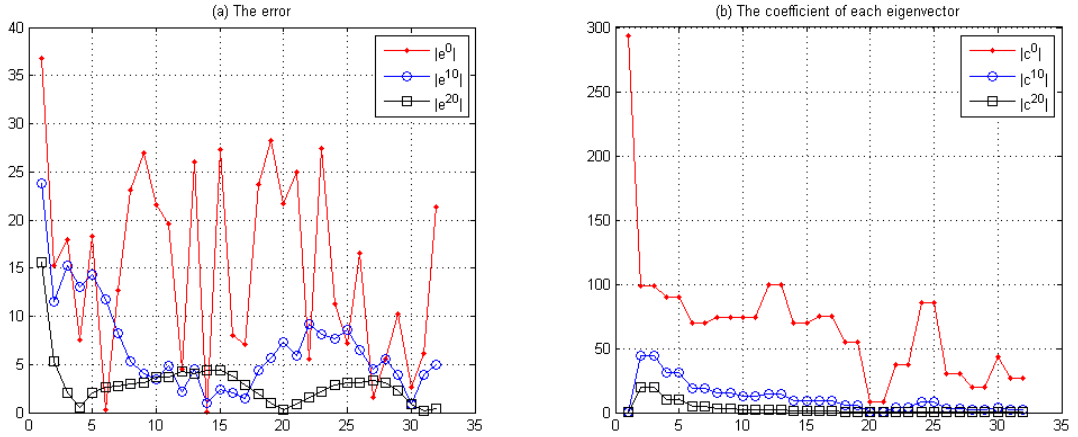


Figure 2.7: GS: (a) shows the error $|e^i| = |\mathbf{p} - \mathbf{p}^i|$ after i GS iterations ($i = 0, 10, 20$), where \mathbf{p} and \mathbf{p}^i are the exact solution and the iterates, respectively. The error can be expressed as $\mathbf{e} = \sum_j c_j \mathbf{v}^j$, $j = 1, \dots, 32$, where \mathbf{v}^j is the j th eigenvector of the iteration matrix, with coefficients c_j giving the contribution of this eigenvector to the error. These coefficients are stored in one vector \mathbf{c} , shown in (b) for different errors e^i in (a).

Fig. 2.7(a) shows the error $|e^i| = |\mathbf{p} - \mathbf{p}^i|$ after the i th GS iteration, where $i = 0, 10, 20$ and \mathbf{p} is the exact solution. As expected, the error gets smoother after more relaxations. The error \mathbf{e} can be expressed as a linear combination of

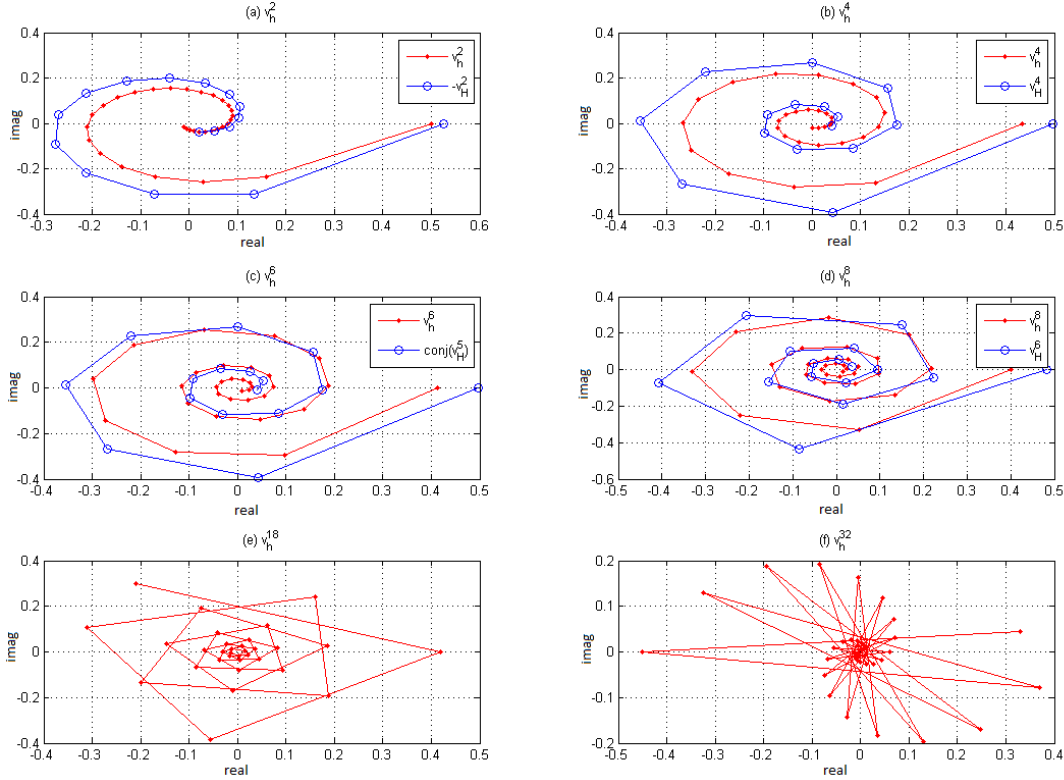


Figure 2.8: GS: the eigenvectors \mathbf{v}_h^i with $i = 2, 4, 6, 8, 18, 32$ are given in red dots in the complex plane, h denotes the fine grid 32×1 . The blue circles shows the corresponding eigenvectors of GS on coarser 16×1 grid.

the eigenvectors, i.e. $\mathbf{e} = \sum_j c_j \mathbf{v}^j$. We plot these c_j coefficients in Fig. 2.7(b) for the three errors in (a). As can be seen, when the error becomes smooth after 20 iterations, the coefficients for most eigenvectors are almost zero, except for the coefficients $c_j, j = 2, \dots, 9$.

Figs. 2.8(a)-(d) present in red dots the corresponding eigenvectors. Because of the conjugate property of GS eigenvalues in Fig. 2.6(b), the pairs of eigenvectors have symmetric shape. So we do not show them here. These modes are smooth. They correspond to the large eigenvalues in Fig. 2.6(c). The small eigenvalues are connected to eigenvectors with complicated shapes, e.g., \mathbf{v}_h^{18} and \mathbf{v}_h^{32} in Figs. 2.8(e) and (f), respectively. Therefore, similarly to the high- and low-frequency Fourier modes, we can also separate the eigenvectors into rough and smooth modes, where the former can be reduced by the smoother efficiently, and the latter remain after smoothing.

A question is, whether the smooth modes \mathbf{v}_h^i with $i = 2, 4, 6, 8$ can be represented on a coarser grid. Fig. 2.8(a)-(d) gives in blue circles the eigenvectors on a 16×1 grid, that have similar shapes as the fine grid smooth modes. We see that the fine grid smooth modes that can not be reduced efficiently, can be represented on the coarse grid, where they look rougher and can be reduced

efficiently.

This shows numerically that GS may be a satisfactory smoother for our integral problem.

2.5 Numerical Experiment and Results

In this section, we perform two numerical experiments, on a smooth and on a rough surface, respectively. We start by selecting a smoother and its damping parameter for the MG solver. Then, three V-cycles are compared. The performance of the MG+active set algorithm and the FMG method are investigated, comparing with the original method. The efficiency of interpolation and restriction of the active set is also discussed.

The stopping criterion is chosen as

$$\frac{\|d\|_{rms}}{\|h\|_{rms}} \leq 10^{-6}, \quad (2.18)$$

where d is the defect, and the root-mean square norm is $\|\cdot\|_{rms} = \|\cdot\|_2/\sqrt{n}$, with n the number of unknowns. Moreover, a 4×4 grid is used as the coarsest grid. We begin with the smooth contact surface experiment.

2.5.1 Test I: Hertzian Smooth Contact

The first test is a ‘‘Hertzian’’ case [44]. The undeformed distance function reads

$$h(x, y) = \frac{1}{2R_x}x^2 + \frac{1}{2R_y}y^2 - \delta,$$

where $R_x = 300 \text{ mm}$ and $R_y = 163 \text{ mm}$ are the principal relative radii of curvature, and $\delta = 0.1091 \text{ mm}$ is the maximum inter-penetration of the undeformed profiles. The potential contact area is $(x, y) \in [-10, 10] \times [-6, 6] \text{ mm}^2$. Steel material with shear modulus $G = 82000 \text{ N/mm}^2$ and Poisson’s ratio $\nu = 0.28$ is used to calculate the coefficient matrix A in system (2.12). We solve for the contact area, C , and the corresponding pressures, \mathbf{p} .

An analytic solution to this normal contact problem is given in [44]. The contact area is elliptic. For steel material in this test problem, the semi-axes of the contact ellipse are $a = 6.0 \text{ mm}$, $b = 4.0 \text{ mm}$, where a and b are half widths of the contact ellipse in the x - and y -directions, respectively. The normal pressure is zero outside the contact ellipse. In the contact area, it has ‘‘semi-ellipsoidal’’ form:

$$p(x, y) = p_{max} \cdot \sqrt{1 - \left(\frac{x}{a}\right)^2 - \left(\frac{y}{b}\right)^2},$$

where the maximum pressure, $p_{max} = 3F_z/(2\pi ab) = 1631 \text{ N/mm}^2$, with total load $F_z = 82000 \text{ N}$.

Results by the Multigrid Solver

First, we analyze the multigrid solver numerically to determine the efficiency of the multigrid components. The system we solve here is the C^0 system, which is (2.12) defined on the initial contact area C^0 . In this test, the C^0 system will be solved to a high accuracy given by (2.18).

As analyzed before, the Jacobi relaxation is not a good smoother for MG, particularly not when a fine discretization is used. Moreover, the damped Jacobi and red-black GS also exhibit worse convergence. We therefore do not show these results here.

The Gauss-Seidel relaxation with underrelaxation is required. To verify this, we tested a damped GS smoother with $\omega = 0.8$, $\omega = 1$ (standard GS) and $\omega = 1.1$ (SOR). The iteration numbers and convergence factors are given in Table 2.1 for two discretization resolutions. The convergence factor is defined as in [88]: $\hat{q}^{(m)} = \sqrt[m]{\frac{\|d_h^m\|_{rms}}{\|d_h^0\|_{rms}}}$. Quantity $\hat{q}^{(m)}$ is an average defect reduction factor over m iterations, so it may be an accurate estimate for the multigrid convergence, if m is sufficiently large. The smaller the convergence factor is, the better the convergence performance. In our tests, m is chosen to be the total number of iterations when the desired accuracy is achieved.

Discretization	$\omega = 0.8$	$\omega = 1$	$\omega = 1.1$
64×64	9 (0.19)	9 (0.20)	10 (0.24)
128×128	9 (0.20)	10 (0.24)	11 (0.28)

Table 2.1: *Hertzian contact: The number of V(1,1)-cycles by multigrid with GS smoothers to solve C^0 system. The convergence factors of multigrid are given in brackets.*

From Table 2.1, we find that the GS method with $\omega = 0.8$ is a fine choice, since its convergence factor is the smallest and it requires only 9 iterations on a 128×128 grid. In [91], Jacobi and Gauss-Seidel were regarded as unstable smoothers. However, the latter with underrelaxation works well for our integral problems. Hence, we will use GS with $\omega = 0.8$ as the smoother in our multigrid method.

Remember that in our MG solver also the contact area is restricted to the coarse grid. The results in the table imply that this special strategy is efficient and results in a mesh-independent solver.

Results by the MG+Active Set Algorithm

We arrive at the analysis of the MG+active set algorithm to solve the LCP (2.9)-(2.11). Three V-cycles, V(1,1), V(1,0) and V(0,1) cycles, are implemented in the MG+active set algorithm. Table 2.2 shows the numbers of active set iterations by the MG+active set algorithm (one active set iteration is combined with one MG cycle) and by the original method. As can be seen the former

requires slightly more active set iterations than the latter. Moreover, the iteration numbers increase as the problem gets larger by the two methods.

Discretization	V(1,1)	V(1,0)	V(0,1)	Active set+CG
32×32	9 (35)	15 (39)	15 (39)	5 (74)
64×64	9 (36)	16 (43)	16 (43)	7 (123)
128×128	10 (40)	18 (48)	16 (43)	8 (182)
256×256	12 (48)	21 (56)	20 (53)	9 (260)

Table 2.2: *Hertzian contact: the total number of active set iterations by the MG+active set algorithm and the original method, with the numbers of work units in brackets.*

However, the dominant effect of the computational time is the total number of matrix-vector products (MVPs). Therefore, we make a comparison for these between the MG+active set algorithm and the original method. In our algorithm, the damped GS smoother can be regarded as one MVP. Hence, one V(1,1)-cycle calculates three MVPs on each grid level (except for the coarsest grid, where the cost can be ignored): one for both pre- and post-smoothing, and one to compute the defect. Similarly, V(1,0)- and V(0,1)-cycles have two MVPs on each grid level. The cost of an MVP reduces by a factor 4 when going to a next coarser grid. Let the cost for one MVP on the target grid be one *work unit*. Then, the total numbers of work units are rounded and given in brackets. It can be observed that the V(1,1)-cycle requires fewer MVPs than the other two V-cycles.

As a comparison, the number of work units by the original method is also shown in this table, where one MVP is equal to one CG iteration. We can find that the MG+active set algorithm shows a great reduction of the number of work units compared with the original method.⁵

Results by the FMG Method

Now, we combine the MG+active set algorithm with the FMG technique, where we choose $\beta = 0.003$ in Eq. (2.17) to determine the initial contact area on finer grids, and present the convergence results in Table 2.3. We find that the numbers of active set iterations on the target grid are fewer when the FMG technique is used, comparing with the results in Table 2.2. Moreover, these numbers stay stable or even decrease as the problem size gets larger.

The numbers of work units are given in brackets. FMG presents a significant reduction of work units comparing with the original method shown in the last column of this table. Among these three V-cycles, the V(0,1)-cycle has the smallest cost for the larger problem 256×256 .

A comparison is made of the difference between the initial contact area and the converged contact area when solving on the 32×32 grid, which is shown in Fig. 2.9, where the dots of dark color denote the elements in the contact area

⁵Note that MVPs have complexity $\mathcal{O}(n \log n)$ if FFTs can be applied, but are $\mathcal{O}(n^2)$ if this is not the case.

Discretization	V(1,1)	V(1,0)	V(0,1)	Active set+CG
32×32	6 (24)	13 (36)	13 (36)	5 (74)
64×64	6 (27)	11 (32)	11 (32)	7 (123)
128×128	5 (23)	9 (27)	8 (24)	8 (182)
256×256	7 (34)	9 (29)	7 (24)	9 (260)

Table 2.3: *Hertzian contact: the total number of active set iterations by FMG on the target grid and by the original method, with the numbers of work units in brackets.*

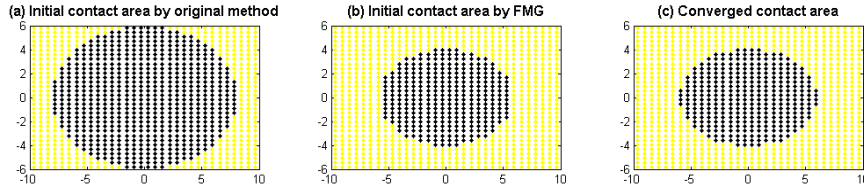


Figure 2.9: *Hertzian contact: contact areas on a 32×32 grid: (a) the initial contact area obtained by the original method, selecting all the elements I with $h_I \leq 0$, (b) the initial contact area by the FMG scheme, (c) the converged contact area.*

and light color dots represent the exterior area. The V(1,1)-cycle is used in the FMG method.

It can be seen from the figure that the initial contact area, using the FMG scheme (in (b)) resembles better the converged contact area (in (c)), than the initial contact area (in (a)) based on the original approach using the undeformed distance. This implies that the approach to “interpolate” the contact area, based on the deformed distance, is efficient in this case.

FMG and the Discrete Error

The FMG method has the interesting property that it can achieve the discrete error after several MG iterations on the target grid. Here, we also check this property for our problem and our FMG method. For different grid scenarios, Table 2.4 shows the discrete error $\|\mathbf{p} - \mathbf{p}_h\|_{rms}$ and the numerical error after the k th MG iteration on the target grid ($k = 0, 1, \dots, 5$). $h = \max(h_x, h_y)$ is the maximum mesh size. “(C)” indicates that after this iteration, the contact area is modified, and “–” means that the numerical error does not change.

As can be seen, discrete error $\|\mathbf{p} - \mathbf{p}_h\|_{rms}$ for different discretization gives $\mathcal{O}(h)$ accuracy of the pressure, which is also indicated in [51]. FMG achieves the discrete error after a few iterations. Then, more MG iterations do not help to reduce the discrete error. Moreover, for the 256×256 problem, the error already reaches the discrete error even when small modifications of contact area are still to be made.

Discretization	8×8	16×16	32×32	64×64	128×128	256×256
h	2.5	1.25	0.625	0.3125	0.1563	0.0781
0	315.19	239.54	142.09	78.97	41.04	22.84
1	56.73	81.92 (C)	44.12 (C)	17.51 (C)	8.14 (C)	10.00 (C)
2	44.90	33.73	18.57 (C)	7.63	4.55 (C)	4.07 (C)
3	—	28.31	16.74	6.82	3.67 (C)	2.36 (C)
4	—	—	—	—	—	1.97 (C)
5	—	—	—	—	—	—
$\ \mathbf{p} - \mathbf{p}_h\ _{rms}$	45.29	27.69	16.59	6.77	3.70	1.97

Table 2.4: *Hertzian contact: the norm $\|\mathbf{p} - \mathbf{p}_h^{FMG}\|_{rms}$. The first column gives the number of iterations, where 0 indicates the initial error. $h = \max(h_x, h_y)$ is the maximum mesh size. “(C)” means that after this iteration, the contact area is modified, and “—” means that the numerical error does not change. The $V(1,1)$ -cycle is used in FMG. $\|\mathbf{p} - \mathbf{p}_h\|_{rms}$ shown in the last row is discrete error.*

2.5.2 Test II: Rough Surface Contact

We implemented a problem of friction-less contact between a smooth, rigid flat ended circular punch and a rough, elastic half-space, see also [52]. Fig. 2.10(a) shows a general profile of this problem. The data are given, as follows:

- Potential contact area: $[-6, 6] \times [-5, 5] \text{ mm}^2$.
- Diameter of the punch: $D = 8.97 \text{ mm}$.
- Decay factor: $\gamma = 0.8$.
- Material parameters of the rigid punch: $G_1 = E_1 = \infty$.
- Material parameters of the half-space: $\nu_2 = 0.28$ (steel), $G_2 = 40000 \text{ N/mm}^2$.
- Coefficient of friction: $\mu = 0$.

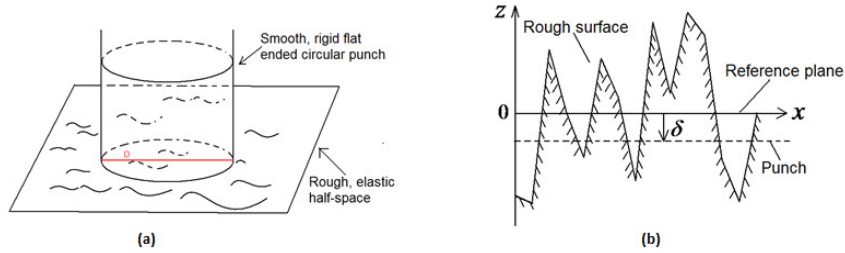


Figure 2.10: (a) *The general profile of rough surface contact; (b) Undeformed state in 1D situation.*

In a friction-less contact with an elastic solid, the contact stresses only depend on the shape of the gap between the two contacting bodies before loading, i.e. on the undeformed distance [44]. Fig. 2.10(b) gives the configuration in a 1D

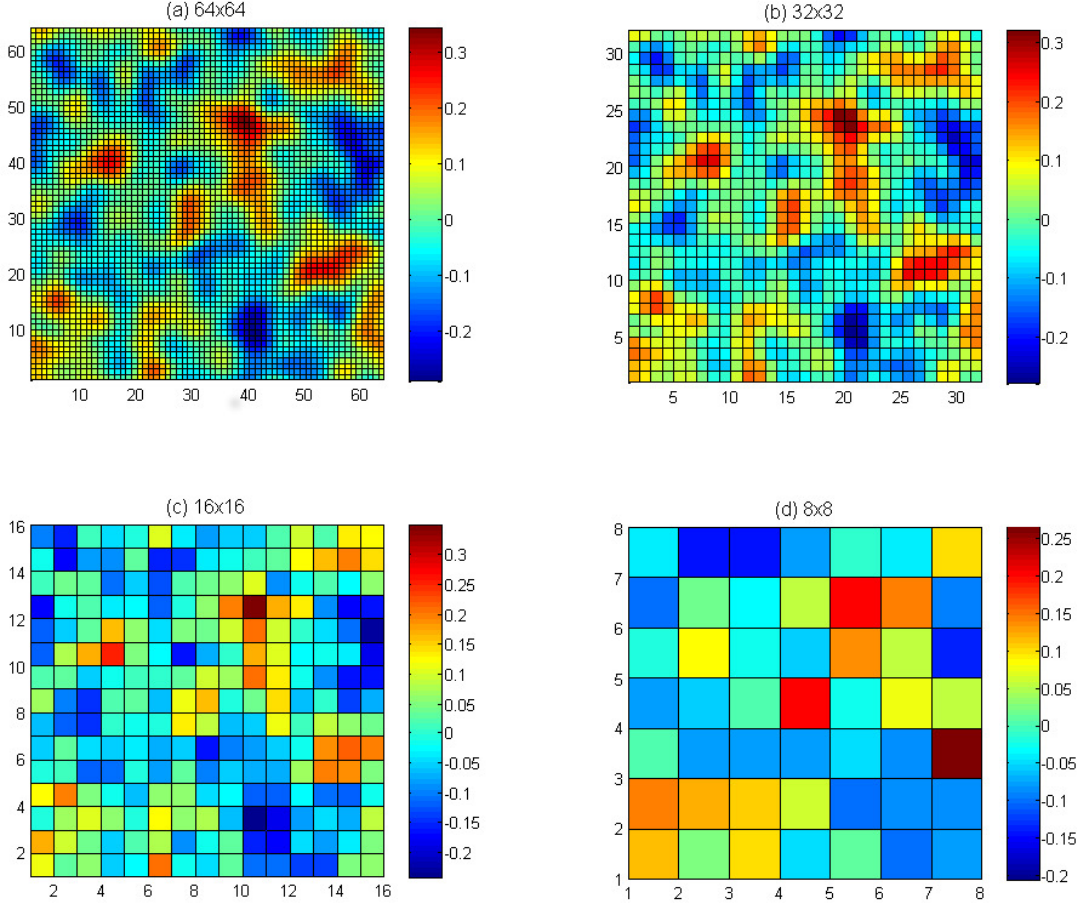


Figure 2.11: *The rough surfaces for four successive grids. They indicate similar shapes.*

situation, where the reference plane coincides with $z = 0$, since the mean of the surface heights, generated by the method discussed in Section 2.2.5, is equal to zero. According to this figure the undeformed distance h can be calculated, as follows:

$$h(x, y) = \begin{cases} -[z(x, y) + \delta], & \text{when } x^2 + y^2 \leq (D/2)^2 \\ \infty, & \text{otherwise} \end{cases}$$

where δ is the penetration, which is negative when the punch is above the reference plane and positive below the reference plane.

The rough surface generated in Fig. 2.3(b) by the technique explained in Section 2.2.5 is used. For the FMG algorithm, the surface height on each grid level is computed. Fig. 2.11 presents the surfaces on four successive grids, showing similar shapes.

The experiments are implemented for Cases 1, 2, 3 with δ equal to -0.11 mm , 0.01 mm and 0.25 mm , respectively. The penetration δ governs the percentage of the actual contact area w.r.t the surface of the punch, which is 5.0%, 12.0% and 28.2% in the Cases 1-3, respectively.

Results by FMG

Table 2.5 gives the total number of active set iterations by FMG on the target grid, and by the original method to solve Case 1. FMG needs fewer iterations than the MG+active set algorithm, the results of which are not shown here. As can be seen from the table, FMG does not show a superiority compared to the original method when considering the iteration numbers. However, as expected, a reduction of work units by FMG is found. Moreover, as the problem size gets larger, the number of work units increases for the original method whereas it stays stable for the FMG method. The V(1,1)-cycle requires fewer work units than the other two cycles for each problem size.

Case 1: $\delta = -0.11$, 5.0% contact				
Discretization	V(1,1)	V(1,0)	V(0,1)	Active set+CG
32×32	7 (32)	14 (40)	14 (39)	3 (35)
64×64	7 (31)	14 (40)	14 (40)	5 (69)
128×128	7 (31)	14 (40)	14 (41)	6 (112)
256×256	7 (32)	13 (39)	13 (39)	8 (167)

Table 2.5: *Rough contact Case 1: the total number of active set iterations by FMG and the original method, with the numbers of work units in brackets.*

Case 2: $\delta = 0.01$, 12.0% contact				
Discretization	V(1,1)	V(1,0)	V(0,1)	Active set+CG
32×32	7 (31)	13 (36)	13 (36)	4 (57)
64×64	8 (36)	14 (40)	15 (43)	5 (90)
128×128	8 (36)	14 (41)	14 (41)	8 (148)
256×256	7 (33)	12 (37)	12 (38)	8 (221)

Table 2.6: *Rough contact Case 2: the total number of active set iterations by FMG and the original method, with the numbers of work units in brackets.*

Case 3: $\delta = 0.25$, 28.2% contact				
Discretization	V(1,1)	V(1,0)	V(0,1)	Active set+CG
32×32	8 (35)	14 (40)	15 (42)	5 (72)
64×64	8 (37)	15 (45)	14 (42)	6 (107)
128×128	7 (34)	13 (39)	14 (42)	8 (169)
256×256	7 (35)	12 (38)	12 (38)	9 (244)

Table 2.7: *Rough contact Case 3: the total number of active set iterations by FMG and the original method, with the numbers of work units in brackets.*

The results in Tables 2.6 and 2.7 for Case 2 and Case 3, respectively, show the same behavior. Comparing these three cases, the work units by the original method grow as the penetration δ increases, i.e. as more elements come in contact. However, FMG shows stable work units for the three cases.

Case 3: $\delta = 0.25$, 28.2% contact					
#it	20	15	10	5	4
32×32	5(110)	5(85)	5(60)	6(42)	7(42)
64×64	6(132)	6(102)	6(72)	8(56)	10(60)
128×128	8(176)	8(136)	8(96)	11(77)	14(84)
256×256	9(198)	9(153)	10(120)	15(105)	DIV

Table 2.8: *Rough contact Case 3: the total number of active set iterations by inaccurate CG in the original method, with the numbers of work units in brackets. The second row is the number of CG iterations in each active set iteration.*

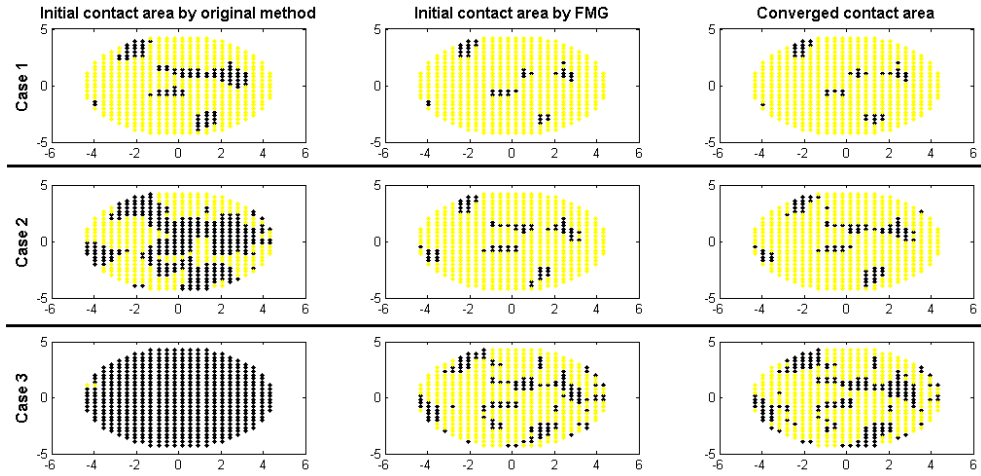


Figure 2.12: *Rough contact: contact area on a 32×32 grid for Cases 1, 2 and 3. The first column: the initial contact area obtained by the original method, selecting all the elements I with $h_I \leq 0$; the second column: the initial contact area by the FMG scheme; the last column: the converged contact area.*

The above comparison shows inefficiency of the original method, which solves the system in each active set iteration to high accuracy by CG. One may solve the system approximately by several CG iterations. We show the results for Case 3 in Table 2.8, where the second row “#it” gives the number of CG iterations in each active set iteration. We find that a larger number of active set iterations is needed when fewer CG iterations are used. Divergence even occurs when only four CG iterations are used to modify the active set for the 256×256 problem. Comparing with Table 2.7, this method requires fewer work units than the original method, but still more than the FMG method.

Fig. 2.12 displays the contact areas obtained on a 32×32 grid for these three cases, where the i th row denotes Case i , $i = 1, 2, 3$. The first and second columns are the initial contact areas obtained by the original method and by FMG with a V(1,1)-cycle, respectively. The last column gives the converged contact area in each case. It can be seen again that the initial contact areas, obtained by the FMG scheme resemble much better the converged contact areas,

than those obtained by the original method for these three cases. It turns out that, the strategy to obtain the initial contact area is efficient even when the area is irregular.

2.6 Conclusion

In this chapter we presented a full multigrid method, which is a combination of a multigrid method, an active set algorithm and the nested iteration approach, to solve linear complementarity problems arising from elastic normal contact problems.

Multigrid using a damped GS smoother is efficient for solving the corresponding integral equations, that give rise to a dense coefficient matrix. One multigrid cycle within an active set iteration appears to be sufficient to give an approximate solution to modify the contact areas efficiently and accurately. In particular, we do not only restrict the defects but also the contact area, and we set quantities outside the contact area equal to zero.

The FMG scheme provides accurate initial pressures when solving system (2.12), and also a highly satisfactory initial contact area, resembling the converged contact area much better than the initial contact area in the original method. The number of active set iterations and work units by FMG stays stable or even decreases as the problem size gets larger. Moreover, the FMG method achieves discrete error accuracy in a few iterations on the target grid.

According to the results for the rough surface contact, the FMG method shows robustness for the LCP on irregular computational domains. This is different from [78], where difficulties of FMG are reported for rough contact problems since the coarse grid could not represent the roughness well. In our FMG similar rough surfaces are generated on the successive grids.

The GS smoother is not a simultaneous displacement relaxation, hence a fast computing technique for MVPs, such as FFT computations or MLMI, can not be applied. In the next chapter, we will search for a more efficient smoother where the FFT can be applied for acceleration.

Multigrid with FFT smoother for a 2D Frictional Contact Problem

In this chapter we present the first part of the study to frictional contact problems, where we consider a 2D simplified model which is based on linear constraints. A Fredholm integral equation of the first kind is to be solved, for which we develop a fast multigrid solver. After discretization on a rectangular contact area, the integral equation gives rise to a linear system with the coefficient matrix being dense, symmetric, positive definite and Toeplitz. A so-called FFT smoother is proposed, which is based on a preconditioner M that approximates the inverse of the original coefficient matrix, and which is determined using the fast Fourier transform (FFT) technique. The iterates are then updated by a Richardson iteration. The FFT smoother significantly reduces most components of the error, but enlarges several smooth components. This causes divergence of the multigrid method. Two approaches are studied to remedy this feature: subdomain deflation and row sum modification.¹

3.1 Introduction

3.1.1 Physics

The frictional contact problem has attracted interest from many researchers, due to its applications in the industry and engineering fields, e.g. rolling contact fatigue (RCF) [29], the fatigue life of machine elements [89], friction and wear [22, 77, 41]. This problem concerns two elastic bodies. When they are pressed together, the forces they obtain from each other result in elastic deformation. This yields a contact area where the surfaces of the two bodies coincide, and exert stresses on each other. These stresses are composed of normal stress (pressure)

¹The contents of this chapter have been published in paper [113]: J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee. Multigrid with FFT smoother for a simplified 2D frictional contact problem. *Numerical Linear Algebra with Applications*, 21(2):256-274, 2014.

and frictional stress (traction) acting in the tangential direction. When and where the frictional stress is small, the two bodies stick to each other. However, local sliding occurs where the frictional stress is large enough. The challenge is to find the distribution of the frictional stress, and the subdivision of the contact area: which part is an adhesion area and in which part does slip occur.

The model for frictional contact starts with a known contact area and pressure distribution. Then the frictional stress should satisfy:

1. In the adhesion area, the magnitude of the tractions should not exceed the traction bound, and there is no slip.
2. In the slip area, the traction bound is reached, and the resulting slip points in the opposite direction of the tractions.

The traction bound comes from the friction law that is used. For this one may take Coulomb's law locally, which states that the traction bound equals the product of the normal pressure and a friction coefficient. The magnitude of the tangential tractions should be less or equal to the traction bound. This gives rise to inequality constraints. Moreover, when slip occurs, equality should hold, and the directions of the tangential tractions and the resulting slip should be opposite. This brings linear constraints in 2D problems, and nonlinear constraints in 3D problems. As a result the solution of frictional contact problems is quite involved, takes considerable computing time, so that faster solution methods need to be sought.

3.1.2 Solution Strategies

The classic solutions to frictional contact problems with partial sliding stem from the work by Cattaneo [13] and Mindlin [64]. In the last decades, other solution techniques have been studied, for example Johnson [44] and Kalker [51] contributed with fundamental work.

Numerical solution techniques typically employ variational inequalities [44, 51, 107]. They are generally divided into two classes. One class is based on the finite element method (FEM) [53, 107, 56, 19, 6, 39], which is widely used, especially in the case of large deformations, and nonlinear elastic materials. This method typically focuses on overall behavior. Due to the discretization of the contacting bodies, this method can be computationally expensive. The other class is governed by the boundary element method (BEM) [51, 2, 62, 55], which is well-suited for "concentrated contact" and efficient for homogeneous elastic problems. The boundary value problem is transformed to a boundary integral equation. The dimensionality of the problem decreases, i.e., the 3D contact problem is solved by considering 2D contact regions where only the boundary is discretized. Hence, this method reduces the computational time significantly.

When the contacting bodies are of different materials, the tangential tractions and normal displacements interact with each other. In this case, the normal and tangential problems cannot be easily separated. A straightforward way to

process it is to solve a fully coupled formulation [106]. Another popular approach is via the so-called “Panagiotopoulos process” [69, 28, 51]. In each iteration, the normal problem is solved first followed by the tangential problem. When contacting bodies are of the same material, i.e. a so-called quasi-identity case, these two problems can be decoupled, and one iteration is sufficient [51].

3.1.3 Solution Algorithms

Kalker’s variational approach [51], which is a prominent method for the rolling contact problem, employs the Green’s function for the elastic half-space. This is a BEM, where Coulomb’s law is applied locally. A fine discretization is used inside the contact area. Dense matrices need to be solved for the elements in the contact area.

As a solution algorithm, the TANG algorithm was proposed in [50]. It applies an active set strategy [65], which leads to systems of nonlinear equations that are solved using Newton’s method and Gauss-elimination. This approach has $\mathcal{O}(n^{3.5})$ complexity, with n the number of contact elements. Another method is the ConvexGS method [95]. It reduces the global problem to a small-sized optimization problem on each element, and solves these by a block Gauss-Seidel iteration. This method is incorporated into the software CONTACT [101]. However, the Gauss-Seidel process is also relatively slow for fine discretizations with a complexity of about $\mathcal{O}(n^{2.3})$ [96].

Different from the BEM, the FEM is based on a large number of elements covering whole contacting bodies, while much fewer elements are placed in the contact area. Sparse matrices are solved, but the size of the matrices is much larger than the dense matrices involved in the BEM. Algorithms include the penalty approach, the augmented Lagrangian technique, etc. Comparing with the BEM method, we encounter similar approaches for the nonlinear equations, like Newton-based methods [107], or a nonlinear Gauss-Seidel method [45] that is similar to ConvexGS.

Solving the surface integral problem has always been the most time-consuming part of the tangential contact problem. Within the context of a prescribed contact area, this integral equation is a Fredholm integral of the first kind. Using rectangular elements for discretization on a rectangular domain, the normal contact problem leads to a linear system with its coefficient matrix being dense, symmetric and positive definite (SPD). The 2D tangential contact problem is also linear, but in 3D the tangential problem becomes nonlinear. Moreover, the underlying matrix is a Toeplitz matrix for 2D problems and a block Toeplitz matrix with Toeplitz blocks (BTTB) for 3D problems. This structure can be exploited using fast algorithms.

Toeplitz matrices occur in various engineering fields, e.g. image deblurring and signal processing (e.g. [34, 79]). Besides the direct solution methods, e.g. [1], a popular iterative method for these matrices is the preconditioned Conjugate Gradient (PCG) algorithm, where circulant and Toeplitz preconditioners were

studied in [85, 38, 15, 30]. As an efficient iterative method, multigrid has also been applied and analyzed, e.g. in [14, 18, 7, 82]. From the point of computational cost, Lubrecht and Venner [91] applied a multi-level multi-summation (MLMS) technique. Fast Fourier transforms (FFT) have been used successfully for contact problems too [84], for speeding up matrix-vector multiplications for Toeplitz matrices.

3.1.4 Contents of This Chapter

The motivation of our work in this chapter is to improve the computational speed for solving a 2D frictional contact problem. In the previous chapter, we presented a full multigrid method for the normal contact problem [112]. An important component there involved the restrictions and interpolations of functions on changing, irregular domains. There we used the Gauss-Seidel iteration as a smoother, because of its robust convergence properties. However, this smoother leads to $\mathcal{O}(n^2)$ complexity for the overall solution algorithm. The use of FFTs for speeding up the NORM algorithm was addressed in [100], as well as a way to use FFTs for creating a preconditioner. Here we investigate the possibilities of this algorithm as a smoother in the multigrid algorithm. This appears to be not so straight-forward as one might expect. Therefore in this chapter we first focus on a 2D problem in which no slip occurs, and aim to expose the difficulties and remedies that can be used. The 3D frictional contact problem will be solved in Chapter 4.

This chapter consists of six sections. Section 3.2 formulates the 2D frictional rolling contact problem. Furthermore, it specifies the first model problem for later discussions on efficiency and convergence. Our FFT smoother is proposed in Section 3.3, including the construction and discussion of its smoothing properties. The investigation of the FFT smoother in the multigrid method shows its ability to significantly reduce most of the components of the error except for several problematic modes. Section 3.4 studies two different remedies for these modes: subdomain deflation (SD) and row sum modification (RSM). Three numerical experiments on different model problems are presented in Section 3.5. In addition, the FFT+RSM as a solver, rather than as a smoother, is also discussed. The last section concludes the chapter.

3.2 Formulation of the 2D Frictional Contact Problem

This section gives a general introduction of the 2D frictional contact problem along the lines of the presentation in [102]. More details can be found in [51]. A first model problem, called “MP1”, is defined at the end of this section. It will be used in the investigation and discussion in the sections to follow.

3.2.1 Physics of the Frictional Contact Problem

Let's consider the train wheel rolling on a rail. Both the wheel and rail experience deformations due to the forces they obtain from each other. After deformation, the surfaces of the two bodies coincide and an area of contact occurs. With the position information before deformation, we can solve for the resulting deformation, including the contact area and the traction on it.

This traction includes pressures in the normal direction and frictional stresses acting tangentially. The former can be obtained by the classical Hertzian theory [36] if the contacting surface is smooth and quadratic or by a numerical method (e.g. [101]) if the situation is non-Hertzian. It is compressive in the contact area and zero outside. The latter results from “creepage”. This is the average relative velocity between the surfaces, and is given by the difference between the forward velocity V and the circumferential velocity ωR . Creepage is accommodated partly by elastic deformation and partly by local sliding in the contact patch. As illustrated by the example of tug of war in Fig. 1.1, in both cases frictional stress is invoked, either to create deformation for keeping the surfaces sticking together, or opposing the local slip velocity.

A typical size for the contact between steel wheel and rails is $10 \times 10 \text{ mm}$, and the corresponding penetration is 0.01 mm . The elastic deformations decrease by $1/r$, with r the distance to the contact area, and the stresses and strains decrease in proportion to $1/r^2$. Therefore the stresses and strains are negligible at distances of a few centimeters outside the contact patch. This is the reason why the overall and local problems may be decoupled [51].

3.2.2 Mathematical Model of the Frictional Contact

For a detailed study of the components of the contact solver, in this chapter we focus on a 2D frictional contact problem. Fig. 3.1 depicts a cylinder rolling over a flat surface, with its axle parallel to the y -direction. This is a situation of “line contact” where “plane strain” occurs. In this situation there's no dependence on the y -coordinate, so that we focus from a 3D on a 2D situation.

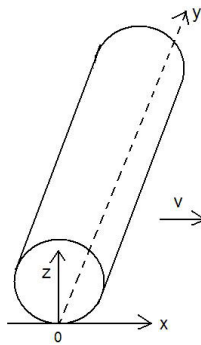


Figure 3.1: *The 2D line contact problem.*

Modeling starts from the assumptions of linear elasticity, small displacements and displacement gradients, and uses a quasi-static approach. The basic equations obtained from these assumptions describe equilibrium of the forces everywhere in the contacting bodies at all times (ignoring effects of inertia), give the relation between displacements and strains, and describe the strain of the material as a function of the stresses applied (constitutive equation). We are interested mostly in the quantities at the surfaces of the contacting bodies. Since the tractions on the two bodies are the same and opposite, we need only consider the tractions acting on one of them. Moreover, if the bodies are of equal elastic materials, the displacements are of equal size and opposite too, and can be combined into a single “displacement difference”, called displacement for short.²

The 3D tractions acting on the upper surface are denoted by \mathbf{p} . The component in normal direction is $p_z = \mathbf{p} \cdot \mathbf{n}$ with $\mathbf{n} = (0, 0, 1)^T$ the inward normal (coordinate system aligned with the contact plane). The pressure p_z is compressive in the contact area C and vanishes outside. Assuming that the normal contact problem has been solved before, p_z and C are prescribed, we concentrate on the tangential problem. Further, since we focus on a 2D situation the unknowns in the contact plane are functions of the position x and the time t .

Three aspects, i.e. creepage, elastic deformation and friction between two contacting bodies are taken into account in the model. First of all, the rigid slip velocity w_x in the undeformed state must add up with the actual slip velocity s_x and the change of deformation Du_x/Dt [102]:

$$s_x(x, t) = w_x(x, t) + \frac{Du_x(x, t)}{Dt}, \quad (3.1)$$

where subscript x denote the x -direction in Fig. 3.1. A material time derivative $\frac{D}{Dt}$ is used for the change of deformation so that the same equation can be used with moving coordinates. In case of wheel-rail contact with a world-fixed coordinate system this term equals $\partial u_x / \partial t$.

Concerning the third aspect, friction, the contact area C is divided into an adhesion and a slip area, determined by the following contact conditions, amounting to Coulomb’s friction law applied locally in each point $x \in C(t)$:

$$\begin{cases} \text{In adhesion area } H(t) : |s_x(x, t)| = 0, |p_x(x, t)| \leq g(x, t), \\ \text{In slip area } S(t) : |s_x(x, t)| > 0, p_x(x, t) = -g(x, t) \operatorname{sgn}(s_x(x, t)). \end{cases} \quad (3.2)$$

These conditions state that the tangential traction is bounded by a traction bound $g(x, t) = \mu p_z(x, t)$, where μ is the friction coefficient. When the tangential traction reaches the maximum, local slip occurs, and the traction opposes the slip. In 2D this simply requires us to find the sign of the slip velocity, which is typically the same as the sign of w_x . In 3D, this problem becomes much more complicated and we will discuss it in Chapter 4.

The second aspect mentioned above, concerning the deformation as a function of stresses p_x , is addressed below.

²The situation of equal elastic constants is called quasi-identity [51], and is assumed here for simplicity; the same theory works for bodies of different materials.

3.2.3 The Half-Space Approach

The slip velocity s_x in (3.1) depends on the overall motion (the rigid slip velocity w_x) and the rate of change of the deformation Du_x/Dt . To calculate the deformation $u_x(x, t)$, we use the *half-space approach*, which was discussed in Section 2.2.3. The idea is to approximate the elastic field in the contacting bodies, by considering each body as a half-space, a semi-infinite elastic solid bounded by a plane surface. Based on the classical solutions by Boussinesq and Cerruti (see Johnson [44] or Kalker [51]), the relation between surface tractions and deformations is the following:

$$u_x(x, t) = \int_{C(t)} A^{xx}(x, x') p_x(x', t) dx'. \quad (3.3)$$

This integral indicates that at one time instant, the deformation u_x at one point x is influenced by the tractions at all points x' in the whole contact area. Note that relation (3.3) is instantaneous, there's no dependence on the prior history. This is because of the quasi-static approach in which the effects of inertia on the elastic field are ignored. The kernel function $A^{xx}(x, x')$ is the influence function for the x -component deformation at a surface point x , due to the contribution of a unit x -direction traction at another surface point x' . Similarly as stated in Eq. (2.6), we have

$$A^{xx}(x, x') = A^{xx}(x - x'). \quad (3.4)$$

3.2.4 Discretization

In the 2D problem, shown in Fig. 3.1, the contact area is a strip of infinite length in the y -direction. Hence, the real contact area is $[-a, a] \times [-\infty, \infty]$. One way of dealing with this “infinite region” is by truncating the contact area to $[-a, a] \times [-b, b]$ with $a \ll b$. We only use a single element for discretization in the y -direction, and n elements in the x -direction. Quantities in the y -direction can then again be ignored due to plane strain. An advantage of this approach is that 2D problems can be solved with the same code as 3D problems, without any reference to the influence functions for 2D. Since the following discussion concentrates on the quantities in the x -direction, as a reminder, we drop subscript x from u_x, p_x, s_x, w_x , and superscript xx from influence coefficients A^{xx} .

Now, we discuss the discretization of the integral (3.3) and the slip Eq. (3.1) on a rectangular domain. A cell-centered mesh is used. The surface traction is approximated by an element-wise constant function. Then, the integral (3.3) is discretized as:

$$u_I = \sum_{J=1}^n A_{IJ} p_J, \quad I = 1, \dots, n, \quad (3.5)$$

where I, J are element indices. In a matrix form, we obtain:

$$\mathbf{u} = A \mathbf{p}, \quad \mathbf{u}, \mathbf{p} \in \mathbb{R}^n, \quad A \in \mathbb{R}^{n \times n}. \quad (3.6)$$

The coefficient matrix A is dense and SPD. In 2D it has a Toeplitz structure due to the space-invariance property in (3.4). In 3D problems, the coefficient matrix is a block Toeplitz with Toeplitz blocks (BTTB).

In our multigrid approach for the 3D normal contact problem in Chapter 2, we used a Gauss-Seidel smoother. Because A is dense, this smoother has arithmetic complexity $\mathcal{O}(n^2)$ and is difficult to parallelize. This solver is therefore not competitive to algorithms based on FFTs (exploiting the Toeplitz structure) or multi-level summation (also exploiting (3.4)) that are of complexity $\mathcal{O}(n \log(n))$. Therefore our target is here to construct a smoother that is parallelizable and has $\mathcal{O}(n \log(n))$ complexity, for solving the 2D tangential contact problem.

The slip Eq. (3.1) in rolling problems can be discretized with a sequence of time steps with step length $\delta t = t - t'$, where t and t' are the current and previous time instants, respectively. Then Eq. (3.1) yields [96]:

$$\mathbf{s} = \mathbf{w} + (\mathbf{u} - \mathbf{u}')/\delta t. \quad (3.7)$$

This uses the implicit Euler approach.

3.2.5 The First Model Problem

We specify a model problem, “MP1”, for the following discussion about efficiency and convergence of the smoother in our MG solver. It is defined as follows:

- The cylinder in Fig. 3.1 is pressed first to a flat surface and is then shifted, instead of rolled. Hence the deformation at a previous time step \mathbf{u}' is zero and the time-dependence is not present.
- We assume that the friction coefficient $\mu = \infty$. In this case, the traction will not reach the traction bound and no slip occurs. Eq. (3.7) becomes:

$$\begin{aligned} \mathbf{s} &= \mathbf{w} + \mathbf{u}/\delta t \quad (\text{since } \mathbf{u}' = \mathbf{0}) \\ &= \mathbf{w} + A\mathbf{p}/\delta t \quad (\text{due to Eq. (3.6)}) \\ &= \mathbf{0} \quad (\text{due to } \mu = \infty). \end{aligned} \quad (3.8)$$

Therefore, the system to be solved is

$$A\mathbf{p} = \mathbf{u}, \quad (3.9)$$

where $\mathbf{u} = -\delta t \mathbf{w}$ with $(\delta t \mathbf{w})$ the rigid shift.

- The cylinder and the flat surface have Poisson’s ratio $\nu = 0.28$ and shear modulus $G = 82000 \text{ N/mm}^2$, typical values for steel. These two parameters are used to compute the influence coefficients.
- The potential contact area is $[-4, 4] \times [-50, 50] \text{ mm}^2$. We use an $n \times 1$ grid for discretization, yielding a Toeplitz coefficient matrix A in Eq. (3.9).

- We set the rigid shift ($\delta t \mathbf{w}$) to be constant with 0.0008 mm and we solve for the traction \mathbf{p} .

Summarizing, MP1 reduces the commonly 3D problem to a 2D problem. For simplicity, it excludes time-dependence and contact conditions. We solve Eq. (3.9) to get the tractions \mathbf{p} . The right-hand side of this system is constant in MP1, but we will also consider a space-dependent right-hand side in MP2 in Section 3.5.

3.3 FFT Smoother for Multigrid Method

The multigrid method shows its great efficiency particularly when solving elliptic PDEs. It has also been developed for integral equations of the first kind (e.g. [104, 10]) and of the second kind (e.g. [33, 3, 35]). The integral (3.3) in the 2D problem is of the first kind. Since a smoother is essential for a multigrid solver, we focus on finding a satisfactory smoother in this chapter. Our target is to design a solver of $\mathcal{O}(n \log(n))$ complexity, with n the number of unknowns.

This section consists of three subsections. We construct the FFT smoother in the first subsection and analyze its smoothing properties in the second subsection. The last subsection shows multigrid convergence with this smoother.

3.3.1 Construction of the FFT Smoother

Consider solving a system $A\mathbf{p} = \mathbf{u}$ where \mathbf{p} and \mathbf{u} are the tractions and deformations, respectively. The FFT smoother is inspired by the idea of using FFTs for matrix-vector products (MVPs) for Toeplitz matrices. It is based on a Richardson iteration without relaxation parameter, which reads:

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \hat{\mathbf{d}}^k, \quad (3.10)$$

where

$$\hat{\mathbf{d}}^k = M\mathbf{d}^k, \quad (3.11)$$

i.e. the correction is based on the current defect $\mathbf{d}^k = \mathbf{u}^k - A\mathbf{p}^k$, preconditioned by matrix M . The challenge is to construct the preconditioner M . Since M is required to be an accurate approximation for A^{-1} , we analyze the columns of A^{-1} first. From the mechanics point-of-view, the K th column of A^{-1} represents the traction pattern, which results in a unit deformation on element K and zero deformation on the other elements.

Consider a 16×1 problem in MP1 as an example. Fig. 3.2(a) depicts columns \mathbf{c}^K of A^{-1} , where $K = 1, 3, 8, 12$. The pattern in the K^{th} column reveals the largest positive component at element K , and negative components for other elements. Such a pattern physically implies that on element K , a traction is arranged to obtain a unit deformation, with opposite tractions on the other elements to prevent the deformation there.

This figure also shows that, except for the column at the boundary, the other columns of A^{-1} have similar patterns. This implicitly is an indication for the

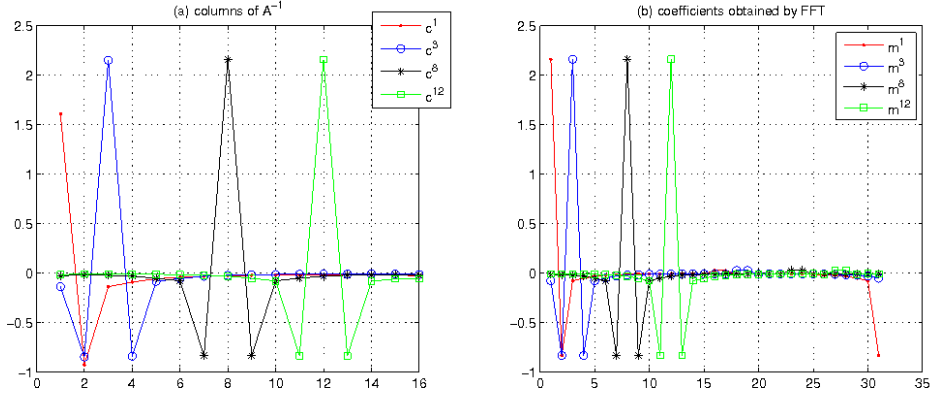


Figure 3.2: (a) The first, third, eighth and twelfth columns of A^{-1} , and (b) the coefficients obtained by FFTs to approximate them (for a 16×1 problem).

Toeplitz-like structure of matrix A^{-1} . Therefore, we construct a Toeplitz preconditioner M , using the coefficients obtained by approximating only one interior column of matrix A^{-1} .

The approximation of such a column can be done by the FFT technique. It is a powerful technique for efficiently computing with specially structured matrices, e.g. circulant and Toeplitz matrices. Both FFT and inverse FFT (IFFT) reduce the cost of an MVP, $A\mathbf{x} = \mathbf{b}$, from $\mathcal{O}(n^2)$ to $\mathcal{O}(n \log(n))$ operations, with n the number of unknowns. The main idea is to transform the coefficients of matrix A and vector \mathbf{x} by FFTs to the Fourier domain, where point-wise multiplication is implemented, after which we transform back by the IFFT to the physical domain. We refer to [90, 66] for more details. If \mathbf{b} is given, \mathbf{x} is approximated by transforming \mathbf{b} and A to the Fourier domain and then by back transformation after point-wise division.

By this approach we approximate the K th column of A^{-1} in our model problem. The algorithm is given as follows:

1. Denote the vector \mathbf{a} with all the influence coefficients, i.e.

$$\mathbf{a} = [a_{-(n-1)}, \dots, a_{-1}, a_0, a_1, \dots, a_{n-1}],$$

which uniquely determines this Toeplitz coefficient matrix A .

2. Extend the vector $\mathbf{u} = [\mathbf{0}, \mathbf{e}]$ of the same size with \mathbf{a} , where $\mathbf{e} = [e_0, e_1, \dots, e_{n-1}]$ is a unit vector of size n and the K th component $e_K = 1$.
3. The coefficients to approximate the K th column can be obtained by:

$$\mathbf{m}^K = IFFT[FFT(\mathbf{u}) ./ FFT(\mathbf{a})].$$

This formula means that both \mathbf{u} and \mathbf{a} are transformed by FFTs into the Fourier domain, where pointwise division ($./$) is done, after which the result is transformed back by the IFFT.

Here, steps 1 and 2 are illustrated in Fig. 3.3. Fig. 3.2(b) shows the resulting approximations for the columns in (a) by this algorithm. As can be seen, the left half part of the patterns in (b) have similar shapes as those in (a).

$$\begin{array}{c}
 a = \begin{array}{|c|c|c|c|c|c|c|c|c|c|}
 \hline
 a_{-(n-1)} & a_{-(n-2)} & \dots & a_{-1} & a_0 & a_1 & \dots & a_{n-2} & a_{n-1} \\
 \hline
 \end{array} \\
 \\
 u = \begin{array}{|c|c|c|c|c|c|c|c|c|c|}
 \hline
 0 & 0 & \dots & 0 & 0 & \dots & 1 & \dots & 0 & 0 \\
 \hline
 \end{array} \\
 \begin{array}{c}
 \underbrace{\hspace{10em}}_{\text{padding with zeros}} \quad \underbrace{\hspace{4em}}_{\text{the } K\text{th column of the identity matrix } I_n}
 \end{array}
 \end{array}$$

Figure 3.3: Steps 1 and 2 in the algorithm for approximating the K th column of A^{-1} by FFTs.

We choose to approximate the center column $K = n/2$ of A^{-1} in our method. The left half part of the resulting coefficients are denoted by

$$\mathbf{m} = [m_{-(n-1)}, \dots, m_{-1}, m_0, m_1, \dots, m_{(n-1)}],$$

with $m_i = m_{-i}$, $i = 1, \dots, (n-1)$ and m_0 the largest positive component. The dense, symmetric Toeplitz preconditioner M is defined as follows:

$$M = \begin{pmatrix}
 m_0 & m_{-1} & \dots & m_{-(n-2)} & m_{-(n-1)} \\
 m_1 & m_0 & m_{-1} & \dots & m_{-(n-2)} \\
 \vdots & m_1 & m_0 & \ddots & \vdots \\
 m_{n-2} & \dots & \ddots & \ddots & m_{-1} \\
 m_{n-1} & m_{n-2} & \dots & m_1 & m_0
 \end{pmatrix}.$$

3.3.2 Discussion about Smoothing Analysis

To analyze the performance of the FFT smoother, we decompose the error \mathbf{e} by the eigenvectors of iteration matrix $R_F := (I - MA)$. A smoother for a multigrid method should be able to efficiently eliminate the oscillatory eigenvectors. The smooth parts that are left can be represented and resolved on a coarse grid. Fig. 3.4(a) displays the eigenvalues of iteration matrix R_F of the FFT smoother for a 32×1 grid. As can be seen, all eigenvalues are very close to zero, except for two isolated eigenvalues that yield a spectral radius larger than 1. The eigenvectors corresponding to these two problematic eigenvalues are plotted in Fig. 3.4(b). They show either symmetry or anti-symmetry. Significant jumps are found near the boundaries, since the boundary effect is ignored in preconditioner M but is represented in A^{-1} . This observation indicates that the FFT smoother is able to significantly reduce most eigenvectors. However, the two eigenvectors that correspond to the isolated eigenvalues are magnified.

Local Fourier analysis (LFA) is widely used for smoothing analysis with PDE problems [88], since the Fourier modes are often the same as the eigenvectors

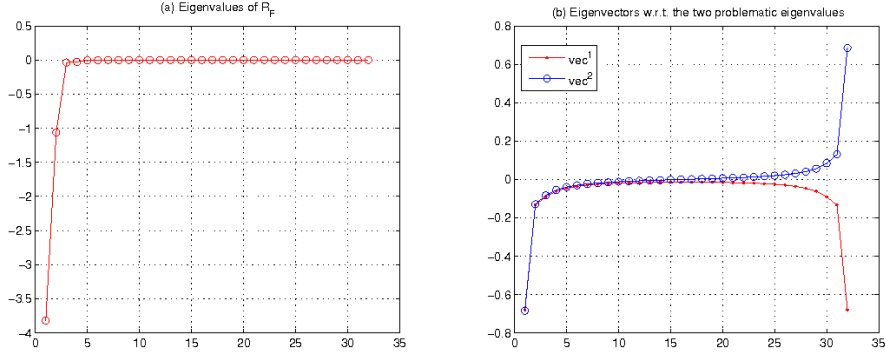


Figure 3.4: (a) Eigenvalues of $R_F = (I - MA)$ of the FFT smoother on a 32×1 grid. (b) Two eigenvectors corresponding to the problematic eigenvalues.

of the iteration matrix of the smoother. However, the Fourier modes do not resemble the eigenvectors in our problem! For instance, if we define the error in MP1 in a discretized form using an anti-symmetric extension [87]:

$$\mathbf{e} = \sum_{k=1}^{n-1} b_k \sin\left(\frac{k\pi\mathbf{x}}{L}\right),$$

where \mathbf{x} represents the centers of all elements, and $2L$ is the period of the extended error. Fourier modes are given by $\mathbf{v}_k = \sin\left(\frac{k\pi\mathbf{x}}{L}\right)$.

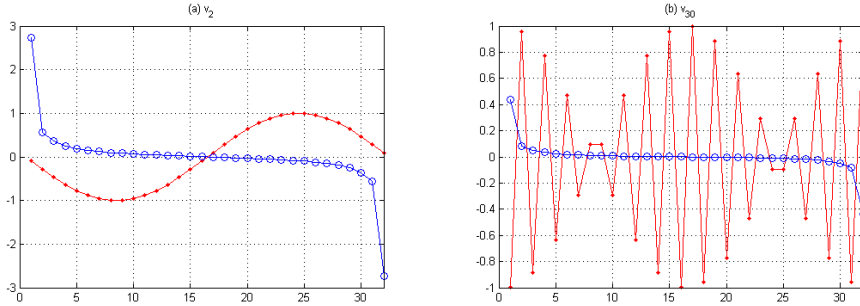


Figure 3.5: The Fourier modes (in red dots) and after the FFT smoother (in blue circles) for a 32×1 grid: (a) a smooth mode \mathbf{v}_2 , (b) an oscillatory mode \mathbf{v}_{30} .

Fig. 3.5 shows a low-frequency mode, \mathbf{v}_2 , in (a) and a high-frequency mode \mathbf{v}_{30} in (b). These two modes are denoted by red dots, and their shapes after the application of the FFT smoother are given by blue circles. Both modes after smoothing display the shape of the anti-symmetry eigenvector, shown in Fig. 3.4(b). (This is due to the Fourier modes we choose. They are sine functions with anti-symmetry shapes. The decomposition of the modes by the eigenvectors exhibits a large portion of the anti-symmetry eigenvector.) It is clear that the Fourier modes are not eigenfunctions. Therefore, we analyze the eigenvectors instead of the Fourier modes in our integral problem.

3.3.3 Multigrid Method Based on FFT Smoother

The above discussion indicates a difficulty of the FFT smoother when dealing with the two eigenvectors in Fig. 3.4(b). To see how this influences the multigrid convergence, we specify a multigrid method with a basic coarse grid correction. Standard coarsening which doubles the element size is illustrated in Fig. 3.6: two elements on the fine grid are combined into one coarse element.

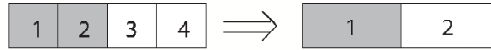


Figure 3.6: *The coarsening approach for a “2D problem”. Take the shaded elements on a 4×1 grid as an example. Two elements on the fine grid are combined to be one coarse grid element.*

The components of a two-grid method are defined as follows. (It is easily extended to a multigrid method.) The subscripts h and H denote the fine and coarse grid, respectively.

1. Smoother: the FFT smoother, as explained.
2. Restriction: use the average of defects on the two grid elements on the fine grid for the corresponding coarse grid elements. Taking the shaded elements in Fig. 3.6 as an example, the restriction reads:

$$d_H^1 = \frac{1}{2}(d_h^1 + d_h^2). \quad (3.12)$$

3. On the coarse grid: the coefficient matrix is obtained according to the grid resolution and a direct method is used to solve the defect equation. (We employ the backslash command in Matlab.)
4. Interpolation: copy the correction from the coarse grid element to the two corresponding fine grid elements. For instance in Fig. 3.6,

$$v_h^1 = v_h^2 = v_H^1. \quad (3.13)$$

Since multigrid is an iterative method, it converges when the spectral radius of the iteration matrix is less than 1. The spectral radii of the MG operators with the FFT smoother are given in Table 3.1, where V(1,0)- and V(0,1)-cycles display the same results due to the symmetry of matrices A and M . As can be seen, the spectral radii of these three V-cycles grow as the problem size increases. And they get larger than 1 for larger problems. The V(1,1)-cycle exhibits a much worse spectral radius than other two V-cycles. This is an indication of the difficulty of the FFT smoother. The reason may be explained as follows. The problematic eigenvectors contain steep gradients near the boundaries. This non-smoothness is amplified by the FFT smoother and the coarse grid correction,

multi-grid cycle	8×1	16×1	32×1	64×1	128×1	256×1
V(1,1)	1.80e0	1.74e1	2.49e2	5.06e3	1.41e5	5.20e6
V(1,0)	2.20e-1	7.1e-1	2.60e0	1.18e1	6.19e1	3.75e2
V(0,1)	2.20e-1	7.1e-1	2.60e0	1.18e1	6.19e1	3.75e2

Table 3.1: *The spectral radii of MG operators with the FFT smoother for different discretizations.*

because restriction by using averaging and interpolation just copying can not efficiently reduce them.

Although the FFT smoother results in multigrid divergence, it possesses the advantages of easy construction, fast computation of MVPs, and great reduction of major parts of the error. In the next section, we will discuss two remedies to overcome the difficulty with the problematic eigenvectors in Fig. 3.4(b).

3.4 Remedies for the FFT Smoother

This section describes two techniques to make multigrid with the FFT smoother convergent, subdomain deflation and row sum modification.

3.4.1 Subdomain Deflation

The first technique is to apply the FFT smoother to a defect which does not contain the problematic eigenvectors of R_F , i.e. of $\bar{A} := MA$. In order to remove them, we make use of the *deflation* technique, which preferably leaves the remaining part of the spectrum unchanged [67, 26]. The deflation matrix $P \in \mathbb{R}^{n \times n}$ is defined as:

$$P : = I - \bar{A}Q, \quad \text{with} \quad (3.14)$$

$$Q : = ZE^{-1}Z^T, \quad \text{and} \quad (3.15)$$

$$E : = Z^T \bar{A}Z. \quad (3.16)$$

We need to determine a deflation-subspace matrix Z , the columns of which should preferably be approximations of the “bad” eigenvectors. Moreover, computing with the deflation matrix should be cheap. One easy and often used approach is *subdomain deflation (SD)* [86]. It divides the computational domain Ω into m non-overlapping subdomains Ω_j , $j = 1, 2, \dots, m$, and we presume one deflation vector corresponding to one subdomain. This results in a deflation-subspace matrix Z with m columns. Using a piecewise constant approximation, the entries of Z are defined as:

$$z_{ij} = \begin{cases} 1 & i \in \Omega_j \\ 0 & \text{otherwise} \end{cases},$$

where $i = 1, \dots, n$ denote the grid elements, and $j = 1, \dots, m$ are the indices of the subdomains.

Fig. 3.4 showed only two problematic eigenvalues. Hence we divide the computational domain into $m = 2$ identical subdomains, i.e. the left half part and right half part. For instance, if there are $n = 8$ elements in total, with elements $\{1, 2, 3, 4\}$ and $\{5, 6, 7, 8\}$ in the left and right subdomains, respectively, then the resulting matrix Z reads:

$$Z = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}^T.$$

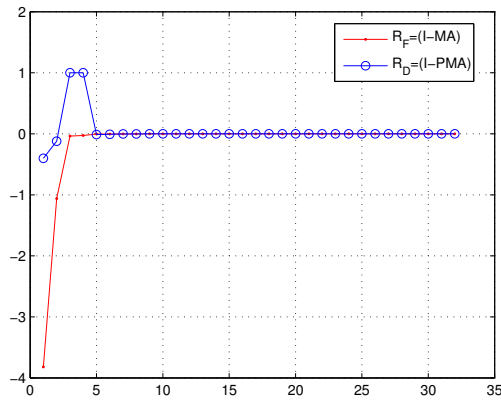


Figure 3.7: *Eigenvalues of R_F with the FFT smoother (denoted by red dots), and of R_D with the FFT+SD smoother (given by blue circles) for the 32×1 problem.*

Fig. 3.7 shows the eigenvalues of the iteration matrix based on this approach, $R_D := (I - PMA)$, for the 32×1 problem. They are denoted by blue circles. Comparing to the eigenvalues of the original iteration matrix R_F (given by red dots), the deflation technique shifts the two problematic eigenvalues towards 1. However, two other eigenvalues are enlarged, as the piecewise constant deflation vectors may not properly approximate the steep gradients near the boundaries of the problematic eigenvectors.

Fig. 3.8 displays by red dots the first four eigenvectors of iteration matrix R_D , which is based on the FFT+SD smoother. The blue circles in the figure show the shapes of these eigenvectors after multiplication by R_D . The two eigenvectors in (a) and (b) show symmetry and anti-symmetry. They are reduced but still contain steep gradients at the boundaries. The other two eigenvectors in (c) and (d) come from the piecewise constant deflation vectors, and they are preserved because the corresponding eigenvalues are 1.

Consider the computational work of the FFT+SD smoother. First, we look at the cost of an MVP with deflation matrix P , i.e. $P\mathbf{x}$. Assume that matrix

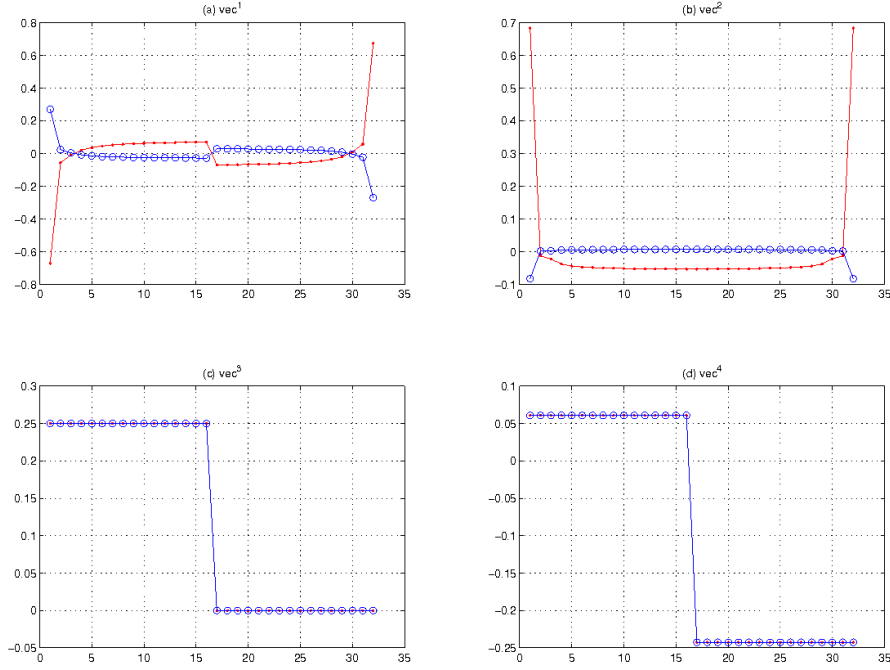


Figure 3.8: *The FFT+SD smoother: The first four eigenvectors of iteration matrix R_D (red dots) and the shapes of these eigenvectors after multiplying by R_D (blue circles), for the 32×1 problem.*

Z consists of m columns with $m \ll n$. Multiplying by Z or Z^T involves n operations.

- The inverse $E_{m \times m}^{-1}$ given in Eq. (3.15) can be computed beforehand. Since $m \ll n$, the cost for $E_{m \times m}^{-1}$ can be ignored.
- For an MVP $\mathbf{y} := Q\mathbf{x}$ according to (3.15), $Z^T\mathbf{x}$ requires n operations. The cost for $E^{-1}(Z^T\mathbf{x})$ can be ignored, and $ZE^{-1}(Z^T\mathbf{x})$ requires n operations. Hence, one MVP with Q is $\mathcal{O}(n)$ complexity.
- For the MVP $P\mathbf{x}$ in (3.14), $\bar{A}Q\mathbf{x} = \bar{A}\mathbf{y} = M(A\mathbf{y})$ requires 2 MVPs, involving $\mathcal{O}(n \log(n))$ operations.

Therefore, $\mathcal{O}(n \log(n))$ operations are needed for one MVP $P\mathbf{x}$, which, roughly speaking, involves 2 MVPs in total. The FFT+SD smoother is defined by:

$$\mathbf{p}^{k+1} = (I - PMA)\mathbf{p}^k + PM\mathbf{u}, \quad (3.17)$$

where, $(I - PMA)\mathbf{p}^k$ costs 4 MVPs, with 2 for matrix P and 2 for MA . The computation of $PM\mathbf{u}$ requires 3 MVPs, however, on the finest grid it can be computed in advance.

3.4.2 Row Sum Modification

The second technique for improvement comes from the fact that the row sum of matrix A^{-1} is related to the elastic energy $J(\mathbf{p}) := \frac{1}{2}\mathbf{p}^T A \mathbf{p} - \mathbf{p}^T \mathbf{u}$, since the solution $\mathbf{p} = A^{-1}\mathbf{u}$ minimizes energy. It is expected that matrix M in the FFT smoother is an accurate approximation of A^{-1} . We compare the row sums of these two matrices in Fig. 3.9 for the 32×1 problem.

We denote the row sums of any matrix A by $\text{sum}(A)$. It can be found from this figure that, first of all, $\text{sum}(A^{-1})$ exhibits almost constant values except for the rows at boundaries. Secondly, they are smaller than $\text{sum}(M)$ with especially large differences at the domain boundaries. It is due to the large difference in the boundary columns of A^{-1} and M , seen in Fig. 3.2. This can be explained from a mechanics point-of-view, as follows.

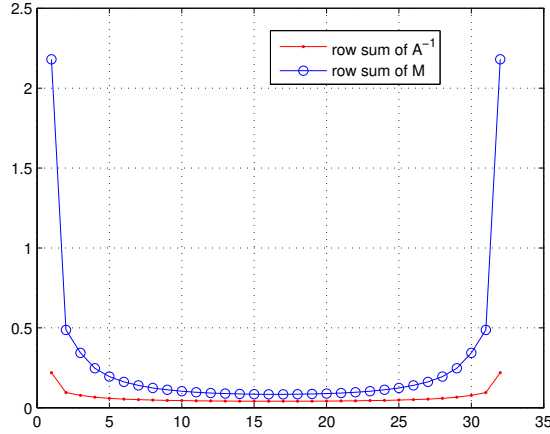


Figure 3.9: Row sums of A^{-1} (denoted by red dots), and of M (denoted by blue circles), for the 32×1 problem.

We number the elements in the contact area from the left to the right by 1 to n . When computing the coefficients \mathbf{m}^K for M by the FFT technique, the contact area is “extended” into an infinite domain. Suppose that there is one ghost element numbered by -1 at the left, next to boundary element 1. According to the coefficient pattern in M , to achieve a unit deformation at element 1, a large positive traction on element 1 and negative tractions on other elements should be arranged. Element -1 also gets a negative traction and contributes negative deformation to element 1. On the other hand, no ghost elements are present in A^{-1} . Hence, the elements at the boundaries do not receive the negative contributions of the deformation from them, and smaller traction values are required at these boundary elements to have unit deformation. This boundary effect gives rise to the difference between A^{-1} and M .

A modification of M is based on the observation above. The technique is to shift $\text{sum}(M)$ to a constant level, as $\text{sum}(A^{-1})$ appears constant at the interior

rows. Since $\text{sum}(M)$ is larger than $\text{sum}(A^{-1})$, the constant is chosen to be the minimum of $\text{sum}(M)$, which is the nearest to $\text{sum}(A^{-1})$. Shifting $\text{sum}(M)$ can be performed by only modifying diagonal entries, as they contribute most. Therefore, we will iterate with a shifted matrix \bar{M} , defined as:

$$\bar{M} = M - \text{diag}(\text{sum}(M) - \gamma), \quad (3.18)$$

where $\text{diag}(\mathbf{x})$ is a diagonal matrix with \mathbf{x} the main diagonal entries, and the scalar

$$\gamma = \min[\text{sum}(M)]. \quad (3.19)$$

This leads to a matrix \bar{M} , with $\text{sum}(\bar{M})$ equal to γ .

To investigate the quality of the choice of γ in (3.19), we perform a numerical experiment and take the 32×1 problem, plot the spectral radii of the shifted iteration matrices, $R_R = I - \bar{M}A$, with the values $\gamma \in [0, 0.15]$, in Fig. 3.10(a). As can be seen, $\gamma \approx 0.08$ results in the smallest spectral radius, and we zoom in around this value in Fig. 3.10(b), where $\gamma = 0.087$ appears an optimal value. Based on Eq. (3.19), we obtain $\gamma = 0.0837$, which is very close to this optimal value. Hence, we expect that this technique may work well.

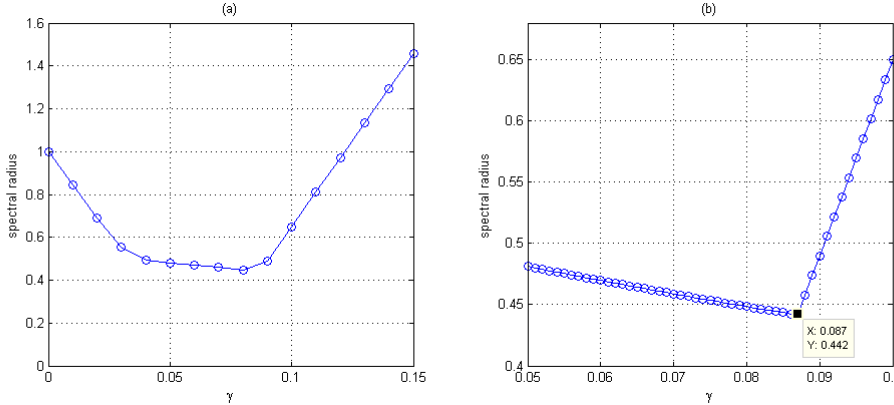


Figure 3.10: *FFT+RSM smoother: the spectral radii of iteration matrices $R_R = (I - \bar{M}A)$ with (a): $\gamma \in [0, 0.15]$, for the 32×1 problem. (b) displays the situation around $\gamma = 0.08$.*

Fig. 3.11(a) presents the eigenvalues of iteration matrix $R_R = (I - \bar{M}A)$, with \bar{M} defined by Eq. (3.18). All eigenvalues are now smaller than 1. The first eigenvector of R_R is denoted by red dots in Fig. 3.11(b), where the blue circles represent its shape after multiplying by R_R . The function is smooth and can be dealt with by the coarse grid correction. Fig. 3.11(c) shows the second eigenvector. It still exhibits a certain steepness at the boundaries.

Considering the computational cost, we check Definition (3.18) again. Write the coefficients of M as $\mathbf{m} = [m_{-(n-1)}, \dots, m_{-1}, m_0, m_1, \dots, m_{(n-1)}]$. Here, $\text{sum}(M)$ can be computed in an efficient way due to the symmetry of \mathbf{m} and the Toeplitz structure of M . First of all, we compute the first row sum f_1 , involving $n - 1$

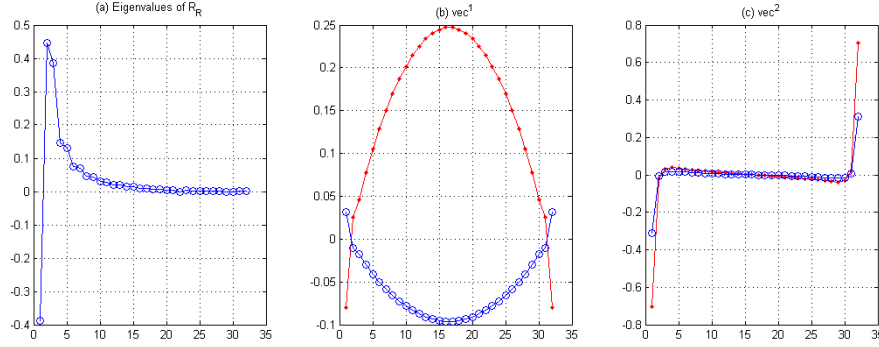


Figure 3.11: *FFT+RSM smoother: Eigenvalues of the iteration matrix of R_R for the 32×1 problem are shown in (a). The first and the second eigenvector are denoted by red dots in (b) and (c), respectively. Their shapes after multiplying by R_R are denoted by blue circles.*

operations. For the rows $i = 2, \dots, n$, the sum f_i can be computed recursively by $f_i = f_{i-1} - m_{n-(i-1)} + m_{i-1}$, where 2 operations are required for each f_i . So, $3(n-1)$ operations are needed to determine $\text{sum}(M)$. The arithmetic complexity is reduced from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$. An alternative which has $\mathcal{O}(n \log(n))$ complexity is to multiply M by a vector with all entries equal to one, using the FFT technique.

In each MVP with \bar{M} , the first part M needs $\mathcal{O}(n \log(n))$ operations by FFTs due to the Toeplitz structure. The second part requires $\mathcal{O}(n)$ operations because it is a diagonal matrix. Hence, $\mathcal{O}(n \log(n))$ operations are required for this MVP. The iteration scheme of the FFT+RSM smoother reads:

$$\mathbf{p}^{k+1} = (I - \bar{M}A)\mathbf{p}^k + \bar{M}\mathbf{u}. \quad (3.20)$$

The first term at the right-hand side involves 2 MVPs and the second involves 1 MVP, which can be computed in advance on the target grid. So in one FFT+RSM smoothing step, 2 MVPs are required on the target grid, and 3 MVPs on the coarser grids. The complexity $\mathcal{O}(n \log(n))$ is due to the Toeplitz structure.

3.5 Numerical Results

In this section, we solve three model problems, including MP1 presented in Section 3.2.5. MP2 is concerned with a different material as well as a different right-hand side of $A\mathbf{p} = \mathbf{u} = -\delta t \mathbf{w}$. It resembles a 3D case, with spatially varying rigid shift ($\delta t \mathbf{w}$). This is due to the so-called spin creepage, which results from relative rotation of the cylinder around the normal to the contact area [4]. Solving MP2 serves as a preparation for the work on 3D problems in Chapter 4.

MP3 has the same setting as MP1, except that the plane is not a smooth but a wavy surface. It resembles a rough surface contact problem, which has an

irregular domain as the contact areas. This type of problems stems from railway applications [11, 52]. In MP3, the true contact area consists of two strips, which destroys the Toeplitz structure of the resulting system, and is one of the reasons why we do not consider existing direct Toeplitz solvers for our contact problems.

Summarizing:

- MP1: the rigid shift $(\delta t \mathbf{w})$ is constant.
- MP2: the contacting bodies are rubber with shear modulus and Poisson's ratio being $G = 0.3 \text{ N/mm}^2$ and $\mu = 0.49$, respectively. The rigid shift has a linear relation with position: $(\delta t \mathbf{w}) = -\mathbf{x}$.
- MP3: the contact area is $[-50, 50] \text{ mm}$ in the y -direction, but consists of two intervals in x -direction, i.e. $[-4, -2] \cup [0, 4] \text{ mm}$, due to the wavy surface.

We first solve MP1 and MP2. Since MP3 loses the Toeplitz structure, it is discussed separately at the end of this section.

Our solver is the multigrid method with the FFT smoother, enhanced by either SD or RSM. The initial guess is chosen randomly. The iterations terminate when $\frac{\|\mathbf{u} - A\mathbf{p}^k\|_{rms}}{\|\mathbf{u}\|_{rms}} \leq 10^{-8}$, where we define the norm as $\|\mathbf{x}\|_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}$. The coarsest grid is chosen to be a 2×1 grid. To investigate the performance, we use the convergence factor as a measure. It is defined as [88]: $\hat{q}^k = \sqrt[k]{\frac{\|\mathbf{d}^k\|_{rms}}{\|\mathbf{d}^0\|_{rms}}}$, where \mathbf{d}^k is the defect after the k th iteration.

3.5.1 Results by MG with the FFT+SD Smoother

The numbers of iterations and the convergence factors by MG with the FFT+SD smoother for the first two model problems are shown in Tables 3.2 and 3.3, respectively. A V(1,1)-cycle does not converge with the components chosen, but V-cycles with one smoothing step perform well. Usually, a V(1,1)-cycle converges faster than a V-cycle with one smoothing step, which is not the case here. One reason is the inefficient approximation of some eigenvectors of \bar{A} by piecewise constant deflation vectors, that are not removed from the defect and influence the performance of the smoother.

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
V(1,0)	6(0.036)	5(0.018)	6(0.037)	7(0.064)	7(0.063)
V(0,1)	5(0.023)	5(0.023)	5(0.023)	6(0.040)	7(0.068)

Table 3.2: MG with the FFT+SD smoother for MP1: the numbers of iterations with the convergence factors in brackets for different discretizations.

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
V(1,0)	11(0.149)	12(0.162)	12(0.177)	13(0.196)	14(0.203)
V(0,1)	10(0.110)	10(0.109)	10(0.109)	10(0.110)	10(0.114)

Table 3.3: *MG with the FFT+SD smoother for MP2: the numbers of iterations with the convergence factors in brackets for different discretizations.*

3.5.2 Results by MG with the FFT+RSM Smoother

The results by MG with the FFT+RSM smoother for the first two model problems are presented in Tables 3.4 and 3.5, respectively. As can be seen, all three V-cycles converge fast and show mesh-independence. Moreover, the V(1,1)-cycle requires almost half of the iterations that are needed by the other two V-cycles. Also for other right-hand side functions very similar results were obtained. The FFT+RSM smoother works efficiently in the basic MG framework defined in Section 3.3.3.

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
V(1,1)	3(0.002)	3(0.002)	3(0.001)	3(0.001)	3(0.001)
V(1,0)	6(0.039)	5(0.024)	5(0.022)	5(0.019)	4(0.009)
V(0,1)	5(0.023)	5(0.018)	5(0.008)	4(0.007)	4(0.007)

Table 3.4: *MG with the FFT+RSM smoother for MP1: the numbers of iterations with the convergence factors in brackets for different discretizations.*

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
V(1,1)	4(0.004)	4(0.004)	4(0.003)	3(0.001)	3(0.001)
V(1,0)	7(0.049)	7(0.044)	6(0.031)	6(0.028)	6(0.025)
V(0,1)	7(0.043)	6(0.029)	6(0.026)	5(0.015)	5(0.014)

Table 3.5: *MG with the FFT+RSM smoother for MP2: the numbers of iterations with the convergence factors in brackets for different discretizations.*

Regarding the computational cost, we consider the number of work units, that are defined as *the work of one MVP on the finest grid*. (All MVPs are done by $\mathcal{O}(n \log(n))$ operations using FFTs due to the Toeplitz structure.) The work is halved on a coarser grid, due to the standard coarsening in Fig. 3.6. In one V(1,0)- or V(0,1)-cycle on the finest grid, 3 MVPs are required, including 2 for FFT+RSM smoothing, and 1 for defect computation. Similarly, 5 MVPs are involved in one V(1,1)-cycle. One more MVP is required on coarser grid levels. The total number of work units is obtained by multiplying the cost in one cycle by the number of cycles.

We take the $2^{20} \times 1$ problem as an example, which consists of more than one million unknowns. Table 3.6 shows the (rounded) number of work units by MG with the FFT+RSM smoother for the MP1 and MP2. The V(0,1)-cycle

appears favorable. Codes are written in Matlab 8.0 (R2012b) on Windows 7 64-bit platform with Intel(R) Core(TM)2 Duo CPU, E8500@3.16GHz, 3.17GHz. The CPU time by V(0,1)-cycle for these two problems is around 20 seconds.

Multigrid-cycle	MP1	MP2
V(1,1)	37	37
V(1,0)	29	43
V(0,1)	29	36

Table 3.6: *MG with the FFT+RSM smoother: the (rounded) number of work units for a $2^{20} \times 1$ problem in MP1 and MP2.*

3.5.3 Results by the FFT+RSM Used as a Solver

The iteration matrix R_R had an eigenvector which was not smoothly varying near the boundary, as shown in Fig. 3.11(c). We thus expect that standard coarse grid correction does not help significantly for convergence acceleration. Here, we also apply the FFT+RSM technique as an iterative solver *on only one grid*. The results are given in Table 3.7 for the two tests, showing stable iteration numbers as the problem size gets larger.

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
MP1	28(0.511)	28(0.513)	28(0.514)	28(0.515)	29(0.519)
MP2	34(0.537)	34(0.538)	34(0.538)	33(0.535)	33(0.535)

Table 3.7: *The FFT+RSM solver: the number of iterations with the convergence factors in brackets for different discretizations.*

Regarding the computational cost, we again take the $2^{20} \times 1$ problem as an example. Since 2 MVPs are involved in each iteration, the numbers of work units are 58, 66 for MP1 and MP2, respectively. Compared to the numbers in Table 3.6 using MG with the FFT+RSM smoother, we see that the MG is favorable.

3.5.4 Model Problem 3

The contact area in Model Problem 3 (MP3) consists of two strips, $[-4, -2]$ and $[0, 4]$ in the x -direction, and $[-50, 50]$ in the y -direction. We define the *potential contact area* as $[-4, 4] \cup [-50, 50]$, which contains these two strips. The influence coefficient matrix A on this potential contact area is still of Toeplitz structure. The reduced coefficient matrix A_r for these two strips is obtained by removing from A the rows and columns corresponding to the elements that are not in these two strips. Hence, A_r is not a Toeplitz matrix anymore. However, it is still a dense and SPD matrix.

To benefit from the FFT technique, we still employ a Toeplitz matrix A when we implement matrix-vector products $\mathbf{u}_r = A_r \mathbf{x}$. This is done as explained in [100]. First, we put the entries of \mathbf{x} that are not in contact to zero, and apply the FFT to compute $A\mathbf{x} = \mathbf{u}$. Then, we set the entries of \mathbf{u} that are not in the two contact strips to zero, which leads to the desired result \mathbf{u}_r .

Since we lost the Toeplitz structure, the RSM approach requires the following adaptation:

$$\bar{M}_{imp} = M_r - \text{diag}(\text{sum}(M_r) - \gamma), \quad (3.21)$$

where, $\gamma = \min[\text{sum}(M)]$ is again the minimal row sum of the original preconditioner M which is constructed by the FFT approach on the potential contact area. The matrix M_r is obtained from M in the same way as we got A_r from A . The resulting method is still $\mathcal{O}(n \log(n))$.

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
V(1,1)	4(0.005)	3(0.002)	3(0.001)	3(0.001)	3(0.001)
V(1,0)	6(0.041)	6(0.037)	5(0.023)	5(0.020)	4(0.010)
V(0,1)	6(0.035)	5(0.022)	5(0.019)	4(0.009)	4(0.008)

Table 3.8: *MG with the FFT+RSM smoother for MP3: the numbers of iterations with the convergence factors in brackets for different discretizations.*

Discretization	$2^{16} \times 1$	$2^{17} \times 1$	$2^{18} \times 1$	$2^{19} \times 1$	$2^{20} \times 1$
	36(0.598)	36(0.598)	36(0.598)	37(0.600)	37(0.602)

Table 3.9: *FFT+RSM solver for MP3: the iteration numbers with convergence factors in brackets for different discretizations.*

Table 3.8 presents the results by MG with the FFT+RSM smoother. The three V-cycles converge rapidly, and show mesh-independent multigrid convergence. Table 3.9 gives the results by the FFT+RSM solver, which also indicates highly satisfactory convergence and stable iteration numbers on different grids.

Regarding the computational costs, we again take $2^{20} \times 1$ as the example. The numbers of work units for V(1,1)-, V(1,0)- and V(0,1)-cycles are 37, 29 and 29, respectively. The CPU time for the V(0,1)-cycle is around 21 seconds. The FFT+RSM solver requires 78 work units, and 58 seconds CPU time. MG with the FFT+RSM smoother compares favorably.

3.6 Conclusion

In this chapter we investigated an FFT smoother in a multigrid method, to solve a 2D frictional contact problem which is a no-slip case. The discretization of the integral equation gives rise to a linear system with a coefficient matrix being dense, SPD and of Toeplitz structure. A Toeplitz preconditioner M is used

in the smoother, whose construction and implementation are relatively cheap by the use of the FFT technique. This smoother reduces many components of the error, but enlarges several smooth components. In order to overcome this drawback, two techniques have been studied: subdomain deflation (SD) and row sum modification (RSM). MG with the FFT+RSM smoother shows a rapid convergence and mesh-independence in the numerical tests, particularly in MP3 where the Toeplitz structure is lost. In addition, the FFT+RSM smoother is applied as a stand-alone solver, which also converges well. Moreover, both solvers are $\mathcal{O}(n \log(n))$ methods.

It may be interesting to extend this FFT smoother within MG to 3D contact problems, where an BTTB matrix is to be solved. An FFT preconditioner has been proposed by Vollebregt [100], and may be incorporated as an FFT smoother. More attention is required to deal with the increasing error components, such as RSM.

Chapter 4

A Fast Nonlinear Conjugate Gradient-based Method for 3D Concentrated Frictional Contact Problems

In the previous chapter, we solved a 2D frictional contact problem without slip. In this chapter we focus on 3D concentrated frictional shift and rolling contact problems with dry Coulomb friction, where a nonlinear constrained optimization problem arises. For such problems, a fast numerical solver is presented, which combines an active set strategy with a nonlinear conjugate gradient method. One novelty is to consider the tractions of each slip element in a polar coordinate system, using azimuth angles as variables instead of conventional traction variables. The new variables are scaled by the diagonal of the underlying Jacobian. The fast Fourier transform (FFT) technique accelerates all matrix-vector products encountered, exploiting the matrix' Toeplitz structure. Numerical tests demonstrate a significant reduction of the computational time compared to existing solvers for concentrated contact problems.¹

4.1 Introduction

We have already described the physical phenomenon of frictional contact in Section 3.1.1, where two contact conditions arose. In the adhesion area, the magnitude of the frictional tractions does not exceed the traction bound, and no slip occurs. In the slip area, it attains the traction bound and the resulting slip points to the opposite direction of the tractions. The traction bound can deter-

¹The contents of this chapter have been published in paper [115]: J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee. A fast nonlinear conjugate gradient based method for 3D concentrated frictional contact problems. *Journal of Computational Physics*, 288:86-100, 2015.

mined by a frictional law, such as the Coulomb's law. These conditions bring in inequalities and nonlinearity in 3D problems.

The solution strategies and algorithms for the frictional contact problem have been discussed in Sections 3.1.2 and 3.1.3, respectively. As mentioned there, the boundary element method (BEM) is particularly well fitted for the concentrated contact problem. It employs a Green's function for the response of material due to a load, typically using an elastic half-space. Existing algorithms based on the BEM with the half-space approach include the TANG algorithm [50] and the ConvexGS algorithm [95]. The complexity of these two methods are $\mathcal{O}(n^{3.5})$ and $\mathcal{O}(n^{2.3})$ [96], respectively, with n the number of unknowns in a contact plane.

The motivation of the work in this chapter is to develop a fast solver for the 3D frictional contact problem, especially for the so-called shift problem, e.g. the Cattaneo shift [13]. It is a transient contact problem, and concerns one object pressed onto another, and shifted tangentially. It plays an important role in the study of rolling contact problems, since these can be modeled generally as a sequence of shift problems. In this chapter we consider the tangential problem, with the solutions from the normal problem already available. It can be easily incorporated into the Panagiotopoulos process [69, 51], to deal with more complicated contact.

Our new method contributes to the BEM solvers. It is called "TangCG", since it searches for the tangential tractions and is based on the nonlinear conjugate gradient method. The constraint that the magnitude of tractions should be equal to the traction bound on each slip element inspires to place the traction vector at a circle in a polar coordinate system, with the radius being the traction bound. We use azimuth angles as variables in the slip area, which is a significant difference from conventional solvers.

The TangCG algorithm is a so-called bound-constrained conjugate gradient (BCCG) method, which was proposed for linear complementarity problems in normal contact [99]. The BCCG method uses an active set strategy, and employs the conjugate gradient (CG) method for the governing linear system. Differently, the governing system in frictional contact problems is mainly nonlinear, hence, we employ a nonlinear conjugate gradient (NLCG) method [76]. The TangCG algorithm is combined with a diagonal scaling preconditioner.

The most time consuming part of the solution procedure relates to a Boussinesq integral [44, 51], e.g. Eq. (3.3), which gives the relation between tractions and deformation. It results in a dense, symmetric and positive definite coefficient matrix. This matrix is block Toeplitz matrix with Toeplitz blocks (BTTB). Such structure can be exploited by the fast Fourier transform (FFT), which has been applied in the field of contact problems, e.g. [84, 100, 16]. The complexity is reduced by the FFT to $\mathcal{O}(n \log n)$ with n the number of unknowns.

Section 4.2 gives the mathematical formulation of the 3D frictional contact problem and explains existing solvers. The TangCG method is illustrated and numerically analyzed in Section 4.3. Section 4.4 shows the numerical results of our new method, and also compares the efficiency with an existing method.

Section 4.5 concludes.

4.2 Formulation of the 3D Frictional Contact Problem

This section introduces the physical problem and the mathematical model of frictional contact. For more details, we refer to [51]. Subsequently, the discretization and existing methods for the model are given.

4.2.1 Notations

The tractions on element I are defined by $\mathbf{p}_I = [p_{Ix}, p_{Iy}]$, including two traction components in the x - and y -directions. Without the subscript I , e.g. \mathbf{u} is defined by $\mathbf{u} = [\mathbf{u}_x, \mathbf{u}_y]$. Here, $\mathbf{u}_x = [u_{1x}, u_{2x}, \dots]$ indicates the x -direction deformation of all elements, and a similar definition applies to \mathbf{u}_y . Vectors $\bar{\mathbf{s}}_I$ and \mathbf{s}_I are slip velocity and distance, respectively, on element I . Superscript k is an iteration index.

4.2.2 Mathematical Model of the 3D Frictional Contact Problem

The normal problem can be solved by Hertz theory [36] if the contacting surfaces are smooth and quadratically, or by a numerical method (e.g. [101]) if the profile is non-Hertzian. The solution pressure is compressive in the contact area, and vanishes outside of it.

Tangential tractions result from the overall motion of the two bodies, causing a tendency of the surfaces to slip with respect to each other, called “rigid slip”. In rolling contact it is often characterized via “creepage”, which is the average relative velocity between the surfaces, and is given by the difference between the forward rolling velocity and the circumferential velocity.

The formulation concerns three aspects, i.e., slip, elastic deformation, and the friction between the two contacting bodies. First of all, the relation between slip, rigid slip and deformation is defined by [101]:

$$\bar{\mathbf{s}} := \bar{\mathbf{w}} + \frac{1}{V} \frac{D\mathbf{u}}{Dt}, \quad (4.1)$$

where, $\bar{\mathbf{s}}$ is the slip velocity of two opposing particles on the contacting surfaces with respect to each other. It is the summation of the relative rigid slip velocity $\bar{\mathbf{w}}$ and the change of deformation $\frac{D\mathbf{u}}{Dt}$ scaled by the rolling velocity V .

Concerning the effect of friction, the contact area C is divided into an adhesion area H and a slip area S , according to the contact conditions:

$$\begin{cases} \text{In adhesion area } H(t) : & \|\bar{\mathbf{s}}(\mathbf{x}, t)\| = 0, \quad \|\mathbf{p}(\mathbf{x}, t)\| \leq g(\mathbf{x}, t), \\ \text{In slip area } S(t) : & \|\bar{\mathbf{s}}(\mathbf{x}, t)\| > 0, \quad \mathbf{p}(\mathbf{x}, t) = -g(\mathbf{x}, t) \frac{\bar{\mathbf{s}}(\mathbf{x}, t)}{\|\bar{\mathbf{s}}(\mathbf{x}, t)\|}. \end{cases} \quad (4.2)$$

The norm $\|\cdot\|$ is the Euclidean norm. These conditions state that the magnitude of the tangential tractions at position \mathbf{x} cannot exceed the corresponding

traction bound g in the adhesion area. When the traction bound is reached, local slip occurs with its direction opposite to the tractions.

The traction bound g in this model is obtained by applying the Coulomb's frictional law locally

$$g(\mathbf{x}, t) = \mu p_z(\mathbf{x}, t), \quad (4.3)$$

where μ is a constant friction coefficient, and p_z is the normal pressure.

The deformation \mathbf{u} in Eq. (4.1) is obtained by the *half-space* approach, which was discussed in Section 2.2.3. The two contacting bodies are approximated by two semi-infinite half-space, bounded by the contacting plane. The relation between the deformation \mathbf{u} and traction \mathbf{p} is given by:

$$\mathbf{u}(\mathbf{x}, t) = \int_{C(t)} A(\mathbf{x}, \mathbf{x}') \mathbf{p}(\mathbf{x}', t) dx'. \quad (4.4)$$

Similar as Eq. (2.6), the kernel function $A(\mathbf{x}, \mathbf{x}')$ also possesses the property of space-invariance, i.e.:

$$A(\mathbf{x}, \mathbf{x}') = A(\mathbf{x} - \mathbf{x}'). \quad (4.5)$$

4.2.3 Discretization

The discretization focuses on a rectangular potential contact area, which contains the true contact area. A rectangular mesh is placed on this 2D region, with $n = n_x \times n_y$ elements of size $\delta x \times \delta y$. The coordinates of the center of the element I are denoted by $[x_I, y_I]$.

Tractions \mathbf{p} in the surface integral (4.4) are approximated by element-wise constant functions. The cell-centered discretization of the integral (4.4) yields, for each contacting element I :

$$u_{Ix} = \sum_{J=1}^n (A_{IJ}^{xx} p_{Jx} + A_{IJ}^{xy} p_{Jy}), \quad u_{Iy} = \sum_{J=1}^n (A_{IJ}^{yx} p_{Jx} + A_{IJ}^{yy} p_{Jy}), \quad (4.6)$$

where u_{Ix} is the deformation in x -direction of element I . Influence coefficient A_{IJ}^{xy} is computed by integrating (4.4) over a single element J , with respect to an observation point at the center of element I (see [51] for the detailed formulas).

Writing (4.6) in matrix form, we obtain:

$$\mathbf{u} = A\mathbf{p}, \quad A \in \mathbb{R}^{2n \times 2n}, \quad \mathbf{u}, \mathbf{p} \in \mathbb{R}^{2n}, \quad (4.7)$$

where, the global coefficient matrix A has 2×2 blocks:

$$A = \begin{pmatrix} A^{xx} & A^{xy} \\ A^{yx} & A^{yy} \end{pmatrix}, \quad (4.8)$$

with each block $A^{xx}, A^{xy}, A^{yx}, A^{yy} \in \mathbb{R}^{n \times n}$. Blocks A^{xx} and A^{yy} are symmetric and positive definite (SPD). Blocks A^{xy} and A^{yx} are not symmetric and they

satisfy $A^{xy} = (A^{yx})^T$. Moreover, due to property (4.5), these blocks are BTTB if the contact area is rectangular². The whole matrix A is SPD.

To discretize slip Eq. (4.1), a sequence of time instances is used, with length $\delta t = t - t'$, where t and t' are the current and previous time points, respectively. Applying a backward Euler method for the derivative, we obtain:

$$\bar{\mathbf{s}}_I = \bar{\mathbf{w}}_I + \frac{\mathbf{u}_I - \mathbf{u}'_I}{\delta t V}. \quad (4.9)$$

Define the traversed distance per time step as $\delta q = V \cdot \delta t$, which is called time step for short. The shift on one element $\mathbf{s}_I := \delta q \cdot \bar{\mathbf{s}}_I$ represents the aggregated slipped distance over δq . Then, discretized Eq. (4.9) reads:

$$\mathbf{s}_I = \mathbf{w}_I + (\mathbf{u}_I - \mathbf{u}'_I), \quad 1 \leq I \leq N, \quad (4.10)$$

where rigid shift \mathbf{w}_I is defined as:

$$\mathbf{w}_I = [\xi - \phi y_I, \eta + \phi x_I]. \quad (4.11)$$

Here ξ , η are longitudinal and lateral shifts, respectively, and ϕ is the rotation shift [101]. For simplicity we restrict to a one-step shift, where the deformation at a previous time \mathbf{u}'_I vanishes. Then Eq. (4.10) yields:

$$\mathbf{s}_I = \mathbf{w}_I + \mathbf{u}_I. \quad (4.12)$$

After discretization, condition (4.2) becomes:

$$\begin{cases} \text{In the adhesion area } H : & \|\mathbf{s}_I\| = 0, \quad \|\mathbf{p}_I\| \leq g_I, \\ \text{In the slip area } S : & \|\mathbf{s}_I\| > 0, \quad \mathbf{p}_I = -g_I \cdot \mathbf{s}_I / \|\mathbf{s}_I\|. \end{cases} \quad (4.13)$$

In the solution procedure, these two conditions are equivalent to solving the following equations:

- For adhesion element $I \in H$:

$$\mathbf{s}_I = \mathbf{0}. \quad (4.14)$$

- For slip element $I \in S$:

$$\begin{cases} p_{Ix}^2 + p_{Iy}^2 = g_I^2, & (4.15) \\ p_{Ix} s_{Iy} - p_{Iy} s_{Ix} = 0, & (4.16) \end{cases}$$

where Eq. (4.16) is derived from the fact that slip \mathbf{s}_I and traction \mathbf{p}_I are in opposite directions, and hence, $\mathbf{p}_I^\perp \perp \mathbf{s}_I$ with $\mathbf{p}_I^\perp = [-p_{Iy}, p_{Ix}]$. Eqs. (4.15) and (4.16) give rise to nonlinearity.

²Otherwise, they are not BTTB anymore. An approach of applying FFTs in this case is given in [100].

The frictional contact problem now consists of Eqs. (4.7), (4.12) and (4.13). It was shown that this problem can also be considered as a variational inequality [51]:

$$\min_{\mathbf{p}} \frac{1}{2} \mathbf{p}^T \mathbf{A} \mathbf{p} + \mathbf{w}^T \mathbf{p}, \quad \text{s.t.} \quad \|\mathbf{p}_I\| \leq g_I \text{ for } I \in C. \quad (4.17)$$

This is a nonlinear convex optimization problem [8], for which the corresponding Karush–Kuhn–Tucker (KKT) conditions [57] provide both sufficient and necessary conditions for the existence of a unique solution. The KKT conditions also result in Eqs. (4.7), (4.12), and (4.13).

4.2.4 Existing Algorithms

Although many existing solution strategies for convex optimization problems may be applied in principle, only two approaches appear to be used in practice for wheel-rail contact problems. One is Kalker’s “TANG” algorithm [50, 51]. It is an active set method and consists of inner and outer iterations. In each outer iteration it fixes the subdivision of the contact area, which determines the corresponding governing systems (4.14)-(4.16). The resulting solution is used to modify the subdivision according to contact condition (4.13). Newton’s method is applied for the nonlinear system, and the linearized system in every inner iteration is solved by Gauss-elimination.

The limitation of this method shows when the problem size is large. Gauss-elimination, with complexity $\mathcal{O}(n^3)$, is time-consuming and memory intensive.

An alternative is the “ConvexGS” algorithm [95], which is a block Gauss-Seidel iteration. The elements in contact are processed one by one, with the recently updated tractions fixed. The subproblem defined on each element is derived based on the element-wise constraints in the convex optimization formulation (4.17). The corresponding nonlinear system (4.15)-(4.16) for the element is solved by Newton’s method.

However, this method cannot benefit from an implementation using FFTs exploiting the BTTB structure, because the elements are processed one by one so that ConvexGS does not employ matrix-vector products w.r.t. the global matrix.

4.3 New Method: TangCG Algorithm

We present our new algorithm TangCG for frictional contact in this section. Its main components are introduced in Section 4.3.1. Section 4.3.2 illustrates this method based on a small test, for additional insight. Section 4.3.3 gives the resulting algorithm. Moreover, the performance of the inner NLCG iterations is discussed in Section 4.3.4.

4.3.1 Main Components of the TangCG Algorithm

The TangCG algorithm consists of six main components, shown in Fig. 4.1, where the adjacent pieces are related to each other.

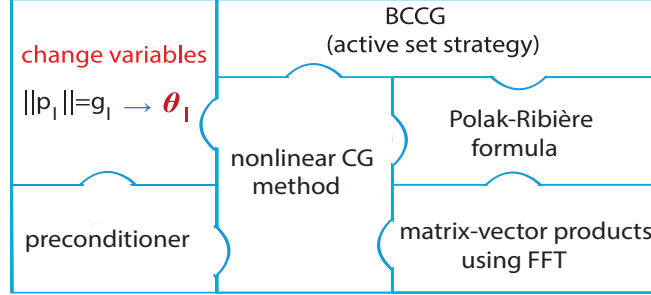


Figure 4.1: *The main components of the TangCG method. The adjacent pieces are related to each other.*

The TangCG method adopts the framework of an active set strategy from the BCCG method [99]. In each outer iteration, the adhesion and slip areas are fixed and systems (4.14)-(4.16) are solved approximately; then we modify the subdivision of the contact area, according to condition (4.13). A reduced system of equations is considered in the inner iterations, where the constraints for elements in the slip area are automatically enforced.

The most pronounced component of the TangCG algorithm is as follows. Different from conventional methods that use traction variables, we change variables in the slip area, where the magnitude of tractions should equal the traction bound. On each slip element I , traction \mathbf{p}_I lies at a circle with radius g_I . Therefore, we can define \mathbf{p}_I in a polar coordinate system with its two components p_{Ix} and p_{Iy} along the horizontal and vertical axes, respectively. Since g_I is known, \mathbf{p}_I can be uniquely determined by the azimuth angle θ_I as: $\mathbf{p}_I = g_I \cdot [\cos(\theta_I), \sin(\theta_I)]$. Thus, Eq. (4.15) is automatically satisfied. Only Eq. (4.16) is to be solved on each slip element.

For the resulting nonlinear system, TangCG employs a nonlinear conjugate gradient (NLCG) method. In each NLCG iteration, the system is linearized, and a CG process is applied. Concerning the construction of the search directions, the conjugacy may be lost as the iterations proceed. The Polak-Ribière formula [72] is in our case the preferable choice compared to the Fletcher-Reeves formula [25] to improve the search directions. The former sets the search direction automatically towards the steepest descent direction, when the search direction which is based on the previous defect is almost orthogonal to this defect. The latter gives rise to small changes in defects and keeps the problematic situation in the subsequent iterations. This slows down the progress and requires to restart the solution process [75].

Preconditioning is done using the diagonal of the reduced system's Jacobian matrix. This preconditioner is necessary for the TangCG method, which employs different variables in the adhesion and slip areas. These quantities are brought to

the same measure by this preconditioner. Without preconditioner the solver may largely disregard the defect in either the adhesion or the slip area, depending on the material parameters.

Since the TangCG algorithm avoids generating Jacobian matrices, the matrix-vector products (MVPs) encountered only depend on the influence coefficient matrix A in (4.7), which includes submatrices being block Toeplitz with Toeplitz blocks (BTTB). The fast Fourier transform (FFT) technique is applied to accelerate such MVPs with complexity $\mathcal{O}(n \log(n))$.

4.3.2 Illustration of the TangCG Algorithm

This subsection aims at showing how the pieces given in Fig. 4.1 are combined to result in an efficient algorithm. A very small test case is considered. It consists of only two elements in contact, with elements 1 and 2 in the adhesion and slip areas, respectively. The traction bounds $g_I, I = 1, 2$ are given.

Change of Variables in Slip Area

Let's start with a prominent component of the method, which is the change of variables in the slip area [98]. The following discussion is for the k th TangCG iteration, with the tractions \mathbf{p}^{k-1} and slip \mathbf{s}^{k-1} known from the previous iteration. For clarity of the equations, we avoid the iteration indices $k-1$ and k where possible.

When the slip area S is empty, i.e. in the full adhesion case, the governing system $\mathbf{s} = A\mathbf{p} + \mathbf{w} = \mathbf{0}$ is linear. When the slip area S is not empty, it becomes nonlinear. Considering the two-element test, the governing system reads:

$$\mathbf{F}_f = \mathbf{0}, \quad (4.18)$$

with

$$F_1 = s_{1x}, \quad (4.19)$$

$$F_2 = s_{1y}, \quad (4.20)$$

$$F_3 = p_{2x}^2 + p_{2y}^2 - g_2^2, \quad (4.21)$$

$$F_4 = p_{2x}s_{2y} - p_{2y}s_{2x}, \quad (4.22)$$

where the first two equations correspond to adhesion element 1. Eqs. (4.21) and (4.22) for slip element 2 bring in nonlinearity, and TangCG applies an NLCG method. The nonlinear system (4.18) is linearized by a truncated Taylor expansion, which yields:

$$J_f \cdot \delta \mathbf{p}_f = -\mathbf{F}_f(\mathbf{p}^{k-1}), \quad (4.23)$$

with Jacobian matrix $J_f = \nabla \mathbf{F}_f(\mathbf{p}^{k-1})$ having the following form:

$$J_f(\mathbf{p}^{k-1}) = \begin{pmatrix} A_{11}^{xx} & A_{11}^{xy} & A_{12}^{xx} & A_{12}^{xy} \\ A_{11}^{yx} & A_{11}^{yy} & A_{12}^{yx} & A_{12}^{yy} \\ 0 & 0 & p_{2x} & p_{2y} \\ a_1 & a_2 & a_3 & a_4 \end{pmatrix}, \quad (4.24)$$

and:

$$\begin{aligned} a_1 &= p_{2x}A_{21}^{yx} - p_{2y}A_{21}^{xx}, & a_2 &= p_{2x}A_{21}^{yy} - p_{2y}A_{21}^{xy}, \\ a_3 &= p_{2x}A_{22}^{yx} - p_{2y}A_{22}^{xx} + s_{2y}, & a_4 &= p_{2x}A_{22}^{yy} - p_{2y}A_{22}^{xy} - s_{2x}. \end{aligned}$$

This Jacobian corresponds to variables $\delta\mathbf{p}_f = [\delta p_{1x}, \delta p_{1y}, \delta p_{2x}, \delta p_{2y}]^T$. It is not symmetric and thus the CG method cannot be applied.

A change of variables in the slip area can overcome this difficulty. Since traction \mathbf{p}_2 is placed at a circle of radius g_2 (e.g. see Fig. 4.2(a)), the nonlinear Eq. (4.21) is satisfied automatically and can be eliminated. The linearized version of this equation gives:

$$\begin{aligned} p_{2x}\delta p_{2x} + p_{2y}\delta p_{2y} &= -(p_{2x}^2 + p_{2y}^2 - g_2^2) \\ &= 0 \text{ (since assuming } \|\mathbf{p}_2\| = g_2), \end{aligned} \quad (4.25)$$

which indicates $\delta\mathbf{p}_2 \perp \mathbf{p}_2$. It further yields:

$$\delta p_{2x} = -p_{2y}\delta\theta_2, \quad \delta p_{2y} = p_{2x}\delta\theta_2. \quad (4.26)$$

This relation satisfies $\|\delta\mathbf{p}_2\| = \|\mathbf{p}_2\|\delta\theta_2$, which is the definition of the arc length corresponding to the angle $\delta\theta_2$. This arc length is approximated by vector $\delta\mathbf{p}_2$ which is orthogonal to \mathbf{p}_2 .

Reduced System

Eliminating Eq. (4.21) yields a reduced system written as:

$$\mathbf{F}_r = \mathbf{0}, \quad (4.27)$$

and the linearized system with relation (4.26) is given by:

$$J_r\delta\mathbf{p}_r = -\mathbf{F}_r, \quad (4.28)$$

with a 3×3 Jacobian matrix:

$$J_r = \begin{pmatrix} A_{11}^{xx} & A_{11}^{xy} & a_1 \\ A_{11}^{yx} & A_{11}^{yy} & a_2 \\ a_1 & a_2 & a_4p_{2x} - a_3p_{2y} \end{pmatrix}. \quad (4.29)$$

The corresponding variables are $\delta\mathbf{p}_r = [\delta p_{1x}, \delta p_{1y}, \delta\theta_2]^T$ satisfying

$$\delta\mathbf{p}_f = T\delta\mathbf{p}_r, \quad (4.30)$$

with transformation matrix T defined by:

$$T = \begin{pmatrix} 1 & & \\ & 1 & \\ & & -p_{2y} \\ & & & p_{2x} \end{pmatrix}, \quad (4.31)$$

whose transpose can convert $\delta\mathbf{p}_f$ to $\delta\mathbf{p}_r$. The reduced Jacobian matrix (4.29) is symmetric and positive definite (SPD), so we can apply CG to it.

Solution Method for the Reduced System

The reduced system (4.27) is solved by an NLCG method. In each of its iterations, the CG method is applied to the linearized system (4.28), with initial iterate $\delta \mathbf{p}_r^0 = \mathbf{0}$.

First of all, we compute the defect of linearized system (4.28) by:

$$\begin{aligned} \mathbf{d}_a &= -\mathbf{F}_r - J_r \cdot \delta \mathbf{p}^0, \\ &= -\mathbf{F}_r, \quad (\text{because } \delta \mathbf{p}_r^0 = \mathbf{0}), \end{aligned} \quad (4.32)$$

which is equal to the defect of the nonlinear system (4.27)³. The search direction \mathbf{v}_a adopts the steepest descent direction when the area subdivision changes. Otherwise, it employs the Polak-Ribière formula. The line search aims at determining a step length α , such that the defect resulting from the updated iterate is orthogonal to the current search direction \mathbf{v}_a . This yields a formula for α given by:

$$\alpha = \frac{(\mathbf{d}_a, \mathbf{v}_a)}{(\mathbf{v}_a, \mathbf{q}_a)}, \quad (4.33)$$

where \mathbf{q}_a is the change of the defect caused by \mathbf{v}_a , defined by:

$$\mathbf{q}_a = J_r \mathbf{v}_a. \quad (4.34)$$

Subsequently, the iterate of linear system (4.28) is corrected by $\delta \mathbf{p}_r^1 = \alpha \mathbf{v}_a$. This further gives:

$$T \delta \mathbf{p}_r^1 = \alpha T \mathbf{v}_a \quad \Rightarrow \quad \delta \mathbf{p}_f^1 = \alpha \cdot \mathbf{v}_s, \quad (4.35)$$

where \mathbf{v}_s is the auxiliary search direction⁴, defined by:

$$\mathbf{v}_s = T \mathbf{v}_a. \quad (4.36)$$

The new traction iterate is given by:

$$\mathbf{p}^k = \mathbf{p}^{k-1} + \delta \mathbf{p}_f^1 = \mathbf{p}^{k-1} + \alpha \cdot \mathbf{v}_s^k. \quad (4.37)$$

Moreover, the tractions on each slip element need to be projected onto the corresponding circle. This represents one NLCG iteration.

Matrix-vector Products

To compute \mathbf{q}_a in (4.34), TangCG does not directly implement this multiplication of Jacobian matrix J_r and vector \mathbf{v}_a , but takes a detour, which allows the use of

³Subscript *a* denotes the *primary variables*. On an adhesion element they have two entries for the *x*- and *y*-directions, but we use only one entry on each slip element, carrying the information about angles.

⁴Subscript *s* denotes *auxiliary variables*, that have two entries for the *x*- and *y*-directions on each adhesion and slip element.

FFTs. Vector \mathbf{v}_a is transformed to the auxiliary search direction \mathbf{v}_s by (4.36). Then its influence on slip $\delta\mathbf{s}$ can be computed by

$$\delta\mathbf{s} = A\mathbf{v}_s, \quad (4.38)$$

where the FFT technique can be applied due to the matrix A in (4.8). Eq. (4.34) with relation (4.26) yields:

$$\mathbf{q}_a = \begin{pmatrix} \delta s_{1x} \\ \delta s_{1y} \\ p_{2x}\delta s_{2y} - p_{2y}\delta s_{2x} + s_{2y}v_{2x} - s_{2x}v_{2y} \end{pmatrix}. \quad (4.39)$$

This indicates that on adhesion element 1 we have $\delta\mathbf{s}_1 = \mathbf{q}_{a1}$, and on slip element 2, $\delta\mathbf{s}_2$ also has its influence as seen in the last formula in (4.39).

Active Set Framework from BCCG(K)

An outer iteration is processed after every K inner iterations. It validates the contact conditions, and adjusts the subdivision of the contact area. The adhesion elements with $\|\mathbf{p}_I\| > g_I$ are moved to the slip area, and are projected to $\|\mathbf{p}_I\| = g_I$. Remember that tractions \mathbf{p}_I of each slip element should have opposite directions to the slip \mathbf{s}_I . The elements are moved to the adhesion area if these two vectors lie in the same half plane. Finally, the tolerance of the solution is checked.

Illustration by Angles

Since TangCG uses angle variables in the slip area, it can be explained from the view of angles as well. We define the anti-clockwise angle to be positive. Fig. 4.2 shows the k th TangCG iteration on slip element I . (We drop the superscripts $k-1$ and k in this figure.) In this discussion, we will use six angles $\tau_i, i \in \{1, 2, 3, 4, 5, 6\}$.

In Fig. 4.2(a), the traction \mathbf{p}_I lies at the circle with radius g_I . It is expected that

$$\theta_p = \theta_s + \pi, \quad (4.40)$$

i.e., the azimuth angle of slip \mathbf{s}_I plus π is equal to azimuth angle of \mathbf{p}_I . We notice that $\tau_2 = \theta_s + \pi - \theta_p$ is the defect of Eq. (4.40); τ_2 represents the angle along which \mathbf{p}_I should move, in order to be located opposite to \mathbf{s}_I . However, a change in \mathbf{p}_I also causes a change in \mathbf{s}_I . So τ_2 is not the exact angle that \mathbf{p}_I needs to be changed with.

The angle defect computed by Eq. (4.32) is given by:

$$\begin{aligned} d_{aI} &= -(p_{Ix}s_{Iy} - p_{Iy}s_{Ix}) \\ &= \|-\mathbf{p}_I^\perp\| \|\mathbf{s}_I\| \cos(\tau_1) \\ &= g_I \|\mathbf{s}_I\| \sin(\tau_2) \quad (\text{since } \tau_2 = \pi/2 - \tau_1) \\ &\approx g_I \|\mathbf{s}_I\| \tau_2, \quad \text{if } \tau_2 \rightarrow 0, \end{aligned} \quad (4.41)$$

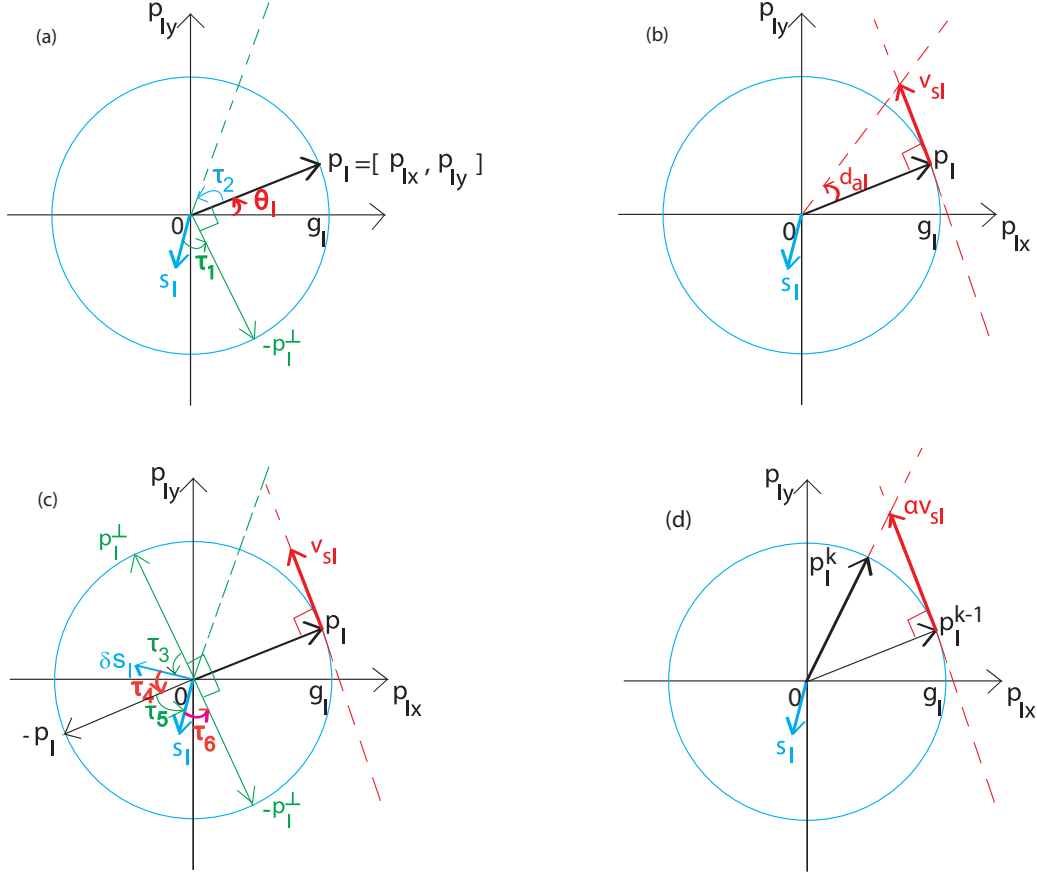


Figure 4.2: First TangCG inner iteration on slip element I . (a) At the beginning, traction \mathbf{p}_I lies at the circle with radius g_I . It should be opposite to slip \mathbf{s}_I . (b) Primary defect d_{aI} (equal to search direction v_{aI}) is computed. The auxiliary search direction \mathbf{v}_{sI} is set to be orthogonal to \mathbf{p}_I , due to linearization. (c) Line search is performed to get a step length $\alpha = \frac{(\mathbf{v}_a, \mathbf{d}_a)}{(\mathbf{v}_a, \mathbf{q}_a)}$, where q_{aI} is obtained from the information of two angles τ_4, τ_6 . (d) Update the traction by $\mathbf{p}_I^{k-1} + \alpha \mathbf{v}_{sI}^k$, which is then projected onto the circle, obtaining \mathbf{p}_I^k .

where $-\mathbf{p}_I^\perp = [p_{Iy}, -p_{Ix}]$ is orthogonal to \mathbf{p}_I , and τ_1 is the angle between $-\mathbf{p}_I^\perp$ and \mathbf{s}_I . Defect d_{aI} can be regarded as the angle τ_2 scaled by $g_I \|\mathbf{s}_I\|$. It gives a preliminary angle by which the traction \mathbf{p}_I^k should move.

In the first iteration, the search direction is equal to d_{aI} . As a result of linearization, the auxiliary search direction \mathbf{v}_{sI} is orthogonal to \mathbf{p}_I , as seen in Fig. 4.2(b). It is a linear approximation of the arc length, corresponding to the angle d_{aI} .

By the line search, the TangCG method computes the change of slip δs_I , as

shown in Fig. 4.2(c). It calculates the change of angle defect \mathbf{q}_a as in (4.39):

$$\begin{aligned}
q_{aI} &= p_{Ix} \cdot \delta s_{Iy} - p_{Iy} \cdot \delta s_{Ix} + s_{Iy} \cdot v_{Ix} - s_{Ix} \cdot v_{Iy} \\
&= \|\mathbf{p}_I^\perp\| \|\delta \mathbf{s}_I\| \cos(\tau_3) + \|\mathbf{v}_I^\perp\| \|\mathbf{s}_I\| \cos(\tau_5) \\
&= g_I \|\delta \mathbf{s}_I\| \sin(\tau_4) + \|\mathbf{v}_I^\perp\| \|\mathbf{s}_I\| \sin(\tau_6) \\
&\approx g_I \|\delta \mathbf{s}_I\| \tau_4 + \|\mathbf{v}_I^\perp\| \|\mathbf{s}_I\| \tau_6, \text{ if } \tau_4, \tau_6 \rightarrow 0,
\end{aligned} \tag{4.42}$$

where $\mathbf{v}_I^\perp = [-v_{Iy}, v_{Ix}]$, and $\tau_4 = \pi/2 - \tau_3, \tau_6 = \pi/2 - \tau_5$. The two angles τ_4, τ_6 are presented in Fig. 4.2(c). They, together with the concrete scaling, give the angle change q_{aI} .

Then, the TangCG method computes the step length. The updating is shown in Fig. 4.2(d): after adding $\alpha \mathbf{v}_I$ to \mathbf{p}_I^{k-1} , the result is projected onto the circle again, which yields new iterate \mathbf{p}_I^k .

Preconditioning

The idea of preconditioning for Krylov subspace methods is to reduce the condition number of the coefficient matrix. In the TangCG algorithm, the preconditioner is applied to the reduced Jacobian matrix (4.29) in each NLCG iteration.

We use a diagonal matrix as the preconditioner, which scales the main diagonal of the Jacobian, to result in a matrix with constant diagonal entries. This constant can be chosen to be any diagonal component of Jacobian, corresponding to an adhesion element. The defects in the adhesion and slip areas have different meanings: defects are “slip” in the former and “angle” in the latter area. Hence, the preconditioner scales the quantities to the same measure, e.g. the measure in the adhesion area. Regarding computational time, a preconditioner should not cost much.

In the two element case, preconditioner M for J_r in (4.29) is given by:

$$M = \begin{pmatrix} 1 & & \\ & A_{11}^{xx}/A_{11}^{yy} & \\ & & A_{11}^{xx}/(a_4 p_{2x}^k - a_3 p_{2y}^k) \end{pmatrix}. \tag{4.43}$$

With $M = Q^2$, the preconditioned matrix QJ_rQ has values of A_{11}^{xx} on its main diagonal.

4.3.3 The Resulting Algorithm

We detail the resulting TangCG algorithm with preconditioner M below. (The unpreconditioned algorithm can be obtained by setting $M = I$.)

0. Given normal pressure \mathbf{p}_z and contact area C , we solve for the tangential tractions and the subdivision of the contact area. The initial adhesion and slip areas are set as: $H^0 = C$ and $S^0 = \emptyset$. We use zero initial tractions $\mathbf{p}^0 = \mathbf{0}$.

1. Start iterations $k = 1, 2, \dots$ by computing slip $\mathbf{s}^{k-1} = \mathbf{w} + A\mathbf{p}^{k-1}$.
2. Compute corresponding primary defect \mathbf{d}_a^{k-1} , as Eq. (4.32):

$$\begin{cases} \mathbf{d}_{aI}^{k-1} = -\mathbf{s}_I^{k-1}, & I \in H^{k-1}, \\ d_{aI}^{k-1} = -(p_{Ix}^{k-1}s_{Iy}^{k-1} - p_{Iy}^{k-1}s_{Ix}^{k-1}), & I \in S^{k-1}. \end{cases} \quad (4.44)$$

3. Preconditioning by: $\mathbf{z}_a^{k-1} = M\mathbf{d}_a^{k-1}$.

4. Compute primary search direction \mathbf{v}_a^k :

- 4.1 If the subdivision did not change, i.e. $H^{k-1} = H^{k-2}$, then choose \mathbf{v}_a^k conjugate to \mathbf{v}_a^{k-1} as:

$$\mathbf{v}_a^k = \mathbf{z}_a^{k-1} + \beta^k \cdot \mathbf{v}_a^{k-1}, \quad \text{with } \beta^k = \max(0, \beta_{PR}^k), \quad (4.45)$$

where β_{PR}^k is the Polak-Ribière formula: $\beta_{PR}^k = \frac{(\mathbf{z}_a^{k-1}, \mathbf{d}_a^{k-1} - \mathbf{d}_a^{k-2})}{(\mathbf{z}_a^{k-2}, \mathbf{d}_a^{k-2})}$.

- 4.2 Else, restart CG by using $\mathbf{v}_a^k = \mathbf{z}_a^{k-1}$, the steepest descent direction.

5. Compute auxiliary search direction \mathbf{v}_s^k by Eq. (4.36):

$$\mathbf{v}_{sI}^k = \begin{cases} \mathbf{v}_{aI}^k, & I \in H^{k-1}, \\ v_{aI}^k \cdot [-p_{Iy}^{k-1}, p_{Ix}^{k-1}], & I \in S^{k-1}. \end{cases} \quad (4.46)$$

Here, search direction \mathbf{v}_{sI}^k can be regarded as the desired change of traction $\delta\mathbf{p}^k$. For a slip element, this corresponds to the desired change of angle v_{aI}^k .

6. Compute the change in the slip by Eq. (4.38), i.e.: $\delta\mathbf{s}^k = \mathbf{q}_s^k = A\mathbf{v}_s^k$.
7. Compute the change in primary defect \mathbf{q}_a^k by Eq. (4.39):

$$\begin{cases} \mathbf{q}_{aI}^k = \mathbf{q}_{sI}^k, & I \in H^{k-1}, \\ q_{aI}^k = p_{Ix}^{k-1} \cdot \delta s_{Iy}^k - p_{Iy}^{k-1} \cdot \delta s_{Ix}^k + s_{Iy}^{k-1} \cdot v_{Ix}^k - s_{Ix}^{k-1} \cdot v_{Iy}^k, & I \in S^{k-1}. \end{cases}$$

8. Compute the step length by Eq. (4.33), i.e.: $\alpha^k = \frac{(\mathbf{d}_a^{k-1}, \mathbf{v}_a^k)}{(\mathbf{q}_a^k, \mathbf{v}_a^k)}$.

9. Update tractions: $\tilde{\mathbf{p}}^k = \mathbf{p}^{k-1} + \alpha^k \cdot \mathbf{v}_s^k$ for all elements. Then use:

$$\mathbf{p}_I^k = \begin{cases} \tilde{\mathbf{p}}_I^k, & I \in H^{k-1}, \\ g_I \cdot \tilde{\mathbf{p}}^k / \|\tilde{\mathbf{p}}^k\|, & I \in S^{k-1}. \end{cases} \quad (4.47)$$

10. In the inner iteration, i.e. $\text{mod}(k, K) \neq 0$, the subdivision of the contact area is fixed, i.e., $H^k = H^{k-1}, S^k = S^{k-1}$, go to Step 1.

11. At the end of each K th inner iteration, i.e. $\text{mod}(k, K) = 0$, do the following:

- 11.1 Check elements $I \in H^{k-1}$. If $\|\mathbf{p}_I^k\| > g_I$, then move I to the slip area, and set $\mathbf{p}_I^k = g_I \cdot \mathbf{p}_I^k / \|\mathbf{p}_I^k\|$. This gives intermediate \tilde{H}^k and \tilde{S}^k .

- 11.2 Compute $\mathbf{s}^k = \mathbf{w} + \mathbf{A}\mathbf{p}^k$, and check the elements $I \in \tilde{S}^k$. If $(\mathbf{p}_I^k, \mathbf{s}_I^k) > 0$, then move I to the adhesion area. This results in H^k and S^k .
- 11.3 Check for convergence. If the subdivision of the contact area does not change, and the stopping criterion on the solution is satisfied, then we are done. Else go to Step 1.

Remark 4.1. Steps 1-10 present one inner iteration, where a nonlinear system is solved approximately by K iterations of NLCG. Line search is performed in Steps 5-8. Step 11 checks the contact conditions and convergence, according to the BCCG(K) algorithm. Moreover, Steps 9 and 11.1 ensure that $\|\mathbf{p}_I\| = g_I, I \in S$ is always satisfied.

Remark 4.2. In the above algorithm, there are K inner iterations before checking the conditions. It may happen that the tolerance is already reached within the K inner iterations. An improved version is to leave the inner iteration as soon as a tolerance is reached, and check the conditions in the outer iteration.

4.3.4 2×2 Test

To investigate the performance of NLCG within the preconditioned TangCG algorithm, we enlarge the test to a 2×2 grid, where Elements 1 and 2 are in adhesion and 3 and 4 are in slip. The contact area is $[-1, 1] \times [-1, 1] \text{ mm}^2$. The material parameters are: $G = 200 \text{ N/mm}^2, \nu = 0.42$. The traction bound is $\mathbf{g} = [0.4, 0.8, 0.282843, 0.5]$. The true solution is: $\mathbf{p}_1^* = [0.2, 0.1], \mathbf{p}_2^* = [0.1, 0.2], \mathbf{p}_3^* = [0.2, 0.2], \mathbf{p}_4^* = [0.3, 0.4]$. The true slip is $\mathbf{s}_1^* = [0, 0], \mathbf{s}_2^* = [0, 0], \mathbf{s}_3^* = [-0.1, -0.1], \mathbf{s}_4^* = [-0.3, -0.4]$.

We linearize the nonlinear system at true solution \mathbf{p}^* . By setting $\delta p_{3x} = -\beta_3 \cdot p_{3y}^*, \delta_{3y} = \beta_3 \cdot p_{3x}^*$, and $\delta p_{4x} = -\beta_4 \cdot p_{4y}^*, \delta_{4y} = \beta_4 \cdot p_{4x}^*$, the Jacobian matrix is reduced to a 6×6 matrix:

$$J_r^* = \begin{pmatrix} A_{11}^{xx} & A_{11}^{xy} & A_{12}^{xx} & A_{12}^{xy} & a_{31}^x & a_{41}^x \\ A_{11}^{yx} & A_{11}^{yy} & A_{12}^{yx} & A_{12}^{yy} & a_{31}^y & a_{41}^y \\ A_{21}^{xx} & A_{21}^{xy} & A_{22}^{xx} & A_{22}^{xy} & a_{32}^x & a_{42}^x \\ A_{21}^{yx} & A_{21}^{yy} & A_{22}^{yx} & A_{22}^{yy} & a_{32}^y & a_{42}^y \\ a_{31}^x & a_{31}^y & a_{32}^x & a_{32}^y & a_{33}^y p_{3x}^* - a_{33}^x p_{3y}^* & a_{34}^y p_{4x}^* + a_{34}^x p_{4y}^* \\ a_{41}^x & a_{41}^y & a_{42}^x & a_{42}^y & a_{43}^y p_{3x}^* - a_{43}^x p_{3y}^* & a_{44}^y p_{4x}^* + a_{44}^x p_{4y}^* \end{pmatrix}. \quad (4.48)$$

Here, for elements $I \in S$ and the elements in contact $J \in C$:

$$\begin{aligned} \text{if } I \neq J: & \quad a_{IJ}^x = A_{IJ}^{yx} \cdot p_{Ix}^* - A_{IJ}^{xx} \cdot p_{Iy}^*, & \quad a_{IJ}^y = A_{IJ}^{yy} \cdot p_{Ix}^* - A_{IJ}^{yx} \cdot p_{Iy}^*, \\ \text{if } I = J: & \quad a_{II}^x = A_{II}^{yx} \cdot p_{Ix}^* - A_{II}^{xx} \cdot p_{Iy}^* + s_{Iy}^*, & \quad a_{II}^y = A_{II}^{yy} \cdot p_{Ix}^* - A_{II}^{yx} \cdot p_{Iy}^* - s_{Ix}^*. \end{aligned}$$

The Jacobian matrix (4.48) is SPD. Its condition number is found to be 88.

The TangCG algorithm employs an NLCG process, whose convergence should be similar to the linear CG algorithm when the iterates are close to the true solution \mathbf{p}^* . We test whether this is also true for the TangCG algorithm. We also check whether the preconditioner improves the convergence near \mathbf{p}^* .

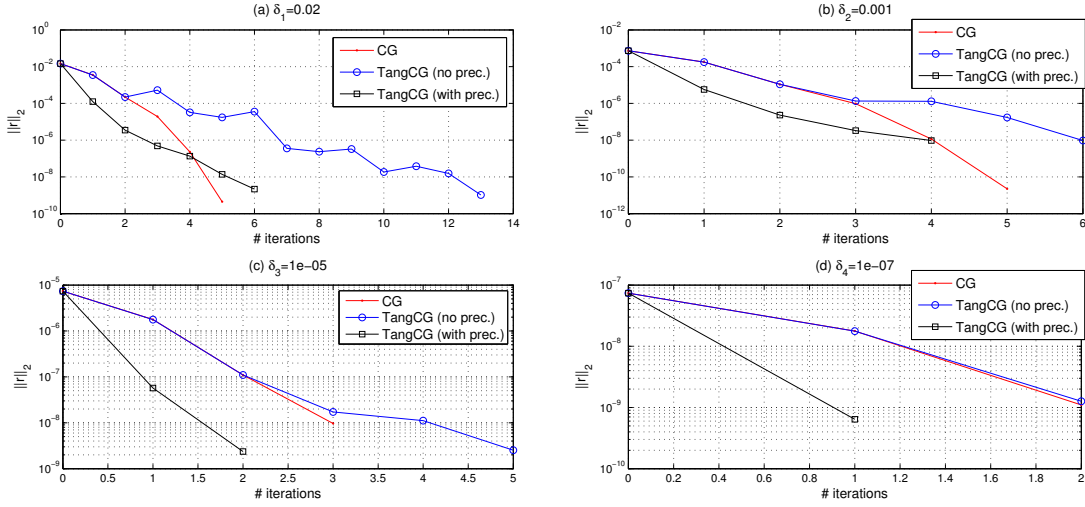


Figure 4.3: The convergence behavior of TangCG with and without preconditioning, compared to CG, for four different initial points.

As a test, four different initial points are implemented, respectively, with their absolute distances to \mathbf{p}^* being $\delta_i = 0.02, 0.001, 1 \times 10^{-5}, 1 \times 10^{-7}$ ($i = 1, 2, 3, 4$). CG is applied to the linearized system $J_r^* \delta \mathbf{p}_r^* = \mathbf{0}$, with initial $\delta \mathbf{p}_r^0 = T^T(\mathbf{p}^* - \mathbf{p}^0)$. The iterate is converted to traction by (4.30) and further the nonlinear defect is compared with those by the unpreconditioned and preconditioned TangCG algorithms.

These results are shown in Fig. 4.3. It can be seen that the closer the initial point is at the true solution, the more similar the convergence by TangCG is to the CG convergence. As the iterate approaches the true solution, the Jacobian matrix obtained by linearization in each TangCG iteration is close to the exact Jacobian J_r^* . In this case, the NLCG method within TangCG is the same as CG, as seen in Fig. 4.3(d). Moreover, we find that the preconditioner indeed improves the convergence of TangCG.

4.4 Numerical Results

Various Cattaneo shift problems are tested here with different patterns of slip. A sphere is pressed onto a plane, and is then shifted tangentially. The sphere and plane are of the same material, with shear modulus $G = 200 \text{ N/mm}^2$, and Poisson's ratio $\nu = 0.42$. The potential contact area is $[-a, a] \times [-b, b] \text{ mm}^2$ where $a = b = 1.2857 \text{ mm}$. The total normal force is $F_z = 9.1954 \text{ N}$. The radii of the sphere are $R_x = R_y = 50 \text{ mm}$, and the profile of the sphere is $h = \frac{1}{2R_x}x^2 + \frac{1}{2R_y}y^2$. Coulomb's law is applied with friction coefficient $\mu = 0.4$.

Since the two contacting bodies are quasi-identical, the solution process is decoupled to first solve the normal problem, followed by the tangential problem.

The former is solved by Hertz theory [44] or a numerical method [101], which results in a contact area C and normal pressure p_z . Applying Coulomb's law, we thus obtain the traction bound $g = \mu p_z$. Below we focus on solving the tangential problems, with seven different slip cases, as specified in Table 4.1.

Subdivision of contact area		ξ	η	ϕ
Case 1	full adhesion	0	0	5×10^{-7}
Case 2	0.6% slip	1.5×10^{-5}	0	1×10^{-5}
Case 3	20% slip	0	0.0015	0.0012
Case 4	40% slip	0.0021	0.0010	0.003
Case 5	60% slip	0.0037	0.0048	0.004
Case 6	80% slip	0	0.005	0.009
Case 7	full slip	0.0044	0	0.08

Table 4.1: *Test setting: seven cases with different percentages of slip. Longitudinal shift ξ , lateral shift η and rotation shift ϕ are prescribed. They are used to compute rigid shift \mathbf{w} (see Eq. (4.11)).*

The stopping criterion on the accuracy of the tractions is given by

$$\frac{\|\mathbf{p}^k - \mathbf{p}^{k-1}\|}{\|\mathbf{p}^k\|} < \varepsilon, \quad (4.49)$$

where, if not specified differently, we use $\varepsilon = 10^{-5}$. The norm used in this section is the “root-mean-square” norm. This stopping criterion on traction p is chosen since engineers pay most attention to tractions, rather than to the slip, which is the defect of nonlinear system (4.27). Moreover, when (4.49) is satisfied, the defect of this nonlinear system is smaller than 10^{-8} in our numerical tests.

The tangential solvers are TangCG(K) algorithms with and without preconditioning, using the improved version mentioned in Remark 4.2. Remember that K is the number of NLCG iterations for one nonlinear system before checking the contact conditions. We implemented these two algorithms for the seven problem cases, in order to find an optimal value for K . They are then compared with each other, yielding the most efficient algorithm. It is also compared with the ConvexGS method [95], from the perspective of iteration numbers and CPU time.

4.4.1 Optimal K

First we use a 120×100 grid, which involves 5732 contacting elements with $5732 \times 2 = 11464$ traction unknowns. We apply $K \in \{1, 2, 3, 4, 5\}$ for the seven cases of Table 4.1.

Table 4.2 gives the total number of inner NLCG iterations by TangCG(K) without preconditioning. For Case 1, the full adhesion problem, the results are independent of the values of K . Remember that the TangCG(K) method starts

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7
$K = 1$	30	46	135	100	148	85	48
$K = 2$	30	52	66	94	119	81	48
$K = 3$	30	54	86	83	105	77	65
$K = 4$	30	50	94	104	117	95	57
$K = 5$	30	53	78	104	165	87	37

Table 4.2: *Unpreconditioned TangCG(K): the total number of inner iterations with different values of K for the seven slip cases, using a 120×100 grid (with $5732 \times 2 = 11464$ unknowns).*

with the subdivision $H^0 = C$, and $S^0 = \emptyset$. This is already the correct subdivision in the full adhesion case. The governing system is linear, and TangCG is actually equivalent to the CG method.

When slip occurs, different K values show different results. $K = 1$ is the fastest in the slight slip Case 2. It gets worse when more slip is included, as seen in Cases 3-6, where $K = 3$ seems a proper choice. The reason is that the slip area has to be found from initial area $S^0 = \emptyset$. If $K = 1$, then only one inner iteration is performed and the traction iterates may not be accurate enough to perform a useful adjustment of the subdivision of the contact area: in Step 11 of the TangCG algorithm (see Section 4.3.3), many elements that are first moved from the adhesion to the slip area, may be immediately moved back. Using a larger value for K can improve this situation. However, if K is too large, there would be unnecessarily many inner iterations.

In the full slip Case 7, $K = 5$ results in the fastest convergence. The initial slip area S^0 tends to be $S = C$. More precise solutions can move more elements into the slip area at once, which accelerates the convergence. Moreover, the results of other values for K are also acceptable.

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7
$K = 1$	30	43	52	59	66	52	27
$K = 2$	30	50	47	52	77	60	32
$K = 3$	30	48	49	55	49	46	37
$K = 4$	30	45	56	52	56	50	34
$K = 5$	30	54	52	57	56	47	22

Table 4.3: *Preconditioned TangCG(K): the total number of inner iterations with different values of K for the seven slip cases, using a 120×100 grid (with $5732 \times 2 = 11464$ unknowns).*

From the above discussion, we find $K = 3$ is an optimal choice for the unpreconditioned TangCG(K) method in different cases. The same conclusion can be found for TangCG(K) with preconditioning, from the results in Table 4.3. Therefore, we will use $K = 3$ in the following tests.

4.4.2 Comparison between TangCG(3) with and without Preconditioning

Comparing these two tables, we find that the preconditioner helps to accelerate the convergence, depending on the different slip situations. One exception is the full adhesion Case 1, where the main diagonal entries of the Jacobian are almost the same. Hence, the condition number after preconditioning hardly reduces.

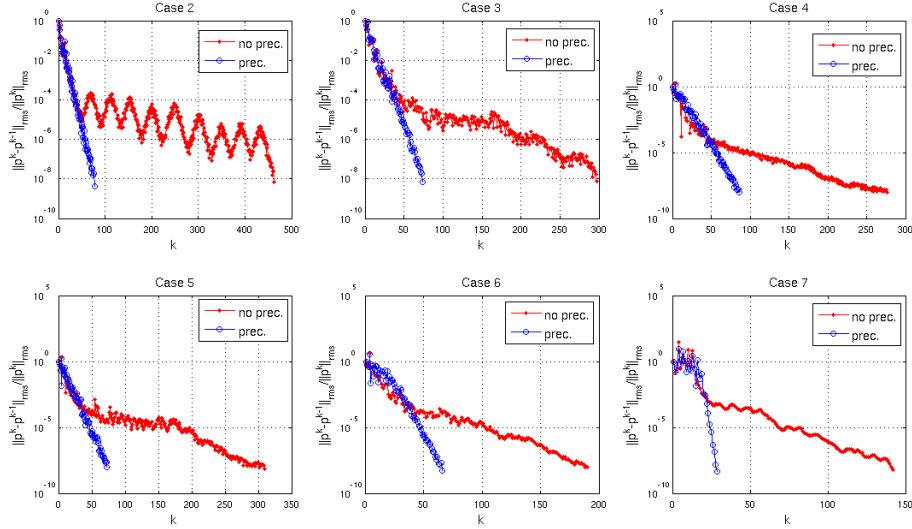


Figure 4.4: Convergence of TangCG(3) with and without preconditioning for Cases 2-7, with $\varepsilon = 10^{-8}$ in the stopping criterion (4.49). The horizontal axis is number of inner iterations, and the vertical axis is the relative difference between two iterates.

Fig. 4.4 displays the convergence behavior of the two methods for Cases 2-7, that include slip. In this figure, we use $\varepsilon = 10^{-8}$ in the stopping criterion (4.49), to examine the behavior of these two methods near the true solution. We find that TangCG(3) with preconditioning greatly speeds up the convergence and makes the method much more robust. It gives an almost linear reduction of the difference between two iterates, when approaching the true solution. TangCG(3) without preconditioning, on the other hand, displays a slower convergence speed, and even an oscillatory reduction in Case 2.

4.4.3 Comparison with ConvexGS Method

We compare TangCG(3) with preconditioning to the performance of the ConvexGS method in the software CONTACT. The codes of the former were written in Matlab 7.13 (R2011b). The latter was programmed in Fortran. Both methods are implemented on a Linux desktop PC (Intel Core I5-2400 processor, quad-core, 3.1 GHz). First of all, iteration numbers by these two methods are compared in Table 4.4, where the iteration numbers by ConvexGS are given in brackets. As can be seen, iteration numbers of TangCG(3) are of $\mathcal{O}(n^{0.4})$, with n the number

of unknowns ⁵. Moreover, TangCG(3) requires fewer iterations than ConvexGS for the test cases with a small percentage of slip. With more slip, the iteration numbers are comparable.

Discretization	30×25	60×50	120×100	240×200
# traction unknowns	728	2880	11464	45736
Case 1	14 (23)	23 (32)	30 (50)	49 (77)
Case 2	20 (26)	33 (32)	48 (48)	62 (72)
Case 3	33 (31)	39 (37)	49 (56)	65 (88)
Case 4	33 (32)	41 (33)	55 (48)	76 (73)
Case 5	33 (31)	42 (32)	49 (45)	70 (69)
Case 6	34 (29)	34 (30)	46 (35)	73 (53)
Case 7	14 (18)	12 (18)	37 (18)	25 (18)

Table 4.4: Numbers of inner iterations by preconditioned TangCG(3), and numbers of iterations by ConvexGS (shown in brackets).

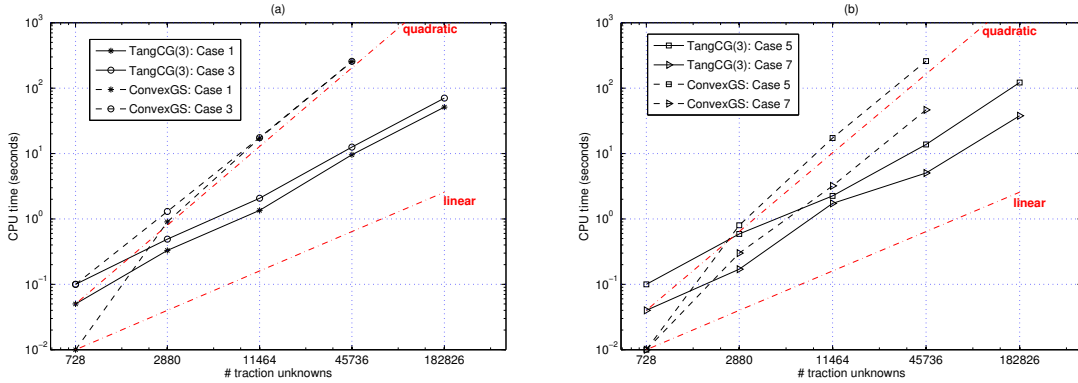


Figure 4.5: CPU time in seconds by TangCG(3) with preconditioning and ConvexGS for (a) Cases 1 and 3, (b) Cases 5 and 7.

Fig. 4.5 shows the CPU time (in seconds) for some of the test cases, for TangCG(3) with preconditioning (denoted by solid lines), and ConvexGS (denoted by dashed lines). For the smallest problem, both methods require approximately the same computing time. The superiority of TangCG(3) shows when more unknowns are involved. The *speedup factor* is defined as the ratio of time by ConvexGS and by TangCG(3). For around 10^4 unknowns, the speedup factors for Cases 1,3,5,7 are about 13,8,8,2, respectively. They grow to 27,20,19,9 when the number of unknowns is multiplied by a factor of 4. Fig. 4.5 also displays curves of quadratic complexity $\mathcal{O}(n^2)$ and linear complexity $\mathcal{O}(n)$ (denoted by the dash-dot lines). The CPU time curves for TangCG(3) remain between

⁵There is no good estimate for Case 7, since the corresponding iteration numbers do not gradually increase but oscillate as unknown n grows.

these two complexity lines. We find that the complexity of TangCG(3) is $\mathcal{O}(n^{1.7})$, which is faster than ConvexGS with $\mathcal{O}(n^{2.3})$. Note that ConvexGS cannot benefit from the favorable Toeplitz matrix structure.

4.5 Conclusion

In this chapter we proposed the TangCG algorithm to solve a nonlinear constrained optimization formulation, arising from the 3D frictional contact problem. The corresponding KKT conditions provide the governing equations and contact conditions. The TangCG algorithm applies an active set strategy. The subdivision of the contact area is fixed and the resulting governing system is solved approximately. The resulting solution is used to modify the subdivision based on these contact conditions.

Because the magnitude of the tractions on a slip element equals the traction bound, we can change the conventional traction variables to azimuth angles, when placing the traction vector of each slip element in a polar coordinate system. This changes the constraints to simple bounds such that the BCCG strategy can be used. The resulting nonlinear equations are solved by several NLCG iterations, which is based on linearization and applies a CG method in each iteration. A diagonal scaling preconditioner is necessary to bring the defects of adhesion and slip areas to the same scale, and all matrix-vector products in this algorithm are speeded up by FFTs.

The TangCG algorithm is tested for Cattaneo shift problems, with different amounts of slip. The preconditioner is found to improve robustness and accelerate convergence when slip occurs. The computational time is reduced dramatically compared to ConvexGS, a state-of-the-art method for concentrated frictional contact problems. The corresponding speedup factor grows as the problem size increases. This confirms the efficiency of our new method. Moreover, since rolling contact problems can be transformed into a sequence of shift problems, our TangCG method can also be applied in this case.

Chapter 5

Extending the Range of Applicability of the Contact Approach

In this chapter we extend the range of applicability of the boundary element method (BEM) for concentrated elastic contact problems by computing the influence coefficients (ICs) numerically. These ICs represent the Green's function of the problem, i.e. the surface deformation due to unit loads. For the half-space they are analytically available. An elastic model is proposed to compute these coefficients numerically, by the finite element method (FEM). We recommend proper strategies of FEM meshing and element types, considering accuracy and computational cost. The effects of computed ICs to contact solutions are examined for a Cattaneo shift problem. The work in this chapter aims at providing a reference to study fast solvers for the conformal contact problem where ICs are not analytically known.¹

5.1 Introduction

5.1.1 Strategies for Solving Concentrated Contact Problems

In previous chapters, we developed fast solvers for concentrated contact problems. In such problems, the contact area is considered flat since it is small compared to the dimensions of the contacting bodies. Examples can be found in Figs. 5.1(a) and (b), where we depict the contact between the wheel tread and rail crown, and between wheel flange and rail gauge face. The half-space approach is used, which approximates the contacting bodies as two semi-infinite solids bounded by the contacting plane [51]. The boundary element method (BEM) transforms the three-dimensional (3D) boundary value problem to a two-dimensional (2D) boundary integral problem. The integral gives the relation be-

¹The contents of this chapter have been published in technical report [114] and submitted for publication: J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee. Extending the BEM for elastic contact problems beyond the half-space approach. *Delft University of Technology*, 2015.

tween displacements and tractions. In the half-space approach, the corresponding Green’s function is based on a solution by Boussinesq and Cerruti (see [44]). It is the so-called influence function, which expresses the displacements at one point caused by a load at another point on the half-space.

Solving such problems starts with the definition of the potential contact area, which contains the true contact region. Usually rectangular elements are placed in this 2D region, and piecewise constant functions are used to represent the tractions to be solved. This results in *influence coefficients (ICs)* from the influence function. Analytic coefficients are available [51], and have been incorporated with solvers for concentrated contact, e.g. [100, 115, 51].

5.1.2 Strategies for Solving Conformal Contact Problems

Conformal contact occurs when the rail and wheel are seriously worn so that a curved and much larger surface comes into contact, as seen in Fig. 5.1(c). Other applications can be found, for instance, journal-bearing contact [105], microcontact printing [63], etc. The half-space approach is prohibited, since its assumptions of a flat and small contact area are not valid in these cases. The finite element method (FEM) is usually employed for such problems, e.g. in [93, 61, 23], due to its generally applicability.

However, since discretization by FEM covers whole contacting bodies, this method can be computationally slow and memory intensive. Moreover, focusing on the contact region, FEM often employs coarser meshes than BEM, which implies less detailed information in the region of interest.

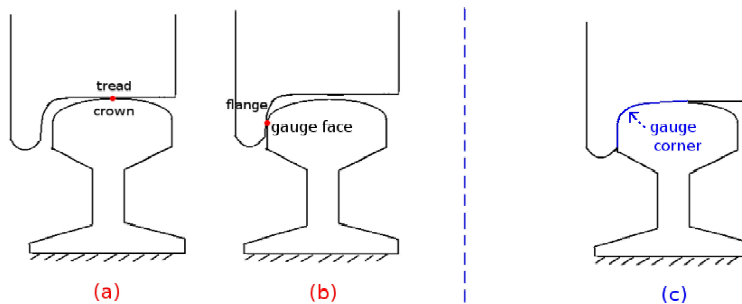


Figure 5.1: *Two types of contact problems in wheel-rail contact [60]: concentrated contact in (a) and (b) with “flat contact areas”, and conformal contact in (c) with a “curved contact area”.*

One idea of fast solvers for conformal contact problems is to apply the BEM, which means to only discretise the contact surface, and work on the surface integral similar as in concentrated contact [60, 103]. Those fast solvers for concentrated contact can then also be applied. A difficulty is that the influence function (and further the influence coefficients) is not theoretically known in this case.

Therefore, the influence function may need to be computed numerically, resulting in numerical influence coefficients. The first question is how we can compute these coefficients efficiently, considering the accuracy and computational cost. This requires to build a model, to choose good strategies of FEM meshing and element types, and to find the accuracy of the resulting coefficients. The second question is about how these influence coefficients affect the final solution of the conformal contact problem.

This approach has been developed by Li in [60], where the rail was approximated by a quasi-quarter space with a distributed load and the FEM was applied. However, this work did not present detailed strategies for the mesh or element types. Moreover, there was no discussion of the error propagation from influence coefficients to contact solutions. The approach was extended by Vollebregt [103], who presented initial results on meshing strategies and error propagation. In the present work, this research is continued, such that more detailed guidance can be provided. In addition, the technique of replacing the half-space solution by a finite element solution has also been applied in elastohydrodynamic lubrication problems in [32].

5.1.3 Content and Structure of This Chapter

We investigate the influence coefficients in a concentrated contact setting, where analytic influence coefficients are known when a piecewise constant approximation is used as discretization of the tractions [51, 44]. Secondly, there are analytic solutions such as the Hertz theory for normal contact [36] and the Cattaneo theory for the tangential shift problem [13]. The corresponding BEM results can provide a reference for studying conformal contact problems, for which there are no analytic influence coefficients, neither analytic contact solutions.

For conformal contact seen in Fig. 5.1(c), one solves the displacements on the curved contact region of the rail. The FEM is then employed. Prismatic elements are generally applicable with their triangular faces on the curved boundaries of the cross section of the rail. Regarding such application, prismatic elements are also investigated for the model in this chapter, which solves the displacements on a half-space.

In the model a unit load is applied on the region of one element, which brings a jump at the edge of this element to its neighbors. This implies a significant discontinuity in the traction boundary condition for the model as well, and may disturb the FEM solution. An alternative is to impose a bilinear load, which equals a “tent” shape at this region. Analytic investigation for this test setting can be found in [21], however, the implicit expressions for influence coefficients are difficult to process directly. A closed form solution is given in [59], however, on a triangular mesh. This is not useful for our work, which mainly focuses on rectangular meshes since these result in matrices of Toeplitz² structure so that the fast Fourier transform (FFT) can be applied to accelerate the corresponding

²A Toeplitz matrix is the matrix with constant diagonals.

matrix-vector products [100]. Therefore, numerical influence coefficients computed with a bilinear load on rectangular meshes are of interest.

The structure of this chapter is as follows. In Section 5.2, an elastic model is built to compute influence coefficients. The results of this model are discussed in Section 5.3 with three different FEM mesh settings, from which insight is provided on the choice of meshing and element types, in view of accuracy and computational cost. In Section 5.4, a specific contact problem is solved, with numerical influence coefficients obtained by the recommended strategies. Error propagation is investigated separately in normal and tangential contact problems. The last section concludes this chapter.

5.2 An Elastic Model for Computing Influence Coefficients

5.2.1 Influence Coefficients on Half-Space

Influence coefficients are equivalent to displacements caused by a specific load. Evaluating the displacements under a given load is a basic problem in elasticity. It can be done by solving three elastic equations with some boundary conditions. These equations, in a quasi-static case, read [40]:

1. Equilibrium equation:

$$\sigma_{ij,j} + F_i = 0. \quad (5.1)$$

2. Strain-displacement equation:

$$\varepsilon_{i,j} = \frac{u_{i,j} + u_{j,i}}{2}. \quad (5.2)$$

3. Stress-strain equation (constitutive equation, or Hooke's law):

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl}. \quad (5.3)$$

In the above equations, subscripts i, j, k, l take on values x, y, z . The Einstein summation convention is used when repeated subscripts occur. Notations $\sigma, \varepsilon, F, u$ denote stress, strain, body force per volume, and displacement, respectively. Subscript $(,j)$ means differentiating w.r.t. the corresponding coordinate direction j .

In the case of concentrated contact problems, an alternative way can be used to obtain displacements. It is based on four assumptions. First of all, the contacting region should be small compared to the dimensions of the contacting bodies. Secondly, these contacting bodies should be of homogenous linear elastic material. The third is small displacements and displacement gradients. Fourthly, inertial effects are ignored, which is a quasi-static case.

These assumptions allow for the use of the *half-space approach*, which approximates the two contacting bodies by two semi-infinite elastic bodies, bounded by

the contacting plane. Usually a Cartesian coordinate system $(O; x, y, z)$ is placed, with the origin at the center of the contact region, and the z -direction pointing to the outward normal of one contacting body. A half-space is at one side of the bounding plane and can be denoted by, for instance, $\{(x, y, z) | z \leq 0\}$.

We only consider one half-space here. Displacements at one point on this half-space caused by a point load at another point were given as a function of relative distance between these two points by Boussinesq and Cerruti (see [44]). For instance, normal displacement u_z at surface point (x, y) resulting from a normal point load P at another surface point (ξ, η) is given by [51]:

$$u_z(x, y) = \frac{P}{2\pi G} \cdot \frac{(1 - \nu)}{\rho}, \quad (5.4)$$

where G is shear modulus and ν is Poisson's ratio. $\rho = \sqrt{(x - \xi)^2 + (y - \eta)^2}$ is the distance between points (x, y) and (ξ, η) . We drop the z -component for point coordinates here since all points of interest are on the surface of the half-space, i.e. $z = 0$.

This equation has a singularity point when $\rho = 0$, at the point where the load is applied. This is not a serious problem since in practice the load is often distributed on a finite region [5]. The displacement caused by a distributed load can be obtained by integration of the displacement by the point load. Replacing normal load P by $p_z(\xi, \eta)d\xi d\eta$ in Eq. (5.4) and integrating on a certain contact area C , we obtain the relation between normal displacement and pressure (normal traction):

$$u_z(x, y) = \iint_C A^{zz}(x, y, \xi, \eta) p_z(\xi, \eta) d\xi d\eta, \quad (5.5)$$

where the kernel

$$A^{zz}(x, y, \xi, \eta) = \frac{(1 - \nu)}{2\pi G} \cdot \frac{1}{\rho} \quad (5.6)$$

is called the *influence function*. It describes the influence of a unit normal load at point (ξ, η) on the normal displacement at point (x, y) . As mentioned in Section 2.2.3, the value of $A^{zz}(x, y, \xi, \eta)$ depends on the relative distance of these two points, rather than their own locations. This means that:

$$A^{zz}(x, y, \xi, \eta) = A^{zz}(x - \xi, y - \eta). \quad (5.7)$$

Discretization is done by placing $n := n_x \cdot n_y$ rectangular elements on a potential contact area which contains the real contact region. The corresponding element sizes are δx and δy , respectively. A cell-centered mesh arrangement is chosen. Using $I, J \leq n$ as element indices, a widely used approximation for pressure p_z is by *piecewise constant* function $f_J(\xi, \eta)$:

$$p_z(\xi, \eta) = \sum_{J=1}^n p_J \cdot f_J(\xi, \eta), \quad (5.8)$$

where $f_J(\xi, \eta) = 1$ for (ξ, η) on element J and $f_J(\xi, \eta) = 0$ elsewhere. Plugging Eq. (5.8) into integral (5.5), we arrive at:

$$u_z(x_I, y_I) = \sum_{J=1}^n \left[p_J \iint_{e_J} A^{zz}(x_I, y_I, \xi, \eta) f_J(\xi, \eta) d\xi d\eta \right], \quad I = 1, \dots, n, \quad (5.9)$$

where e_J denotes element J . Eq. (5.9) leads to a linear relationship:

$$\mathbf{u}_z = A^{zz} \mathbf{p}_z, \quad \mathbf{p}_z, \mathbf{u}_z \in \mathbb{R}^n, A^{zz} \in \mathbb{R}^{n \times n}, \quad (5.10)$$

where matrix A^{zz} stores all the *influence coefficients (IC)* A_{IJ}^{zz} . Each of them is defined by:

$$A_{IJ}^{zz} = \iint_{e_J} A^{zz}(x_I, y_I, \xi, \eta) f_J(\xi, \eta) d\xi d\eta, \quad (5.11)$$

which results from integrating Eq. (5.5) on a single element e_J w.r.t. an observation on element I . It indicates the physical meaning of the IC: A_{IJ}^{zz} is the normal displacement on element I , caused by a load f_J on element J . In this case, the applied (normal) load f_J is a unit piecewise constant function, and analytic solutions of integral (5.11) can be found in [51].³ Moreover, Eq. (5.11) also implies that the value of the IC is related to the area of element e_J and material parameters G, ν .

A similar derivation can be done for other ICs, such as A^{yx} , the y -component of displacement caused by a unit load in the x -direction.

5.2.2 Model

The model for computing the ICs is based on the contact problem to be solved. Assume that the contact problem considers two bodies of homogeneous linearly elastic materials, with shear modulus G and Poisson's ratio ν . Recall that $n = n_x \cdot n_y$ rectangular elements are placed in the potential contact area, with corresponding mesh sizes δx and δy .

This model is inspired by the fact that an IC is equivalent to the displacement at one surface point resulting from a unit load at another surface point. We have to deal with two issues. One is to approximate the semi-infinite half-space, and the other is to provide a proper boundary value problem.

To approximate the half-space which represents one contacting body, we consider a block of size $[-L, L] \times [-L, L] \times [-L, 0]$ mm³ in 3D space [103] (see Fig. 5.2(a)). This block is of the same material as the contacting body, and $L \rightarrow \infty$ gives the half-space model. A unit load is applied at the center of the top surface, and we solve for the resulting displacements on this surface, which are the corresponding ICs. Due to the symmetry and anti-symmetry of the displacements, the computational domain can be reduced to a quarter of the block, $[-L, 0] \times [-L, 0] \times [-L, 0]$, which is shown in Fig. 5.2(b).

³The analytic ICs are displacement differences of the two contacting bodies. When the same material is adopted for the bodies, the IC divided by 2 leads to the displacement of one body.

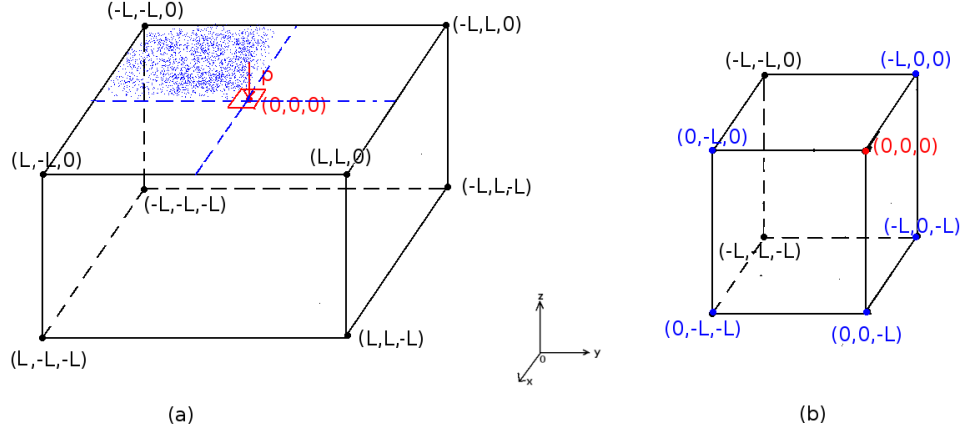


Figure 5.2: (a) The original 3D block for approximating the half-space. It occupies space of $[-L, L] \times [-L, L] \times [-L, 0]$ mm^3 . Due to the symmetry and anti-symmetry of displacements on the top surface, we consider the left upper quarter (blue shadow), which is shown in (b).

ICs	Front surface ($x = 0$)	Right surface ($y = 0$)	Top surface ($z = 0$)
A^{xz}, A^{yz}, A^{zz}	$u_x = 0,$ $\frac{\partial u_y}{\partial x} = 0,$ $\frac{\partial u_z}{\partial x} = 0.$	$\frac{\partial u_x}{\partial y} = 0,$ $u_y = 0,$ $\frac{\partial u_z}{\partial y} = 0.$	$\sigma_z = 1, (x, y) \in [-\frac{\delta x}{2}, 0] \times [-\frac{\delta y}{2}]$ $= 0, \text{ otherwise.}$
A^{xx}, A^{yx}, A^{zx}	$\frac{\partial u_x}{\partial x} = 0,$ $u_y = 0,$ $u_z = 0.$	$\frac{\partial u_x}{\partial y} = 0,$ $u_y = 0,$ $\frac{\partial u_z}{\partial y} = 0.$	$\sigma_x = 1, (x, y) \in [-\frac{\delta x}{2}, 0] \times [-\frac{\delta y}{2}]$ $= 0, \text{ otherwise.}$
A^{yy}, A^{xy}, A^{zy}	$u_x = 0,$ $\frac{\partial u_y}{\partial x} = 0,$ $\frac{\partial u_z}{\partial x} = 0.$	$u_x = 0,$ $\frac{\partial u_y}{\partial y} = 0,$ $u_z = 0.$	$\sigma_y = 1, (x, y) \in [-\frac{\delta x}{2}, 0] \times [-\frac{\delta y}{2}]$ $= 0, \text{ otherwise.}$

Table 5.1: Boundary conditions at the front surface ($x = 0$), at the right surface ($y = 0$) and at the top surface ($z = 0$) for different ICs. On other three surfaces the boundary condition is given by $\mathbf{u} = \mathbf{0}$.

Governing equations for this model are the basic equations in elasticity, as given by Eqs. (5.1)-(5.3). Boundary conditions can be different for different ICs. Define displacement by $\mathbf{u} = (u_x, u_y, u_z)^T$, and stress by $\sigma = [\sigma_x, \sigma_y, \sigma_z]^T$, then Table 5.1 gives boundary conditions at the front, right and top surfaces of the block. At the other three surfaces the boundary condition is given by $\mathbf{u} = \mathbf{0}$.

We explain the boundary conditions on the top surface in Table 5.1. The top surface of the block is shown in Fig. 5.3. A 2×2 grid with $h_c = \delta x = \delta y$ is used for the contact problem. Due to the cell-centered mesh, the ICs for the center points of these four elements need to be computed. Since the value of the IC depends only on the relative distance of a pair of points, we can place the right-bottom

element of the contact grid centered at the origin and compute displacements for these four points w.r.t. the origin. Moreover, a piecewise constant representation of tractions is employed, which implies that a unit load is applied on the region of the center element (the shaded area in Fig. 5.3). Since we only consider a quarter of the block, the load in the model is only given on a quarter of the center element.

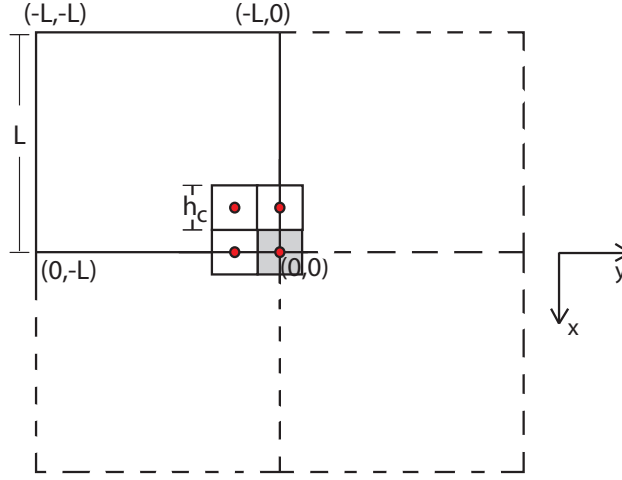


Figure 5.3: *The top surface of the block in the model. In the contact problem a 2×2 grid with mesh size $h_c = \delta x = \delta y$ is used. The center of the right-bottom element is at the origin. A unit load is applied on the shaded element region. When solving the resulting displacements by FEM, different meshes can be applied.*

With governing equations (5.1)-(5.3) and boundary conditions, we can employ the finite element method (FEM) to solve for the displacements. In Fig. 5.3, differently from the 2×2 contact grid, other meshes can be used when solving this model by the FEM. (Be aware that the meshes for the contact problem, and for the FEM model are different.) A linear interpolation technique is applied to convert the resulting displacements to the center points of the contact mesh, which leads to the required coefficients.

Remark 5.1. *When using a bilinear approximation of pressures, i.e. the applied load is bilinear in the model, boundary conditions on the top surface $z = 0$ are different. For instance, for the IC caused by a normal load, i.e. A^{xz}, A^{yz}, A^{zz} , the condition reads [96]:*

$$\sigma_z(x, y, 0) = \begin{cases} (1 - |x|/\delta x)(1 - |y|/\delta y), & (x, y) \in [-\delta x, 0] \times [-\delta y, 0], \\ 0, & \text{otherwise.} \end{cases}$$

For the IC caused by a load in x - or y -direction, one needs to replace the left-hand side of this condition by $\sigma_x(x, y, 0)$ or $\sigma_y(x, y, 0)$, respectively.

5.3 Influence Coefficients A^{zz} by a Piecewise Constant Load

We know the analytic ICs when pressures are represented by piecewise constant functions [44, 51]. In this section, the model proposed in Section 5.2 is used to compute the IC A^{zz} , which is the normal displacement caused by a unit normal load. The resulting IC value can be compared with the analytic IC. Tests in this section aim at providing insight on mesh strategies and element types, considering the accuracy and computational cost.

The contact problem which requires the numerical IC is defined as follows. Consider a sphere pressed on a plane. Both bodies are of the same material, with shear modulus and Poisson's ratio $G = 82000 \text{ N/mm}^2, \nu = 0.28$, respectively. In this case, the analytic ICs divided by 2 is the displacement on one of the bodies. The potential contact area is $[-4, 4] \times [-4, 4] \text{ mm}^2$. On this area, four $n_c \times n_c$ grids are placed: $10 \times 10, 20 \times 20, 40 \times 40, 80 \times 80$, with mesh sizes $h_c = 0.8, 0.4, 0.2, 0.1$, respectively.

As a reminder, subscript c refers to the contact problem, subscript s in the following refers to the model, which is solved by finite element package SEPRAN [81]. For the solution of the FEM discretization, a preconditioned conjugate gradients method is employed with stopping criterion $\frac{\|\mathbf{u}^{k+1} - \mathbf{u}^k\|}{\|\mathbf{u}^{k+1}\|} \leq 10^{-6}$, where \mathbf{u} is displacement and the superscripts are iteration indices. To compare with the analytic IC on a certain contact grid, linear interpolation is used. We define our target of approximation by:

$$\frac{|A_{\mathbf{0}}^{zz} - F_{\mathbf{0}}^{zz}|}{|A_{\mathbf{0}}^{zz}|} < 0.01, \quad (5.12)$$

where $A_{\mathbf{0}}^{zz}$ is the analytic coefficient (divided by 2) at the origin, and its approximation by the FEM is $F_{\mathbf{0}}^{zz}$.

5.3.1 Mesh Strategies

Remember that our FEM model needs to approximate the semi-infinite half-space. Hence, the FEM mesh should be easily extended to a very large block without involving too many elements. Since the displacements decrease outside the loaded region, more elements can be used near the origin, and much fewer elements far away from it.

Regarding the type of elements, we first use the hexahedron element which is easy to be generated for a 3D block. The corresponding mesh strategy is called "Mesh1", shown in Fig. 5.4(a). The loaded region is covered by a small cube $[-l, 0] \times [-l, 0] \times [-l, 0]$. Within this cube, a uniform mesh is used with element size $h_s = l/m$, where m is the number of elements in each direction. Outside this cube, the element size increases by a factor f leaving the loaded area. Moreover, edges that do not contain the small cube are divided uniformly into m intervals.

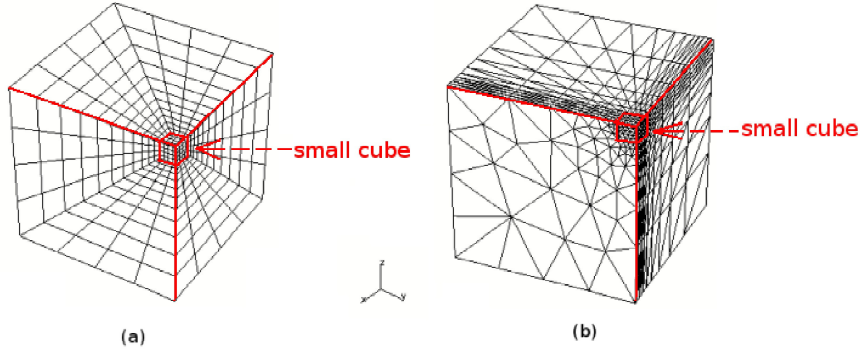


Figure 5.4: Meshes generated by SEPRAN: (a) hexahedron Mesh1. (b) prismatic Mesh2 (where diagonal edges on top and right surfaces are a plotting artefact).

In this mesh, stretching factor f , cube size l and its element size h_s are to be determined, so that accurate numerical ICs result.

Another employed element is the prismatic element. As mentioned in Section 5.1, in conformal contact, one computes ICs considering a load on the curved contact surface of the rail, see Fig. 5.1(c). The cross section of the rail has curved boundaries, and it is well fitted by triangles. Moreover, the rail can be regarded as infinitely long, so rectangles are used for the other surfaces. This yields the prismatic elements.

Assuming that the front and back surfaces of the 3D block correspond to the cross section of the rail in Fig. 5.1(c), these two surfaces should consist of the triangular faces of the prismatic elements, as shown in Fig. 5.4(b). (One may see that the top and right surfaces in this figure also show triangles. In fact the diagonal line in each rectangle does not exist. This is just a plotting issue.)

This prismatic mesh is called “Mesh2”, which also has the small cube configuration with a uniform mesh to cover the loaded region. The mesh strategy for the front and back surfaces is the same as the front surface of hexahedron Mesh1, except for using triangles instead of rectangles. The top surface can be divided into four parts. One corresponds to the small cube. The second is a rectangle along the x -direction, with its width equal to cube size l . Its element size, along the x -direction, increases by a factor f outside the cube, but the element size in the y -direction is always h_s . This will result in extremely stretched elements when the block size L gets large. The rectangle along the y -direction has similar structure. In the fourth part of the top surface, the element size increases in both directions. The right surface has a similar structure.

Next, we will test different mesh parameters on Mesh1 and Mesh2. Three tests are defined in Table 5.2, w.r.t. h_c , the element size of the grid for the contact problem. TESTs 2 and 3 are based on a smaller cube near the origin than TEST 1. Moreover, TEST 3 has a smaller stretching factor f than TEST 2.

Mesh parameters	Cube size l	Cube element size h_s	Stretching factor f
TEST 1	$l/h_c = 2.5$	$h_c/h_s = 4$	$f = 1.5$
TEST 2	$l/h_c = 1$	$h_c/h_s = 4$	$f = 1.5$
TEST 3	$l/h_c = 1$	$h_c/h_s = 4$	$f = 1.2$

Table 5.2: *Settings of three tests.*

5.3.2 A^{zz} on Hexahedron Mesh1

We start by approximating the IC A^{zz} for the 10×10 contact grid. Fig. 5.5 shows results for TEST 1 using linear and quadratic elements, as block size L increases. Linear elements fail to achieve the target (5.12), denoted by the dashed line. However, quadratic elements exhibit very satisfactory convergence. *Therefore, quadratic elements will be employed in the subsequent tests.*

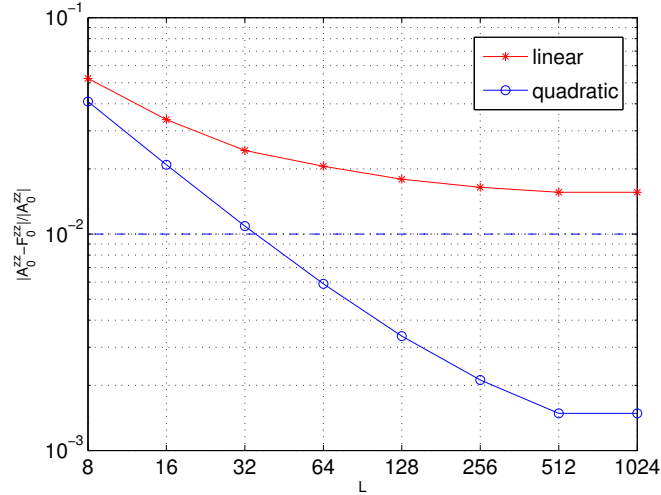


Figure 5.5: *TEST 1 on hexahedron Mesh1: relative errors at the origin, using linear and quadratic elements. The dashed line is the target accuracy defined by Eq. (5.12).*

Focusing on quadratic elements, the corresponding errors decrease as L is doubled, and they stabilize for $L \geq 512$. The reason for this is that the error consists of two parts. One is the error of the continuous model, which approximates the infinite domain, i.e. block size L approximates ∞ . The other is the FEM discretization approximating the true solution of the model on the finite domain. The former is dominant when L is small, until $L \geq 512$, when the latter error is remaining.

TEST 2 and TEST 3 use a smaller uniform cube, and the latter has a smaller stretching factor than the former. Fig. 5.6 shows the results of these two tests. Since TEST 3 employs a finer mesh, it results in smaller errors than TEST 2. At the same time, it requires more elements and thus CPU time. But the cost is acceptable. Comparing with TEST 1, we observe that TEST 3 can achieve

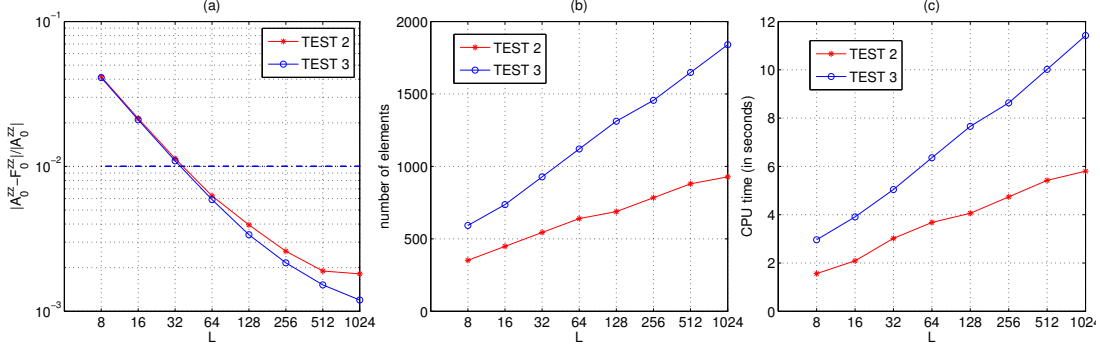


Figure 5.6: *TEST 2* and *TEST 3* on hexahedron *Mesh1* for a 10×10 contact grid. (a) Relative error at the origin for different domain sizes L . The dashed line indicates the target accuracy defined by Eq. (5.12). (b) Numbers of elements. (c) CPU time in seconds.

similar errors, however at a lower cost (since it has a coarser mesh).

We need to balance the accuracy and the computational cost. On the one hand, a coarser grid yields cheaper solutions, for instance, using larger stretching factors such as $f = 2$ and $f = 2.5$, or a smaller uniform cube $l/h_c = 0.5$. However, the corresponding accuracy may be unsatisfactory. A finer grid gives rise to a better accuracy but is more expensive.

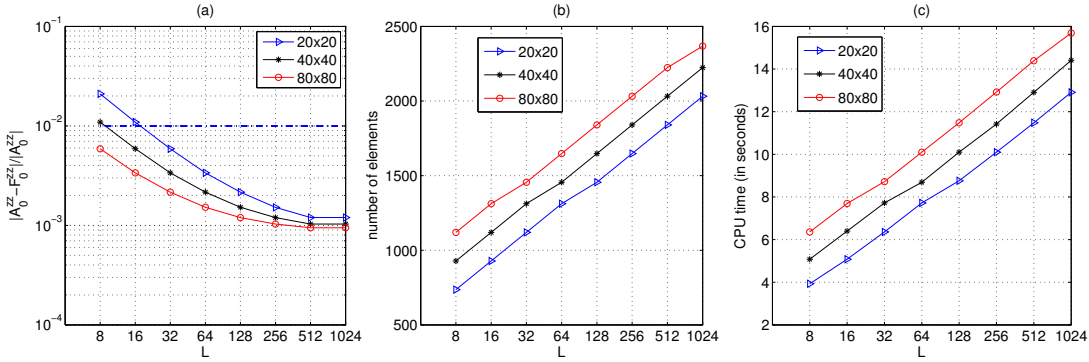


Figure 5.7: *TEST 3* on hexahedron *Mesh1*, for 20×20 , 40×40 and 80×80 contact grids. (a) Relative error at the origin for different domain sizes L . The dashed line indicates the target accuracy defined by Eq. (5.12). (b) Numbers of elements. (c) CPU time in seconds.

Fig. 5.7 gives the results using *TEST 3* on the 20×20 , 40×40 and 80×80 grids with h_c halved. It can be seen from Fig. 5.7(a) that all computations reach the target accuracy when $L/h_c \geq 80$ is satisfied. Moreover, when L/h_c is the same for these three grids, the corresponding errors are also similar. In other words, by shifting the line with red circles to the right, it will coincide with the line with black stars. The same observation can be found in Figs. 5.7(b) and (c).

The reason is that the parameters are determined based on the scaling factors w.r.t. the mesh size of the contact problem h_c .

5.3.3 A^{zz} on Prismatic Mesh2

We only show results with prismatic elements for a 10×10 contact problem in Fig. 5.8. It can be found that TEST 3 leads to the smallest errors when L increases. However, the corresponding cost is the highest (the CPU time with $L = 1024$ is so high that we do not show it in this figure). The errors by TEST 1 and TEST 2 are comparable. Since the latter requires fewer elements and less CPU time, it is the better choice for prismatic elements on Mesh2.

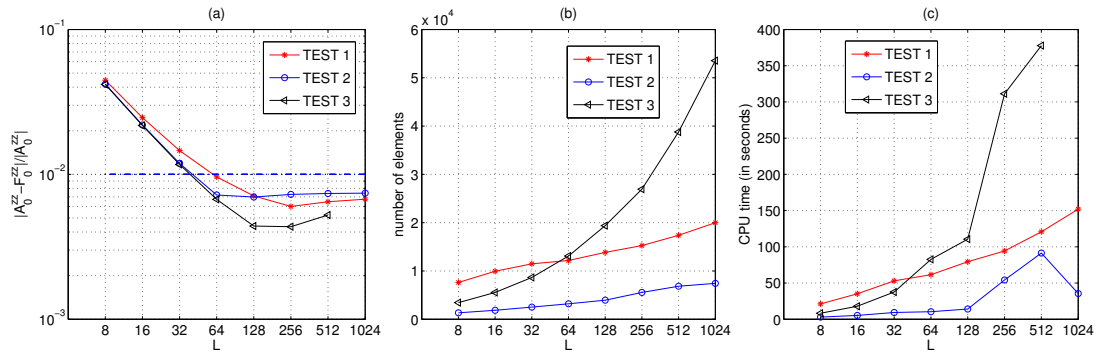


Figure 5.8: Three tests on prismatic Mesh2 for 10×10 contact grid. (a) Relative error at the origin for different domain sizes L . The dashed line indicates the target accuracy defined by Eq. (5.12). (b) Numbers of elements. (c) CPU time in seconds.

A difference with the behavior for the hexahedron mesh observed in Fig. 5.6 is that the errors at the origin with prismatic elements are decreasing less regularly when L increases, and stagnate at a higher level than when hexahedron elements are used. This is attributed to the large stretching of the prismatic elements when a large L value is used, which yields an ill-conditioned matrix from the FEM discretization, and hence the results are spoiled to some extent.

5.3.4 Discussion on Mesh Strategies

Based on the above results, we conclude regarding the choices of the mesh parameters in Table 5.3 for hexahedron Mesh1 and prismatic Mesh2. With these parameters, target accuracy (5.12) can be achieved. We can see that the same ranges apply for domain size L , cube size l and cube element size h_s between the hexahedron and the prismatic meshes. Stretching factor f is optimal in different ranges. The optimal choices for Mesh1 and Mesh2 are TEST 3 and TEST 2, respectively, as defined in Table 5.2.

Other conclusions are as follows:

- Quadratic elements are preferable on both Mesh1 and Mesh2.

Parameters	Domain size L	Cube size l	Cube element size h_s	Stretching factor f	Optimal
Mesh1	$L/h_c \geq 80$	$1 \leq l/h_c \leq 1.25$	$h_c/h_s \geq 4$	$1 \geq f \geq 1.5$	TEST 3
Mesh2	$L/h_c \geq 80$	$1 \leq l/h_c \leq 1.25$	$h_c/h_s \geq 4$	$1.5 \geq f \geq 2$	TEST 2

Table 5.3: Recommended mesh parameters for hexahedron Mesh1 and prismatic Mesh2.

- The mesh parameters in Table 5.3 are also satisfactory for other contact grids, such as the 20×20 , 40×40 , 80×80 meshes.
- The errors become stable when $L/h_c \geq 640$ on Mesh1, and when $L/h_c \geq 160$ on Mesh2.
- The same mesh strategies can be used to compute other ICs such as A^{xx} , A^{yy} , and A^{xy} , using the corresponding boundary conditions as defined in Table 5.1. Similar results are found for the numerical accuracy and the computational time.
- The same mesh strategies are also applicable to compute ICs when prescribing a unit bilinear load.
- Smaller errors can be attained with Mesh1 than with Mesh2, as described in Section 5.3.3. A proper value of domain size L needs to be chosen for the prismatic Mesh2.

5.4 Error Propagation in a Cattaneo Shift Problem

Errors in influence coefficients (ICs) can disturb the final solution of a contact problem. To investigate this, a Cattaneo shift contact problem [13] is solved, which is based on a sphere pressed on a plane and then shifted tangentially. This problem was specified in Section 5.2.1.1 of [51] and in Section 5.1 of [101], and has been solved in Chapter 4. Both sphere and plane are of the same elastic material, a quasi-identical case. In this case, the problem can be decoupled into a normal and tangential problem, since the results of the latter do not influence the former [51].

We solve these two subproblems separately in this section. Analytic and numerical ICs are used in the solution procedure, and the contact solutions are compared.

5.4.1 The Normal Problem

The normal problem is based on a normal force applied to the sphere on the plane so that a contact area is formed due to the deformation. The sphere and plane have the same shear modulus and Poisson's ratio, $G = 200 \text{ N/mm}^2$ and $\mu = 0.42$, respectively. The radii of the sphere are $R_x = R_y = 50 \text{ mm}$. The contact area is a circle with radius 1 mm . According to the Hertz theory [44],

the normal force is $F_z = 9.1954 N$, the approach, i.e., the maximal penetration, is $\delta = 0.02 mm$, and the maximal pressure is $p_{max} = 4.3905 N$.

In the numerical implementation, approach $\delta = 0.02 mm$ is prescribed. The contact area and pressure on it are solved for. We show the results for the normal force F_z , which is the integral of the pressure on the contact area. Moreover, the errors in the maximal pressure p_{max} and in the contact area are also analyzed. The potential contact area is set to $[-1.2, 1.2] \times [-1.2, 1.2] mm^2$. The undeformed distance function reads $h(x, y) = \frac{1}{2R_x}x^2 + \frac{1}{2R_y}y^2 - \delta$. The algorithm NORM [51] is used, with stopping criterion chosen as

$$\frac{\|d\|_{rms}}{\|h\|_{rms}} \leq 10^{-6}, \quad (5.13)$$

where d is defect, and the root-mean square norm is used.

Only IC A^{zz} is required in this normal problem. Both analytic and numerical ICs based on our model are employed in the numerical experiment. Relative errors w.r.t. the Hertzian solutions are shown. For example, the error in the normal force is defined by $\frac{|\bar{F}_z - F_z|}{F_z}$, where \bar{F}_z and F_z are forces provided by the numerical method and analytic solution, respectively.

We firstly examine the IC obtained by a unit piecewise constant load, then the ICs by a bilinear load. The results are also compared.

Analytic ICs

First of all, we use the analytic ICs in this normal problem. Relative errors in the normal force F_z , in the maximal pressure p_{max} and in the area of contact region are presented in Fig. 5.9. It is found that the errors in the normal force are already quite small. Errors in the maximal pressure show a linear reduction when the grid is refined. The errors in the contact area are the largest, which can be explained by the addition or removal of complete elements in the numerical treatment where the contact area changes.

ICs by a Piecewise Constant Load on Hexahedron Mesh1

In this experiment numerically computed ICs on Mesh1 are used. Fig. 5.10(a) shows relative errors in the resulting normal force for three different discretizations of the contact problems: 10×10 , 40×40 and 160×160 grids. It can be seen that these errors decrease as domain size L increases on the three grids. Moreover, the errors using 40×40 and 160×160 grids are very similar, and they are smaller than using the 10×10 grid when L increases.

Fig. 5.10(b) presents the relative errors in the maximal pressure p_{max} . As L increases, the error for the 10×10 decreases until it stagnates, then it increases and stabilizes. Such stagnation also occurs with the errors related to the 40×40 and 160×160 grids (the latter is not clearly shown since stagnation is at $L = 1024$. If L is larger, the curve will increase again). However, it is difficult to know

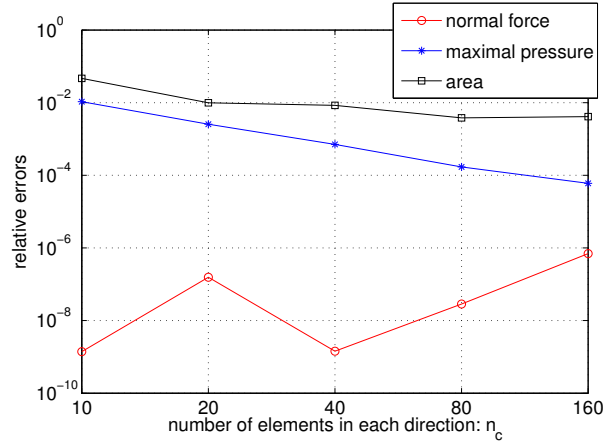


Figure 5.9: Normal problem with analytic ICs: relative errors in the normal force F_z , in the maximal pressure p_{max} and in the contact area. The x-axis shows the number of element in each direction of contact grid $n_c \times n_c$.

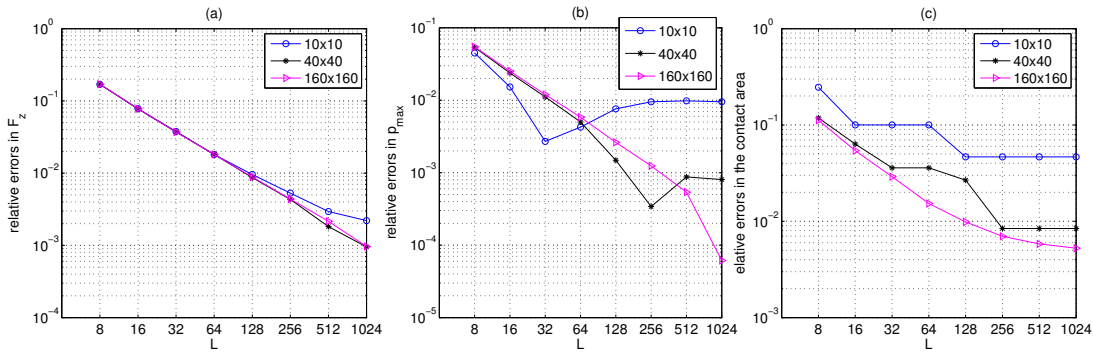


Figure 5.10: Normal problem with ICs by a piecewise constant load on hexahedron Mesh1: (a) errors in the normal force, (b) errors in the maximal pressure, (c) errors in the contact area.

beforehand the value of L for which the maximal pressure stagnates. Moreover, when L reaches the largest value, i.e. $L = 1024$, the finer the grid is, the smaller the resulting error.

Errors in the contact area are presented in Fig. 5.10(c), where the finest grid always yields the smallest errors for different values of L . Moreover, as L increases, errors on these three grids reduce and then stagnate.

Based on the above discussion, we come to the conclusion that it is optimal to choose L as large as possible on this mesh.

ICs by a Piecewise Constant Load on Prismatic Mesh2

ICs computed on prismatic Mesh2 are employed here. In Fig. 5.11(a), errors decrease as L increases, and then they increase a bit when L gets larger. Using

40×40 and 160×160 grids results in smaller errors than the 10×10 grid. Moreover, these two fine grids exhibit the same errors. This situation is the same as when using the IC on Mesh1 in Fig. 5.10(a).

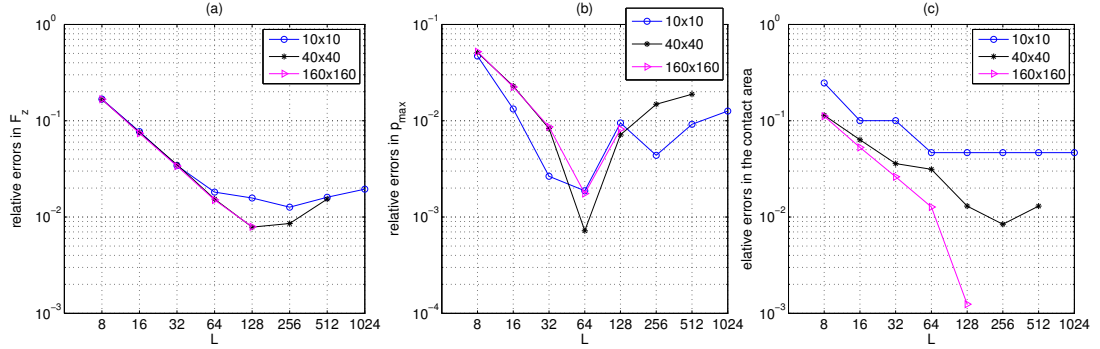


Figure 5.11: Normal problem with ICs by a piecewise constant load on prismatic Mesh2: (a) errors in the normal force, (b) errors in the maximal pressure, (c) errors in the contact area.

Fig. 5.11(b) presents a stagnation in each curve for the error in the maximal pressure p_{max} . Errors in the contact area in Fig. 5.11(c) indicate that for a fixed value of L , the finer the discretization, the smaller the error will be.

Notice that in Fig. 5.11, the results on 40×40 and 160×160 grids are lacking when L gets very large. The reason is that the ICs are not computed in this case. In fact, highly distorted elements occur on Mesh2. As mentioned in Section 5.3.3, large L values give rise to over-stretched elements and ill-conditioned FEM matrices. Therefore, it is important to carefully find a proper value for L on Mesh2, neither too large nor too small.

ICs by a Bilinear Load

Here, the ICs have been computed also with the bilinear load as described in Remark 5.1. Since the analytic values cannot be accessed easily, no comparison was made in Section 5.3 between analytic ICs and numerical ICs. However, it is possible to use these ICs in the contact model and study the propagation of errors, as the above discussion for the ICs computed using the piecewise constant load. The resulting figures are very similar to Figs. 5.10 and 5.11, and are therefore omitted for brevity.

Discussion on the Normal Problem

We compare the influence of all ICs on the normal contact problem. Numerical ICs are obtained on Mesh1 with $L = 1024$ and on Mesh2 with $L = 128$. Relative errors in the normal force are shown in Fig. 5.12(a). We find that, on the one hand, both piecewise constant and bilinear loads result in similar errors on a very fine discretization. On the other hand, errors on Mesh1 are smaller than

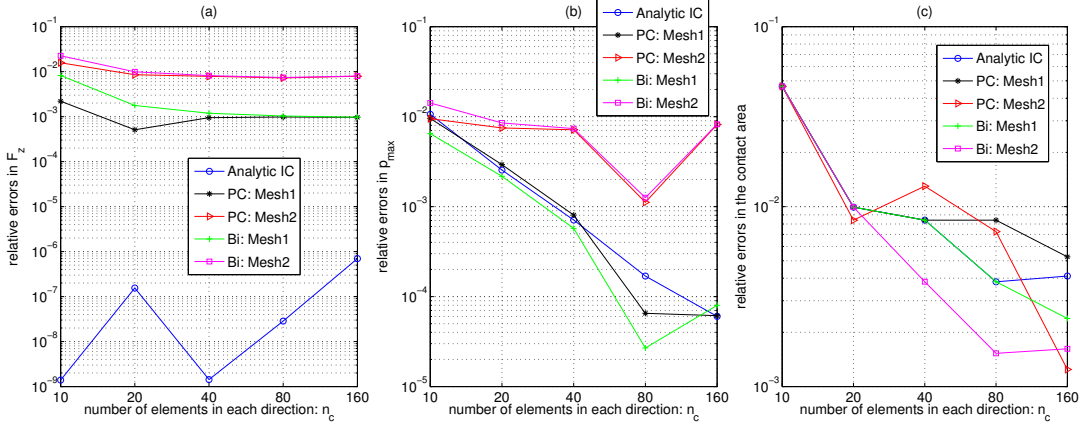


Figure 5.12: Normal problem with different ICs: (a) errors in the normal force, (b) errors in the maximal pressure, (c) errors in the contact area. Here, “PC” and “Bi” denote piecewise constant and bilinear loads, respectively.

those on Mesh2. For instance, on the finest 160×160 grid, the former reaches a relative error of 0.1% and the latter around 0.8%. Moreover, these numerical ICs cannot yield normal forces as accurate as the analytic ICs, which results in errors smaller than 10^{-6} .

Fig. 5.12(b) gives the errors in p_{max} , where the Mesh1 results in a similar accuracy as using analytic ICs. These errors are again smaller than those on Mesh2. Errors in the contact area are presented in Fig. 5.12(c), where we notice that the ICs based on a bilinear load on Mesh2 can even result in better accuracy than the analytic ICs.

5.4.2 The Tangential Problem

In the tangential problem, the contacting sphere is shifted in the x -direction by a tangential force $F_x = (7/8)\mu F_z$. Here, $\mu = 0.4$ is the friction coefficient of Coulomb’s frictional law. The sphere sticks to the plane where the tangential traction is small. Local sliding occurs when the traction reaches the frictional traction bound. According to Cattaneo’s theory [13, 101, 51], the adhesion area is a circle with radius $\theta = 0.5$. In this case, the rigid shift of the sphere is given by $w_x = 0.00817 \text{ mm}$.

In the implementation, we have prescribed the rigid shift $w_x = 0.00817 \text{ mm}$ and friction coefficient $\mu = 0.4$. The tangential traction, the adhesion and slip areas are to be solved. Integration of the tangential traction over the contact area yields tangential forces F_x, F_y . The required ICs are $A^{xx}, A^{yx}, A^{yy}, A^{xy}$, where $A^{yx} = (A^{xy})^T$. The solver is the TangCG algorithm, proposed in [115] and presented in Chapter 4. The iterations of TangCG terminate when $\frac{\|\mathbf{p}_t^{k+1} - \mathbf{p}_t^k\|}{\|\mathbf{p}_t^{k+1}\|} < 10^{-6}$, where \mathbf{p}_t is the tangential traction consisting of x - and y -components. The solution procedure starts with the computed contact area and the corresponding

pressures resulting from the normal problem where the same strategies are used to compute the ICs for both the normal and tangential problems. This means that, for instance, if A^{zz} for the normal problem is obtained by a bilinear load on Mesh1, this strategy is also employed to compute A^{xx} , A^{xy} , A^{yy} for the tangential problem.

We only compare tangential force F_x in this case. The analytic Cattaneo solution for F_x is not completely relevant here, since it does not involve the force in the lateral direction w.r.t. the shift direction. Therefore, rather than using this solution as the reference for comparison, we use $F_x = 3.2216 \text{ N}$, which is obtained by the TangCG algorithm with analytic ICs when $h_c \rightarrow 0$.

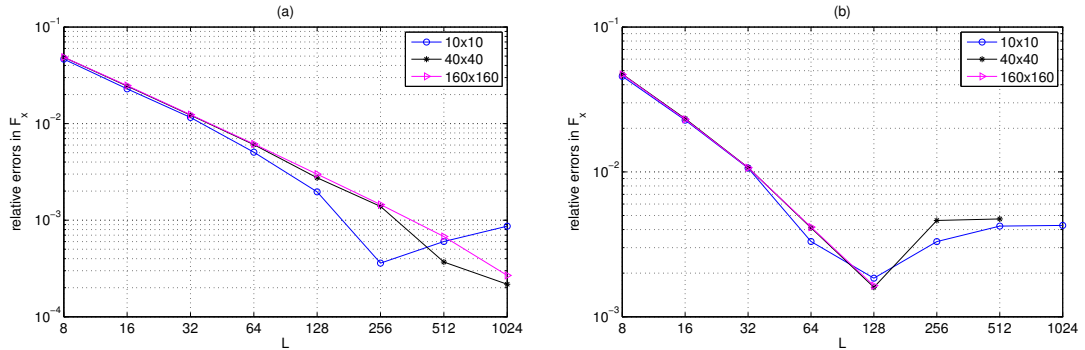


Figure 5.13: Tangential problem: errors in tangential force F_x , with ICs by a piecewise constant load on (a) hexahedron Mesh1 and on (b) prismatic Mesh2.

First, we use the ICs computed by a unit constant load on hexahedron Mesh1, and the resulting errors in the tangential force are shown in Fig. 5.13(a). These errors decrease as domain size L increases. Errors on the 10×10 grid sometimes are smaller than those on finer grids. This may be due to the fact that a finer grid requires a larger number of ICs and hence errors may accumulate in the evaluation. Fig. 5.13(b) presents errors when the ICs are computed on prismatic Mesh2. As the domain size increases, these errors first are reduced, and then increase somewhat again. The same observation can be found when the ICs are obtained by a bilinear load, so we do not show the corresponding results here.

To compare these ICs, again we choose $L = 1024$ for Mesh1 and $L = 128$ for Mesh2. A piecewise constant load as well as a bilinear load are used. Fig. 5.14 shows the errors in the tangential force with the resulting ICs. Using analytic IC yields a linear reduction of errors, and the numerical ICs obtained by hexahedron Mesh1 give rise to smaller errors than those by prismatic Mesh2. For instance, the former achieves around 0.03% and the latter possesses 0.17% on the 160×160 grid. The piecewise constant load and bilinear load yield similar results. However, we conclude that “engineering accuracy” can also be achieved on the prismatic mesh.

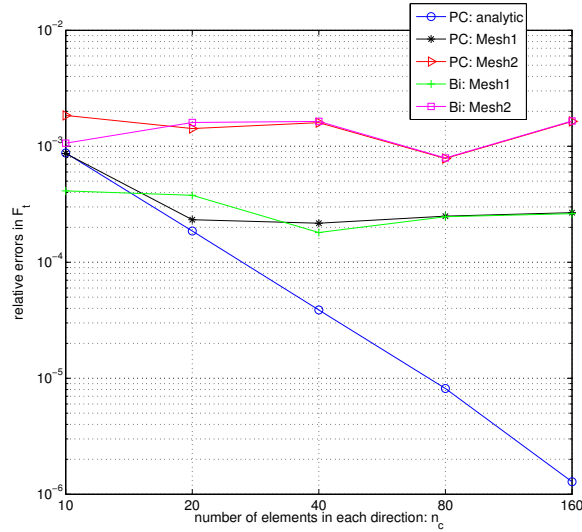


Figure 5.14: *Tangential problem: errors in tangential force F_x with different ICs. Here, “PC” and “Bi” denote piecewise constant and bilinear loads, respectively.*

5.5 Conclusion

In this chapter, we studied the numerical calculation of the influence coefficients (ICs) that are used in fast solvers for concentrated contact problems. These ICs are displacements caused by a unit load, either a piecewise constant load or a bilinear load, depending on the representation of the unknown tractions. To compute these ICs, an elastic model was provided, including governing equations and boundary conditions for different ICs. The finite element method is employed via package SEPRAN [81]. Considering the accuracy and computational cost, we recommend hexahedron Mesh1 and prismatic Mesh2, with their own specific mesh parameters after several numerical tests. The former is easily extended to a large computational domain that approximates the half-space. The latter is more generally applicable, while a proper domain size needs to be chosen carefully in order to prevent over-stretched elements.

The numerically computed ICs are incorporated into the solution procedure of a Cattaneo shift problem, which is decoupled into a normal and a tangential problem. The effects of these ICs on the contact solutions are discussed. ICs obtained on hexahedron Mesh1 show better performance than those on prismatic Mesh2. However, the prismatic mesh is favorable for engineering applications. The performance does not differ much between ICs that are computed by a piecewise constant load or a bilinear load. This means that using piecewise constant and bilinear representations for unknowns results in the same accuracy, in agreement with the corresponding statements in [96, 51].

The work in this chapter provided a guidance for developing fast solvers for conformal contact problems. Similar model and meshing strategies can be used

there to compute ICs on curved domains. Similar errors in the ICs and similar error propagation may be expected, such that meshing parameters may be chosen along the lines of the results presented here.

Chapter 6

Conclusions and Outlook

6.1 Introduction

The objective of our work in this thesis was to develop fast numerical solvers for concentrated elastic contact problems, with the application of wheel-rail contact in the railway industry as the motivating example. The contact problem concerns the elastic deformation of two bodies when they are brought into contact and further into relative motion. Generally speaking, it can be modeled by a 3D boundary value problem, consisting of the basic elastic equations and specific boundary conditions for the contact phenomenon. In this thesis, we focused on the contact between bodies of linear homogenous elastic material. In the case of concentrated contact where the contact area is small compared to the dimensions of the contacting bodies, models and discretization have been provided by a variational formulation [51], based on the boundary element method (BEM). The BEM transforms the 3D boundary value problem to a 2D boundary integral problem, where the corresponding Green's function expresses the relation between the tractions and deformations on the surface of the half-space. This method only requires discretization of the boundary region, which reduces the computational cost.

The variational formulation represents a convex optimization problem, with linear or nonlinear constraints for different types of contact problems. The existence of a unique solution for this formulation has been provided by the corresponding Karush-Kuhn-Tucker (KKT) conditions. The boundary integral gives rise to a system with symmetric, positive definite coefficient matrices of dense Toeplitz structure, for which fast computing techniques such as the fast Fourier transform (FFT) have been explored. Furthermore, an active set strategy has been incorporated to deal with the contact conditions. The research in the present thesis has resulted in efficient numerical techniques and fast algorithms in this context.

In this chapter, we summarize and conclude our work in Section 6.2. Suggestions for future work are provided in Section 6.3.

6.2 Conclusions

Conclusion on Chapter 2

Chapter 2 was devoted to the 3D normal contact problem, which was based on contact caused by a force applied perpendicularly to the contact surface. We have searched for the contact area, which was formed due to the elastic deformation, and the pressures on it. As the model, a linear complementarity problem was used, where the complementarity conditions stated that the gap between the two body surfaces should be equal to zero in the contact area and positive outside. Moreover, the pressures were compressive (≥ 0) in the contact region but vanished outside.

We have presented a full multigrid (FMG) method, that combines an active set strategy, a multigrid (MG) approach and a nested iteration technique. The active set strategy partitions the potential contact region into a contact area and an exterior area, and iteratively modifies them based on the solution of the governing equations defined on the current contact area. The governing equations are solved approximately by one multigrid cycle. It employs a damped Gauss-Seidel smoother to smooth highly oscillatory error modes with the coefficient matrices being dense, symmetric and positive definite, and, moreover, of block Toeplitz with Toeplitz block (BTTB) structure. Different from standard multigrid is that the contact area is restricted to coarser grids as well.

This FMG method has been implemented for a smooth Hertzian contact. We examined MG as a solver. Then, for the complete problem, we checked the performance of the active set strategy with MG, and further the improvement by the nested iteration. After this, the FMG method has been employed for a rough surface contact. The results indicate the efficiency of MG for the governing equations, and of using one MG cycle to modify the contact areas in each active set iteration. The FMG scheme provides accurate initial pressures for solving the governing equations on the target grid, and a highly satisfactory initial contact area which resembles the converged contact area better than the original method. The FMG method has been compared with the original NORM [51] method in terms of the number of iterations and work units. Those numbers by the former are low and sometimes even reduced when the problem size increases. Moreover, the FMG method attains the discretization error accuracy in a few iterations on the target grid. From the experiment for the rough surface contact, the FMG also demonstrates robustness for problems on irregular computational domains.

Concerning the BTTB coefficient matrices, a Gauss-Seidel smoother cannot benefit from fast computing techniques for matrix-vector products, such as the fast Fourier transform (FFT) or multi-level multi-integration techniques. The former has been explored by Vollebregt in [100] for the normal contact, and in our work it has also been incorporated for tangential contact problems in the subsequent chapters.

Conclusion on Chapter 3

In Chapter 3, we started to work on the tangential contact problem, where frictional effects are involved when the two contacting bodies are brought into relative motion. The two bodies stick to each other when the tangential tractions do not exceed the traction bound. Local slip occurs when the traction bound is attained. In such problems, the adhesion and slip areas, and the corresponding tangential tractions are to be found. We have applied locally Coulomb's friction law to determine the traction bound, which is equal to the pressure multiplied by a friction coefficient. The model for the 3D tangential problem is based on inequalities and nonlinear constraints, which makes it quite complicated.

Therefore, we have first focused on a 2D line contact problem in this chapter. The problem considers a cylinder shifted over a flat surface. The friction coefficient is assumed infinity so that no slip occurs and further the contact conditions are excluded. Due to the case of plane strain, quantities along the cylinder axle direction are ignored. As a result, the potential contact area is discretized by an $n \times 1$ grid, yielding a Toeplitz coefficient matrix for the governing equation.

We have developed an FFT smoother incorporated in an MG method. This smoother is based on a Richardson iteration with a Toeplitz preconditioner that approximates the inverse of the coefficient matrix. We have illustrated the approach to construct this smoother by FFTs, and numerically analyzed the smoothing property. It was shown that this smoother reduces many components of the error, but enlarges several smooth components. This causes convergence difficulties for the MG method. To overcome this drawback, techniques such as subdomain deflation and row sum modification (RSM) have been studied. They are incorporated into the FFT smoother within the MG method, to solve two model problems that have constant and spatially varying right-hand sides. The resulting iteration numbers and convergence factors indicate that MG with the FFT+RSM smoother is a preferable choice. Among the three V-cycles, the V(0,1)-cycle appears favorable regarding the CPU time. The RSM approach has also been adopted for a wavy contact where the Toeplitz structure is lost, but the MG with FFT+RSM still works efficiently. Furthermore, the FFT+RSM method has been applied as a stand-alone solver, showing rapid convergence. Both methods related to the RSM have complexity $\mathcal{O}(n \log(n))$, with n the number of unknowns.

Conclusion on Chapter 4

In Chapter 4, we have proposed the TangCG algorithm for 3D tangential contact problems, where a nonlinear constrained optimization formulation is solved. Different from conventional solvers, the TangCG method applies two types of variables: tractions in the adhesion area and angles in the slip area. On each slip element, the traction is defined at a circle with radius equal to the traction bound, which automatically satisfies the condition that the magnitude of the tractions should equal the traction bound in the slip area. This results in non-

linear governing equations, and a nonlinear conjugate gradient (NLCG) method is employed, with a diagonal preconditioner in order to bring the quantities of the adhesion and the slip areas to the same scales. An active set strategy is combined again to deal with the contact conditions. We have described this algorithm in detail based on a simple problem, and also illustrated it by figures. The convergence of the NLCG and a linear CG methods have been discussed as well.

We have tested the TangCG algorithm for various Cattaneo shift problems carrying different amounts of slip. Since each governing equation was solved approximately by K NLCG iterations, first of all, we analyzed different values of K and found $K = 3$ to be optimal. The convergence behavior was compared between TangCG with and without preconditioning, where it turns out that preconditioning dramatically accelerates the convergence. Moreover, this method has been compared with the existing ConvexGS method [95], in terms of iteration numbers and CPU time. The TangCG method achieved a complexity of $\mathcal{O}(n^{1.7})$ while the ConvexGS showed $\mathcal{O}(n^{2.3})$, with n the number of unknowns in the contact area.

Conclusion on Chapter 5

We have developed numerical solvers for concentrated contact problems where the BEM was used and the corresponding influence coefficients (ICs) represent the relation between tractions and deformations in the contact region. In Chapter 5, we aimed at extending the range of applicability of the contact approach towards the conformal contact problem, which possesses a possibly curved and larger contact area. Hence the half-space approach is not valid, which gives difficulties for fast solvers based on the BEM formulation since the ICs are not analytically known.

In this chapter, we have focused on a concentrated contact setting where analytic ICs and contact solutions are available. We have proposed an elastic model to numerically compute the ICs, based on the half-space approach. This model considers a unit load applied on a 3D block. It is formed with the basic elastic equations in a quasi-static case, and different boundary conditions for different ICs. Moreover, this model employs a large computational domain to approximate the semi-infinite half-space. The finite element method in the package SEPRAN is used. With this model, the ICs for the normal contact problem were computed. Considering the accuracy and computational cost, we recommend a specific hexahedron mesh as well as a prismatic mesh. The former is easily applied on a large computational domain and results in small errors. The latter is generally applicable, however, the domain size has to be chosen properly and with care. Suggestions of the involved mesh parameters were provided, moreover, quadratic elements are found preferable over linear elements. The same meshing strategies can be used to obtain other ICs.

The computed ICs were incorporated into the solution methods for a Cattaneo

shift problem, that is decoupled into a normal and a tangential problem. In the normal problem we compared the contact solutions on different contact grids, with the ICs obtained by a piecewise constant load on a hexahedron mesh and on a prismatic mesh. The errors in the normal force and in the contact area decrease as the model domain size increases. At the same time, the errors in the maximal pressure decrease and then increase a bit. Similar behavior was found with ICs based on a bilinear load on these two meshes. The comparison of the errors in the normal force shows that both piecewise constant and bilinear loads give rise to similar errors on very fine discretizations. Moreover, the hexahedron mesh results in smaller errors than the prismatic mesh. Using numerical ICs cannot achieve the accuracy of using analytic ICs. However, concerning the errors in the maximal pressure, the ICs on the hexahedron mesh can yield similar accuracy as analytic ICs, and they are smaller than using the ICs on the prismatic mesh.

In the tangential contact problem, adopting the ICs obtained by piecewise constant and bilinear loads shows similar behavior when the domain size increases: errors in the tangential force decrease for ICs on the hexahedron mesh, while they increase a bit after reaching their smallest values using ICs on the prismatic mesh. Choosing specific domain sizes for the two meshes and comparing the solution errors with these numerical ICs, we found that the ICs obtained on the hexahedron mesh show a better convergence than those on the prismatic mesh. The latter ones can achieve however the requested engineering accuracy. As expected, the behavior does not differ much between ICs computed by a piecewise constant load and a bilinear load, in agreement with the corresponding statement in [96, 51].

6.3 Outlook

The present work in this thesis inspires towards the following interesting topics for future research:

- For the Toeplitz matrix arising from the 2D tangential contact problem, a multigrid method with the FFT+RSM smoother has been proposed in Chapter 3. This method may be applied to Toeplitz matrices in other applications, such as image deblurring and signal processing (e.g. [34, 79]).
- The 3D contact problem requires to solve a block Toeplitz matrix with Toeplitz blocks (BTTB), for which the multigrid method with the FFT smoother needs to be extended. The corresponding FFT smoother can be constructed, adopting the FFT preconditioner proposed by Vollebregt in [100]. More attention should be given to a technique for the increasing error component resulting from the applications of the smoother, similarly as the RSM. The resulting method, incorporated into a full multigrid method as presented in Chapter 2, can yield a fast solver for 3D normal contact problems. Moreover, it can also be applied to BTTB matrices in other applications as mentioned above.

- The TangCG algorithm proposed in Chapter 4 has been applied to transient rolling contact problems where the deformation and the tractions at the previous time are known beforehand. A big challenge is to develop fast solvers for the steady-state rolling contact, with a computing speed similar as for TangCG. In steady-state rolling contact problems, tractions are time-independent and to be solved, but the particles on each contacting surface changes with time. This gives rise to different coefficient matrices in two successive time steps. The resulting governing matrix is non-symmetric and the positive definite property is not guaranteed [97]. There is no precise variational formulation for such problems [51, 95].

As a solution strategy, the DUVOROL approach [51] has been used, which computes tractions at each time until the difference of the tractions is small. Another method is SteadyGS [97], which reformulates the problem to keep the symmetric and positive definite structure for the coefficient matrix, and a nonlinear Gauss-Seidel process is employed. However, these methods are not satisfactorily fast and more efficient methods are demanded.

- The future research can focus on the extension of the work in this thesis towards conformal contact problems, for which fast solvers are also demanded. Chapter 5 can be a good reference since the resulting model and meshing strategies can be used to compute influence coefficients, and similar error propagation may also be expected. Moreover, it is worthwhile to combine the fast solvers presented for concentrated contact problems in Chapters 2, 3 and 4, with these computed influence coefficients for conformal contact problems.
- In this thesis, we have focused on model problems mainly. It may be very interesting to see the performance of all solvers and resulting strategies in true rail-wheel engineering applications and in other involved engineering applications.

Bibliography

- [1] G. S. Ammar. Classical foundations of algorithms for solving positive definite Toeplitz equations. *Calcolo*, 33(1-2):99–113, 1996.
- [2] T. Andersson. The boundary element method applied to two-dimensional contact problems with friction. In *Proceedings of the 3rd International Seminar on Recent Advances in Boundary Element Methods*. Springer, Berlin, 1981.
- [3] K. E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, UK, 1997.
- [4] J. B. Ayasse and H. Chollet. Wheel–rail contact. In S. Iwnicki, editor, *Handbook of Railway Vehicle Dynamics*, chapter 4, pages 85–120. CRC Press, Boca Raton, 2006.
- [5] J. R. Barber. *Elasticity*. Kluwer, Dordrecht, The Netherlands, 1992.
- [6] M. Barboteu, P. Alart, and M. Vidrascu. A domain decomposition strategy for nonclassical frictional multi-contact problems. *Computer Methods in Applied Mechanics and Engineering*, 190(37):4785–4803, 2001.
- [7] M. Bolten, M. Donatelli, and T. Huckle. Aggregation-based multigrid methods for circulant and Toeplitz matrices. Bergische Universität Wuppertal, 2012. Preprint BUW-IMACM 12/10.
- [8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, UK, 2004.
- [9] A. Brandt and C. W. Cryer. Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems. *SIAM Journal on Scientific and Statistical Computing*, 4(4):655–684, 1983.
- [10] A. Brandt and A. A. Lubrecht. Multilevel matrix multiplication and fast solution of integral equations. *Journal of Computational Physics*, 90(2):348–370, 1990.

- [11] F. Bucher, K. Knothe, and A. Theiler. Normal and tangential contact problem of surfaces with measured roughness. *Wear*, 253(1):204–218, 2002.
- [12] F. W. Carter. On the action of a locomotive driving wheel. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 112, pages 151–157. The Royal Society, 1926.
- [13] C. Cattaneo. Sul contatto di due corpi elastici: distribuzione locale degli sforzi. *Rendiconti dell' Accademia nazionale dei Lincei*, 27(3424348):4344436, 1938.
- [14] R. H. Chan, Q. S. Chang, and H. W. Sun. Multigrid method for ill-conditioned symmetric Toeplitz systems. *SIAM Journal on Scientific Computing*, 19(2):516–529, 1998.
- [15] R. H. F. Chan and X. Q. Jin. *An Introduction to Iterative Toeplitz Solvers*. SIAM, Philadelphia, USA, 2007.
- [16] W. W. Chen and Q. J. Wang. A numerical model for the point contact of dissimilar materials considering tangential tractions. *Mechanics of Materials*, 40(11):936–948, 2008.
- [17] M. Ciavarella. Frictional energy dissipation in hertzian contact under biaxial tangential harmonically varying loads. *The Journal of Strain Analysis for Engineering Design*, 49(1):27–32, 2014.
- [18] M. Donatelli. An algebraic generalization of local Fourier analysis for grid transfer operators in multigrid based on Toeplitz matrices. *Numerical Linear Algebra with Applications*, 17(2-3):179–197, 2010.
- [19] Z. Dostál, T. Kozubek, A. Markopoulos, T. Brzobohatý, V. Vondrák, and P. Horyl. A theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction. *Computer Methods in Applied Mechanics and Engineering*, 205:110–120, 2012.
- [20] G. Duvaut and J. L. Lions. *Les inéquations en mécanique et en physique*. Dunod, Paris, 1972.
- [21] J. R. Dydo and H. R. Busby. Elasticity solutions for constant and linearly varying loads applied to a rectangular surface patch on the elastic half-space. *Journal of Elasticity*, 38(2):153–163, 1995.
- [22] R. Enblom. Deterioration mechanisms in the wheel–rail interface with focus on wear prediction: a literature review. *Vehicle System Dynamics*, 47(6):661–700, 2009.
- [23] M. J. Fagan and J. McConnachie. A review and detailed examination of non-layered conformal contact by finite element analysis. *The Journal of Strain Analysis for Engineering Design*, 36(2):177–195, 2001.

- [24] F. Fichera. Problemi elastostatici con vincoli unilaterali: II problema di signorini con ambigue condizioni al contorno. *Mem. Accad. Naz. Lincei.*, 8(7):91, 1964.
- [25] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *The Computer Journal*, 7(2):149–154, 1964.
- [26] J. Frank and C. Vuik. On the construction of deflation-based preconditioners. *SIAM Journal on Scientific Computing*, 23(2):442–462, 2001.
- [27] H. Fromm. Berechnung des Schlupfes beim Rollen deformierbarer Scheiben (English title: Analysis of creep of deformable discs during rolling). *Zeitschrift für angewandte Mathematik und Mechanik*, 7:27–58, 1927.
- [28] J. Á. González and R. Abascal. Solving 2D rolling problems using the NORM–TANG iteration and mathematical programming. *Computers & Structures*, 78(1):149–160, 2000.
- [29] S. L. Grassie. Rolling contact fatigue on the British railway system: treatment. *Wear*, 258(7):1310–1318, 2005.
- [30] R. M. Gray. Toeplitz and circulant matrices: A review. *Communications and Information Theory*, 2(3):155–239, 2005.
- [31] J. A. Greenwood and J. B. P. Williamson. Contact of nominally flat surfaces. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 295(1442):300–319, 1966.
- [32] W. Habchi. *A full-system finite element approach to elastohydrodynamic lubrication problems: application to Ultra-Low-Viscosity Fluids*. PhD thesis, University of Lyon, France, 2008.
- [33] W. Hackbusch. *Multi-grid Methods and Applications*. Springer-Verlag, Berlin, Heidelberg, 1985.
- [34] P. C. Hansen, J. G. Nagy, and D. P. O’leary. *Deblurring Images: Matrices, Spectra, and Filtering*. SIAM, Philadelphia, USA, 2007.
- [35] P. W. Hemker and H. Schippers. Multiple grid methods for the solution of Fredholm integral equations of the second kind. *Mathematics of Computation*, 36(153):215–232, 1981.
- [36] H. Hertz. Über die Berührung fester elastischer Körper. *Journal für Reine und Angewandte Mathematik*, 92:156–171, 1882.
- [37] Y. Z. Hu and K. Tonder. Simulation of 3-D random rough surface by 2-D digital filter and Fourier analysis. *International Journal of Machine Tools and Manufacture*, 32(1):83–90, 1992.

- [38] T. Huckle. Circulant and skew circulant matrices for solving Toeplitz matrix problems. *SIAM Journal on Matrix Analysis and Applications*, 13(3):767–777, 1992.
- [39] S. Hübner, G. Stadler, and B. I. Wohlmuth. A primal-dual active set algorithm for three-dimensional contact problems with Coulomb friction. *SIAM Journal on Scientific Computing*, 30(2):572–596, 2008.
- [40] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Englewood Cliffs, NJ, 1987.
- [41] M. Ignesti, A. Innocenti, L. Marini, E. Meli, and A. Rindi. A numerical procedure for the wheel profile optimisation on railway vehicles. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 228(2):206–222, 2014.
- [42] K. L. Johnson. The effect of a tangential contact force upon the rolling motion of an elastic sphere on a plane. *Journal of Applied Mechanics*, pages 339–346, 1958.
- [43] K. L. Johnson. The effect of spin upon the rolling motion of an elastic sphere on a plane. *Journal of Applied Mechanics*, 25(3):332–338, 1958.
- [44] K. L. Johnson. *Contact Mechanics*. Cambridge University Press, UK, 1985.
- [45] F. Jourdan, P. Alart, and M. Jean. A Gauss-Seidel like algorithm to solve frictional contact problems. *Computer Methods in Applied Mechanics and Engineering*, 155(1):31–47, 1998.
- [46] J. J. Kalker. *On the rolling contact of two elastic bodies in the presence of dry friction*. PhD thesis, Delft University of Technology, 1967.
- [47] J. J. Kalker. Simplified theory of rolling contact. *Delft Progress Rep.*, 1(1):1–10, 1973.
- [48] J. J. Kalker. A fast algorithm for the simplified theory of rolling contact. *Vehicle System Dynamics*, 11(1):1–13, 1982.
- [49] J. J. Kalker. The principle of virtual work and its dual for contact problems. *Ingenieur-Archiv*, 56(6):453–467, 1986.
- [50] J. J. Kalker. Contact mechanical algorithms. *Communications in Applied Numerical Methods*, 4(1):25–32, 1988.
- [51] J. J. Kalker. *Three-dimensional Elastic Bodies in Rolling Contact*. Kluwer Academic Publisher, The Netherlands, 1990.
- [52] J. J. Kalker, F. M. Dekking, and E. A. H. Vollebregt. Simulation of rough, elastic contacts. *Journal of Applied Mechanics*, 64(2):361–368, 1997.

- [53] N. Kikuchi and J. T. Oden. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*. SIAM, Philadelphia, USA, 1988.
- [54] K. Knothe. History of wheel/rail contact mechanics: from Redtenbacher to Kalker. *Vehicle System Dynamics*, 46(1-2):9–26, 2008.
- [55] F. Kosior, N. Guyot, and G. Maurice. Analysis of frictional contact problem using boundary element method and domain decomposition method. *International Journal for Numerical Methods in Engineering*, 46(1):65–82, 1999.
- [56] R. Krause. A nonsmooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency. *SIAM Journal on Scientific Computing*, 31(2):1399–1423, 2009.
- [57] H. W. Kuhn. Nonlinear programming: a historical view. *Traces and Emergence of Nonlinear Programming*, pages 393–414, 2014.
- [58] T. A. Laursen. *Computational Contact and Impact Mechanics: Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis*. Springer Science & Business Media, 2002.
- [59] J. Li and E. J. Berger. A Boussinesq-Cerruti solution set for constant and linear distribution of normal and tangential load over a triangular area. *Journal of Elasticity*, 63(2):137–151, 2001.
- [60] Z. Li. *Wheel-rail rolling contact and its application to wear simulation*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2002.
- [61] C. S. Liu, K. Zhang, and R. Yang. The FEM analysis and approximate model for cylindrical joints with clearances. *Mechanism and Machine Theory*, 42(2):183–197, 2007.
- [62] K. W. Man. *Contact Mechanics Using Boundary Elements*. Computational Mechanics Publisher, Boston MA, 1994.
- [63] B. Michel, A. Bernard, A. Bietsch, E. Delamarche, M. Geissler, D. Juncker, H. Kind, J. P. Renault, H. Rothuizen, H. Schmid, et al. Printing meets lithography: Soft approaches to high-resolution patterning. *IBM Journal of Research and Development*, 45(5):697–719, 2001.
- [64] R. D. Mindlin. Compliance of elastic bodies in contact. *Journal of Applied Mechanics*, 16:259–268, 1949.
- [65] K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Heldermann Berlin, 1988.
- [66] M. K. Ng. *Iterative Methods for Toeplitz Systems*. Oxford University Press, UK, 2004.

- [67] R. A. Nicolaidis. Deflation of conjugate gradients with applications to boundary value problems. *SIAM Journal on Numerical Analysis*, 24(2):355–365, 1987.
- [68] D. Nowell. Analysis of a rocking and walking punch-part II: General numerical solution using quadratic programming. *Journal of Applied Mechanics*, 71(2):234–239, 2004.
- [69] I. P. D. Panagiotopoulos. A nonlinear programming approach to the unilateral contact, and friction-boundary value problem in the theory of elasticity. *Ingenieur-Archiv*, 44(6):421–432, 1975.
- [70] N. Patir. A numerical procedure for random generation of rough surfaces. *Wear*, 47(2):263–277, 1978.
- [71] A. Pieringer, W. Kropp, and D. J. Thompson. Investigation of the dynamic contact filter effect in vertical wheel/rail interaction using a 2D and a 3D non-hertzian contact model. *Wear*, 271(1):328–338, 2011.
- [72] E. Polak and G. Ribière. Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, 3(R1):35–43, 1969.
- [73] I. A. Polonsky and L. M. Keer. A numerical method for solving rough contact problems based on the multi-level multi-summation and conjugate gradient techniques. *Wear*, 231(2):206–219, 1999.
- [74] V. L. Popov. *Contact Mechanics and Friction: Physical Principles and Applications*. Springer, Berlin, 2010.
- [75] M. J. D. Powell. Restart procedures for the conjugate gradient method. *Mathematical Programming*, 12(1):241–254, 1977.
- [76] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press, New York, 1992.
- [77] E. Rabinowicz. *Friction and Wear of Materials*, volume 2. Wiley, New York, 1995.
- [78] P. Sainsot and A. A. Lubrecht. Efficient solution of the dry contact of rough surfaces: a comparison of fast Fourier transform and multigrid methods. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 225(6):441–448, 2011.
- [79] L. L. Scharf. *Statistical Signal Processing. Detection, Estimation, and Time Series Analysis*. Addison-Wesley Reading MA, 1991.
- [80] H. Schippers. Multigrid methods for boundary integral equations. *Numerische Mathematik*, 46(3):351–363, 1985.

- [81] A. Segal. *SEPRAN users manual*. Den Haag, The Netherlands, 2013. <http://ta.twi.tudelft.nl/sepran/sepran.html>.
- [82] S. Serra-Capizzano. Convergence analysis of two-grid methods for elliptic Toeplitz and PDEs matrix-sequences. *Numerische Mathematik*, 92(3):433–465, 2002.
- [83] A. Signorini. Sopra alcune questioni di elastostatica. *Atti della Societa Italiana per il Progresso delle Scienze*, 1933.
- [84] H. M. Stanley and T. Kato. An FFT-based method for rough surface contact. *Journal of tribology*, 119(3):481–485, 1997.
- [85] G. Strang. A proposal for Toeplitz matrix calculations. *Studies in Applied Mathematics*, 74(2):171–176, 1986.
- [86] J. M. Tang. *Two-level Preconditioned Conjugate Gradient Methods with Applications to Bubbly Flow Problems*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2008.
- [87] K. T. Tang. *Mathematical Methods for Engineers and Scientists 3: Fourier Analysis, Partial Differential Equations and Variational Methods*, volume 3. Springer-Verlag, Berlin, Heidelberg, 2007.
- [88] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, London, 2000.
- [89] A. van Beek. *Advanced Engineering Design: Lifetime Performance and Reliability*. Delft University of Technology, 2006.
- [90] C. van Loan. *Computational Frameworks for the Fast Fourier Transform*. Society for Industrial and Applied Mathematics, Philadelphia, USA, 1987.
- [91] C. H. Venner and A. A. Lubrecht. *Multi-level Methods in Lubrication*. Elsevier Science, Amsterdam, The Netherlands, 2000.
- [92] C. H. Venner and W. E. ten Napel. Multilevel solution of the elasto-hydrodynamically lubricated circular contact problem part I: Theory and numerical algorithm. *Wear*, 152(2):351–367, 1992.
- [93] S. Vijayakar, H. Busby, and L. Wilcox. Finite element analysis of three-dimensional conformal contact with friction. *Computers & Structures*, 33(1):49–61, 1989.
- [94] P. Vila, L. Baeza, J. Martínez-Casas, and J. Carballeira. Rail corrugation growth accounting for the flexibility and rotation of the wheel set and the non-hertzian and non-steady-state effects at contact patch. *Vehicle System Dynamics*, 52(sup1):92–108, 2014.

- [95] E. A. H. Vollebregt. A Gauss-Seidel type solver for special convex programs, with application to frictional contact mechanics. *Journal of Optimization Theory and Applications*, 87(1):47–67, 1995.
- [96] E. A. H. Vollebregt. Refinement of Kalker’s rolling contact model. In: *Proceedings of the 8th International Conference on Contact Mechanics and Wear of Rail/Wheel Systems, Firenze*, pages 149–156, 2009.
- [97] E. A. H. Vollebregt. Improving the speed and accuracy of the frictional rolling contact model “CONTACT”. In *Proceedings of the 10th International Conference on Computational Structures Technology, Stirlingshire, United Kingdom*, 2010.
- [98] E. A. H. Vollebregt. CG-type solver for the tangential problem. Technical Report m11018, VORtech BV, 2011.
- [99] E. A. H. Vollebregt. The bound-constrained conjugate gradient method for non-negative matrices. *Journal of Optimization Theory and Applications*, pages 1–23, 2013.
- [100] E. A. H. Vollebregt. A new solver for the elastic normal contact problem using conjugate gradients, deflation, and an FFT-based preconditioner. *Journal of Computational Physics*, 257:333–351, 2014.
- [101] E. A. H. Vollebregt. User’s guide for CONTACT, Vollebregt & Kalker’s rolling and sliding contact model. Technical Report TR09-03, version 14.1, VORtech BV, The Netherlands, 2014.
- [102] E. A. H. Vollebregt and H. M. Schuttelaars. Quasi-static analysis of two-dimensional rolling contact with slip-velocity dependent friction. *Journal of Sound and Vibration*, 331(9):2141–2155, 2012.
- [103] E. A. H. Vollebregt and A. Segal. Solving conformal wheel–rail rolling contact problems. *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, 52.sup1:455–468, 2014.
- [104] T. von Petersdorff and E. P. Stephan. Multigrid solvers and preconditioners for first kind integral equations. *Numerical Methods for Partial Differential Equations*, 8(5):443–450, 1992.
- [105] Q. Wang. Seizure failure of journal-bearing conformal contacts. *Wear*, 210(1):8–16, 1997.
- [106] K. Willner. Fully coupled frictional contact using elastic halfspace theory. *Journal of Tribology*, 130(3):031405, 2008.
- [107] P. Wriggers. *Computational Contact Mechanics*. Springer-Verlag, Berlin, Heidelberg, 2006.

-
- [108] H. Wu and N. Wilson. Chapter 8: Railway vehicle derailment and prevention. *Handbook of Railway Vehicle Dynamics*, page 209, 2006.
- [109] J. J. Wu. Simulation of rough surfaces with FFT. *Tribology International*, 33(1):47–58, 2000.
- [110] J. J. Wu. Simulation of non-Gaussian surfaces with FFT. *Tribology International*, 37(4):339–346, 2004.
- [111] V. A. Yastrebov. *Numerical Methods in Contact Mechanics*. ISTE/Wiley, New York City, NY, USA, 2013.
- [112] J. Zhao, E. A. H. Vollebregt, and C. W. Oosterlee. A full multigrid method for linear complementarity problems arising from elastic normal contact problems. *Mathematical Modelling and Analysis*, 19(2):216–240, 2014.
- [113] J. Zhao, E. A. H. Vollebregt, and C. W. Oosterlee. Multigrid with FFT smoother for a simplified 2D frictional contact problem. *Numerical Linear Algebra with Applications*, 21(2):256–274, 2014.
- [114] J. Zhao, E. A. H. Vollebregt, and C. W. Oosterlee. Extending the BEM for elastic contact problems beyond the half-space approach. Technical Report 15-04, Delft Institute of Applied Mathematics, Delft University of Technology, 2015.
- [115] J. Zhao, E. A. H. Vollebregt, and C. W. Oosterlee. A fast nonlinear conjugate gradient based method for 3D concentrated frictional contact problems. *Journal of Computational Physics*, 288:86–100, 2015.

Curriculum Vitae

Jing Zhao was born in Baoji (Shaanxi Province, China) on 8th December 1986. She obtained her BSc with specialization in Information and Computing Science, at Northwestern Polytechnical University (Xi'an, China) in 2009. In the same year, she was recommended to become a postgraduate student in Applied Mathematics. In 2010, she started her PhD work at Delft University of Technology (The Netherlands), supported by China Scholarship Council (CSC). Her early study period in Delft also included research for her MSc, which was obtained at Northwestern Polytechnical University, in 2012.

As a PhD candidate, Jing focused on developing fast numerical solvers for linear and nonlinear optimization problems, arising from the wheel-rail contact in railway industry, under the supervision of Dr. ir. E.A.H. Vollebregt and Prof. dr. ir. C.W. Oosterlee. She has three journal papers published, and one submitted. One of her papers received the *Honorable Mention* in the Student Paper Competition at the 12th Copper Mountain Conference in Colorado (USA) in 2012. She has presented her work in different international conferences, and also worked as a teaching assistant in the course “Applied Numerical Mathematics” for master students in Chemical Engineering at Delft University of Technology. Her present research interests are in the field of Applied Mathematics, particularly in Scientific Computing with industrial applications.

List of Publications

1. **J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee.** Extending the BEM for elastic contact problems beyond the half-space approach, *Submitted for publication*, 2015.
2. **J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee.** A fast nonlinear conjugate gradient based method for 3D concentrated frictional contact problems. *Journal of Computational Physics*, 288:86-100, 2015.
3. **J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee.** Multigrid with FFT smoother for a simplified 2D frictional contact problem. *Numerical Linear Algebra with Applications*, 21(2):256-274, 2014.
4. **J. Zhao, E.A.H. Vollebregt and C.W. Oosterlee.** A full multigrid method for linear complementarity problems arising from elastic normal contact problems. *Mathematical Modelling and Analysis*, 19(2):216-240, 2014.

List of Presentations

- **Oral Presentations:**

1. Student Krylov Day organized by SIAM Student Chapter Delft, Delft, The Netherlands, February. 2015.
2. The 11th World Congress on Computational Mechanics, the 5th European Conference on Computational, Barcelona, Spain, July 2014.
3. Spring Meeting (invited speaker), Amsterdam, The Netherlands, May 2014.
4. The 16th Copper Mountain Conference on Multigrid Methods, Colorado, United States, March 2013.
5. European Multigrid Conference, Schwetzingen, Germany, August 2012.
6. The 12th Copper Mountain Conference on Iterative Methods, Colorado, United States, March 2012.

- **Poster Presentations:**

1. The 39th Woudschoten Conference, Zeist, The Netherlands, October 2014.
2. The 38th Woudschoten Conference, Zeist, The Netherlands, October 2013.
3. The 37th Woudschoten Conference, Zeist, The Netherlands, October 2012.
4. NWO-JSPS Joint Seminar, Delft, The Netherlands, April 2012.