Dissertation Thesis

Czech
Technical
University
in Prague

F2

Faculty of Mechanical Engineering
Department of Mechanics, Biomechanics and Mechatronics

Efficient and Robust Numerical Solution of Contact Problems by the Finite Element Method

Ing. Ján Kopačka

Supervisor: Ing. Jiří Plešek, CSc.
Supervisor–specialist: Ing. Dušan Gabriel, Ph.D.
Field of study: Mechanics of Solids, Deformable Bodies and Continua
January 2017

Acknowledgements

First of all, I would like to wholeheartedly thank my wife, Markéta, for her unwavering support during writing this thesis. I would also like to thank my parents that their upbringing instilled in me the respect for knowledge and that always put my education in the first place. I want to thank my supervisor Jiří Plešek for the opportunity to work with him and to the supervisor-specialist, Dušan Gabriel, for stimulating discussion of advanced topics. Special thanks go to my co-authors and friends from Stuttgart University, Manfred Bischoff and Anton Tkachuk, for inspiration. I appreciate their professional and human qualities. Last but not least, I would like to thank colleague Radek Kolman for his motivation and encouragement for further work.

This work was supported by the European Regional Development Fund under Grant No. CZ.02.1.01/0.0/0.0/15_003/0000493 (Centre of Excellence for Nonlinear Dynamic Behaviour of Advanced Materials in Engineering) and by the Czech Science foundation under grant number 16-03823S within institutional support RVO:61388998.

Declaration

I declare, that I wrote my dissertation thesis autonomously and exclusively by using the sources I have cited. I agree with lending and publishing of this work.

In Prague, 31. January 2017

Abstract

The work deals with the numerical solution of contact problems by the finite element method. The symmetry preserving contact formulation is improved on two levels. Firstly, the stability and robustness of the BFGS algorithm for constrained systems is increased. And secondly, a new efficient and robust method for local contact search for finite elements of higher order is proposed. This symmetry-preserving contact formulation based on the penalty method is also used as a basis for development of an isogeometric contact formulation. Isogeometric analysis is a modern method for spatial discretization based on the finite element method. This new isogeometric contact formulation is further extended to solve dynamic problems utilizing an explicit, conditionally stable, temporal integration scheme. Special attention is paid to the influence of the mass matrix lumping on the oscillations of the contact pressure. The negative impact of the penalty parameter on the critical time step is treated by employing the bipenalty method.

Keywords: computational contact-impact mechanics, quasi-Newton methods, local contact search, isogeometric analysis, bipenalty method

Supervisor: Ing. Jiří Plešek, CSc.
Institute of Thermomechanics of the CAS, v. v. i.,
Dolejškova 1402/5,
182 00 Praha 8

Abstrakt

Práce se zabývá numerickým řešením kontaktních úloh metodou konečných prvků. Symetrii zachovávající kontaktní formulace je vylepšena na dvou úrovních. Za prvé je zvýšena stabilita a robustnost BFGS algorithm pro systémy s vázaným extrémem. A za druhé je navržená nová efektivní a robustní metoda pro lokální vyhledávání kontaktu na konečných prvcích vyššího řádu. Tato symetrii zachovávající kontaktní formulace založená na penaltové metodě byla dále využita k návrhu kontaktní formulace pro isogeometrickou analýzu, což je moderní metoda prostorové diskretizace na bázi metody konečných prvků. Tato nová isogeometrická kontaktní formulace byla dále rozšířena pro řešení dynamických úloh pomocí explicitní, podmíněně stabilní, časové integrace. Zvláštní pozornost je věnována vlivu diagonalizace matice hmotnosti na oscilace kontaktního tlaku. Negativní vliv pokutového parametru na kritický časový krok je řešen použitím bipenaltové metody.

Klíčová slova: výpočtová mechanika kontaktu, kvazinewtonské metody, lokální vyhledávání kontaktu, isogeometrická analýza, bipenaltová metoda

Překlad názvu: Efektivní a robustní numerické řešení kontaktních problémů metodou konečných prvků

Contents

Appendices

Figures

Tables

Nomenclature

Acronyms

2D	Two-dimensional
3D	Three-dimensional
ANSYS	Analysis System
AVRG	Average
BFGS	Broyden-Fletcher-Goldfarb-Shenno
CAD	Computer aided design
CAM	Computer aided manufacturing
CDM	Central difference method
CPP	Closest point projection
CPR	Critical penalty ration
CPU	Central processing unit
DNF	Did not finish
DOP	Discrete orientation polytopes
FEA	Finite element
FEM	Finite element method
FFS	Free form surface
GM	Global minimum
GPTS	Gauss-point-to-segment
HHT	Hilbert-Hughes-Taylor
HMH	Huber-von Mises-Hencky
HP	High pressure
HRZ	Hinton-Rock-Zienkiewicz
IGA-C	Isogeometric analysis-collocation
IGA	Isogeometric analysis
KTS	Knot-to-segment
LM	Local minimum
MAX	Maximum

NASTRAN	NASA STructure ANalysis
NTN	Node-to-node
NTS	Node-to-segment
NURBS	Non-uniform rational B-splines
PDASS	Primal-dual active set strategy
PDE	Partial differential equations
PMD	Package for machine design
SAP	Structural analysis package
SMS	Selective mass scaling
SP	Saddle point
STS	Segment-to-segment
`TOL`	Tolerance

Greek Symbols

α, β, γ	Algorithmic parameters of temporal integration schemes
β_{m}	Dimensionless mass penalty parameter
β_{s}	Dimensionless stiffness penalty parameter
$\boldsymbol{\chi}$	Regular mapping called motion
Δ	Increment
δ	First variation
δ_{α}^{β}	Krönecker's delta symbol
Δt^n	Time step at time t^n
Δt_c	Critical time step
$\boldsymbol{\varepsilon}$	Infinitesimal strain tensor
ϵ_{m}	Mass penalty parameter
ϵ_{N}	Normal penalty parameter
ϵ_{s}	Stiffness penalty parameter
ϵ_{T}	Tangent penalty parameter
$\Gamma_{\mathrm{c}}^{(i)}$	Portion of the boundary in the initial configuration where contact (traction) vectors are prescribed

$\gamma_{\mathrm{c}}^{(i)}$	Portion of the boundary in the current configuration where contact (traction) vectors are prescribed
$\Gamma_{\sigma}^{(i)}$	Portion of the boundary in the initial configuration where stress (traction) vectors are prescribed
$\gamma_{\sigma}^{(i)}$	Portion of the boundary in the current configuration where stress (traction) vectors are prescribed
$\Gamma_{\mathrm{u}}^{(i)}$	Portion of the boundary in the initial configuration where displacements are prescribed
$\gamma_{\mathrm{u}}^{(i)}$	Portion of the boundary in the current configuration where displacements are prescribed
$\boldsymbol{\kappa}$	Curvature tensor
κ_{α}	Principal curvatures
κ_{n}	Normal curvature
$\dot{\lambda}$	Rate of plastic slip
λ_{N}	Lagrange multiplier of the contact pressure
$\boldsymbol{\lambda}_{\mathrm{T}}$	Lagrange multiplier of the tangent contact traction
μ	Coefficient of friction
ν	Poisson's ratio
ω	Eigenfrequency
$\bar{\Omega}_0$	Closed set of position vectors at the initial configuration
Ω_e	Domain of finite element
$\Omega_{\square}$	Bi-unit region
$\bar{\Omega}_t$	Closed set of position vectors at the current configuration
Φ	Slip function
ϕ	Weighting parameter of the contact interface
Φ_A	Basis function of the A-th global node
ψ_0	Mapping of convective coorinates on the boundary of the initial configuration
ψ_t	Mapping of convective coorinates on the boundary of current configuration
ρ_0	Reference density

$\boldsymbol{\sigma}$	Cauchy stress tensor
σ_{Y}	Yield stress
$\boldsymbol{\tau}$	Tangent vector at the current configuration
θ	Step-length of the line-search technique
$\vartheta_{1,2,\ldots}$	Angular dimensions
$\boldsymbol{\xi}$	Convective coordinates
$\boldsymbol{\Xi}^i$	Knot vector
$\boldsymbol{\zeta}$	Parametric coordinates of the finite element
$C_{\mathrm{r}}^{\mathrm{crit}}$	Critical Courant dimensionless number
n_{nod}	Number of nodes
$\mathbf{t}_{\mathrm{N}}$	Normal contact traction vector
$\mathbf{t}_{\mathrm{T}}$	Tangent contact traction vector
$\mathcal{W}_{\mathrm{con}}$	Work done by contact forces

Roman Symbols

$\mathbf{A}$	Material acceleration
$\mathcal{A}$	Domain of convective coordinates
A	Cross-sectional area
A_s	Size of the associated contact boundary that belongs to the s-th active node
$\mathcal{B}$	Body
$\mathbf{c}$	Global vector of nodal weighting coefficients of displacement
C^n	Continuity of the n-th order
c_0	Wave speed in 1D continuum
c_1	Line-search parameter for the slope of linear function
c_2	Line-search parameter for the curvature of minimized function
$\mathbf{c}_a$	Nodal weighted displacement vector of the a-th node
C_{r}	Courant dimensionless number
$\ddot{\mathbf{d}}$	Global vector of nodal accelerations
$\mathbf{D}$	Secant matrix

D	Directional derivative
d	Squared distance function
d_1	Coefficinet of cubic interpolation during the step-length procedure
d_2	Coefficinet of cubic interpolation during the step-length procedure
$\mathbf{d}_a$	Nodal displacement vector of the a-th node
$\det$	Determinant operator
$\mathrm{d}\mathbf{f}_\mathrm{c}$	Differential of the contact force
Div	Divergence operator with respect to initial configuration
E	Young's modulus
$\mathbf{e}_\alpha$	Principal directions
$\mathcal{E}_\mathrm{kin}$	Kinetic energy
$\mathcal{E}^\mathrm{p}_\mathrm{kin}$	Penalised kinetic energy
$\mathcal{E}_\mathrm{pot}$	Potential energy
$\mathcal{E}_\mathrm{tot}$	Total energy
$\mathbf{F}$	Deformation gradient
F	A functional
f	A function of one variable for line-searching
$\mathbf{f}_0$	Body force prescribed per unit reference volume
$\mathbf{F}_\mathrm{c}$	Global vector of contact nodal forces
$\mathbf{F}_\mathrm{ext}$	Global vector of external nodal forces
$\mathbf{F}_\mathrm{int}$	Global vector of internal nodal forces
f_q	Quadratic interpolation of the 1D function minimized during step-length procedure
G	A function of multiple variables
g_N	Normal gap function
Grad	the gradient operator with respect to initial configuration
$\dot{\mathbf{g}}_\mathrm{T}$	Relative tangential velocity
$\dot{\mathbf{g}}_\mathrm{T}^{\mathrm{sl}(i)}$	Slip part of the relative tangential velocity
$\dot{\mathbf{g}}_\mathrm{T}^{\mathrm{st}(i)}$	Stick part of the relative tangential velocity

$\mathbf{H}$	Hessian matrix
h	Characteristic length of the element
h	Characteristic element length
$\mathbf{I}$	Identity matrix
$\mathbf{I}_{n_{\mathrm{dof}}}$	n_{dof}-dimensional identity matrix
J	Jacobian determinant
$\mathbf{K}$	Tangent matrix
$L_{1,2,\ldots}$	Geometric parameter
$\mathcal{L}_{\mathrm{p}}$	Penalised Lagrangian
$\mathbf{M}$	Global mass matrix
m	Vertex position counter of the simplex method
$\mathbf{m}$	Matric tensor
n	Number of basis functions
n_{sd}	Number of spatial dimensions
N	Shape function
n_{c}	Number of active contact points
n_{dof}	Number of degrees of freedom
n_{el}	Number of elements
n_{en}	Number of element nodes
n_{nes}	Number of nodes of the element segment
$\mathbf{N}$	Normal vector to the initial boundary
$\mathbf{n}$	Normal vector to the current boundary
n_{pd}	Number of parametric dimensions
n_{s}	Number of contact segments
n_{scp}	Number of segment control points
$\mathbf{P}$	The first Piola-Kirchhoff stress tensor
$\mathbf{p}$	Search direction of the line-search technique
P	Material particle

p	Polynomial order
p_c	Contact pressure
$\mathbf{Q}$	Torus point projection
$\mathbf{q}$	point projection on the osculating sphere
q	Pressure load
$\mathbf{q}_1$	Auxiliary vector
$\mathbf{q}_2$	Point lying on the major circle of the torus
$\mathbb{R}$	Set of real numbers
$\mathbf{R}$	Residual vector
$\mathbf{r}$	Auxiliary vector of quasi-Newton methods
R	Penalty ratio
r	Dimensionless penalty ration
r_c	Contact area radius
R_m	Major radius of the torus
r_m	Minor radius of the torus
$\mathbf{s}$	Auxiliary vector of quasi-Newton methods
$\mathbb{T}$	Time interval
$\mathbf{t}$	Auxiliary tangent vector
t	Time
$\boldsymbol{\mathcal{T}}$	Tangent vector at the initial configuration
$\mathbf{T}_\mathrm{c}$	Contact traction vector acting on the initial contact boundary
$\mathbf{t}_\mathrm{c}$	Contact traction vector acting on the current contact boundary
$\hat{\mathbf{T}}$	Prescribed traction vector field at the boundary Γ_σ
t_N	Length of the normal contact traction vector
$\mathbf{u}$	Displacement field
$\mathcal{U}$	Function space of the trial functions
$\mathbf{u}_0$	Prescribed initial displacement field
$\mathcal{U}^h$	Finite-dimensional subspaces of the function space of the trial functions

$\hat{\mathbf{u}}$	Prescribed displacement field at the boundary Γ_{u}
$\mathcal{E}^{\mathrm{p}}_{\mathrm{pot}}$	Penalised strain energy
$\mathbf{V}$	Material velocity
$\mathcal{V}$	Function space of the weighting (test) functions
$\mathbf{V}_0$	Prescribed initial material velocity field
$\mathcal{V}^h$	Finite-dimensional subspaces of the function space of the weighting (test) functions
w_j	Weight of the NURBS basis function
$\mathbf{X}$	Position vector of a particle at the initial configuration
$\mathbf{x}$	Position vector of a particle at the current configuration
$\mathbf{x}_a$	Nodal position vectors in the current configuration of the a-th node
$\mathbf{x}_{\mathrm{c}}$	Centre point

Superscripts

$\bullet^*$	Dimensionless quantities
$\bullet^{\alpha,\beta,\gamma}$	Index of contravariant coordinates
$\bullet^{(i)}$	Index of the slave body
$\bullet^{(k)}$	Index of the master body
$\bullet^n$	Index of time step

Subscripts

$\bullet_{A,B}$	Global indices of the nodal points
$\bullet_{a,b}$	Local indices of the element nodal points
$\bullet_{\alpha,\beta,\gamma}$	Index of covariant coordinates
$\bullet_e$	Element index
$\bullet_\infty$	Converged solution
$\bullet_k$	Iteration index of the nonlinear solver
$\bullet_\ell$	Iteration index of the step-length procedure
$\bullet_s$	Index of active contact point

Other Symbols

$\bar{\bullet}$	Variable which depends on the closest point projection

$\ddot{\bullet}$	Second time derivatives
$\dot{\bullet}$	First time derivatives
$\|\bullet\|$	Euclidean norm
$\langle\bullet\rangle$	Macaulay's brackets
$\emptyset$	Empty set

Chapter 1

Introduction

It would seem that finite element analysis involving contact is now commonplace. However, it is reported that up to half of the tasks encountered in engineering practice has problems with convergence. Therefore, the development of robust and stable contact algorithms is still in the forefront of researcher's interest in the field of computational mechanics. Witness the fact that the contact analysis has been earmarked as a separate branch of computational science — computational contact mechanics.

Nails, bolds, clamp connections, limited slip mechanism, in all these contact applications, and uncountable others, we rely on the contact with friction. On the other hand, consider for instance bearings, ball screw, linear guides, these are examples where we would like to completely get rid of friction. There are also applications where we would like to control contact and friction. Let us consider an everyday life example — automobile tires. An ideal tire should primarily generate sufficient adhesion to avoid sliding when braking or turning. Next, the rolling resistance should be minimal, because it directly affects fuel consumption, which we want to be as low as possible. Further, the rate of abrasion of the tread should be adequate to achieve reasonable durability. Last but not least, the noise of rolling tire should be minimized. These contradictory requirements have to be simultaneously met in order to obtain an optimal design. The computational contact mechanics is one of the tools without which these challenging tasks would be hardly achievable.

The main purpose of a contact algorithm is to enforce a natural physical fact that bodies in motion can only touch their surfaces but their mutual penetration cannot occur. A specific way in which the contact algorithm enforces the impenetrability condition is called a contact formulation. Types of contact formulations can be classified according to various criteria: according to the dimensions of the problem on the two-dimensional (2D) and three-dimensional (3D); according to inertial forces on quasi-static and dynamic, which can be further subdivided according to temporal integration scheme into implicit and explicit formulations; by considering the friction on the frictionless and frictional; according to the magnitude of relative movement between the contact boundaries on the small sliding and finite sliding; by the type of contact discretization on the node-to-node (NTN), node-to-segment (NTS) Gauss-point-to-segment (GPTS), segment-to-segment (STS); by the methods for contact constraints enforcement on formulations based on the penalty method, barrier method, method of Lagrange multipliers, Augmented Lagrangian method, perturbed Lagrangian method; by the treatment of contact bias on the one-pass, two-pass, two-half-pass; by the geometric description of the contact boundary to C^0-continuous elements with p-th order Lagrange polynomials as shape functions, C^p-continuous hierarchical-type elements, contact boundaries smoothed by different types of splines.

A general contact algorithm consists of several procedures. Firstly, it is necessary to detect whether bodies are in contact for a given displacement field and if so, by which parts of boundaries. A subroutine called global contact search is the procedure that is responsible for this task. Secondly, it is necessary to evaluate the contact residual vector and, optionally, the contact tangent stiffness matrix. An integral part of this procedure is to address the problem of closest point projection at a discrete set of points of contact boundaries. A subroutine that is responsible for the resolution of the closest point projection problem is called local contact search. Third, the contact residual vector and the contact tangent matrix are employed within one iteration of the non-linear solver. If convergence criteria are not achieved or contact constraints are violated, the whole process is repeated for a new displacement field.

The reason why so many quasi-static contact problems have difficulties with convergence, as mentioned at the beginning, lies in the instability of non-linear solver. Contact algorithms of conventional finite element analysis (FEA) softwares utilize non-linear solvers based on the Newton-Raphson method and, therefore, depend on the consistent linearization of the contact formulation. This work is motivated by the contact formulation proposed in [1], which was implemented in the FEA softwate PMD [2]. It is a symmetry-preserving GPTS penalty-based frictionless large deformations contact formulation. The contact algorithm of FEA software PMD obviates the need for consistent tangent matrix with non-linear solver from the class of quasi-Newton methods, namely the Broyden-Fletcher-Goldfarb-Shenno (BFGS) method. This method is based on a rank-two update of the previous secant matrix, which is significantly faster then calculation of the consistent tangent matrix. The updates must be preceded by the initial choice of secant matrix. In the work [1, p. 2625] it was proposed to use as an initial secant matrix the elastic stiffness matrix, which seems to be the natural choice. Practical experience, however, revealed that this choice of the initial matrix leads the approximation of the solution away from the final solution or worse the calculation can even diverge. The similar problem may be encountered even in conventional FEA softwares, where this problem is resolved by an automatic load sub-incrementation. In this work we propose to stabilize the BFGS based non-linear solver by adding an approximative linearization of the contact tangent matrix to the initial elastic stiffness matrix.

Another issue that was revealed by solving problems of engineering practice is divergence of the local contact search procedure for a specific type of finite elements; in particular, the 8-node quadratic serendipity square element. A detailed analysis showed that the reason lies in the applied method for the problem of closest point projection — the Newton-Raphson method. In this work, a set of unconstrained optimization methods will by critically assessed and then, based on this assessment, a new procedure for the local contact search will be proposed.

In classical FEA, the problem of closest point projection is complicated by the fact that after discretization, contact boundaries are only C^0-continuous which leads to ambiguous definition of normal vectors. In practice, this means that one must algorithmically treat the case when the projected closest point lies close to the common boundary of two elements. This lack of smoothness has also an impact on the accuracy of calculation of the contact pressure. In the past, this problem was solved by the so called contact smoothing techniques, where the C^0-continuous FE contact boundary is replaced by a special parameterization with greater smoothness. The disadvantage of this procedure is that it leads to an additional data layer. In this work, we propose to use the isogeometric finite element analysis, a modern method of spatial discretization, which utilizes as basis functions various types of splines, most frequently NURBS (abbreviation for Non-Uniform Rational B-Splines).

The newly developed isogeometric contact algorithm based on the symmetry-preserving contact formulation will be further extended into the explicit contact-impact algorithm for the numerical solution of dynamic problems and wave propagation. A particular attention will be paid to the influence of the mass matrix lumping on the contact pressure oscillations.

The explicit temporal integration schemes are known to be conditionally stable. Furthermore, it is well known that the penalty method has a negative effect on the critical time step. This problem will be addressed by the bipenalty method.

1.1 Historical remarks

To find the roots of computational contact mechanics, one must go back to the late 19th Century, when a German physicist Heinrich Rudolf Hertz was attempting to understand how the optical properties of multiple, stacked lenses might change with the force holding them together. He considered frictionless contact between two linearly elastic ellipsoidal bodies undergoing small deformations. He published his findings in a paper entitled "On contact between elastic bodies" in 1881 [3], and so laid the foundations for contact mechanics. To this day Hertz's formulas for contact stress have still served as a basis for the evaluation of load capabilities and fatigue life in bearings and gears. But analytical solution is achievable solely for a few simple geometries and, therefore, only rough approximation for complex engineering problems was possible.

It was only in 1933 when an Italian mathematician Antonio Signorini outlined a problem which he called ”the problem with ambiguous boundary conditions” [4]. He considered frictionless static contact between a rigid surface and an anisotropic non-homogeneous elastic body subjected only to its mass forces. What made this problem so challenging were contact constraints in the form of inequalities and the fact that the active contact boundary was a priori unknown. In [5], Signorini explicitly invited young analysts to study the problem and to answer the question of existence and uniqueness of the solution. His student, Gaetano Fichera, came with positive answer exactly a week before Signorini's death. Fichera published his proof of the "Signorini problem", as he entitled it to honor his teacher, in 1964 [6]. Signorini and Fichera thus gave rise to a new branch of mathematics called variational inequalities. A pioneering work which opened the door for intensive research in this field was the paper of Lions and Stampacchia [7]. In the book of Duvaut and Lions [8] contact with Coulomb friction, dynamic contact problems or contact problems in viscoelasticity were formulated. A comprehensive text on the topic are books of Kinderlehler and Stampacchia [9] or Kikuchi and Oden [10].

Variational inequalities serves as a rigorous mathematical framework for computational contact mechanics. But without its numerical resolution, an analysis of complex industrial contact problems were still unachievable. Fortunately, there was one particular numerical method that has gained considerable popularity with the development of computer technology. It was the finite element method (FEM). FEM is a spatial discretization method which enables to compute numerical approximations of the real solutions to partial differential equations (PDE) formulated in the week sense. An introduction to the finite element analysis one can find for example in monographs [11, 12, 13, 14]. By coincidence, first finite element programs were founded at the same time as variational inequalities were formulated, i.e. at the end of 1950s and early 1960s. One of the first widely accepted FE program was ”SAP” (an abbreviation for Structural Analysis Package) [15] developed at Berkeley by E.L. Wilson at the end of 1950s. The first general purpose finite element software was NASTRAN (an acronym for NASA STRucture ANalysis) released to NASA in 1968 and to public in 1971.

Motivated by the turbine blade fastenings, Chan and Tuba from Westinghouse Research Laboratories were probably the first who, in 1971, developed a general purpose finite element quasi-static contact algorithm [16, 17]. It was two dimensional small strain frictional contact formulation based on the method of Lagrange multipliers. An interesting fact is that their contact formulation belongs to the note-to-segment category, which is considered to be more complex than the node-to-node discretization. The first node-to-node contact discretization for quasi-static problems was published by Francavilla and Zinkiewitcz [18] in 1975 and for contact-impact problems one year later by Hughes et al. [19]. All the aforementioned contact formulations were based on the Lagrange multiplier method. 2D node-to-node discretization by the so called finite element gaps was developed by Stadler and Weiss [20] in 1979. This formulation could be understood as the penalty method, described more rigurously by Oden [21] in 1980.

The short history excursion of the computational contact mechanics will end in 1985, during which it was published several important works that were heading further development. The milestone in the development of contact algorithms was the inclusion of large deformations and frictional conditions. Bathe and Chaudhary [22] and Hallquist et al. [23] were the first who published it. The first named used Lagrange multipliers and the second the penalty method. Both of them utilized node-to-segment contact discretization. Last but not least, Simo et al. [24] came up with a new type of contact discretization — segment-to-segment — and laid the foundation for further development, which lasts until today.

1.2 Outline of the thesis

The thesis is organized in the following way. The state of the art of numerical solution of contact problems by the finite element method is presented in Chapter 2. Based on the critical overview in Chapter 2, the aims and objectives of the thesis are formulated in Chapter 3. Chapter 4 explains in detail the methods mentioned in the state of the art which have been selected as the basis for the improvement of the general finite element contact algorithm. Chapter 5 describes and discusses improvements and newly proposed methods, followed by Chapter 6 where these improvements are clearly demonstrated on the numerical examples. Finally, theoretical and practical outcomes of the thesis are summarized in Chapter 7, and some conclusions are drawn in Chapter 8, summarizing results and pointing to further research.

Chapter 2

State of the Art

The intent of this chapter is to provide the comprehensive up-to-date information about the field of computational contact mechanics. After reviewing the basic concepts of continuum mechanics and the definition of contact kinematic variables, the normal and tangential contact conditions are formulated. Then, the contact initial-boundary value problem is formulated first, in the strong and next, in the weak sense, to provide the basis for further spatial discretization by the finite element method, with particular attention being paid to the enforcement of contact constraints. In particular the penalty method, the Lagrange multipliers method, and the Augmented Lagrangian method are described. Also, the linearization of the resulting system of nonlinear ordinary differential equations is addressed. The chapter closes the section on specific procedures, which are necessary for the numerical solution of the contact problems.

2.1 Contact kinematics

Without loss of generality, let us consider two deformable bodies $\mathcal{B}^{(i)}, i = \{1, 2\}$ in n_{sd} dimensional Euclidean space $\mathbb{R}^{n_{\mathrm{sd}}}$ coming into contact, which are depicted in Figure 2.1. The position of a particle $P^{(i)}$ of the body $\mathcal{B}^{(i)}$ at time $t = 0$ is described by the position vector $\mathbf{X}^{(i)} \in \bar{\Omega}_0^{(i)} \subset \mathbb{R}^{n_{\mathrm{sd}}}$. Further, let us consider existence of a regular mapping $\boldsymbol{\chi}^{(i)} : \bar{\Omega}_0^{(i)} \times \mathbb{T} \mapsto \bar{\Omega}_{\mathrm{t}}^{(i)}$, which will be called *motion* [25, p. 61]. Its image is the position vector $\mathbf{x}^{(i)} \in \bar{\Omega}_{\mathrm{t}}^{(i)} \subset \mathbb{R}^{n_{\mathrm{sd}}}$ of the particle $P^{(i)}$ at time $t \in \mathbb{T} = [0, T]$. The displacement field is defined as

$$\mathbf{u}^{(i)}(\mathbf{X}^{(i)}, t) := \boldsymbol{\chi}^{(i)}(\mathbf{X}^{(i)}, t) - \mathbf{X}^{(i)}. \tag{2.1}$$

The boundary of domains $\Omega_0^{(i)}$ and $\Omega_{\mathrm{t}}^{(i)}$ are covered by disjoint sets

$$\partial\Omega_0^{(i)} = \Gamma_{\mathrm{u}}^{(i)} \cup \Gamma_{\sigma}^{(i)} \cup \Gamma_{\mathrm{c}}^{(i)}, \tag{2.2}$$

$$\partial\Omega_{\mathrm{t}}^{(i)} = \gamma_{\mathrm{u}}^{(i)} \cup \gamma_{\sigma}^{(i)} \cup \gamma_{\mathrm{c}}^{(i)} \tag{2.3}$$

where $\Gamma_{\mathrm{u}}^{(i)}, \gamma_{\mathrm{u}}^{(i)}$ and $\Gamma_{\sigma}^{(i)}, \gamma_{\sigma}^{(i)}$ are those portions of the boundary in the initial and current configuration where displacements and tractions are prescribed, respectively. The sets $\Gamma_{\mathrm{c}}^{(i)}, \gamma_{\mathrm{c}}^{(i)}$ represent the unknown part of the boundary in the initial and current configuration where contact can possibly take place. Its determination is a part of the solution.

The physical fact that two bodies cannot penetrate each other during the motion can be expressed by the principle of impenetrability [26, p. 244]

$$\Omega_{\mathrm{t}}^{(1)} \cap \Omega_{\mathrm{t}}^{(2)} = \emptyset, \quad \forall t \in \mathbb{T}. \tag{2.4}$$

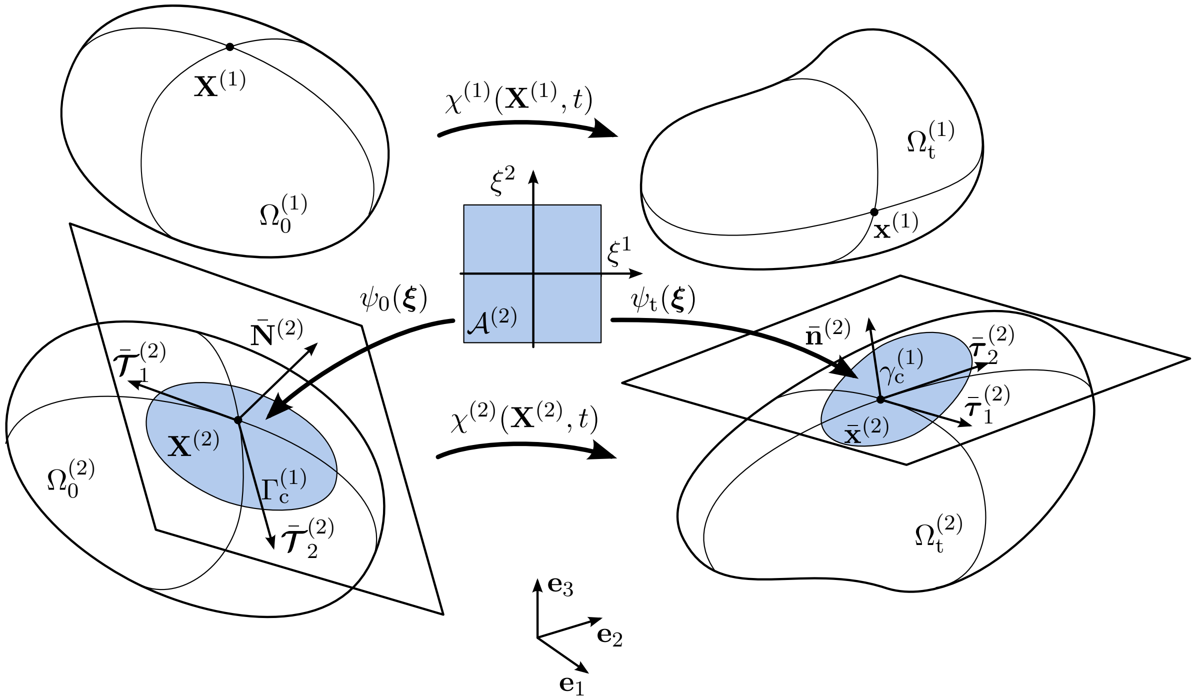


Figure 2.1: Two deformable bodies coming into contact depicted in the initial and current configuration.

This certainly elegant term, however, is not suitable for numerical purposes and therefore, it will be useful to reformulate it slightly. To generalize the notation, let us introduce another body index, $\bullet^{(k)}$, as

$$k = \{1, 2\} \setminus i, \tag{2.5}$$

i.e. $\bullet^{(k)}$ is the complement body to the body $\bullet^{(i)}$. The closest point $\bar{\mathbf{X}}^{(k)} \in \Gamma_{\mathrm{c}}^{(k)}$ is assigned to each point $\mathbf{X}^{(i)} \in \Gamma_{\mathrm{c}}^{(i)}$ in the sense

$$\bar{\mathbf{X}}^{(k)}(\mathbf{X}^{(i)}, t) := \arg \min_{\mathbf{X}^{(k)} \in \Gamma_{\mathrm{c}}^{(k)}} \left\| \mathbf{x}^{(i)}(\mathbf{X}^{(i)}, t) - \mathbf{x}^{(k)}(\mathbf{X}^{(k)}, t) \right\|, \tag{2.6}$$

where $\|\bullet\|$ is the Euclidean norm. It should be noted that the body $\mathcal{B}^{(i)}$ is usually termed as the slave body and $\mathcal{B}^{(k)}$ as the master body [23]. It is probably due to the dependence implied by the closest point projection. From now on, where it will be useful, the variables which depend on the closest point projection will be denoted with the bar $\bar{\bullet}$.

To find the position vector $\bar{\mathbf{X}}^{(k)}$ the contact boundary $\Gamma_{\mathrm{c}}^{(k)}$ is parametrized by convective coordinates $\boldsymbol{\xi} \in \mathcal{A} \subset \mathbb{R}^{n_{\mathrm{pd}}}$ and described by the mapping $\psi_0^{(k)} : \mathcal{A} \mapsto \Gamma_{\mathrm{c}}^{(k)}$, see Figure 2.1. Further, a squared distance function $d(\mathbf{X}^{(i)}, \boldsymbol{\xi}) : \Gamma_{\mathrm{c}}^{(i)} \times \mathcal{A} \mapsto \mathbb{R}$ is introduced as

$$d(\mathbf{X}^{(i)}, \boldsymbol{\xi}) := \frac{1}{2} \left\| \mathbf{x}^{(i)}(\mathbf{X}^{(i)}, t) - \mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\boldsymbol{\xi}), t) \right\|^2 . \tag{2.7}$$

Minimization of this function (see Appendix (A.1)) yields system of non-linear algebraic equations for unknown convective coordinates $\bar{\boldsymbol{\xi}}$ of the nearest point $\bar{\mathbf{X}}^{(k)}$

$$\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \frac{\partial \bar{\mathbf{x}}^{(k)}}{\partial \mathbf{X}^{(k)}} \frac{\partial \bar{\mathbf{X}}^{(k)}}{\partial \xi^{\alpha}} = 0, \tag{2.8}$$

where $\alpha = 1, \ldots, n_{\mathrm{pd}}$ is the index of the convective coordinate component. Here, $n_{\mathrm{pd}} = n_{\mathrm{sd}} - 1$ is the number of parametric dimensions. From now on, it will be assumed that the greek letters indices α, β, γ can take vaules $1, \ldots, n_{\mathrm{pd}}$. The solution of this system of non-linear algebraic equations, $\bar{\boldsymbol{\xi}}$, determines the desired position vector $\bar{\mathbf{X}}^{(k)} = \psi_0^{(k)}(\bar{\boldsymbol{\xi}})$. Equation (2.8) can be further recast considering the definition of the deformation gradient $\mathbf{F}^{(k)} \in \mathbb{R}^{n_{\mathrm{sd}} \times n_{\mathrm{sd}}}$

$$\mathbf{F}^{(k)} := \frac{\partial \mathbf{x}^{(k)}}{\partial \mathbf{X}^{(k)}}, \tag{2.9}$$

and the vectors tangent to the initial contact boundary $\Gamma_{\mathrm{c}}^{(k)}$ at a point $\mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}})$

$$\boldsymbol{\mathcal{T}}_{\alpha}^{(k)} := \frac{\partial \mathbf{X}^{(k)}}{\partial \xi^{\alpha}}. \tag{2.10}$$

The tangent vector $\boldsymbol{\mathcal{T}}_{\alpha}^{(k)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ can be transformed to the current configuration $\boldsymbol{\tau}_{\alpha}^{(k)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ by the push forward operation

$$\boldsymbol{\tau}_{\alpha}^{(k)} = \mathbf{F}^{(k)} \boldsymbol{\mathcal{T}}_{\alpha}^{(k)}. \tag{2.11}$$

Equation (2.8) now gains a particularly simple form

$$\boxed{\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} = 0} \tag{2.12}$$

which can be interpreted as the orthogonality condition. Usually, rather then the initial position vector $\bar{\mathbf{X}}^{(k)}$ its spatial counterpart $\bar{\mathbf{x}}^{(k)} = \mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}}), t)$ will be in question. Therefore it is useful to replace the mapping $\psi_0^{(k)}$ by the composite mapping $\psi_t^{(k)} : \mathcal{A} \times \mathbb{T} \mapsto \gamma_{\mathrm{c}}^{(k)}$ defined as

$$\psi_t^{(k)} := \boldsymbol{\chi}^{(k)} \circ \psi_0^{(k)}. \tag{2.13}$$

The spatial position vector $\bar{\mathbf{x}}^{(k)} = \mathbf{x}^{(k)}(\bar{\boldsymbol{\xi}}, t)$ now directly depends on the convective coordinates. Consequently, the vector tangent to the current contact boundary $\gamma_{\mathrm{c}}^{(k)}$ at point $\bar{\mathbf{x}}^{(k)}$ can be defined as

$$\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} := \frac{\partial \bar{\mathbf{x}}^{(k)}}{\partial \xi^{\alpha}}. \tag{2.14}$$

It is worth mentioning that the tangent vectors form a covariant basis of the tangent space at point $\bar{\mathbf{x}}^{(k)}$ which, in general, are not orthogonal. Therefore, it will be useful to define the metric tensor $\mathbf{m}^{(k)} \in \mathbb{R}^{n_{\mathrm{pd}} \times n_{\mathrm{pd}}}$ by its components

$$m_{\alpha\beta}^{(k)} := \boldsymbol{\tau}_{\alpha}^{(k)} \cdot \boldsymbol{\tau}_{\beta}^{(k)}. \tag{2.15}$$

With the metric tensor at hand, the relation between the covariant and contravariant basis can be expressed as

$$\boldsymbol{\tau}_{\alpha}^{(k)} = m_{\alpha\beta}^{(k)} \boldsymbol{\tau}^{(k)\beta}, \tag{2.16}$$

where the Einstein summation convention [25, p. 6] has been used. Let us recall that for the contravariant base vectors, $\boldsymbol{\tau}^{(k)\beta}$, it holds

$$\boldsymbol{\tau}_{\alpha}^{(k)} \cdot \boldsymbol{\tau}^{(k)\beta} = \delta_{\alpha}^{\beta}, \tag{2.17}$$

where δ_{α}^{β} is the Krönecker delta [27, p. 33]. With the aid of tangent vectors, the normal vector $\mathbf{N}^{(k)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ to the initial contact boundary $\Gamma_{\mathrm{c}}^{(k)}$ is constructed as

$$\mathbf{N}^{(k)} := \frac{\mathbf{e}_3 \times \boldsymbol{\mathcal{T}}_1^{(k)}}{\left\| \mathbf{e}_3 \times \boldsymbol{\mathcal{T}}_1^{(k)} \right\|}, \qquad \mathbf{N}^{(k)} := \frac{\boldsymbol{\mathcal{T}}_1^{(k)} \times \boldsymbol{\mathcal{T}}_2^{(k)}}{\left\| \boldsymbol{\mathcal{T}}_1^{(k)} \times \boldsymbol{\mathcal{T}}_2^{(k)} \right\|} \tag{2.18}$$

for $n_{\mathrm{sd}} = 2$ and $n_{\mathrm{sd}} = 3$, respectively, and its spatial counterparts $\mathbf{n}^{(k)} \in \mathbb{R}^{n_{\mathrm{sd}}}$

$$\mathbf{n}^{(k)} := \frac{\mathbf{e}_3 \times \boldsymbol{\tau}_1^{(k)}}{\left\| \mathbf{e}_3 \times \boldsymbol{\tau}_1^{(k)} \right\|}, \qquad \mathbf{n}^{(k)} := \frac{\boldsymbol{\tau}_1^{(k)} \times \boldsymbol{\tau}_2^{(k)}}{\left\| \boldsymbol{\tau}_1^{(k)} \times \boldsymbol{\tau}_2^{(k)} \right\|}. \tag{2.19}$$

Note that on the contact interface, the smoothness and kinematic compatibility of the two contacting surfaces necessitates that [28, p. 1130]

$$\mathbf{n}^{(i)} = -\bar{\mathbf{n}}^{(k)}. \tag{2.20}$$

2.1.1 The normal gap function

The normal vector allows to define an essential contact kinematic quantity, the normal gap function $g_{\mathrm{N}}^{(i)} : \Gamma_{\mathrm{c}}^{(i)} \times \mathbb{T} \mapsto \mathbb{R}$ [29, p. 114], in the form

$$\boxed{g_{\mathrm{N}}^{(i)}(\mathbf{X}^{(i)}, t) := -\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \bar{\mathbf{n}}^{(k)},} \tag{2.21}$$

which can be understood as the signed distance function. According to our definition, (2.21), the gap is open, i.e. bodies are not in contact, when the normal gap function is negative. Conversely, positive value of the normal gap function indicates penetration of the bodies. Finally, the zero value means that bodies are in touch. With the normal gap function at hand the non-penetration condition (2.4) can be reformulated as

$$g_{\mathrm{N}}^{(i)} \leq 0, \qquad \forall \mathbf{X}^{(i)} \in \Gamma_{\mathrm{c}}^{(i)}, t \in \mathbb{T}. \tag{2.22}$$

This condition is essential for the contact problem formulation, which will be presented in Section 2.6. It is the inequality character of the non-penetration condition why the numerical solution of contact problems is so challenging.

2.1.2 The relative tangential velocity

As far as friction is concerned another important kinematic quantity has to be introduced. Namely, the relative tangential velocity [29, p. 118], $\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$, which states the time change of position of point $\mathbf{x}^{(i)}$ with respect to its closest projection point $\bar{\mathbf{x}}^{(k)}$. For the ideal slip, i.e. $g_{\mathrm{N}}^{(i)} = \dot{g}_{\mathrm{N}}^{(i)} = 0$, it must hold

$$\begin{aligned}
\frac{\partial}{\partial t}\left(\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(k)}\right) &= 0, && (2.23)\\
\frac{\partial \mathbf{x}^{(i)}}{\partial t}-\frac{\partial \bar{\mathbf{x}}^{(k)}}{\partial t}-\frac{\partial \bar{\mathbf{x}}^{(k)}}{\partial \mathbf{X}^{(k)}}\frac{\partial \bar{\mathbf{X}}^{(k)}}{\partial \xi^{\alpha}}\frac{\partial \bar{\xi}^{\alpha}}{\partial t} &= 0, && (2.24)
\end{aligned}$$

where the material time derivative $\mathbf{V}^{(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ can be recognized

$$\mathbf{V}^{(i)}(\mathbf{X}^{(i)},t) := \frac{\partial \mathbf{x}^{(i)}(\mathbf{X}^{(i)},t)}{\partial t}. \tag{2.25}$$

Utilizing this definition along with (2.9) and (2.10) one finally gets

$$\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}-\bar{\mathbf{F}}^{(k)}\bar{\boldsymbol{\mathcal{T}}}^{(k)}_{\alpha}\dot{\bar{\xi}}^{\alpha}=0. \tag{2.26}$$

The last term on the left hand side of (2.26) presents the rate of relative tangential slip expressed with the aid of the time derivative of convective coordinates related to the spatial tangent base

$$\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} := \dot{\bar{\xi}}^{\alpha}\bar{\mathbf{F}}^{(k)}\bar{\boldsymbol{\mathcal{T}}}^{(k)}_{\alpha} = \dot{\bar{\xi}}^{\alpha}\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}. \tag{2.27}$$

It means that contravariant components of the slip rate vector, $\dot{g}_{\mathrm{T}}^{(i)\alpha}$, are equal to the time derivative of the convective coordinates, $\dot{\bar{\xi}}^{\alpha}$. Often it will be useful to express components of the relative tangential velocity vector by its covariant components

$$\boxed{\dot{g}_{\mathrm{T}\alpha}^{(i)} = \dot{\bar{\xi}}_{\alpha} = \bar{m}^{(k)}_{\alpha\beta}\dot{\bar{\xi}}^{\beta}.} \tag{2.28}$$

The unknown time derivative of convective coordinates can be obtained by differentiating the orthogonality condition (2.12) (see Appendix A.2)

$$\dot{\bar{\xi}}^{\alpha} = \left(\bar{m}^{(k)}_{\alpha\beta}+g_{\mathrm{N}}^{(i)}\bar{\kappa}^{(k)}_{\alpha\beta}\right)^{-1}\left[\left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}^{(k)}_{\beta}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\frac{\partial \bar{\mathbf{V}}^{(k)}}{\partial \xi^{\beta}}\right], \tag{2.29}$$

where $\kappa^{(k)}_{\alpha\beta}$ stands for components of the curvature tensor $\boldsymbol{\kappa}^{(k)} \in \mathbb{R}^{n_{\mathrm{pd}}\times n_{\mathrm{pd}}}$ in the spatial tangent basis, which are defined as

$$\kappa^{(k)}_{\alpha\beta} := \mathbf{n}^{(k)}\cdot\frac{\partial \mathbf{x}^{(k)}}{\partial \xi^{\alpha}\partial \xi^{\beta}}. \tag{2.30}$$

In the frictional contact the stick and slip state has to be distinguished. The characteristic feature of the stick state is that there is no relative movement between the bodies, i.e.

$$\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} = \mathbf{0}. \tag{2.31}$$

Conversely, during the slip the relative tangential velocity is non-zero and hence relative movement between contacting bodies can occur. It means that the closest projection $\bar{\mathbf{x}}^{(k)}$ of the slave point $\mathbf{x}^{(i)}$ changes its convective coordinates $\bar{\boldsymbol{\xi}}$. The infinitesimal relative tangential displacement of these two points is

$$\mathrm{d}\mathbf{g}_{\mathrm{T}}^{(i)} = \dot{\mathbf{g}}_{\mathrm{T}}^{(i)}\,\mathrm{d}t = \dot{\bar{\xi}}_{\alpha}\bar{\boldsymbol{\tau}}^{(k)\alpha}\,\mathrm{d}t. \tag{2.32}$$

2.2 Contact traction vector

The traction vector $\mathbf{t}_\mathrm{c}^{(k)} \in \mathbb{R}^{n_\mathrm{sd}}$ acting on the current contact boundary $\gamma_\mathrm{c}^{(k)}$ is given by Cauchy's stress theorem [25, p. 137]

$$\mathbf{t}_\mathrm{c}^{(k)} = \mathbf{n}^{(k)} \cdot \boldsymbol{\sigma}^{(k)}, \tag{2.33}$$

where $\boldsymbol{\sigma}^{(k)} \in \mathbb{R}^{n_\mathrm{sd} \times n_\mathrm{sd}}$ is the Cauchy stress tensor (also known as the true stress tensor). For the differential of the contact force $\mathrm{d}\mathbf{f}_\mathrm{c}^{(k)} \in \mathbb{R}^{n_\mathrm{sd}}$ in the current configuration it holds

$$\mathrm{d}\mathbf{f}_\mathrm{c}^{(k)} = \mathbf{t}_\mathrm{c}^{(k)} \mathrm{d}\gamma^{(k)} = \mathbf{n}^{(k)} \cdot \boldsymbol{\sigma}^{(k)} \mathrm{d}\gamma^{(k)}. \tag{2.34}$$

To express the counterpart of $\mathbf{t}_\mathrm{c}^{(k)}$ in the initial configuration, Nanson's formula [30, p. 366]

$$\mathbf{n}^{(k)} = J^{(k)} \frac{\mathrm{d}\Gamma^{(k)}}{\mathrm{d}\gamma^{(k)}} \mathbf{N}^{(k)} \cdot \left(\mathbf{F}^{(k)}\right)^{-1}, \tag{2.35}$$

will be exploited. Here, $J^{(k)} \in \mathbb{R}$ denotes the Jacobian determinant of the mapping from the initial to the current configuration, i.e. $J^{(k)} = \det \mathbf{F}^{(k)}$. Nanson's formula establishes a relation between the normal vector in the initial and the current configurations. Substitution from (2.35) into (2.34) yields

$$\begin{aligned}
\mathrm{d}\mathbf{f}_\mathrm{c}^{(k)} &= J^{(k)} \frac{\mathrm{d}\Gamma^{(k)}}{\mathrm{d}\gamma^{(k)}} \mathbf{N}^{(k)} \cdot \left(\mathbf{F}^{(k)}\right)^{-1} \cdot \boldsymbol{\sigma}^{(k)} \mathrm{d}\gamma^{(k)} && (2.36)\\
&= \mathbf{N}^{(k)} \cdot J^{(k)} \left(\mathbf{F}^{(k)}\right)^{-1} \cdot \boldsymbol{\sigma}^{(k)} \mathrm{d}\Gamma^{(k)} && (2.37)\\
&= \mathbf{N}^{(k)} \cdot \mathbf{P}^{(k)} \mathrm{d}\Gamma^{(k)}, && (2.38)
\end{aligned}$$

where the first Piola-Kirchhoff stress tensor $\mathbf{P}^{(k)} \in \mathbb{R}^{n_\mathrm{sd} \times n_\mathrm{sd}}$ has been introduced as

$$\mathbf{P}^{(k)} := J^{(k)} \left(\mathbf{F}^{(k)}\right)^{-1} \cdot \boldsymbol{\sigma}^{(k)}. \tag{2.39}$$

The first Piola-Kirchhoff traction vector $\mathbf{T}_\mathrm{c}^{(k)} \in \mathbb{R}^{n_\mathrm{sd}}$ acting on the initial contact boundary $\Gamma_\mathrm{c}^{(k)}$ is then given as

$$\mathbf{T}_\mathrm{c}^{(k)} = \mathbf{N}^{(k)} \cdot \mathbf{P}^{(k)}. \tag{2.40}$$

Now, for the differential of contact force it holds

$$\mathrm{d}\mathbf{f}_\mathrm{c}^{(k)} = \mathbf{t}_\mathrm{c}^{(k)} \mathrm{d}\gamma^{(k)} = \mathbf{T}_\mathrm{c}^{(k)} \mathrm{d}\Gamma^{(k)}. \tag{2.41}$$

Thus the contact traction vectors $\mathbf{t}_\mathrm{c}^{(k)}$ and $\mathbf{T}_\mathrm{c}^{(k)}$ have different length but they are pointing in the same direction. Therefore, the components of the first Piola-Kirchhoff stress tensor can be interpreted as the stresses that result from a current force vector which acts on a surface in the initial configuration [30, p. 89].

The both contact traction vectors $\mathbf{t}_\mathrm{c}^{(k)}$ and $\mathbf{T}_\mathrm{c}^{(k)}$ at some particular point $\mathbf{X}^{(k)} \in \Gamma_\mathrm{c}^{(k)}$ can be resolved into the normal and tangential direction. It should be noted that the resolution into the tangent direction can be in principle performed both in the initial [31, p. 673] or in the current

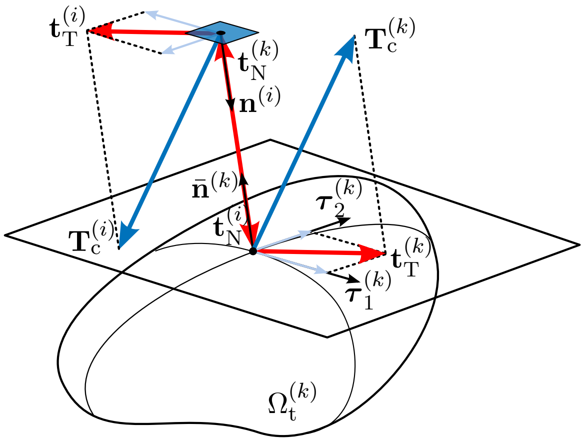


Figure 2.2: Resolution of the contact traction vector into the normal and tangent components in the current (spatial) configuration.

[32, p. 321] tangent basis. In this work only the resolution into the current tangent basis will be presented as it is more common in the numerical implementation. The resolution of the contact traction vector $\mathbf{T}_\mathrm{c}^{(k)}$ into the normal $\mathbf{t}_\mathrm{N}^{(k)} \in \mathbb{R}^{n_\mathrm{sd}}$ and the tangential $\mathbf{t}_\mathrm{T}^{(k)} \in \mathbb{R}^{n_\mathrm{sd}}$ direction in the current basis, as is depicted in Figure 2.2, yields

$$\mathbf{T}_\mathrm{c}^{(k)} = \mathbf{t}_\mathrm{N}^{(k)} + \mathbf{t}_\mathrm{T}^{(k)}. \tag{2.42}$$

Here, the normal component is

$$\begin{aligned} \mathbf{t}_\mathrm{N}^{(k)} &= \left(\mathbf{T}_\mathrm{c}^{(k)} \cdot \mathbf{n}^{(k)}\right) \mathbf{n}^{(k)} && (2.43) \\ &= \mathbf{T}_\mathrm{c}^{(k)} \left(\mathbf{n}^{(k)} \otimes \mathbf{n}^{(k)}\right). && (2.44) \end{aligned}$$

The length of the normal traction vector $t_\mathrm{N}^{(k)} \in \mathbb{R}$ is

$$t_\mathrm{N}^{(k)} = \mathbf{T}_\mathrm{c}^{(k)} \cdot \mathbf{n}^{(k)}. \tag{2.45}$$

Obviously, on the contact interface without adhesion only compression can occur, for which the normal traction is negative. This is the motivation for definition of a new non-negative quantity — the contact pressure $p_\mathrm{c}^{(k)} \in \mathbb{R}^+$

$$p_\mathrm{c}^{(k)} = -t_\mathrm{N}^{(k)}. \tag{2.46}$$

The tangential part of the traction vector can be expressed from (2.42) as

$$\begin{aligned} \mathbf{t}_\mathrm{T}^{(k)} &= \mathbf{T}_\mathrm{c}^{(k)} - \mathbf{t}_\mathrm{N}^{(k)} && (2.47) \\ &= \mathbf{T}_\mathrm{c}^{(k)} - (\mathbf{T}_\mathrm{c}^{(k)} \cdot \mathbf{n}^{(k)})\mathbf{n}^{(k)} && (2.48) \\ &= \mathbf{T}_\mathrm{c}^{(k)} \left(\mathbf{I} - \mathbf{n}^{(k)} \otimes \mathbf{n}^{(k)}\right), && (2.49) \end{aligned}$$

where $\mathbf{I} \in \mathbb{R}^{n_\mathrm{sd} \times n_\mathrm{sd}}$ is the identity matrix. The tangent traction vector can be further resolved in the current tangent basis as

$$\mathbf{t}_{\mathrm{T}}^{(k)} = t_{\mathrm{T}}^{(k)\alpha} \boldsymbol{\tau}_{\alpha}^{(k)}, \tag{2.50}$$

where the contravariant component has been introduced as

$$t_{\mathrm{T}}^{(k)\alpha} = \mathbf{T}_{\mathrm{c}}^{(k)} \cdot \boldsymbol{\tau}^{(k)\alpha}. \tag{2.51}$$

Note that contravariant basis vectors can be easily computed from (2.16) by inverse

$$\boldsymbol{\tau}^{(k)\alpha} = \left[\mathbf{m}^{(k)}\right]_{\alpha\beta}^{-1} \boldsymbol{\tau}_{\beta}^{(k)}. \tag{2.52}$$

A separate statement of the linear momentum balance applies on the contact interface stipulates that [28, p. 1130]

$$\mathbf{T}_{\mathrm{c}}^{(i)} = -\bar{\mathbf{T}}_{\mathrm{c}}^{(k)}. \tag{2.53}$$

It follows that

$$p_{\mathrm{c}}^{(i)} = \bar{p}_{\mathrm{c}}^{(k)}, \tag{2.54}$$

and

$$\mathbf{t}_{\mathrm{T}}^{(i)} = -\bar{\mathbf{t}}_{\mathrm{T}}^{(k)}. \tag{2.55}$$

Consequently, one can write

$$\begin{align} \mathbf{T}_{\mathrm{c}}^{(i)} &= \mathbf{t}_{\mathrm{N}}^{(i)} + \mathbf{t}_{\mathrm{T}}^{(i)} \tag{2.56}\\ &= t_{\mathrm{N}}^{(i)} \mathbf{n}^{(i)} - \bar{\mathbf{t}}_{\mathrm{T}}^{(k)} \tag{2.57}\\ &= -t_{\mathrm{N}}^{(i)} \bar{\mathbf{n}}^{(k)} - \bar{\mathbf{t}}_{\mathrm{T}}^{(k)} \tag{2.58}\\ &= p_{\mathrm{c}}^{(k)} \bar{\mathbf{n}}^{(k)} - t_{\mathrm{T}\alpha}^{(k)} \bar{\boldsymbol{\tau}}^{(k)\alpha}, \tag{2.59} \end{align}$$

where in (2.58) the equality of normal vectors (2.20) has been considered. By utilizing (2.54) and (2.55) one finally arrives to the expression

$$\boxed{\mathbf{T}_{\mathrm{c}}^{(i)} = p_{\mathrm{c}}^{(i)} \bar{\mathbf{n}}^{(k)} + t_{\mathrm{T}\alpha}^{(i)} \bar{\boldsymbol{\tau}}^{(k)\alpha}.} \tag{2.60}$$

In other words, the contact traction vector $\mathbf{T}_{\mathrm{c}}^{(i)}$ acting at some point $\mathbf{X}^{(i)} \in \Gamma_{\mathrm{c}}^{(i)}$ has been resolved into the normal and tangent vectors constructed in the closest point $\bar{\mathbf{X}}^{(k)} \in \Gamma_{\mathrm{c}}^{(k)}$ and its components are $p_{\mathrm{c}}^{(i)}$ and $t_{\mathrm{T}}^{(i)\alpha}$.

2.3 Normal contact conditions

The non-adhesive physical behavior at the contact boundary in the normal direction can be described using the Hertz-Signoriniho-Moreau conditions [29, p. 136]

$$p_{\mathrm{c}}^{(i)} \geq 0, \qquad g_{\mathrm{N}}^{(i)} \leq 0, \qquad p_{\mathrm{c}}^{(i)} g_{\mathrm{N}}^{(i)} = 0, \qquad \forall \mathbf{X}^{(i)} \in \Gamma_{\mathrm{c}}^{(i)}, \quad \forall t \in \mathbb{T}. \tag{2.61}$$

This conditions are known in mathematical optimization as the Karush–Kuhn–Tucker (KKT) conditions [33, p. 328] formulating the first-order necessary conditions for a solution in nonlinear programming to be optimal. The conditions are schematically depicted in Figure 2.3. The first inequality $(2.61)_1$ indicates that only non-negative contact pressure is admissible at the contact interface. The second inequality $(2.61)_2$ represents the non-penetration condition. The third equality $(2.61)_3$, is the so-called complementarity condition which assures that compression is admissible only when the gap is closed and, conversely, the potential contact interface is stress-free only when the gap is open.

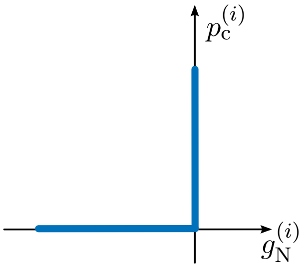
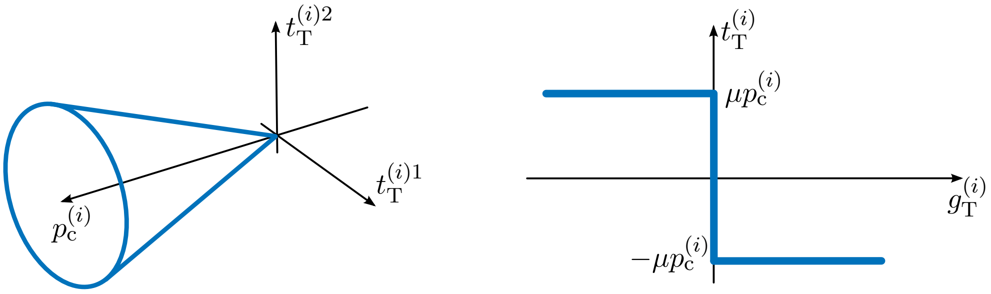


Figure 2.3: A scheme of the normal contact conditions.

2.4 Tangential contact conditions

Although this work is restricted exclusively on Coulomb's friction, the introduced concept will be general enough that the adjustment for other tangential constitutive relation can be achieved without great effort.



(a) : slip function **(b) :** 1d case of KKT conditions for friction

Figure 2.4: The slip function and the KKT conditions for Coulomb's friction law.

Coulomb's friction law can be regarded as the constitutive model for a rigid — perfectly–plastic material with infinity Young's modulus and zero isotropic hardening parameter. At the same time, the yield strength is equal to $\mu p_{\mathrm{c}}^{(i)}$, where $\mu \in \mathbb{R}$ is the coefficient of friction. This analogy between friction and plasticity is commonly exploited in the numerical implementation of friction [34, 35]. Therefore, in the spirit of the plastic analogy let us start by the additive decomposition of the relative slip velocity

$$\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} = \dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{st}(i)} + \dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} \tag{2.62}$$

where $\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{st}(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ is the "elastic" and $\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ is the "plastic" part of the slip rate. Then, the so-called slip function $\Phi^{(i)} : \mathbb{R} \times \mathbb{R}^{n_{\mathrm{sd}}} \mapsto \mathbb{R}$ as a direct analogy to the function of plasticity is introduced. For Coulomb's friction law it has the particular form

$$\Phi^{(i)}(p_{\mathrm{c}}^{(i)}, \mathbf{t}_{\mathrm{T}}^{(i)}) := \left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\| - \mu p_{\mathrm{c}}^{(i)} \leq 0. \tag{2.63}$$

An example of the slip function for the Coulomb friction is depicted in Figure 2.4. The relationship between the tangent stress vector and the slip rate follows from the principle of maximum plastic dissipation [36], under which

$$\left(\mathbf{t}_{\mathrm{T}}^{(i)} - \mathbf{t}_{\mathrm{T}}^{*}\right) \cdot \dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} \leq 0, \qquad \forall \mathbf{t}_{\mathrm{T}}^{*} \in \left\{ \mathbf{t}_{\mathrm{T}}^{*} \in \mathbb{R}^{n_{\mathrm{sd}}} | \Phi(\mathbf{t}_{\mathrm{T}}^{*}) \leq 0 \right\}. \tag{2.64}$$

This inequality implies that

$$\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} = -\dot{\lambda}^{(i)} \frac{\partial \Phi^{(i)}}{\partial \mathbf{t}_{\mathrm{T}}^{(i)}}, \tag{2.65}$$

and for the Coulomb friction law (2.63)

$$\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} = -\dot{\lambda}^{(i)} \frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\|}, \tag{2.66}$$

where $\dot{\lambda}^{(i)} \in \mathbb{R}$ is called the rate of plastic slip. Thus, the consequence of the principle of maximum plastic dissipation is that the slip due to friction occurs in the *opposite direction* to the tangent traction vector. The rate of plastic slip and the slip function are linked by the KKT conditions for the frictional contact problem

$$\dot{\lambda}^{(i)} \geq 0, \qquad \Phi^{(i)} \leq 0, \qquad \dot{\lambda}^{(i)} \Phi^{(i)} = 0, \qquad \forall \mathbf{X}^{(i)} \in \Gamma_{\mathrm{c}}^{(i)}, \quad \forall t \in \mathbb{T}. \tag{2.67}$$

For the numerical solution it will be advantageous to express the condition (2.66) in covariant coordinates

$$\boxed{\dot{g}_{\mathrm{T}\alpha}^{\mathrm{sl}(i)} = -\dot{\lambda}^{(i)} \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\|},} \tag{2.68}$$

where the length of the vector $\mathbf{t}_{\mathrm{T}}^{(i)}$ can be calculated from the identity

$$\left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\|^{2} = t_{\mathrm{T}\alpha}^{(i)} \bar{m}^{(k)\alpha\beta} t_{\mathrm{T}\beta}^{(i)}. \tag{2.69}$$

2.5 Enforcement of contact constraints

The normal (2.61) and tangential (2.67) contact conditions constitute a restriction on the solution. As such, they are the reason why the solution of contact tasks leads to a constrained optimization problem or a nonlinear programming problem, as it is called in the mathematical literature [33]. There is a plethora of numerical methods for solving this problem. The three most common methods used in computational contact mechanics include the penalty method, the Lagrange multipliers method and the augmented Lagrangian method. In this section, the principle of each of these methods will be briefly described.

2.5.1 Penalty method

The method of penalty function is the most commonly used method of regularization of contact traction vector components. The popularity of the penalty method comes from the fact that it is easy to implement and it has the vivid physical interpretation. The penalty function may, in fact, be seen as a spring, which is active only in that part of the contact boundary, where is a mutual penetration of the contacting bodies. The higher the value of penalty function, the less the resultant penetration. Hence the main disadvantage of this method, namely that the contact conditions are fulfilled only approximately for the final value of the penalty function.

The principle of regularization of the contact traction vector components by the penalty method is shown in Figure 2.5. The regularization consists in replacing the unknown components of the contact traction vector with a new ones which are dependent on the displacement field. The normal contact traction is prescribed by the function

$$\boxed{p_{\mathrm{c}}^{(i)} = \epsilon_{\mathrm{N}} \left\langle g_{\mathrm{N}}^{(i)} \right\rangle,} \tag{2.70}$$

where $\epsilon_{\mathrm{N}} \in \mathbb{R}^+$ is the normal penalty function, most commonly a constant, and $\langle \bullet \rangle : \mathbb{R} \to \mathbb{R}$ are the so-called Macaulay's brackets defined as

$$\langle x \rangle := \frac{|x| + x}{2}, \qquad x \in \mathbb{R}. \tag{2.71}$$

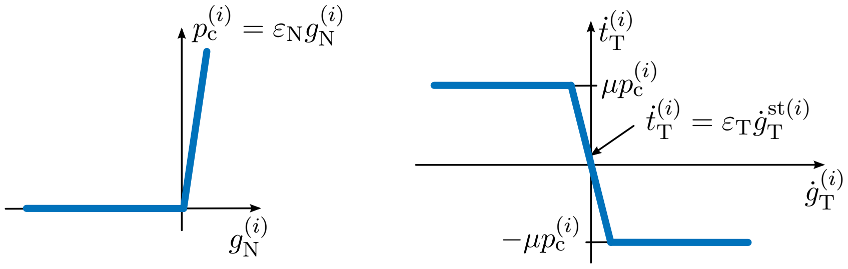
It is thus an operator that for a positive x returns x and for a negative x returns zero. It is now obvious the reason for the negative sign in the definition of the gap function (2.21). Further, the rate of the tangent contact traction is prescribed by the function

$$\dot{\mathbf{t}}_{\mathrm{T}}^{(i)} = -\epsilon_{\mathrm{T}} \dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{st}(i)} \tag{2.72}$$

$$= -\epsilon_{\mathrm{T}} \left(\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} - \dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)} \right) \tag{2.73}$$

$$= -\epsilon_{\mathrm{T}} \left(\bar{m}_{\alpha\beta}^{(k)} \dot{\xi}^{\beta} + \dot{\lambda}^{(i)} \frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\|} \right), \tag{2.74}$$

where $\dot{\mathbf{g}}_{\mathrm{T}}^{(i)}$ and $\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{sl}(i)}$ was substituted with (2.28) and (2.66), respectively. In this function, $\epsilon_{\mathrm{T}} \in \mathbb{R}^+$ is the tangent penalty function, and the expression in parentheses is the "elastic" part of the slip rate, $\dot{\mathbf{g}}_{\mathrm{T}}^{\mathrm{st}(i)}$, as is depicted in Figure 2.5b. The contact constraints are theoretically fulfilled exactly for $\epsilon_{\mathrm{N}}, \epsilon_{\mathrm{T}} \to \infty$ but in practice, however, the penalty function significantly higher than the stiffness of the contacting bodies resulting in the ill-conditioning of the linear system of equations. This problem can be eliminated by the augmented Lagrangian method.



(a) : normal direction **(b) :** tangent direction

Figure 2.5: A scheme of regularization of the normal and tangent contact constraints by the penalty method.

2.5.2 Lagrange multipliers method

Considering the method of Lagrange multipliers one can interpret components of the contact traction vector as the Lagrange multipliers

$$p_{\mathrm{c}}^{(i)} = \lambda_{\mathrm{N}}^{(i)}, \tag{2.75}$$

$$t_{\mathrm{T}\alpha}^{(i)} = \lambda_{\mathrm{T}\alpha}^{(i)}. \tag{2.76}$$

Thus, contact traction components are not regularized but instead left as the independent variables. The advantage of this method is that, in comparison with the penalty method, the contact conditions are satisfied “exactly”. As a disadvantage may be considered the fact that the resulting system of equations constitutes the saddle point problem with a semi-definite system matrix, which requires a special type of linear solver. Moreover, the frequently mentioned drawback is that Lagrange multipliers increase the number of unknowns. This shortcoming can be solved by employing so-called dual mortar method [37, 38] which, thanks to a special choice of the space of Lagrange multipliers, allows to perform the static condensation of Lagrange multipliers.

2.5.3 Augmented Lagrangian method

The augmented Lagrangian method is the robust method, which combines the advantages of the method of Lagrange multipliers and the penalty method. As the name suggests, the principle lies in the extension of the Lagrange functional by a certain term. Its purpose is to convert the problem of constrained optimization in the problem of unconstrained optimization [39]. The result is a regularization of the contact traction components in the form

$$p_{\mathrm{c}}^{(i)} = \left\langle \lambda_{\mathrm{N}}^{(i)} + \epsilon_{\mathrm{N}} g_{\mathrm{N}}^{(i)} \right\rangle, \tag{2.77}$$

$$\dot{\mathbf{t}}_{\mathrm{T}}^{(i)} = \dot{\boldsymbol{\lambda}}_{\mathrm{T}}^{(i)} - \epsilon_{\mathrm{T}} \left(\dot{\mathbf{g}}_{\mathrm{T}}^{(i)} + \dot{\lambda}^{(i)} \frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\| \mathbf{t}_{\mathrm{T}}^{(i)} \right\|} \right), \tag{2.78}$$

where both the Lagrange multipliers and the penalty terms are used. Now, however, the penalty functions can be set smaller in order of magnitude than in the case of standard penalty method. Consequently, it improves the conditioning of the resulting linear systems. In Section 2.10.5 an algorithm for efficient solution of the augmented Lagrangian is presented.

2.6 Large deformation contact problem formulation

After all the necessary variables have been defined, one can proceed to formulate the contact initial-boundary value problem in the strong sense. The Lagrangian description is chosen having regard to the linearization because integration regions will not depend on the displacement field.

2.6.1 Strong form of the contact initial-boundary value problem

At each time instance $t \in \mathbb{T}$, the conservation of momentum [29, p. 111] is required

$$\mathrm{Div}\,\mathbf{P}^{(i)} + \mathbf{f}_0^{(i)} = \rho_0^{(i)}\mathbf{A}^{(i)} \quad \text{in } \Omega_0^{(i)}, \tag{2.79}$$

where Div denotes the divergence operator with respect to the initial configuration, $\mathbf{P}^{(i)} \in \mathbb{R}^{n_{\mathrm{sd}}\times n_{\mathrm{sd}}}$ is the first Piola-Kirchhoff stress tensor, $\mathbf{f}_0^{(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ is the prescribed body force per unit reference volume, $\rho_0^{(i)} \in \mathbb{R}$ is the prescribed reference density and $\mathbf{A}^{(i)} \in \mathbb{R}^{n_{\mathrm{sd}}}$ is the material acceleration field. The equations of momentum conservation are further supplemented by the initial conditions

$$\mathbf{u}^{(i)}\Big|_{t=0} = \mathbf{u}_0^{(i)} \quad \text{in } \bar{\Omega}_0^{(i)}, \tag{2.80}$$

$$\mathbf{V}^{(i)}\Big|_{t=0} = \mathbf{V}_0^{(i)} \quad \text{in } \bar{\Omega}_0^{(i)}, \tag{2.81}$$

the boundary conditions

$$\mathbf{u}^{(i)} = \hat{\mathbf{u}}^{(i)} \quad \text{on } \Gamma_{\mathrm{u}}^{(i)}, \tag{2.82}$$

$$\mathbf{P}^{(i)}\mathbf{N}^{(i)} = \hat{\mathbf{T}}^{(i)} \quad \text{on } \Gamma_{\sigma}^{(i)}, \tag{2.83}$$

and the contact conditions on $\Gamma_{\mathrm{c}}^{(i)}$ (cf. Equation (2.60))

$$\mathbf{P}^{(i)}\mathbf{N}^{(i)} = \mathbf{T}_{\mathrm{c}}^{(i)} = p_{\mathrm{c}}^{(i)}\bar{\mathbf{n}}^{(k)} + t_{\mathrm{T}\alpha}^{(i)}\bar{\boldsymbol{\tau}}^{(k)\alpha}, \tag{2.84}$$

$$p_{\mathrm{c}}^{(i)} \geq 0, \qquad g_{\mathrm{N}}^{(i)} \leq 0, \qquad p_{\mathrm{c}}^{(i)} g_{\mathrm{N}}^{(i)} = 0, \tag{2.85}$$

$$\dot{\lambda}^{(i)} \geq 0, \qquad \Phi^{(i)} \leq 0, \qquad \dot{\lambda}^{(i)}\Phi^{(i)} = 0, \tag{2.86}$$

where $\mathbf{u}_0^{(i)} : \Omega_0^{(i)} \to \mathbb{R}^{n_{\mathrm{sd}}}$ and $\mathbf{V}_0^{(i)} : \Omega_0^{(i)} \to \mathbb{R}^{n_{\mathrm{sd}}}$ are prescribed initial displacement and initial material velocity fields, respectively. $\hat{\mathbf{u}}^{(i)} : \Gamma_{\mathrm{u}}^{(i)} \times \mathbb{T} \to \mathbb{R}^{n_{\mathrm{sd}}}$ and $\hat{\mathbf{T}}^{(i)} : \Gamma_{\sigma}^{(i)} \times \mathbb{T} \to \mathbb{R}^{n_{\mathrm{sd}}}$ are prescribed displacement and traction fields, respectively. To complete the formulation, the normal gap function (2.21), the slip function for Coulomb's friction law (2.63), and the rate of plastic slip (2.68) respectively are recalled

$$g_{\mathrm{N}}^{(i)} = -\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \bar{\mathbf{n}}^{(k)}, \tag{2.87}$$

$$\Phi^{(i)} = \sqrt{t_{\mathrm{T}\alpha}^{(i)} \bar{m}^{(k)\alpha\beta} t_{\mathrm{T}\beta}^{(i)}} - \mu p_{\mathrm{c}}^{(i)}, \tag{2.88}$$

$$\dot{g}_{\mathrm{T}\alpha}^{\mathrm{sl}(i)} = -\dot{\lambda} \frac{t_{\mathrm{T}\alpha}^{(i)}}{\sqrt{t_{\mathrm{T}\beta}^{(i)} \bar{m}^{(k)\beta\gamma} t_{\mathrm{T}\gamma}^{(i)}}}. \tag{2.89}$$

This system of equations is necessary to supplement by the constitutive relation between stress and strain, but in this work, it will not be further specified. The solution of the problem (2.79)-(2.89) is the displacement field, $\mathbf{u}^{(i)}(\mathbf{X}^{(i)}, t)$, which is required to be C^2-continuous, and the contact pressure $p_{\mathrm{c}}^{(i)}(\mathbf{X}^{(i)}, t)$ along with the covariant components of the tangent traction vector, $t_{\mathrm{T}\alpha}^{(i)}(\mathbf{X}^{(i)}, t)$. In what follow, only the penalty method will be considered for enforcement of the contact constraints. As a consequence, after the regularization of the contact pressure (2.70) and the rate of tangent traction (2.74), the only unknown of the problem (2.79)-(2.89) will be the displacement field $\mathbf{u}^{(i)}(\mathbf{X}^{(i)}, t)$. Note that the solution must satisfy (2.79)-(2.89) in every single point of $\bar{\Omega}_0^{(i)}$, hence the name strong form. As will be showed in the next section, this requirement can be weakened by considering the so-called weak form.

2.6.2 Weak form of the contact initial-boundary value problem

As is well known, to address a particular problem by the finite element method (FEM), it is first necessary to convert the strong formulation of the problem to the so-called weak formulation which has lower demands on the smoothness of the solution. It can be shown that the strong and weak formulations are equivalent [40, p. 145].

The older and to engineers usually closer approach is based on the calculus of variations, which establishes the so-called variational formulation. The variational formulation is based on the definition of a suitable energy functional. The solution of the problem is then sought by minimizing the energy functional. This leads to the well-known principle of virtual work for static problems or to the principle of virtual power in the case of dynamic problems.

In this section, the weak formulation of the contact initial-boundary values problem will be derived using the method of weighted residuals. The idea of the method is as follow. The governing equations are first converted into the residual form. Then, a trial solution from a suitable function space is thought substituted into the residual. If the scalar product of the residual for this particular trial solution is equal to zero for all so-called weighting functions, also taken from a suitable function space, then this trial solution is considered as the weak solution of the problem. In other words, the residual is orthogonal to the whole weighting space and, therefore, corresponding trial solution is the best approximation of the exact solution.

Now, appropriate function spaces will be introduced. The weighting space, $\mathcal{V}^{(i)}$, is defined as a class of all kinematically admissible displacement fields [41, p. 547], which are zero on the boundary $\Gamma_{\mathrm{u}}^{(i)}$

$$\mathcal{V}^{(i)} := \left\{ \delta\mathbf{u}^{(i)}(\mathbf{X}^{(i)}) \in H^1(\Omega_0^{(i)}) | \delta\mathbf{u}^{(i)}(\mathbf{X}^{(i)}) = \mathbf{0} \text{ on } \Gamma_{\mathrm{u}}^{(i)} \right\}, \tag{2.90}$$

and the solution space of the trial functions, $\mathcal{U}^{(i)}$, will be defined as the same class of functions but which, in addition, fulfills Dirichlet's boundary conditions on the boundary $\Gamma_{\mathrm{u}}^{(i)}$

$$\mathcal{U}^{(i)} := \left\{ \mathbf{u}^{(i)}(\mathbf{X}^{(i)},t) \in H^1(\Omega_0^{(i)}) | \mathbf{u}^{(i)}(\mathbf{X}^{(i)},t) = \hat{\mathbf{u}}^{(i)} \text{ on } \Gamma_{\mathrm{u}}^{(i)} \right\}. \tag{2.91}$$

Here, H^1 denotes the Sobolev space [42] of all functions defined on $\Omega_0^{(i)}$ that are, as well as all their first partial derivatives, square integrable. The equations of momentum conservation (2.79) in the residual form are

$$\rho_0^{(i)} \mathbf{A}^{(i)} - \operatorname{Div} \mathbf{P}^{(i)} - \mathbf{f}_0^{(i)} = \mathbf{0}, \tag{2.92}$$

and their scalar product with an arbitrary weight function $\delta \mathbf{u}^{(i)} \in \mathcal{V}^{(i)}$ yields

$$\begin{aligned}
\delta \Pi^{(i)}(\mathbf{u}^{(i)}, \delta \mathbf{u}^{(i)}) := & \int_{\Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \left(\rho_0^{(i)} \mathbf{A}^{(i)} - \operatorname{Div} \mathbf{P}^{(i)} - \mathbf{f}_0^{(i)} \right) \mathrm{d}\Omega^{(i)} && (2.93) \\
= & \int_{\Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \rho_0^{(i)} \mathbf{A}^{(i)} \, \mathrm{d}\Omega^{(i)} && (2.94) \\
- & \int_{\Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \operatorname{Div} \mathbf{P}^{(i)} \, \mathrm{d}\Omega^{(i)} && (2.95) \\
- & \int_{\Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \mathbf{f}_0^{(i)} \, \mathrm{d}\Omega^{(i)} && (2.96) \\
= & \ 0. && (2.97)
\end{aligned}$$

Utilizing integration per partes, the stress divergence term (2.95) can be expanded as

$$\int_{\Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \operatorname{Div} \mathbf{P}^{(i)} \, \mathrm{d}\Omega^{(i)} = \int_{\Omega_0^{(i)}} \operatorname{Div} \left(\delta \mathbf{u}^{(i)} \cdot \mathbf{P}^{(i)} \right) \mathrm{d}\Omega^{(i)} - \int_{\Omega_0^{(i)}} \operatorname{Grad} \delta \mathbf{u}^{(i)} : \mathbf{P}^{(i)} \, \mathrm{d}\Omega^{(i)}, \tag{2.98}$$

where Grad denotes the gradient operation with respect to the initial coordinates. Further, with the aid of Gauss-Ostrogradsky theorem one can write

$$\begin{aligned}
\int_{\Omega_0^{(i)}} \operatorname{Div} \left(\mathbf{P}^{(i)} \cdot \delta \mathbf{u}^{(i)} \right) \mathrm{d}\Omega^{(i)} = & \int_{\partial \Omega_0^{(i)}} \delta \mathbf{u}^{(i)} \cdot \mathbf{N}^{(i)} \cdot \mathbf{P}^{(i)} \, \mathrm{d}\Gamma^{(i)} && (2.99) \\
= & \int_{\Gamma_\sigma^{(i)}} \delta \mathbf{u}^{(i)} \cdot \hat{\mathbf{T}}^{(i)} \, \mathrm{d}\Gamma^{(i)} + \int_{\Gamma_{\mathrm{c}}^{(i)}} \delta \mathbf{u}^{(i)} \cdot \mathbf{T}_{\mathrm{c}}^{(i)} \, \mathrm{d}\Gamma^{(i)}, && (2.100)
\end{aligned}$$

where the integral over $\Gamma_{\mathrm{u}}^{(i)}$ is not involved thanks to the weighting space, $\mathcal{V}^{(i)}$, which has been defined in such a way that the weighting functions $\delta \mathbf{u}^{(i)}$ are equal to zero on the boundary $\Gamma_{\mathrm{u}}^{(i)}$. Backward substitution from (2.100) to (2.98) and then into (2.93) yields

$$\begin{aligned}
& \int_{\Omega_0^{(i)}} \left(\delta \mathbf{u}^{(i)} \cdot \rho_0^{(i)} \mathbf{A}^{(i)} + \operatorname{Grad} \delta \mathbf{u}^{(i)} : \mathbf{P}^{(i)} - \delta \mathbf{u}^{(i)} \cdot \mathbf{f}_0^{(i)} \right) \mathrm{d}\Omega^{(i)} \\
& \qquad - \int_{\Gamma_\sigma^{(i)}} \delta \mathbf{u}^{(i)} \cdot \hat{\mathbf{T}}^{(i)} \, \mathrm{d}\Gamma^{(i)} - \int_{\Gamma_{\mathrm{c}}^{(i)}} \delta \mathbf{u}^{(i)} \cdot \mathbf{T}_{\mathrm{c}}^{(i)} \, \mathrm{d}\Gamma^{(i)} = 0.
\end{aligned} \tag{2.101}$$

Introducing definitions for the particular terms in (2.101)

$$\delta\Pi^{(i)}_{\text{int}}(\mathbf{u}^{(i)},\delta\mathbf{u}^{(i)}) := \int_{\Omega^{(i)}_0} \delta\mathbf{u}^{(i)}\cdot\rho^{(i)}_0\mathbf{A}^{(i)}\,\mathrm{d}\Omega^{(i)} + \int_{\Omega^{(i)}_0} \operatorname{Grad}\delta\mathbf{u}^{(i)} : \mathbf{P}^{(i)}\,\mathrm{d}\Omega^{(i)}, \tag{2.102}$$

$$\delta\Pi^{(i)}_{\text{ext}}(\mathbf{u}^{(i)},\delta\mathbf{u}^{(i)}) := -\int_{\Omega^{(i)}_0} \delta\mathbf{u}^{(i)}\cdot\mathbf{f}^{(i)}_0\,\mathrm{d}\Omega^{(i)} - \int_{\Gamma^{(i)}_\sigma} \delta\mathbf{u}^{(i)}\cdot\hat{\mathbf{T}}^{(i)}\,\mathrm{d}\Gamma^{(i)}, \tag{2.103}$$

$$\delta\Pi^{(i)}_{\text{c}}(\mathbf{u}^{(i)},\delta\mathbf{u}^{(i)}) := -\int_{\Gamma^{(i)}_{\text{c}}} \delta\mathbf{u}^{(i)}\cdot\left(p^{(i)}_{\text{c}}\bar{\mathbf{n}}^{(k)} + t^{(i)}_{\text{T}\alpha}\bar{\boldsymbol{\tau}}^{(k)\alpha}\right)\,\mathrm{d}\Gamma^{(i)}, \tag{2.104}$$

where $\mathbf{T}^{(i)}_{\text{c}}$ has been substituted from (2.60). $\delta\Pi^{(i)}_{\text{int}}$ represents the virtual work done by inertia and internal forces, $\delta\Pi^{(i)}_{\text{ext}}$ comprises of terms expressing virtual work due to volume and surface forces, respectively, and $\delta\Pi^{(i)}_{\text{c}}$ denotes virtual work done by contact forces i.e. the contact residual term.

With this definitions at hand, one can proceed to formulate the contact initial-boundary value problem in the weak form: Let $\rho^{(i)}_0$, $\mathbf{f}^{(i)}_0$, $\mathbf{u}^{(i)}_0$, $\mathbf{v}^{(i)}_0$ defined on $\bar{\Omega}^{(i)}$ and $\hat{\mathbf{T}}^{(i)}$, $\hat{\mathbf{u}}^{(i)}$ defined on$\Gamma^{(i)}_\sigma$, $\Gamma^{(i)}_{\text{u}}$, respectively, are given square-integrable functions. In every time instance $t \in \mathbb{T}$, we seek for a displacement field $\mathbf{u}^{(i)} \in \mathcal{U}^{(i)}$ which for all weight functions $\delta\mathbf{u}^{(i)} \in \mathcal{V}^{(i)}$ satisfy

$$\sum_{i=1}^{2}\left(\delta\Pi^{(i)}_{\text{int}} + \delta\Pi^{(i)}_{\text{ext}} + \delta\Pi^{(i)}_{\text{c}}\right) = 0. \tag{2.105}$$

This equation serves as the basis for the numerical solution by the finite element method, which will by presented in Section 2.8.

Contact virtual work term

The contact virtual work term, or contact residual, (2.104) can be expanded into the form

$$\delta\Pi_{\text{c}} = -\int_{\Gamma^{(1)}_{\text{c}}} \delta\mathbf{u}^{(1)}\cdot\left(p^{(1)}_{\text{c}}\bar{\mathbf{n}}^{(2)} + t^{(1)}_{\text{T}\alpha}\bar{\boldsymbol{\tau}}^{(2)\alpha}\right)\,\mathrm{d}\Gamma^{(1)} - \int_{\Gamma^{(2)}_{\text{c}}} \delta\mathbf{u}^{(2)}\cdot\left(p^{(2)}_{\text{c}}\bar{\mathbf{n}}^{(1)} + t^{(2)}_{\text{T}\alpha}\bar{\boldsymbol{\tau}}^{(1)\alpha}\right)\,\mathrm{d}\Gamma^{(2)}. \tag{2.106}$$

This is the most general form of the contact residual which admits two different and independent contact traction vector fields $\mathbf{T}^{(1)}_{\text{c}}$ and $\mathbf{T}^{(2)}_{\text{c}}$. The frictionless variant of this contact residual is utilized in the contact formulation of the FEA software PMD [1, p. 2621]. From equation (2.106) it is apparent that the action-reaction principle is not directly enforced on the contact interface. The advantage of this formulation is that the integrated quantities in both integrals are independent of the counterpart contact boundary, i.e., the integral over $\Gamma^{(1)}_{\text{c}}$ depends only on coordinate $\mathbf{X}^{(1)}$ as well as the integral over $\Gamma^{(2)}_{\text{c}}$ depends only on coordinate $\mathbf{X}^{(2)}$.

If the action-reaction principle is explicitly enforced using (2.53), the contact residual in the form (2.106) can be transformed into the integral only over the slave boundary $\Gamma^{(i)}_{\text{c}}$,

$$\begin{aligned}
\delta\Pi_{\text{c}} &= -\int_{\Gamma^{(i)}_{\text{c}}} \delta\mathbf{u}^{(i)}\cdot\mathbf{T}^{(i)}_{\text{c}}\,\mathrm{d}\Gamma^{(i)} - \int_{\Gamma^{(k)}_{\text{c}}} \delta\mathbf{u}^{(k)}\cdot\left(-\mathbf{T}^{(i)}_{\text{c}}\right)\,\mathrm{d}\Gamma^{(k)} && (2.107)\\
&= -\int_{\Gamma^{(i)}_{\text{c}}} \left(\delta\mathbf{u}^{(i)}\cdot\mathbf{T}^{(i)}_{\text{c}} - \delta\mathbf{u}^{(k)}\cdot\mathbf{T}^{(i)}_{\text{c}}\right)\,\mathrm{d}\Gamma^{(i)} && (2.108)\\
&= -\int_{\Gamma^{(i)}_{\text{c}}} \mathbf{T}^{(i)}_{\text{c}}\cdot\left(\delta\mathbf{u}^{(i)} - \delta\mathbf{u}^{(k)}\right)\,\mathrm{d}\Gamma^{(i)} && (2.109)\\
&= -\int_{\Gamma^{(i)}_{\text{c}}} \left(p^{(i)}_{\text{c}}\bar{\mathbf{n}}^{(k)} + t^{(i)}_{\text{T}\alpha}\bar{\boldsymbol{\tau}}^{(k)\alpha}\right)\cdot\left(\delta\mathbf{u}^{(i)} - \delta\mathbf{u}^{(k)}\right)\,\mathrm{d}\Gamma^{(i)}, && (2.110)
\end{aligned}$$

and taking full advantage of relations for the normal gap function variation, (B.20), and variation of the convective coordinate, (B.30), derived in Appendix B, one arrives to

$$\boxed{\delta\Pi_{\mathrm{c}} = \int_{\Gamma_{\mathrm{c}}^{(i)}} \left(p_{\mathrm{c}}^{(i)} \delta g_{\mathrm{N}}^{(i)} - t_{\mathrm{T}\alpha}^{(i)} \delta\bar{\xi}^{\alpha}\right) \mathrm{d}\Gamma^{(i)}.} \tag{2.111}$$

This is the most common form of the contact residual. Notice that the integration is carried out only over the slave boundary $\Gamma_{\mathrm{c}}^{(i)}$ but the integrand depends on the both contact interfaces. Therefore, a special projection algorithms are needed [43, 44].

After discretization, the boundaries $\Gamma_{\mathrm{c}}^{(1)}, \Gamma_{\mathrm{c}}^{(2)}$ cannot be assumed to coincide, as it is the case for the continuum formulation. Some authors take into account this lack of bias by the so called two-pass algorithm [23], where the contact residual is calculated as the average value of the integral (2.111) evaluated gradually over the both contact boundaries. More generally, it is possible to introduce a parameter $\phi \in \langle 0, 1\rangle$, which governs the amount of the contribution to the contact residual from each of the contact boundaries

$$\delta\Pi_{\mathrm{c}} = (1-\phi) \int_{\Gamma_{\mathrm{c}}^{(i)}} (p_{\mathrm{c}}^{(i)} \delta g_{\mathrm{N}}^{(i)} + t_{\mathrm{T}\alpha}^{(i)} \delta\bar{\xi}^{\alpha})\, \mathrm{d}\Gamma^{(i)} + \phi \int_{\Gamma_{\mathrm{c}}^{(k)}} (p_{\mathrm{c}}^{(k)} \delta g_{\mathrm{N}}^{(k)} + t_{\mathrm{T}\alpha}^{(k)} \delta\bar{\xi}^{\alpha})\, \mathrm{d}\Gamma^{(i)}. \tag{2.112}$$

Another way how to solve the problem with ambiguous definitions of contact boundary Γ_{c} is to integrate the contact residual over a fictitious contact boundary $\Gamma_{\mathrm{c}}^{\mathrm{f}}$ [45, 24]. This contact boundary could be defined for instance as a middle surface (for 3D) or a middle curve (for 2D) of boundaries $\Gamma_{\mathrm{c}}^{(1)}$ and $\Gamma_{\mathrm{c}}^{(2)}$, i.e. each point of the boundary $\Gamma_{\mathrm{c}}^{\mathrm{f}}$ has the same closest distance from $\Gamma_{\mathrm{c}}^{(1)}$ and $\Gamma_{\mathrm{c}}^{(2)}$.

2.7 Linearization

In the previous section, the contact initial-boundary value problem was formulated in the weak sense. Thanks to the regularization by the penalty method, the resulting equation (2.105) non-linearly depends only on the displacement field $\mathbf{u}^{(i)}$. An efficient way to find a solution of system of non-linear equations is the Newton-Raphson method [46, p. 152], which will be described in detail in Section 2.10.4. It is based on iterative solving of the linearized problem. To linearize a non-linear problem, the concept of directional derivative has to be introduced [47, p. 50]. The directional derivative $D_{\Delta\mathbf{u}^{(i)}}[F]$ of a functional F at a point $\mathbf{u}^{(i)}$ and in the direction of $\Delta\mathbf{u}^{(i)}$ is defined as

$$D_{\Delta\mathbf{u}^{(i)}}\left[F\right](\mathbf{u}^{(i)}) := \left.\frac{\mathrm{d}}{\mathrm{d}\theta} F(\mathbf{u}^{(i)} + \theta\Delta\mathbf{u}^{(i)})\right|_{\theta=0}. \tag{2.113}$$

where $\theta \in \mathbb{R}$. With the directional derivative at hand, one can proceed to construct the linearization of the weak form (2.105) at the point $\mathbf{u}^{(i)}$

$$\sum_{i=1}^{2} \left\{ \left.\delta\Pi_{\mathrm{int,ext}}^{(i)}\right|_{\mathbf{u}^{(i)}} + \left.\delta\Pi_{\mathrm{c}}^{(i)}\right|_{\mathbf{u}^{(i)}} + D_{\Delta\mathbf{u}^{(i)}}\left[\delta\Pi_{\mathrm{int,ext}}^{(i)}\right] + D_{\Delta\mathbf{u}^{(i)}}\left[\delta\Pi_{\mathrm{c}}^{(i)}\right] \right\} = 0. \tag{2.114}$$

Further in this section only the linearization of the contact residual term (2.111) will be presented

$$\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[\delta\Pi_{\mathrm{c}}^{(i)}\right] =& D_{\Delta\mathbf{u}^{(i)}}\left[\int_{\Gamma_{\mathrm{c}}^{(i)}}\left(p_{\mathrm{c}}^{(i)}\delta g_{\mathrm{N}}^{(i)} - t_{\mathrm{T}\alpha}^{(i)}\delta\bar{\xi}^{\alpha}\right)\,\mathrm{d}\Gamma^{(i)}\right] && (2.115)\\
=& \int_{\Gamma_{\mathrm{c}}^{(i)}}\left(D_{\Delta\mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{(i)}\right]\delta g_{\mathrm{N}}^{(i)} + p_{\mathrm{c}}^{(i)} D_{\Delta\mathbf{u}^{(i)}}\left[\delta g_{\mathrm{N}}^{(i)}\right]\right.\\
&\left. - D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{(i)}\right]\cdot\delta\bar{\xi}^{\alpha} - t_{\mathrm{T}\alpha}^{(i)}\cdot D_{\Delta\mathbf{u}^{(i)}}\left[\delta\bar{\xi}^{\alpha}\right]\right)\,\mathrm{d}\Gamma^{(i)}. && (2.116)
\end{aligned}$$

Note that in the case of the penalty method, not only the normal gap function but also components of the contact traction vector are dependent on the displacement field. For the contact pressure one gets

$$\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{(i)}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[\epsilon_{\mathrm{N}}\left\langle g_{\mathrm{N}}^{(i)}\right\rangle\right] && (2.117)\\
&= \epsilon_{\mathrm{N}} H\left(g_{\mathrm{N}}^{(i)}\right) D_{\Delta\mathbf{u}^{(i)}}\left[g_{\mathrm{N}}^{(i)}\right]. && (2.118)
\end{aligned}$$

Because the tangent traction vector is defined by the evolution equation (2.74) and, as such, it is time dependent, it has to be integrated in time. In principle, the linearization can be performed both before or after the time integration. In this work, the time integration of the friction model is postponed to Section 2.9.3 and at this point, the linearization of the rate of tangent traction vector, $\dot{\mathbf{t}}_{\mathrm{T}}^{(i)}$, is performed.

The linearization of the tangent traction vector components depends on whether it is in the stick or slip state. In the stick state the directional derivative in the direction of $\mathbf{u}^{(i)}$ yields

$$\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[\dot{t}_{\mathrm{T}\alpha}^{\mathrm{st}(i)}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[-\epsilon_{\mathrm{T}}\bar{m}_{\alpha\beta}^{(k)}\dot{\bar{\xi}}^{\beta}\right] && (2.119)\\
&= -\epsilon_{\mathrm{T}}\left(D_{\Delta\mathbf{u}^{(i)}}\left[\bar{m}_{\alpha\beta}^{(k)}\right]\dot{\bar{\xi}}^{\beta} + \bar{m}_{\alpha\beta}^{(k)} D_{\Delta\mathbf{u}^{(i)}}\left[\dot{\bar{\xi}}^{\beta}\right]\right), && (2.120)
\end{aligned}$$

and in the case of slip

$$\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[\dot{t}_{\mathrm{T}\alpha}^{\mathrm{sl}(i)}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[\dot{t}_{\mathrm{T}\alpha}^{\mathrm{st}(i)}\right] + D_{\Delta\mathbf{u}^{(i)}}\left[-\epsilon_{\mathrm{T}}\dot{\lambda}^{(i)}\frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right] && (2.121)\\
&= D_{\Delta\mathbf{u}^{(i)}}\left[\dot{t}_{\mathrm{T}\alpha}^{\mathrm{st}(i)}\right] - \epsilon_{\mathrm{T}}\left(D_{\Delta\mathbf{u}^{(i)}}\left[\dot{\lambda}^{(i)}\right]\frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} + \dot{\lambda}^{(i)} D_{\Delta\mathbf{u}^{(i)}}\left[\frac{\mathbf{t}_{\mathrm{T}}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right]\right). && (2.122)
\end{aligned}$$

For the sake of completeness, the expressions for all involved variations and directional derivatives of the contact quantities are derived in Appendices B and C.

2.8 Spatial discretization by the finite element method

The aim of the spatial discretization is to construct finite-dimensional subspaces $\mathcal{U}^{h(i)}$ and $\mathcal{V}^{h(i)}$ of the infinite-dimensional function spaces $\mathcal{U}^{(i)}$ and $\mathcal{V}^{(i)}$ defined in Section 2.6.2. From now on discretized quantities will be denoted by the superscripted letter h. First of all, the region

$\Omega^{(i)} \subset \mathbb{R}^{n_\mathrm{sd}}$ is approximated with the aid of the so-called finite elements $\Omega_e^{(i)} \subset \mathbb{R}^{n_\mathrm{sd}}$, as is depicted in Figure 2.6

$$\Omega^{(i)} \approx \Omega^{h(i)} = \bigcup_{e}^{n_\mathrm{el}} \Omega_e^{(i)}, \tag{2.123}$$

where e is the element index and n_el is the number of elements. The geometry of each finite element is described by the mapping $\mathbf{X}^{e\,(i)}(\boldsymbol{\zeta}) : \Omega_\square \mapsto \Omega_e^{(i)}$, where $\Omega_\square \subset \mathbb{R}^{n_\mathrm{sd}}$ indicates the parametric region. (In $\mathbb{R}^1$ the parametric domain is usually the interval $\Omega_\square = [-1, 1]$, in $\mathbb{R}^2$ it is the square $\Omega_\square = [-1, 1] \times [-1, 1]$, etc.) This mapping is defined as

$$\mathbf{X}^{e\,(i)}(\boldsymbol{\zeta}) := \sum_{a=1}^{n_\mathrm{en}} N_a(\boldsymbol{\zeta}) \mathbf{X}_a^{e\,(i)}, \tag{2.124}$$

where $\mathbf{X}_a^{e\,(i)} \in \Omega_e^{(i)}$ is the initial position vector of the a-th nodal point, $\boldsymbol{\zeta} \in \Omega_\square$ is the vector of isoparametric coordinates, $N_a(\boldsymbol{\zeta})$ is the shape function associated with the a-th nodal point, and n_en indicates the number of element nodes. Shape functions are usually chosen as the Lagrange polynomials of the first or second order. Other requirements on shape functions are mainly the linear independence and partition-of-unity. Details can be found e.g. in monographs [11, 12], or [13].

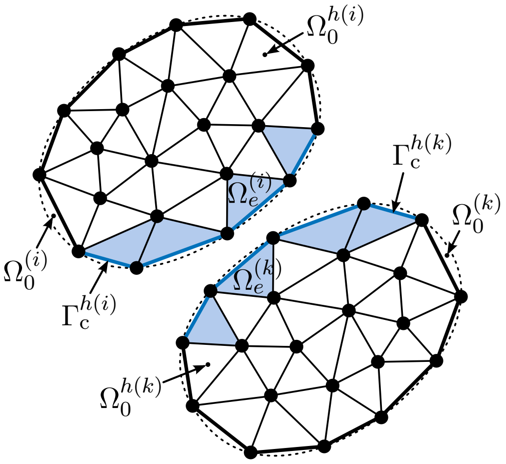


Figure 2.6: Finite element discretization of regions $\Omega_0^{(i)}$ by $\Omega_0^{h(i)}$.

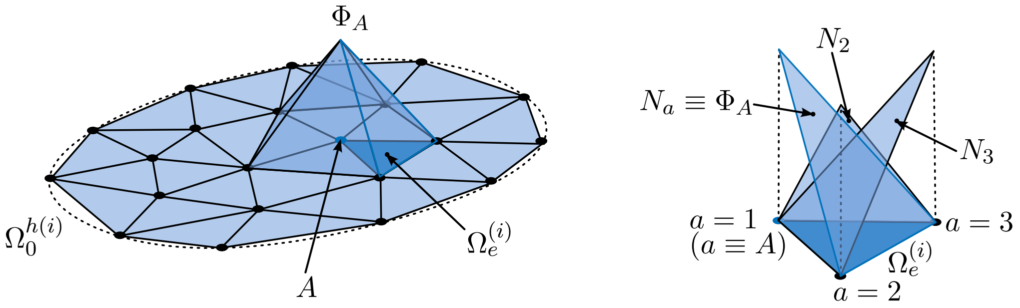
Further, so-called isoparametric concept, first introduced by Irons [48], will be employed. The fundamental idea consists in the fact that any field being discretized on the finite element region $\Omega_e^{(i)}$ uses the same shape functions as the finite element itself in the mapping for $\mathbf{X}^{e\,(i)}(\boldsymbol{\zeta})$ (2.124). Consequently, the discretized displacement field, displacement test functions and the field of spatial coordinates, defined on the finite element $\Omega_e^{(i)}$ can be written as

$$\mathbf{u}^{e\,(i)}(\boldsymbol{\zeta}, t) := \sum_{a=1}^{n_\mathrm{en}} N_a(\boldsymbol{\zeta}) \mathbf{d}_a^{(i)}(\mathbf{X}_a^{e\,(i)}, t), \tag{2.125}$$

$$\delta\mathbf{u}^{e\,(i)}(\boldsymbol{\zeta}) := \sum_{a=1}^{n_\mathrm{en}} N_a(\boldsymbol{\zeta}) \mathbf{c}_a^{(i)}(\mathbf{X}_a^{e\,(i)}), \tag{2.126}$$

$$\mathbf{x}^{e\,(i)}(\boldsymbol{\zeta},t) := \sum_{a=1}^{n_{\text{en}}} N_a(\boldsymbol{\zeta})\mathbf{x}_a^{(i)}(\mathbf{X}_a^{e\,(i)},t), \tag{2.127}$$

where $\mathbf{d}_a^{(i)} \in \mathbb{R}^{n_{\text{sd}}}$ are nodal displacements, $\mathbf{c}_a^{(i)} \in \mathbb{R}^{n_{\text{sd}}}$ are nodal weighted displacements, and $\mathbf{x}_a^{(i)} \in \mathbb{R}^{n_{\text{sd}}}$ are nodal position vectors in the current configuration.



(a) : a basis function

(b) : an element shape functions

Figure 2.7: An example of the basis function Φ_A defined over the whole region $\Omega_0^{h(i)}$ and its counterpart on the finite element $\Omega_e^{(i)}$ — the shape function N_a for $a = 1$.

By the unification of the defined fields over all finite elements $\Omega_e^{(i)}$ one gets fields defined over the whole domain $\Omega^{h(i)}$

$$\mathbf{u}^{h(i)}(\mathbf{X}^{h(i)},t) = \bigcup_{e}^{n_{\text{el}}} \mathbf{u}^{e\,(i)}(\boldsymbol{\zeta},t) = \sum_{A=1}^{n_{\text{nod}}} \Phi_A(\mathbf{X}^{h(i)})\mathbf{d}_A^{(i)}(t), \tag{2.128}$$

$$\delta\mathbf{u}^{h(i)}(\mathbf{X}^{h(i)}) = \bigcup_{e}^{n_{\text{el}}} \delta\mathbf{u}^{e\,(i)}(\boldsymbol{\zeta}) = \sum_{A=1}^{n_{\text{nod}}} \Phi_A(\mathbf{X}^{h(i)})\mathbf{c}_A^{(i)}, \tag{2.129}$$

$$\mathbf{x}^{h(i)}(\mathbf{X}^{h(i)},t) = \bigcup_{e}^{n_{\text{el}}} \mathbf{x}^{e\,(i)}(\boldsymbol{\zeta},t) = \sum_{A=1}^{n_{\text{nod}}} \Phi_A(\mathbf{X}^{h(i)})\mathbf{x}_A^{(i)}(t), \tag{2.130}$$

where the local description using the shape functions $N_a(\boldsymbol{\zeta})$ has been replaced by the global description using the basis functions $\Phi_A(\mathbf{X}^{h(i)})$, where $A = 1,\ldots,n_{\text{nod}}$ are global indices of nodal points and n_{nod} is the total number of nodes. The set of all functions defined in this way forms the finite-dimensional subspaces $\mathcal{V}^{h(i)} \subset \mathcal{V}^{(i)}$ and $\mathcal{U}^{h(i)} \subset \mathcal{U}^{(i)}$

$$\mathcal{V}^{h(i)} := \left\{ \delta\mathbf{u}^{h(i)} = \sum_{A=1}^{n_{\text{nod}}} \Phi_A(\mathbf{X}^{h(i)})\mathbf{c}_A^{(i)} | \delta\mathbf{u}^{h(i)} = \mathbf{0} \text{ on } \Gamma_{\text{u}}^{h(i)} \right\}, \tag{2.131}$$

$$\mathcal{U}^{h(i)} := \left\{ \mathbf{u}^{h(i)} = \sum_{A=1}^{n_{\text{nod}}} \Phi_A(\mathbf{X}^{h(i)})\mathbf{d}_A^{(i)}(t) | \mathbf{u}^{h(i)}(\mathbf{X}^{h(i)},t) = \hat{\mathbf{u}} \text{ on } \Gamma_{\text{u}}^{h(i)} \right\}. \tag{2.132}$$

Note that it is also possible to proceed in reverse. That is, to chose the basis functions, Φ_A, which would be defined on the whole region $\Omega^{h(i)}$ (i.e. the principle of the Ritz method), and then perform their division into finite elements. The advantage of this process is that the basis functions could be chosen in a way that the geometry would be described accurately, i.e. $\Omega^{h(i)} \equiv \Omega_0^{(i)}$. This can be achieved e.g. by choosing the basis functions as some sort of splines. Just described idea is the essential idea behind the isogeometric finite element analysis [49, p. 4138] as a modern method of spatial discretization.

The application of the isogeometric analysis to the solution of contact problems is especially appealing thanks to the guaranteed smoothness of the contact boundaries [50, 51, 52, 53, 54]. In particular, it eliminates the problem of the ambiguous definition of the normal vector field, which for classical finite elements leads to oscillations of the contact traction vector.

A discrete form of the weak formulation, introduced in Section 2.6.2, can be obtained by the Bubnov-Galerkin weighted residual method [12, p. 7]. The principle is simple. It consist in replacement of the infinite-dimensional spaces $\mathcal{U}^{(i)}$ and $\mathcal{V}^{(i)}$ in the weak formulation (2.105) by their finite-dimensional subspaces $\mathcal{U}^{h(i)}$ and $\mathcal{V}^{h(i)}$.

The discrete weak form of the contact initial-boundary value problem can be formulated as follow: Let $\rho_0^{(i)}$, $\mathbf{f}_0^{(i)}$, $\mathbf{u}_0^{(i)}$, $\mathbf{v}_0^{(i)}$ defined on $\bar{\Omega}^{h(i)}$ and $\hat{\mathbf{T}}^{(i)}$, $\hat{\mathbf{u}}^{(i)}$ defined on $\Gamma_\sigma^{h(i)}$, $\Gamma_\mathrm{u}^{h(i)}$, respectively, are given square-integrable functions. In every time instance $t \in \mathbb{T}$, we seek for a displacement field $\mathbf{u}^{h(i)} \in \mathcal{U}^{h(i)}$ which for all weight functions $\delta\mathbf{u}^{h(i)} \in \mathcal{V}^{h(i)}$ satisfies

$$\textstyle\sum_{i=1}^{2} \left\{ \delta\Pi_\mathrm{int}^{h(i)} + \delta\Pi_\mathrm{ext}^{h(i)} + \delta\Pi_\mathrm{c}^{h(i)} \right\} \quad = \quad 0, \tag{2.133}$$

where the discrete equivalents of (2.133) are obtained by the substitution from the definition of spaces $\mathcal{U}^{h(i)}$ (2.132) and $\mathcal{V}^{h(i)}$ (2.131). After some manipulations which can be found e.g. in [13, 29], one can write (2.133) in the matrix notation

$$\mathbf{c}^\mathrm{T} \left[\mathbf{M}\ddot{\mathbf{d}}(t) + \mathbf{F}_\mathrm{int}(\mathbf{d}(t)) - \mathbf{F}_\mathrm{ext}(t) + \mathbf{F}_\mathrm{c}(\mathbf{d}(t)) \right] \quad = \quad 0, \tag{2.134}$$

where $\mathbf{c} \in \mathbb{R}^{n_\mathrm{dof}}$ is the vector of nodal weighting coefficients of displacement, n_dof is the number of degree of freedom (number of unknowns of the problem), $\mathbf{M} \in \mathbb{R}^{n_\mathrm{dof} \times n_\mathrm{dof}}$ is the mass matrix, $\ddot{\mathbf{d}} \in \mathbb{R}^{n_\mathrm{dof}}$ is the vector of nodal accelerations, $\mathbf{F}_\mathrm{int} \in \mathbb{R}^{n_\mathrm{dof}}$ is the vector of internal nodal forces, $\mathbf{F}_\mathrm{ext} \in \mathbb{R}^{n_\mathrm{dof}}$ is the vector of external nodal forces and $\mathbf{F}_\mathrm{c} \in \mathbb{R}^{n_\mathrm{dof}}$ is the vector of nodal contact forces. As was explained in Section 2.6.2, equations (2.134) must hold for all displacement weighting functions from the test space $\mathcal{V}^{h(i)}$. But this means that vector $\mathbf{c}$ of nodal weighting coefficients can be arbitrary. Therefore, the term in square brackets in (2.134) has to be equal to zero vector

$$\boxed{\mathbf{M}\ddot{\mathbf{d}}(t) + \mathbf{F}_\mathrm{int}(\mathbf{d}(t)) - \mathbf{F}_\mathrm{ext}(t) + \mathbf{F}_\mathrm{c}(\mathbf{d}(t)) \quad = \quad \mathbf{0}.} \tag{2.135}$$

Further in this section, the discretization of the contact residual term $\delta\Pi_\mathrm{c}$ (2.111) will be described. Because the discretized version of this term is integrated only over the contact boundary $\Gamma_\mathrm{c}^{h(i)}$ (see Figure 2.6), to establish the global vector of nodal contact forces, $\mathbf{F}_\mathrm{c}$, it is sufficient to consider the restriction of functions $\mathbf{u}^{h(i)}$ and $\delta\mathbf{u}^{h(i)}$ on $\Gamma_\mathrm{c}^{h(i)}$.

All types of contact discretization can be written in a common form

$$\delta\Pi_\mathrm{c}^h = \sum_{s=1}^{n_\mathrm{c}} \left(p_{\mathrm{c}s}^{(i)} \delta g_{\mathrm{N}s}^{(i)} - t_{\mathrm{T}s\alpha}^{(i)} \delta\bar{\xi}_s^\alpha \right) A_s^{(i)}, \tag{2.136}$$

where the integration over contact interface has been replaced by the summation over n_c active contact points (nodes or integration points). $A_s^{(i)}$ means the size of the associated contact boundary that belongs to the s-th active node. Note that superscript h indicating discretization will be omitted in that cases where the discrete character is noted by summation index s.

The variations of the discretized normal gap function and convective coordinates can be written in matrix notation as

$$\delta g_{\mathrm{N}s}^{(i)} = -\mathbf{c}_s^{\mathrm{T}}\mathbf{N}, \tag{2.137}$$

$$\delta\bar{\xi}_s^{\alpha} = \mathbf{c}_s^{\mathrm{T}}\mathbf{D}_{\alpha}, \tag{2.138}$$

where $\mathbf{c}_s \in \mathbb{R}^{n_{\mathrm{nes}}^{(i)}+n_{\mathrm{nes}}^{(k)} \times n_{\mathrm{sd}}}$ is the matrix of nodal weighted displacements, $n_{\mathrm{nes}}^{(i)}$ is the number of nodes of the element segment (i.e. element edge in 2D or element face in 3D), and $\mathbf{N}, \mathbf{D}_{\alpha} \in \mathbb{R}^{n_{\mathrm{nes}}^{(i)}+n_{\mathrm{nes}}^{(k)} \times n_{\mathrm{sd}}}$ are matrices that will be specified by the contact discretization type in the following subsections. Substituting these matrix expressions into the discretized contact residual (2.136) yields

$$\delta\Pi_{\mathrm{c}}^{h} = \sum_{s=1}^{n_{\mathrm{c}}} \left[p_{\mathrm{c}s}^{(i)} \left(-\mathbf{c}_s^{\mathrm{T}}\mathbf{N} \right) - t_{\mathrm{T}s\alpha}^{(i)} \left(\mathbf{c}_s^{\mathrm{T}}\mathbf{D}_{\alpha} \right) \right] A_s, \tag{2.139}$$

$$= \sum_{s=1}^{n_{\mathrm{c}}} \mathbf{c}_s^{\mathrm{T}} \left(\mathbf{F}_{\mathrm{N}s} + \mathbf{F}_{\mathrm{T}s} \right), \tag{2.140}$$

where the vector of normal and tangent components of the local equivalent contact forces has been introduced as

$$\mathbf{F}_{\mathrm{N}s} = -p_{\mathrm{c}s}^{(i)}\mathbf{N}A_s, \tag{2.141}$$

$$\mathbf{F}_{\mathrm{T}s} = t_{\mathrm{T}s\alpha}^{(i)}\mathbf{D}_{\alpha}A_s. \tag{2.142}$$

Note that the global vector $\mathbf{F}_{\mathrm{c}}$ can be easily obtained from the local vectors $\mathbf{F}_{\mathrm{N}s}$ and $\mathbf{F}_{\mathrm{T}s}$ utilizing the standard finite element assembly procedure.

2.8.1 Node-to-node discretization

Node-to-node is the easiest way of contact discretization [18]. The contact constraints are enforced only point-wise in contact nodes, that is the principle of collocation methods [55]. Since the large slip of the contact interface is not permitted, this formulation is suitable only for geometrically linear problems. Normal vectors are defined in master nodes as the average normal vector of adjacent segments.

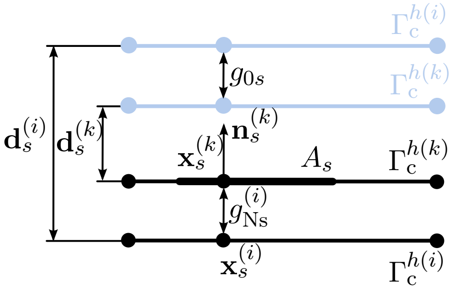


Figure 2.8: Node-to-node discretization.

The indisputable advantage of this formulation is its simplicity. Another advantage is that the node-to-node discretization satisfies the contact patch test [56, 57], which verifies the ability of a contact discretization to transfer a constant pressure load between two bodies. The main disadvantage is the fact that the finite element mesh of a contact interface has to be conform. A typical configuration with the s-th contact pair is shown in Figure 2.8. There is the initial configuration, outlined by the light blue colour, with the initial gap g_{0s}. The current configuration is denoted by the black colour. After deformation, in the kinematically linear case, the normal gap function can be written using the nodal displacements as

$$g_{\mathrm{N}s}^{(i)} = \left(\mathbf{d}_s^{(i)} - \mathbf{d}_s^{(k)}\right) \cdot \bar{\mathbf{n}}_s^{(k)} + g_{0s} = \mathbf{N}^{\mathrm{T}}\mathbf{d}_s + g_{0s}. \tag{2.143}$$

Equation (D.10), governing the variation and linearization of the convective coordinates, simplifies to

$$\mathbf{D}_\alpha = A^{\alpha\beta}\mathbf{T}_\beta, \tag{2.144}$$

where for $A^{\alpha\beta}$ holds (D.6). The main consequence of considering the kinematically linear contact is that the geometric part of the contact tangent matrix is zero. The concrete form of the previously defined matrices is

$$\mathbf{d}_s = \begin{bmatrix} \mathbf{d}_s^{(i)} \\ \mathbf{d}_s^{(k)} \end{bmatrix}, \quad \Delta\mathbf{d}_s = \begin{bmatrix} \Delta\mathbf{d}_s^{(i)} \\ \Delta\mathbf{d}_s^{(k)} \end{bmatrix}, \quad \mathbf{c}_s = \begin{bmatrix} \mathbf{c}_s^{(i)} \\ \mathbf{c}_s^{(k)} \end{bmatrix}, \tag{2.145}$$

$$\mathbf{N} = \begin{bmatrix} \mathbf{n}_s^{(k)} \\ -\mathbf{n}_s^{(k)} \end{bmatrix}, \quad \mathbf{T}_\alpha = \begin{bmatrix} \boldsymbol{\tau}_{s\alpha}^{(k)} \\ -\boldsymbol{\tau}_{s\alpha}^{(k)} \end{bmatrix}. \tag{2.146}$$

2.8.2 Node-to-segment discretization

First note that the term segment is used to denote the portion of the contact boundary per one element, both the 2D and the 3D case. The node-to-segment discretization [58, 59, 60, 61] is currently the most widely used contact discretization in commercial FEA softwares. This type of contact discretization eliminates the drawbacks of the node-to-node discretization. Contact constraints are enforced between the slave node, $\mathbf{x}_s^{(i)}$, and the corresponding point, $\bar{\mathbf{x}}^{(k)}(\boldsymbol{\xi})$, on the master segment, as is depicted in Figure 2.9. The contact reaction in the slave node is then interpolated into the nodes of master segment. Therefore, it is possible to use also non-conforming meshes of the contact interface. Moreover, unlike the node-to-node discretization, the large mutual sliding of contacting bodies are permitted.

Slave nodes are thus prevented from penetration into master segments, however, a master segment can, in principle, penetrate into a slave segment. Therefore, it is recommended selecting as a slave body that one with a finer mesh. This contact bias can be treated with the two-pass algorithm, as was described in Section 2.6.2. Because the node-to-segment discretization simply enforces zero penetration of slave nodes, a localized concentration of contact forces occurs at this nodes, leading to oscillation of the contact pressure.

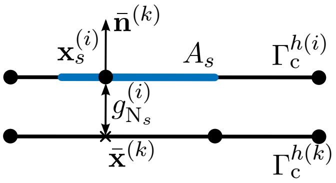


Figure 2.9: Node-to-segment discretization.

The main disadvantage of this formulation is that the one-pass node-to-segment algorithm fails the contact patch test [56], whereas the two-pass node-to-segment algorithm passes the patch test but only if used in conjunction with the Lagrange multiplier method [62, p. 380].

Another drawback of the node-to-segment discretization is that it can lead to the so-called surface locking [63]. In general, locking is a phenomenon, when the mesh refinement does not lead to the convergence of the solution. It is due to the fact that the shape functions of higher orders lead to overdetermined system of linear equations for the calculation of nodal displacements. The locking problem is caused by failure of inf-sup condition, also known as Ladyzhenskaya-Babuška-Breezi (LBB) condition, which establish the condition for the well posedness of the variational problem in the mixed form. For the node-to-segment contact discretization the matrix quantities are

$$\mathbf{x}_s = \begin{bmatrix} \mathbf{x}_s^{(i)} \\ \mathbf{x}_1^{(k)} \\ \vdots \\ \mathbf{x}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \quad \Delta\mathbf{d}_s = \begin{bmatrix} \Delta\mathbf{d}_s^{(i)} \\ \Delta\mathbf{d}_1^{(k)} \\ \vdots \\ \Delta\mathbf{d}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \quad \mathbf{c}_s = \begin{bmatrix} \mathbf{c}_s^{(i)} \\ \mathbf{c}_1^{(k)} \\ \vdots \\ \mathbf{c}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \tag{2.147}$$

$$\mathbf{N} = \begin{bmatrix} \bar{\mathbf{n}}^{(k)} \\ -N_1^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)}}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \quad \mathbf{N}_\alpha = \begin{bmatrix} \mathbf{0} \\ -N_{1,\alpha}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\alpha}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \quad \mathbf{N}_{\alpha\beta} = \begin{bmatrix} \mathbf{0} \\ -N_{1,\alpha\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\alpha\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \tag{2.148}$$

$$\mathbf{T}_\alpha = \begin{bmatrix} \bar{\boldsymbol{\tau}}^{(k)\alpha} \\ -N_1^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}^{(k)\alpha} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)}}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}^{(k)\alpha} \end{bmatrix}, \quad \mathbf{T}_{\alpha\beta} = \begin{bmatrix} \mathbf{0} \\ -N_{1,\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}^{(k)\alpha} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}^{(k)\alpha} \end{bmatrix}. \tag{2.149}$$

2.8.3 Segment-to-segment discretization

As the segment-to-segment discretization is denoted the contact formulation first presented in [24], which was extended for the large deformation case in [64]. This type of discretization enforces contact conditions not pointwise, like the collocation method, but in average over contact segments, i.e. in the weak sense. Contact conditions are evaluated in integration points, as shown in Figure 2.10. The result is in comparison with the node-to-segment discretization a smoother contact pressure. In addition, this type of discretization is not so sensitive to the choice of the master/slave contact boundary, as it is in the case of node-to-segment discretization.

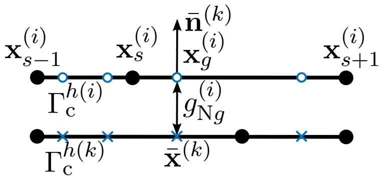


Figure 2.10: Segment-to-segment discretization.

Mortar discretization

Also the mortar method can be understood as the segment-to-segment contact discretization technique. The mortar FE method is a discretization technique for partial differential equations (PDE), which uses special FE spaces of Lagrange multipliers on non-overlapping subdomains to enforce the equality of the solution on the interface. It was originally proposed as a new approach to discretization-driven domain decomposition in application to the spectral element method and its performance was demonstrated on the Navier-Stokes problem [65]. Shortly after that, application to moving-geometry sliding-mesh problem was presented in [66] and fourth-order problem in FE framework was showed in [67]. The extension of the mortar FE method to handle the unilateral contact between deformable bodies was proposed in [68, 69] where among other things an upper bound of the convergence rate was given.

From the computational contact point of view, the key properties of the mortar FE method are: it preserves optimal convergence rate; results in stable discretization, i.e. fulfills the inf-sup condition, and passes the contact patch test. All this provided that a suitable space of Lagrange multiplier is chosen. For instance the dual space to the space of displacements proved itself to be a particularly advantageous choice of the Lagrange multiplier ansatz [70, 71]. This is because it leads to easier implementation as the mortar map can be represented by a diagonal matrix. Consequently, the Lagrange multipliers can be eliminated from the global system of equations by static condensation, and thus avoiding an increase in system size.

Beside continuing rigorous mathematical analysis of the mortar FE method, the papers devoted to the implementation had also appeared. 2D frictionless small deformation case was presented in [72, 73] and finite deformation case in [37, 74], where dual Lagrange space and primal-dual active set strategy (PDASS) were utilized. 2D frictional contact algorithm involving finite deformation and large sliding was presented in [75] where penalty regularization was considered. To the same problem, linear dual Lagrange space and PDASS was applied in [76, 41].

In three dimensions the evaluation of the mortar surface integrals requires special treatment by the projection of the displacement jump across the interface onto the Lagrange multiplier space [77, 78]. This projection was successfully applied on 3D frictionless large deformation problems in [44] and also utilizing PDASS with linear dual Lagrange space in [79, 38] and higher order dual Lagrange space in [80]. 3D frictional large deformation contact was presented in [81], with linear elements, and in [82] with quadratic elements.

Engineers usually understand the mortar FE method as weak enforcement of the contact constraints. This is in contrast to the conventional node-to-segment formulations where the contact constraints are enforced in the sense of collocation method only in the discrete points. Inspired by this reasoning, researchers came with so-called mortar-based contact formulations [83, 84, 85]. Also the contact virtual work discretized by the mortar-based method can be written in the form of summation over active nodal points, (2.136), provided that nodal point contact quantities are defined. In particular, the control point normal gap and its variation are defined as the weighted average, with the basis functions as weights

$$g_{\mathrm{N}s}^{(i)} = \frac{\int_{\Gamma_\mathrm{c}} \Phi_s g_\mathrm{N}^{(i)} \,\mathrm{d}\Gamma}{A_s} \qquad \delta g_{\mathrm{N}s}^{h(i)} = \frac{\int_{\Gamma_\mathrm{c}} \Phi_s \delta g_\mathrm{N}^{(i)} \,\mathrm{d}\Gamma}{A_s}, \tag{2.150}$$

and similarly for frictional contact quantities

$$\boldsymbol{\xi}_s = \frac{\int_{\Gamma_\mathrm{c}} \Phi_s \bar{\boldsymbol{\xi}}_s \,\mathrm{d}\Gamma}{A_s}, \qquad \bar{\mathbf{m}}_s^{(k)} = \frac{\int_{\Gamma_\mathrm{c}} \Phi_s \bar{\mathbf{m}}^{(k)} \,\mathrm{d}\Gamma}{A_s}. \tag{2.151}$$

where A_s is the “area of competence” of the s-th nodal point and is defined as

$$A_s = \int_{\Gamma_\mathrm{c}} \Phi_s \,\mathrm{d}\Gamma. \tag{2.152}$$

Note that an active nodal point is one for which $g_{\mathrm{N}s}^{(i)} \leq 0$. For the sake of completeness, also for the segment-to-segment contact discretization the necessary matrix quantities will be presented:

$$\mathbf{x}_s = \begin{bmatrix} \mathbf{x}_1^{(i)} \\ \vdots \\ \mathbf{x}_{n_\mathrm{nes}^{(i)}}^{(i)} \\ \mathbf{x}_1^{(k)} \\ \vdots \\ \mathbf{x}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \quad \Delta\mathbf{d}_s = \begin{bmatrix} \Delta\mathbf{d}_1^{(i)} \\ \vdots \\ \Delta\mathbf{d}_{n_\mathrm{nes}^{(i)}}^{(i)} \\ \Delta\mathbf{d}_1^{(k)} \\ \vdots \\ \Delta\mathbf{d}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \quad \mathbf{c}_s = \begin{bmatrix} \mathbf{c}_1^{(i)} \\ \vdots \\ \mathbf{c}_{n_\mathrm{nes}^{(i)}}^{(i)} \\ \mathbf{c}_1^{(k)} \\ \vdots \\ \mathbf{c}_{n_\mathrm{nes}^{(k)}}^{(k)} \end{bmatrix}, \tag{2.153}$$

$$\mathbf{N} = \begin{bmatrix} N_1^{(i)}(\boldsymbol{\eta})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ N_{n_\mathrm{nes}^{(i)}}^{(i)}(\boldsymbol{\eta})\bar{\mathbf{n}}^{(k)} \\ -N_1^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)}}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \quad \mathbf{N}_\alpha = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -N_{1,\alpha}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\alpha}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \quad \mathbf{N}_{\alpha\beta} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -N_{1,\alpha\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\alpha\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\mathbf{n}}^{(k)} \end{bmatrix}, \tag{2.154}$$

$$\mathbf{T}_\alpha = \begin{bmatrix} N_1^{(i)}(\boldsymbol{\eta})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \\ \vdots \\ N_{n_\mathrm{nes}^{(i)}}^{(i)}(\boldsymbol{\eta})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \\ -N_1^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)}}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \end{bmatrix}, \quad \mathbf{T}_{\alpha\beta} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -N_{1,\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \\ \vdots \\ -N_{n_\mathrm{nes}^{(k)},\beta}^{(k)}(\bar{\boldsymbol{\xi}})\bar{\boldsymbol{\tau}}_\alpha^{(k)} \end{bmatrix}. \tag{2.155}$$

2.9 Temporal discretization by the finite difference method

The system of non-linear ordinary differential equations (2.135) resulting from the spatial discretization by the finite element method is called semi-discrete, because the vector of equivalent nodal displacements, $\mathbf{d}(t)$, is continuous in time. This section proceeds with the time discretization. The time interval $\mathbb{T} = \langle 0, T \rangle$ is divided into subintervals $\mathbb{T} = \bigcup_{n=0}^{N} \langle t^n, t^{n+1} \rangle$ with the corresponding time step $\Delta t^n = t^{n+1} - t^n$. The unknown displacement field is then calculated at discrete time points employing an appropriate integration schemes. As a representative of temporal integration schemes, the Hilbert-Hughes-Taylor (HHT) method [86], also known as α-method, will be presented. The integration scheme can be summarized as follows:

$$\mathbf{M}\mathbf{a}^{n+1} + \mathbf{F}_{\text{int}}|_{\mathbf{d}^{n+\alpha}} + \mathbf{F}_{\text{c}}|_{\mathbf{d}^{n+\alpha}} = \mathbf{F}_{\text{c}}|_{t^{n+\alpha}}\ , \tag{2.156}$$

$$\mathbf{d}^{n+\alpha} = \alpha \mathbf{d}^{n+1} + (1-\alpha)\mathbf{d}^n, \tag{2.157}$$

$$\mathbf{d}^{n+1} = \mathbf{d}^n + \Delta t^n \mathbf{v}^n + \frac{(\Delta t^n)^2}{2}\left[(1-2\beta)\mathbf{a}^n + 2\beta \mathbf{a}^{n+1}\right], \tag{2.158}$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \Delta t^n \left[(1-\gamma)\mathbf{a}^n + \gamma \mathbf{a}^{n+1}\right], \tag{2.159}$$

where α, β, and γ are algorithmic parameters which determine stability and accuracy of the method. Parameter α controls in which point of the time subinterval, $\langle t^n, t^{n+1} \rangle$, the balance of momentum will be enforced. Parameter β determines the point of subinterval $\langle t^n, t^{n+1} \rangle$, to which the acceleration will be linearly interpolated for the calculation of the new displacement vector $\mathbf{d}^{n+1}$. And finally parameter γ has the same meaning as β but for calculation of the new velocity vector $\mathbf{v}^{n+1}$.

2.9.1 Implicit time integration schemes

If the parameter α is put equal to one, the classical implicit scheme — the Newmark method [87] — is obtained. Further, if $\beta = 1/4$ and $\gamma = 1/2$, one comes to the trapezoidal method. For such given parameters, the acceleration $\mathbf{a}^{n+1}$ can be expressed from equation (2.158) as

$$\mathbf{a}^{n+1} = \mathbf{a}^n - \frac{4}{(\Delta t^n)^2}\left(\mathbf{d}^{n+1} - \mathbf{d}^n\right) + \frac{4}{\Delta t^n}\mathbf{v}^n, \tag{2.160}$$

which can be substituted into the momentum balance equation (2.156) and after some manipulation

$$\mathbf{R}|_{\mathbf{d}^{n+1}} = \mathbf{M}\left[\mathbf{a}^n + \frac{4}{(\Delta t^n)^2}\mathbf{d}^n + \frac{4}{\Delta t^n}\mathbf{v}^n\right] - \mathbf{M}\frac{4}{(\Delta t^n)^2}\mathbf{d}^{n+1} + \mathbf{F}_{\text{int}}|_{\mathbf{d}^{n+1}} - \mathbf{F}_{\text{ext}}|_{t^{n+1}} + \mathbf{F}_{\text{c}}|_{\mathbf{d}^{n+1}} = \mathbf{0}, \tag{2.161}$$

that is the residual vector expressed in the time instance t^{n+1}. It represents a set of non-linear algebraic equations for the new displacement vector $\mathbf{d}^{n+1}$. It is therefore necessary to employ some numerical method for solving nonlinear equations, which will be described in Section 2.10.4. The advantage of the implicit method is unconditional stability. It means that the time step Δt^n can be chosen arbitrarily large, without affecting the stability of the numerical solution. The drawback of the method is the need to solve a set of nonlinear equations in each time step, which can be, especially for large problems, time consuming.

2.9.2 Explicit time integration schemes

The values $\alpha = 1$, $\beta = 0$, and $\gamma = 1/2$ yield a representative of explicit schemes — the central difference method (CDM) . If zero is substituted for the parameter β in the equation (2.158)

$$\mathbf{d}^{n+1} = \mathbf{d}^{n} + \Delta t^{n}\mathbf{v}^{n} + \frac{\left(\Delta t^{n}\right)^{2}}{2}\mathbf{a}^{n}, \tag{2.162}$$

an explicit formula for the new displacement vector $\mathbf{d}^{n+1}$ is immediately obtained. With $\mathbf{d}^{n+1}$ in hand, the new acceleration vector can be expressed for (2.156)

$$\mathbf{a}^{n+1} = \mathbf{M}^{-1}\left(\mathbf{F}_{\mathrm{ext}}|_{t^{n+1}} - \mathbf{F}_{\mathrm{int}}|_{\mathbf{d}^{n+1}} - \mathbf{F}_{\mathrm{c}}|_{\mathbf{d}^{n+1}}\right). \tag{2.163}$$

This system of linear algebraic equations can be solved very efficiently, especially for a diagonal mass matrix $\mathbf{M}$. In this case, the linear system of equations breaks up into linearly independent equations, which can be addressed in parallel. The disadvantage is that the central difference method is conditionally stable. The stable time step is driven by the stability condition

$$\Delta t^{n} \leq \frac{2}{\omega_{\max}}, \tag{2.164}$$

where $\omega_{\max}$ is the largest natural frequency of the linearized system.

2.9.3 Backward Euler method for friction

Both, in the case of Newmark methods and the method of central differences, the vector of contact forces $\mathbf{F}_{\mathrm{c}}|_{\mathbf{d}^{n+1}}$ has to be calculated at time step $n+1$. If the friction is taken into the consideration, driven by the evolution equations (2.63)-(2.67), its temporal integration has to be performed. For this purpose, the most commonly used temporal integration scheme is the backward Euler method, also known as the implicit Euler method. The quantities $\dot{\mathbf{g}}^{(i)}_{\mathrm{T}}$, $\dot{\mathbf{t}}^{(i)}_{\mathrm{T}}$, and $\dot{\lambda}$ are approximated by the backward time differences as

$$\dot{g}^{(i)}_{\mathrm{T}\alpha} = \bar{m}^{(k)}_{\alpha\beta}\dot{\bar{\xi}}^{\beta} \approx \bar{m}^{n(k)}_{\alpha\beta}\frac{\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}}{\Delta t^{n}}, \tag{2.165}$$

$$\dot{t}^{(i)}_{\mathrm{T}\alpha} \approx \frac{t^{n+1(i)}_{\mathrm{T}\alpha} - t^{n(i)}_{\mathrm{T}\alpha}}{\Delta t^{n}}, \tag{2.166}$$

$$\dot{\lambda} \approx \frac{\lambda^{n+1} - \lambda^{n}}{\Delta t^{n}} = \frac{\Delta\lambda^{n+1}}{\Delta t^{n}}. \tag{2.167}$$

These differences are used to approximate the regularized tangential components of the contact traction vector (2.74)

$$\frac{t^{n+1(i)}_{\mathrm{T}\alpha} - t^{n(i)}_{\mathrm{T}\alpha}}{\Delta t^{n}} = -\epsilon_{\mathrm{T}}\left(\bar{m}^{n(k)}_{\alpha\beta}\frac{\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}}{\Delta t^{n}} + \dot{\lambda}\frac{t^{n+1(i)}_{\mathrm{T}\alpha}}{\left\|\mathbf{t}^{n+1(i)}_{\mathrm{T}}\right\|}\right), \tag{2.168}$$

from where

$$\boxed{t_{\mathrm{T}\alpha}^{n+1} = t_{\mathrm{T}\alpha}^{n} - \epsilon_{\mathrm{T}} \left[\bar{m}_{\alpha\beta}^{n(k)} \left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta} \right) + \Delta\lambda^{n+1} \frac{t_{\mathrm{T}\alpha}^{n+1(i)}}{\left\| \mathbf{t}_{\mathrm{T}}^{n+1(i)} \right\|} \right] .} \tag{2.169}$$

Linearization of the tangential components of the contact traction vector, $D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right]$, can be found in Appendix C. To the determination of the time increment of the total plastic slip, $\Delta\lambda^{n+1}$, is dedicated Section 2.10.6, where the method of radial return is described.

2.10 Numerical procedures

In the last section of the state-of-the-art, miscellaneous topics regarding numerical implementation are discussed. The evaluation of the normal gap function (2.21) is the subject of the contact detection. Because it leads to the problem of solving the system of non-linear algebraic equations, great attention is paid to numerical methods for this kind of problem. Also the finite element discretization of the contact initial-boundary value problem leads to the non-linear system of equation, which is discussed in Section 2.10.4. The problem of poor conditioning due to the relatively large value of the penalty function is attacked by the Uzawa iteration as a practical implementation of the Augmented Lagrangian method in Section 2.10.5. The key ingredients of the numerical implementation of Coulomb's friction model is the evaluation of the "plastic" slip rate (2.66) resp. its time difference (2.167). For this purpose, the radial return algorithm is presented in Section 2.10.6. The last two topics that are mentioned in this section are those of contact smoothing in Section 2.10.7 and mass lumping and scaling in Section 2.10.8.

2.10.1 Contact detection

By the contact detection it is understood the evaluation of the normal gap function (2.21) in contact points (nodes or quadrature points) of the potential contact interface $\Gamma_{\mathrm{c}}^{(i)}$. The importance of this procedure results from the fact that this evaluation must be performed in each iteration of the non-linear solver. Computational time of this procedure determines the resulting time demands of the whole contact algorithm. Indeed, the simplest contact detection procedure consists in solving (2.8) by a non-linear solver, such as the Newton-Raphson method, in the loop over slave contact points and master contact segments.

It is obvious that searching for the closest point in a cycle over all contact points is not optimal, because if the bodies are not in contact, the exact value of the gap function is not necessary. Instead, only logical value is sufficient to indicate that no contact occur. Therefore, the contact detection is usually divided into two phases: a global search, which consists of effectively detecting and sorting all potential candidate slave points and their corresponding candidate master segments, and a local search for closest point projections of slave points onto master segments.

Global contact searching

The simplest and least effective global contact searching procedure is the so-called brute force method, when the search is carried out as all-to-all. Obviously there are more effective methods. A frequently used method is the so-called bucket-sorting [88, p. 145–149]. This algorithm performs sorting and searching in 3D by nested loops. Later, Oldenburg and Nilsson [89, p. 368–373] reduced the 3D to only 1D sorting and searching algorithm, called the position code. The complexity of this algorithm is proportional to $\mathcal{O}(N \log_2 N)$, where N is the number of contacting point. A more efficient algorithm with complexity $\mathcal{O}(N)$ is the no binary search (NBS) algorithm, that was proposed in [90, p. 134–142]. Its weakness is that it can only be used for bodies of a spherical shape and a similar size. A general algorithm with complexity $\mathcal{O}(N)$ was proposed in [91, p. 375–378]. This algorithm exploits so-called linked lists, which is a data structure that is commonly used in computer programming for sorting and searching. Recently, the idea of global contact searching based on linked lists was exploited in the algorithm called LC-Grid [92] ("L" after Lei, who devised the algorithm and "C" after Chen, who implemented the algorithm). All contact points and segments are gradually mapped onto layers, rows and cells, while for each of this level the linked-list is constructed. In the context of mortar method, the contact searching algorithm based on the bounding volume hierarchies (BVHs) was proposed in [93]. In this algorithm, master and slave contact interfaces are bounded by the k faced discrete orientation polytopes (k-DOPs) organizing in the tree structure which supports fast browsing throughout the structure.

Local contact searching

When the global search is successfully done, a local one is performed. The most frequently used approach for the local contact search is described in References [23, 88]. A contact point is defined on the master segment as the point closest to the slave point. Its parametric coordinates are calculated by solving the minimization problem. Recently, the uniqueness and existence of the closest point projection were investigated for C^0, C^1 and C^2 continuity of the target segment in [94]. An analytical solution exists only for linear triangular segments, but a bilinear quadrilateral, as well as higher order, elements have to be treated numerically. The Newton-Raphson method had usually been applied to this end.

In order to increase efficiency, in Reference [95] each quadrilateral segment was broken into four triangles with their common vertex at the centre of the master segment. The contact point was calculated by determining which triangle was closest to the slave node and then projecting the slave node onto it. Although the efficiency of this approach is undeniable, it is not adequate for distorted elements, a situation which often occurs in post-buckling computations [88].

For a node-to-segment contact design, an algorithm based on the definition of a meshed-surface normal is proposed in Reference [96]. The mesh normal vector assigned to a node is defined as the average normal vector taken over all surfaces adjacent to the node. The inside-outside algorithm [97] employs this vector to determine whether the projection of a node is located inside or outside the surface. Since no iterations are involved, this algorithm is very fast. However, a lack of continuity on the boundary of surfaces, called the deadzone problem, was observed. In order to overcome such a drawback, the free-form-surface (FFS) algorithm is proposed in Reference [98]. The contact area was approximated by the FFS patch, which ensures smoothness. Accuracy can be increased by subdividing the surface patches.

The closest point projection of a point to the FFS design also poses a common problem to computer graphics. Computational methods for the orthogonal projection of points in CAD/CAM applications have recently been presented in a review paper [99]. In order to guarantee robustness of the calculation, it is recommended to initially employ a subdivision based global scheme, followed by a Newton-type iteration method or the second order algorithms with small sensitivity to the choice of initial estimation such as the sphere approximation method [100] and torus approximation method [101].

The subdivision based global scheme is equivalent to global contact search. One of the first subdivision schemes is presented in Reference [102]. The essence of this technique is similar to the mentioned contact search described in Reference [95]. The so-called base surface is subdivided into quadrilaterals and a point is projected onto the closest quadrilateral. The parametric coordinates of the projection are recovered from the parametric coordinates of the corner point of the closest quadrilateral. Similar concept was proposed in Reference [103] where the base surface is recursively subdivided into Bézier surfaces and their control polygons are utilized to find the projection. An improved version of the method [103] was proposed in Reference [104], where a large number of subdivided parts are excluded from the computation by introducing new elimination criteria, namely the endpoint interpolation, convex hull property and tangent cones. The circular clipping algorithm was suggested in Reference [105], which is more efficient than those cited in References [103, 104]. A sphere is used as an elimination region. After subdivision the surface segments located outside the sphere are repeatedly eliminated until the termination condition is satisfied. The concept of the clipping sphere method was further improved in Reference [106] by using the separating axis and k-DOP (Discrete Orientation Polytopes) type of bounding scheme.

As was mentioned, the advantageous properties of spline representation (Bezier, B-spline, NURBS, etc.) are utilized in the subdivision phase and can be directly applied to modern isogeometric contact analysis [107]. Unfortunately, these properties are not shared by the standard higher-order finite elements which are the primary subject matter of this paper. In any case, the solution to the problem of orthogonal projection remains open since the present solution methods are prone to instability and their performance should also be improved [99].

In sheet forming simulations, a special local contact search procedure was developed to detect the shared position between the tool surface and the sheet [108]. The projected point onto the tool surface along the sheet normal direction was found iteratively, using the projection along the z-direction.

The mortar method has gained substantial popularity in recent years [75, 85, 37, 38, 74], to name some references. A continuous normal field on the slave surface was used to ensure projections. It calculates the intersection of the master segment with a line emanating from the slave point along the vector normal to the slave segment [75].

The closest point projection problem stated in Section (2.1) results is a non-linear algebraic system (2.12). In what follows, various optimization methods for solving this system will be described. Although, a solution, $\bar{\boldsymbol{\xi}}$, may lie outside the feasible region, i.e., outside the element segment, only unconstrained optimization methods are considered. At the same time, these methods are expected to converge to the closest local minimum. Nevertheless, even if the closest local minimum lies outside the feasible region, the current segment is then rejected and a new projection is sought in the next adjacent element segment. It should be noted that another approach formulating an optimization problem with inequality constraints can be found in Reference [109].

Let us introduce some additional notation. Since Equation (2.12) is solved iteratively, the k-th approximation to the solution will be denoted by the iteration counter k, which should not be confused with the body index (k). In particular, the gradient of the squared distance function (2.7) is now given by

$$\left.\frac{\partial d(\boldsymbol{\xi})}{\partial \xi^{\alpha}}\right|_{\boldsymbol{\xi}_k} = \left(\mathbf{x}^{(i)} - \left.\mathbf{x}^{(k)}(\boldsymbol{\xi})\right|_{\boldsymbol{\xi}_k}\right) \cdot \left.\frac{\partial \mathbf{x}^{(k)}(\boldsymbol{\xi})}{\partial \xi^{\alpha}}\right|_{\boldsymbol{\xi}_k} . \tag{2.170}$$

In what follows, an abbreviated notation will be introduced: $d_k = d(\boldsymbol{\xi})|_{\boldsymbol{\xi}_k}$, $\mathbf{x}_k^{(k)}(\boldsymbol{\xi})\Big|_{\boldsymbol{\xi}_k}$.

2.10.2 Line-search technique

In this section, several numerical methods for solving the system of non-linear algebraic equation, (2.170), will be presented. All of these methods fall into the category of so-called line-search methods. Each iteration of a line-search method computes a search direction $\mathbf{p}_k \in \mathbb{R}^{n_{\text{pd}}}$ and then decides how far to move along that direction. The iteration is given by

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + \theta_k \mathbf{p}_k, \tag{2.171}$$

where the positive scalar $\theta_k \in \mathbb{R}^+$ is called the step-length. The success of a line-search method depends on the effective choices of both the direction $\mathbf{p}_k$ and the step-length parameter θ_k. Most line-search algorithms require $\mathbf{p}_k$ to be a descent direction for which $p^k_\alpha \frac{\partial d_k}{\partial \xi^\alpha} < 0$, where the iteration index k was moved to the upper index position of $\mathbf{p}$ without changing the meaning. The search direction often has the form

$$p^k_\alpha = -D^k_{\alpha\beta} \frac{\partial d_k}{\partial \xi^\beta}, \tag{2.172}$$

where $\mathbf{D}_k \in \mathbb{R}^{n_{\text{pd}} \times n_{\text{pd}}}$ is a suitable matrix. Let us consider that $\mathbf{D_k}$ is positive definite. From multiplication of (2.172) by $\frac{\partial d_k}{\partial \xi^\alpha}$ arises

$$p^k_\alpha \frac{\partial d_k}{\partial \xi^\alpha} = -\frac{\partial d_k}{\partial \xi^\alpha} D^k_{\alpha\beta} \frac{\partial d_k}{\partial \xi^\beta} < 0. \tag{2.173}$$

Thus, the positive definiteness of $\mathbf{D}_k$ guarantees a descent direction of $\mathbf{p}_k$. How to compute the matrix $\mathbf{D}_k$ will be discussed in consequence sections. We now give attention to the choice of the step-length parameter θ_k. Its computation is based on the restriction of the minimized function $d(\boldsymbol{\xi})$ to the ray from a point $\boldsymbol{\xi}_k$ in the search direction $\mathbf{p}_k$

$$f(\theta) = d(\boldsymbol{\xi})|_{\boldsymbol{\xi}_j + \theta \mathbf{p}_j}, \quad \theta > 0. \tag{2.174}$$

Apparently, the exact minimization of this function is computationally expensive. To find even a local minimizer of $f(\theta)$ generally requires too many evaluations of the minimized function $d(\boldsymbol{\xi})$. In Reference [33, p. 55], more sophisticated strategies are mentioned to perform an inexact line-search to identify a step-length that achieves reductions in $d(\boldsymbol{\xi})$.

Strong Wolfe Conditions

A suitable step-length θ_k should first of all give sufficient decrease in the minimized function $d(\boldsymbol{\xi})$. Therefore, we insist that the value of $f_k = f(\theta)|_{\theta_k}$ in the candidate for θ_k is less than the value of a linear function. Such an inequality is known as the Armijo condition or *the sufficient decrease condition* that has a form

$$f_k \leq d_k + c_1 \theta_k \frac{\partial d_k}{\partial \xi^\alpha} p^k_\alpha, \quad c_1 \in (0,1). \tag{2.175}$$

The parameter c_1 sets the slope of the linear function (see Figure 2.11). In practice, c_1 is chosen to be quite small. According to [33, p. 38], $c_1 = 10^{-4}$. In Figure 2.11, there are two intervals, denoted by AC, which fulfill the Armijo condition (2.175).

The sufficient decrease condition is not enough to ensure that the algorithm makes reasonable progress. It is satisfied for all sufficiently small values of θ as can be seen from Figure 2.11. Therefore, a second requirement that is called *the curvature condition*, is introduced

$$\left| \left.\frac{\partial d(\boldsymbol{\xi})}{\partial \xi^\alpha}\right|_{\boldsymbol{\xi}_k + \theta_k \mathbf{p}_k} p^k_\alpha \right| \leq c_2 \left| \left.\frac{\partial d(\boldsymbol{\xi})}{\partial \xi^\alpha}\right|_{\boldsymbol{\xi}_k} p^k_\alpha \right|, \quad c_2 \in (c_1, 1). \tag{2.176}$$

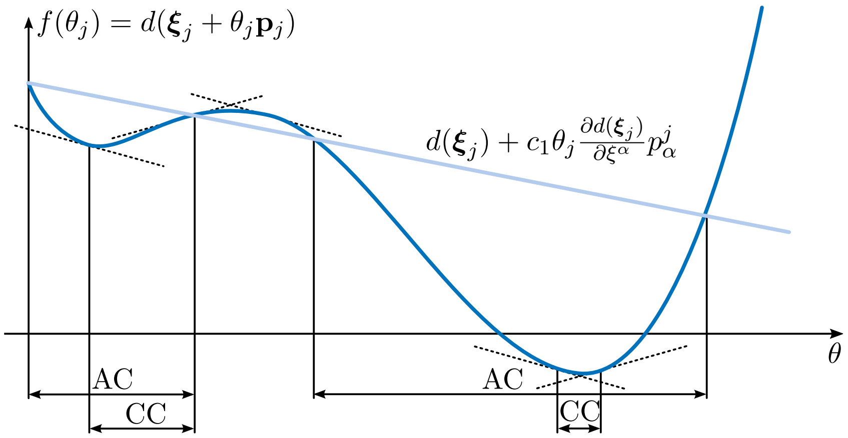


Figure 2.11: The Wolfe condition (AC – Armijo condition, CC – curvature condition).

It is based on the fact, that the gradient of function is close to zero in a neighborhood of a local extremum. Thus, the curvature condition enforces only a slight slope of $f(\theta)$ in the candidate for θ_k. The acceptable slope is set by the parameter c_2. According to [33, p. 39], typical values are 0.9 when the search direction $\mathbf{p}_k$ is computed by the Newton or the quasi-Newton method, and 0.1 when $\mathbf{p}_k$ is obtained from the gradient methods.

In Figure 2.11, the intervals denoted by CC fulfill the curvature condition (2.176). The Armijo condition (2.175) and the curvature condition (2.176) are known as *the strong Wolfe conditions*.

Step length

The determination of the parameter θ_k is based on the restriction of the minimized function in the search direction. Using too accurate line search can substantially increase the total cost of computation. Therefore, the practical strategy consists of adequate reduction of the minimized function at the minimum cost. The derivative-free procedures (e.g. bisection method, golden section, or Fibonacci search) are the simple step-length techniques. A popular method in FE applications is the Illinois algorithm [110]. An overview of sophisticated step-length procedures based on quadratic or cubic interpolations is presented in Reference [33, p. 56].

A particularly effective step-length procedure is based on the interpolation of the known function $f(\theta)$ and their derivations. The procedure generates a decreasing sequence of values θ_ℓ for each of the k-th iteration of the nonlinear solver. Here, the subscript ℓ denotes the iteration counter of the step-length procedure. The initial guess θ_ℓ is simply set to 1. Then the strong Wolfe conditions (2.175), (2.176) are checked. If the conditions are satisfied for this step-length, the procedure is terminated. Otherwise, we know that the interval $[0, \theta_0]$ contains acceptable step-lengths (see Figure 2.11). A quadratic approximation $f_\mathrm{q}(\theta)$ to function $f(\theta)$ can be formed as

$$f_\mathrm{q}(\theta) = \left(\frac{f|_{\theta_0} - \theta_0 \, f'|_0 - f|_0}{\theta_0^2} \right) \theta^2 + f'|_0 \, \theta + f|_0 \, . \tag{2.177}$$

The value of θ_1 is defined as the minimizer of this quadratic function, that is

$$\theta_1 = - \frac{f'|_0 \, \theta_0^2}{2 \left(f|_{\theta_0} - f|_0 - \theta_0 \, f'|_0 \right)} . \tag{2.178}$$

If the strong Wolfe conditions(2.175), (2.176) are satisfied for θ_1, the step-length procedure is terminated. Otherwise, a cubic function is used to interpolate $f|_{\theta_{\ell-1}}$, $f'|_{\theta_{\ell-1}}$, $f|_{\theta_\ell}$, $f'|_{\theta_\ell}$.
In Reference [33, p. 57] the minimizer of the cubic function is given by

$$\theta_{\ell+1} = \theta_\ell - (\theta_\ell - \theta_{\ell-1}) \left(\frac{f'|_{\theta_\ell} + d_2 - d_1}{f'|_{\theta_\ell} - f'|_{\theta_{\ell-1}} + 2d_2} \right), \tag{2.179}$$

where

$$\begin{aligned} d_1 &= f'|_{\theta_{\ell-1}} + f'|_{\theta_\ell} - 3\frac{f'|_{\theta_{\ell-1}} - f'|_{\theta_\ell}}{\theta_{\ell-1} - \theta_\ell}, \\ d_2 &= \sqrt{d_1^2 - f'|_{\theta_{\ell-1}}\, f'|_{\theta_\ell}}. \end{aligned}$$

If the strong Wolfe conditions (2.175), (2.176) are satisfied at $\theta_{\ell+1}$, the step-length procedure is terminated. Otherwise, the interpolation process is repeated by discarding the data at one of the step-length and replacing it by $f|_{\theta_{\ell+1}}$ and $f'|_{\theta_{\ell+1}}$. Then, the repetition of the interpolation process continues until the strong Wolfe conditions are fulfilled.

Gradient methods

The only gradient method which will be dealt with in this section is probably the most famous method in this class — the method of steepest descent.

The method of steepest descent. The steepest descent method, also known as the gradient descent method, is the simplest minimization technique. It is based on the observation that a differentiable function in the neighborhood of a point decreases fastest in the direction of the negative gradient. Thus, one step of this method takes the form

$$\xi^\alpha_{k+1} = \xi^\alpha_k - \theta_k \frac{\partial d_k}{\partial \xi^\alpha} \tag{2.180}$$

where θ_k is the step length parameter. Its proper choice is the objective of the so-called step-length algorithm which was described in the preceding section.

Newton and quasi-Newton methods

Another class of line-search methods consists of the Newton and quasi-Newton methods. What is characteristic of them is that, unlike gradient methods, they use not only the first but also the second derivatives of the minimized function or, in the case of quasi-Newton method, their approximation.

The Newton-Raphson method. Linearizing the residual (2.8) about the k-th approximation gives rise to a popular numerical scheme, the Newton-Raphson method

$$\xi^\alpha_{k+1} = \xi^\alpha_k - \left[\mathbf{H}_k^{-1}\right]_{\alpha\beta} \frac{\partial d_k}{\partial \xi^\beta} \tag{2.181}$$

where $[\mathbf{H}_k]_{\alpha\beta} = \frac{\partial d_k}{\partial \xi^\alpha \xi^\beta}$, $\mathbf{H}_k \in R^{n_{\mathrm{pd}} \times n_{\mathrm{pd}}}$, is the Hessian matrix. A sufficient condition for convergence of the Newton-Raphson method to the nearest local minimum is positive definiteness of the Hessian matrix. The proof of this assertion can be found in [33, p. 138].

It should be noted that even when the Hessian matrix is not positive definite, it is still possible to ensure the Newton-Raphson method convergent. Various techniques exist based on a modification of the Hessian matrix [33, p. 141] or combining it with another gradient method (e.g. the steepest descent method).

The Broyden method. Just as the Newton-Raphson method is a generalization of Newton's procedure to multiple dimensions, Broyden's method is a generalization of the secant method [111]. It belongs to a class of algorithms known as the quasi-Newton methods, which construct an approximation of the Hessian matrix instead of its exact evaluation. The iterative scheme has the form

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k - \theta_k \left[\mathbf{D}_k^{-1}\right]_{\alpha\beta} \frac{\partial d_k}{\partial \xi^\beta} \tag{2.182}$$

where $\mathbf{D}_k \in \mathbb{R}^{n_{\mathrm{pd}} \times n_{\mathrm{pd}}}$ is the representation of the secant matrix. According to Broyden, the update of $\mathbf{D}_k$ is computed as

$$\mathbf{D}_{k+1} = \mathbf{D}_k + \frac{\mathbf{r}_k - \mathbf{D}_k \mathbf{s}_k}{\mathbf{s}_k \cdot \mathbf{s}_k} \otimes \mathbf{s}_k \tag{2.183}$$

in which the auxiliary vectors $r_\alpha^k = \frac{\partial d_{k+1}}{\partial \xi^\alpha} - \frac{\partial d_k}{\partial \xi^\alpha}$, $\mathbf{r}_k \in \mathbb{R}^{n_{\mathrm{pd}}}$ and $s_k^\alpha = \xi_{k+1}^\alpha - \xi_k^\alpha$, $\mathbf{s}_k \in \mathbb{R}^{n_{\mathrm{pd}}}$ have been introduced. The initial secant matrix $\mathbf{D}_0$ is usually set to the identity matrix or its multiple. The method's convergence may again be accelerated by the line search algorithm as indicated in Equation (2.182) by the step-length parameter θ_k.

The BFGS method. The most popular quasi-Newton method for the finite element applications is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [111]. Since the secant matrix $\mathbf{D}_k$ is constructed as the positive definite matrix, the search direction has always a descent character. The iterative scheme is identical to Equation (2.182) but the update of $\mathbf{D}_k$ is computed as

$$\mathbf{D}_{k+1} = \mathbf{D}_k + \frac{\mathbf{r}_k \otimes \mathbf{r}_k}{\mathbf{s}_k \cdot \mathbf{r}_k} - \frac{(\mathbf{D}_k \mathbf{r}_k) \otimes (\mathbf{D}_k \mathbf{r}_k)}{\mathbf{r}_k \cdot (\mathbf{D}_k \mathbf{r}_k)} \tag{2.184}$$

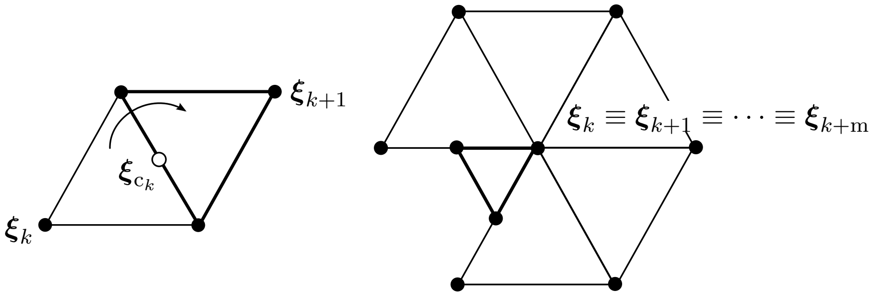
Just as with Broyden's method, the initial secant matrix $\mathbf{D}_0$ is set to identity matrix or its multiple. The BFGS method may also be accelerated by the line search method.

2.10.3 Gradient-free methods

As the gradient-free methods are known those methods which do not require calculation of derivatives, but they work only with function values. The so-called simplex method will be presented as a representative of this class of methods.

The simplex method. The simplex (polytope) method is a robust heuristic unconstrained optimization method. In contrast to previous methods, which operate with the derivatives of the squared distance function, the simplex method utilizes only its function values. The idea of the method was originally proposed in [112]. This should not be confused with another well-known method — Dantzig's simplex algorithm for linear programming [33, p. 361]. In this Section the simplex method will be briefly outlined.

In each iteration, the values of the squared distance function are evaluated at three vertices of a regular polytope (i.e. simplex). A new position of the simplex is obtained by the mirroring of the vertex with the highest function value over the centre of the remaining vertices – see Figure 2.12a. Furthermore, there are two additional rules, which make the algorithm robust. The first rule is applied when the value in the new vertex is the maximum again. It is not allowed to return a vertex back in the subsequent iteration in order to prevent a runaway loop. Instead, the vertex with the second highest function value is moved. The second rule handles the case when one of the vertices remains in position for $m \in \mathbb{N}$ iterations. This situation indicates that the simplex rotates about a local extremum. Then, the length of the simplex edge is halved. The parameter m depends on the problem dimension. In our two-dimensional case $m = 5$ as is apparent in Figure 2.12b.



(a) : 1st criterion **(b) :** 2nd criterion

Figure 2.12: The basic rules of the simplex method.

2.10.4 Solution of the non-linear finite element equations

After formulating the contact initial-coundary value problem in the strong and weak sense, regularization of the contact constraints by the penalty method, linearization, spatial discretization by the finite element method, and temporal discretization, the final step is the solution of the resulting system of non-linear equations. Although already mentioned in the connection with local contact searching, the Newton-Raphson and the BFGS methods are presented here in the context of finite element equations.

Newton-Raphson method

Newton-Raphson method is one of the effective techniques for solving systems of non-linear algebraic equations. Its principle consists in converting the system of equations into the residual form

$$\mathbf{R}(\mathbf{d}) = \mathbf{0}, \tag{2.185}$$

and the subsequent linearization at a point $\mathbf{d}_k$

$$\mathbf{R}|_{\mathbf{d}_k} + \left.\frac{\partial \mathbf{R}}{\partial \mathbf{d}}\right|_{\mathbf{d}_k} \Delta \mathbf{d}_k = \mathbf{0} \tag{2.186}$$

where k indicates iteration counter. For the increment $\Delta \mathbf{d}_k$ one gets from the last equation

$$\Delta\mathbf{d}_k = \left[\left.\frac{\partial \mathbf{R}}{\partial \mathbf{d}}\right|_{\mathbf{d}_k} \right]^{-1} \left(- \mathbf{R}|_{\mathbf{d}_k} \right). \tag{2.187}$$

Then a new estimate of the solution can be calculated from the increment $\Delta\mathbf{d}_k$ as

$$\mathbf{d}_{k+1} = \mathbf{d}_k + \Delta\mathbf{d}_k. \tag{2.188}$$

The whole process is repeated until $\mathbf{R}|_{\mathbf{d}_{k+1}}$ is smaller than the selected tolerance. It can be shown that the Newton-Raphson method converges quadratically to the nearest root of the non-linear system, provided the tangent matrix is positive definite [33, p. 138]. The Newton-Raphson method is used in every time step of the implicit integration scheme for the solution of (2.161), as was mentioned in Section 2.9.1. If the inertial forces in (2.161) can be neglected, the residual vector for quasi-static problems is obtained

$$\mathbf{R}(\mathbf{d}) = \mathbf{F}_{\mathrm{int}}(\mathbf{d}) + \mathbf{F}_{\mathrm{c}}(\mathbf{d}) - \mathbf{F}_{\mathrm{ext}} = \mathbf{0}. \tag{2.189}$$

Otherwise, the non-linear system of equation (2.161) is solved by the Newton-Raphson method for each time step of the temporal integration scheme.

Broyden-Fletcher-Goldfarb-Shenno method

The Broyden-Fletcher-Goldfarb-Shenno (BFGS) method belongs to the class unconstrained optimization methods but in principle it can be utilized also for solving systems of non-linear algebraic equations. To show this, recall the necessary condition for a local extremum of a function $G(\mathbf{d}) : \mathbb{R}^{n_{\mathrm{dof}}} \rightarrow \mathbb{R}$ which demands that the gradient of the function has to be equal to zero vector

$$\left.\frac{\partial G(\mathbf{d})}{\partial \mathbf{d}}\right|_{\mathbf{d}=\mathbf{d}_\infty} = \mathbf{0}. \tag{2.190}$$

If the Newton-Raphson method is utilized to find the extremal point $\mathbf{d}_\infty$

$$\frac{\partial}{\partial \mathbf{d}} \left(\frac{\partial G_k}{\partial \mathbf{d}} \right) \Delta\mathbf{d}_k = -\frac{\partial G_k}{\partial \mathbf{d}}, \tag{2.191}$$

$$\mathbf{K}_k \Delta\mathbf{d}_k = -\mathbf{R}_k, \tag{2.192}$$

where $G_k = G(\mathbf{d})|_{\mathbf{d}_k}$. Now, one can immediately see the connection between the problem of finding an extreme value of the function $G(\mathbf{d})$ and solving the system of non-linear equations $\mathbf{R}(\mathbf{d}) = \mathbf{0}$. The gradient $\frac{\partial G_k}{\partial \mathbf{d}}$ can be in fact viewed as the residual vector $\mathbf{R}$ and the Hessian matrix $\frac{\partial}{\partial \mathbf{d}} \left(\frac{\partial G_k}{\partial \mathbf{d}} \right)$, i.e. the matrix of second partial derivatives, can be understood as the tangent matrix $\mathbf{K} \in \mathbb{R}^{n_{\mathrm{dof}} \times n_{\mathrm{dof}}}$. Therefore, the BFGS method can be employed also for solving systems of non-linear algebraic equations. The BFGS method belongs to the class of quasi-Newton methods, which instead of the exact Hessian/tangent matrix $\mathbf{K}_k$ work with a suitable approximation — a secant matrix $\mathbf{D}_k \in \mathbb{R}^{n_{\mathrm{dof}} \times n_{\mathrm{dof}}}$. The advantage of this method is that it is not necessary to perform the exact linearization and to solve the system of linear algebraic equations in each iteration, as it is the case for the Newton-Raphson method. The price we pay for it is "only" superlinear rate of convergence [33] of the BFGS method.

The initial estimate of the secant matrix $\mathbf{D}_0$ can be chosen as a multiple of the identity matrix, $\mathbf{D}_0 = \mathbf{I}_{n_\text{dof}}$ ($\mathbf{I}_{n_\text{dof}} \in \mathbb{R}^{n_\text{dof} \times n_\text{dof}}$ is the n_dof-dimensional identity matrix), or as the exact tangent matrix $\mathbf{D}_0 = \mathbf{K}|_{\mathbf{d}_0}$. The update of secant matrix is calculated from the relation

$$\mathbf{D}_{k+1} = \mathbf{D}_k + \frac{\Delta\mathbf{R}_k \otimes \Delta\mathbf{R}_k}{\Delta\mathbf{d}_k \otimes \Delta\mathbf{d}_k} - \frac{(\mathbf{D}_k \Delta\mathbf{R}_k) \otimes (\mathbf{D}_k \Delta\mathbf{R}_k)}{\Delta\mathbf{R}_k \cdot (\mathbf{D}_k \Delta\mathbf{R}_k)}, \tag{2.193}$$

where $\Delta\mathbf{R}_k = \mathbf{R}_{k+1} - \mathbf{R}_k$. An implementation according to Matthies and Strang [113] is often used in application to FEM. The utilization of BFGS method for solving the contact problem, as a constrained optimization problem, can be found in [1, 114]. In practice, it appears that due to discontinuous non-linearities, which are characteristic for contact problems, the secant matrix $\mathbf{D}_k$ can become ill-conditioned. In such a case it is recommended to perform the recovery of the secant matrix using the exact tangent matrix $\mathbf{D}_k = \mathbf{K}|_{\mathbf{d}_k}$ [114].

2.10.5 Uzawa iteration method

The augmented Lagrangian method, as well as the Lagrange multiplier method, increases the number of unknowns. This deficiency elegantly removes the Uzawa algorithm. The principle is demonstrated in the calculation of the contact traction vector in time t^{n+1}, where we assume that the solution in time t^n is known. Initially, Lagrange's multiplier estimates are set equal to the values calculated in the previous time step

$$\lambda_{\text{N}\ell}^{n+1} = \lambda_{\text{N}_n}^{n}, \tag{2.194}$$

$$\lambda_{\text{T}\alpha\ell}^{n+1} = \lambda_{\text{T}\alpha}^{n}, \tag{2.195}$$

and the iteration index ℓ is set equal to zero. Then the system of non-linear equations (2.161) where for the contact traction vector $\mathbf{F}_\text{c}$

$$t_\text{N}^{n+1} = \left\langle \lambda_{\text{N}\ell}^{n+1} + \varepsilon_\text{N} g_\text{N}^{(i)n+1} \right\rangle, \tag{2.196}$$

$$t_{\text{T}\alpha}^{n+1} = \lambda_{\text{T}\alpha\ell}^{n+1} + \varepsilon_\text{T} \left[\bar{m}_{\alpha\beta}^{n} \left(\bar{\xi}_\ell^{n+1\beta} - \bar{\xi}_\ell^{n\beta} \right) - \Delta\lambda \frac{t_{\text{T}_\alpha}^{n+1}}{\left\| \mathbf{t}_\text{T}^{n+1} \right\|} \right], \tag{2.197}$$

$$\Delta\lambda \geq 0, \qquad \Phi(t_\text{N}^{n+1}, \mathbf{t}_\text{T}^{n+1}) \leq 0, \qquad \Delta\lambda \Phi(t_\text{N}^{n+1}, \mathbf{t}_\text{T}^{n+1}) = 0, \tag{2.198}$$

The solution is made by radial return algorithm which will be presented in the following subsection. The new approximation of Lagrange multipliers is

$$\lambda_{\text{N}\ell+1}^{n+1} = t_\text{N}^{n+1}, \tag{2.199}$$

$$\lambda_{\text{T}\alpha\ell+1}^{n+1} = t_{\text{T}\alpha}^{n+1}, \tag{2.200}$$

If the difference for iterations ℓ and $\ell+1$ is less than a set tolerance, the calculation is terminated. Otherwise, the iteration index is increased by one and the calculation is repeated. The robustness of this method is compensated by a higher number of iterations than in the case of penalty method or the method of Lagrange multipliers.

2.10.6 Radial return method

The radial return method was originally designed for integration of plasticity constitutive models [115]. The first use of this method for integrating of the constitutive relation for friction can be found in [116, p. 159]. Algorithm 2.1 shows the application of the radial return method on solving discretized evolutionary equation (2.169) to determine the regularized tangential components of the contact traction vector at time t_{n+1}.

Algorithm 2.1 The algorithm of the radial return method.

```
procedure RadialReturn
    $t^{n+1}_{\mathrm{T}\alpha_{\mathrm{trial}}} \leftarrow t^{n}_{\mathrm{T}\alpha} - \epsilon_{\mathrm{T}} \left( \bar{\xi}^{n+1\alpha} - \bar{\xi}^{n\alpha} \right) \bar{m}^{n}_{\alpha\beta}$        ▷ (2.169) for $\Delta\lambda^{n+1} \leftarrow 0$
    $\Phi^{n+1}_{\mathrm{trial}} \leftarrow \left\| \mathbf{t}^{n+1}_{\mathrm{T\,trial}} \right\| - \mu p^{n+1}_{\mathrm{c}}$
    if $\Phi^{n+1}_{\mathrm{trial}} \leq 0$ then
        $\mathbf{t}^{n+1}_{\mathrm{T}} \leftarrow \mathbf{t}^{n+1}_{\mathrm{Ttrial}}$
    else
        $\mathbf{t}^{n+1}_{\mathrm{T}} \leftarrow \mu p^{n+1}_{\mathrm{c}} \frac{\mathbf{t}^{n+1}_{\mathrm{Ttrial}}}{\left\| \mathbf{t}^{n+1}_{\mathrm{Ttrial}} \right\|}$        ▷ (2.169) for $\Delta\lambda^{n+1} \leftarrow \frac{\Phi^{n+1}_{\mathrm{trial}}}{\epsilon_{\mathrm{T}}}$
    end if
end procedure
```

2.10.7 Contact smoothing techniques

Non-smoothness is another difficulty encountered in computational contact analysis which has to be treated. It naturally arises from inequality constraints as well as the geometric discontinuities inducted by spatial discretization. Contact analysis based on traditional finite elements utilizes element facets to describe a contact surface. The facets are C^0-continuous so that surface normals can experience jump across facets boundaries leading to artificial oscillations in normal and tangent contact forces. A global non-linear equation solving scheme is also affected by non-smoothness. It was pointed out in [117] that the Newton-Raphson method admits, in general, only superlinear convergence even for C^2-continuous contact boundary, unless Lipschitz continuity is enforced.

Non-smoothness is usually attacked by contact smoothing techniques, where contact interface is approximated by various species of splines. In 2D, first attempts applied Hermite cubic interpolation [118, 119, 120]. C^1-continuous Overhauser (Catmull-Rom) splines were used in [121], or non-uniform rational B-splines (NURBS) were utilized in [122]. Also in 3D, a lots of different options of contact surface smoothing were presented in the literature: local diffuse approximation with quadratic basis [123], cubic B-Splines [124], Bézier patch [125, 126], NURBS [127], Gregory patches [128, 126], or recently Nagata patches [129]. All these contact smoothing techniques concentrate on the better approximation of the contact boundary. A different approach was proposed in [130], where rather then the contact boundary, the signed distance function was smoothed by a moving least-squares approximation with a polynomial basis.

Contact smoothing techniques introduce an additional geometry on the top of the existing finite element mesh. This adds an extra layer of data management and increases computational overhead. A hierarchical enrichment [131], on the other hand, enriches the shape function space of the contact layer finite element in a way that contact interface geometry is exactly represented but at the same time the bulk of the contacting bodies can be described by lower order shape function. 2D local enrichment with high-order Lagrange and Hermite interpolation was presented in [132], both, 2D and 3D NURBS enrichment was studied in [133], and 3D isogeometrically enriched finite elements were published in [134].

Isogeometric contact analysis

Another remedy to the geometric discontinuity provides isogeometric analysis (IGA) [49, 135]. The fundamental idea is to accurately describe a physical domain of interest by a proper mathematical representation (e.g. B-spline, NURBS, etc.) and then utilize the same basis for analysis. This is in contrast with the classical finite element method where the basis is given in advance by the element type and the geometry of physical domain is then only approximated with the aid of isogeometric mapping.

Isogeometric NURBS-based contact analysis has some additional advantages: preserving geometric continuity, facilitating patch-wise contact search, supporting a variationally consistent formulation, and having a uniform data structure for the contact surface and the underlying volumes.

Geometric basis and formulation for frictionless isogeometric contact were given in [136]. Sharp corners or C^0 edges that can exist on the interface of patches present a challenge to contact detection. A strategy to seamlessly deal with sharp corners was proposed in this reference. Herein, the contact constraints were regularized by penalty method and contact virtual work was discretized by finite strain segment-to-segment contact element. Both one-pass and two-pass algorithm were tested.

In [50], finite deformation frictionless quasi-static thermomechanical contact problems was considered. Two penalty-based contact algorithms were studied herein. The former was called knot-to-segment (KTS) algorithm. It is the straightforward extension of the classical node-to-segment (NTS) algorithm. Since NURBS control points are not interpolatory, contact constraints were enforce directly at the physical points of the quadrature points. It was shown in this reference that this approach is over-constrained and therefore not acceptable if a robust formulation with accurate tractions is desired. The latter was called mortar-KTS algorithm. In this algorithm a mortar projection to control pressures was employed to obtain the correct number of constraints.

The penalty-based mortar-KTS algorithm was extended to frictional contact in [51] and [137]. The mortar-KTS algorithm was also studied in conjugation with augmented Lagrangian method in [53]. As a viable alternative to the augmented Lagrangian approach the interior point method, also known as the barrier method, was applied to frictionless contact in [138]. Isogeometric frictionless contact analysis using non-conforming mortar method in two-dimensional linear elasticity regime was presented in [139].

An isogeometric point-to-segment contact formulation for 2D large elastic deformations was presented in [140]. To collocate the contact integrals it was proposed to use Greville and Botella points. Also two postprocess methods to obtain contact stress distribution from discrete values of the Lagrange multipliers at collocation points were investigated and compared. This concept was recently enhanced by weighting in [141].

Another isogeometric thermomechanical mortar contact algorithms was investigated in [142]. In comparison to [50] where mortar-based formulation was used, in this work a series of simplifications were considered so that an established numerical techniques for mortar methods such as segmentation could be transferred to IGA without modification. Recenlty, a dual mortar formulation for isogeometric analysis was derived and investigated in [143]. Finally, a review of isogeometric contact formulations was given in [107].

Isogeometric collocation methods (IGA-C) [144, 145] have been recently proposed as an alternative to standard Galerkin approaches. They provide a significant reduction in computational cost especially for higher-order discretizations. They can be interpreted as a one point quadrature rule in the IGA context. The imposition of Neumann boundary conditions and the enforcement of contact constraints in IGA-C context was studied in [146]. The application of IGA-C to large deformation frictional contact problems with consistent linearization was presented in [147].

It is well known that accurate evaluation of the contact stresses at the contact interface requires appropriate refinement. As the NURBS multivariate basis functions are defined as tensor product, the number of unknowns increases rapidly as the knot insertion requires control point addition along the whole parametric dimension. The local basis function refinement provides T-spline-based isogeometric analysis [148, 149]. It was first applied to frictionless large deformation contact problem in [150]. Here, the cubic T-splines and cubic NURBS were considered. A Gauss-point-to-surface formulation and the penalty method was used to treat the contact constraints. An alternative to T-splines is the NURBS-based hierarchical refinement scheme [151] applied to contact problems in [152] where large deformation frictionless contact was considered, or in [153] where a mortar contact algorithms applied to a phase-field approach to brittle fracture were studied.

2.10.8 Mass lumping and mass scaling techniques

The performance of the explicit integration method is commonly boosted by the lumping technique [12, 154, 155], i.e. diagonalization of the mass matrix. Diagonal mass matrix results in the orthogonal system of linear equation. Therefore, the mass lumping is the essence which enables the massive parallelization and the reduction of data storage requirements. Two most usual lumping methods are the row sum method [12] and the Hinton-Rock-Zienkiewicz (HRZ) method [156]. The accuracy of the lumping scheme has been recently assessed in Adam *et al.* [157]. A higher order lumping scheme was presented in [158].

Another approach how to improve the performance of the explicit temporal integration is a selective mass scaling (SMS) [159, 160, 161, 162, 163, 155, 154] . While the mass lumping aspires for faster integration via efficient solution of the linear system of equations, SMS trails the same aim through the enlargement of the critical time step. SMS techniques are based on local modification of the material properties, such as density, to increase the stable time step with only a little lose of accuracy. They are particularly useful in analyses which involve the solid shell elements [159, 163, 155] to prevent the time step from being controlled by the thickness. However, SMS usually results in a non-diagonal mass matrix whose inverse is more expensive in comparison with a lumped mass matrix. Therefore, Tkachuk and Bischoff [164] proposed a new SMS technique based on direct variational construction of a sparse inverse mass matrix. They also studied the problem of optimal SMS in [165].

Chapter 3

Aims of the Thesis

In this chapter, based on the state of the art presented in the previous chapter, the aims and objectives of this dissertation will be defined.

3.1 Aim

This work aims only one goal, namely: *To improve efficiency and robustness of the finite element contact algorithm for resolution of the general static and dynamic large deformation contact problems.*

3.2 Objectives

It is intended to achieve the stated aim through the following objectives:

1. *To develop a symmetry preserving isogeometric finite element contact formulation for static and dynamic large deformation problems.*

   Conventional contact formulations are based on the master-slave concept and explicit enforcement of the action-reaction principle (2.111) that introduce bias into the problem, making the solution dependent on the choice of the master and the slave body. We propose symmetry preserving contact formulation based on the contact residual in the form of (2.106).

   Contact analysis based on traditional finite elements utilizes element facets to describe a contact surface. The facets are only C^0-continuous so that surface normals can experience jump across facets boundaries leading to artificial oscillations in normal and tangent contact forces. As an alternative to conventional contact smoothing techniques discussed in Section 2.10.7, we propose to utilize the isogeometric analysis which naturally allows the higher continuity of contact boundaries.

2. *To propose a new local contact search technique based on the geometric iteration methods.*

   Today's implementations of the local contact search procedure most frequently employs the Newton-Raphson method which was presented in Section 2.10.2 . Although this method works passably for linear elements, it suffers from convergence difficulties for higher-order elements. This is due to the fact that the Hessian matrix of the square distance function (2.7) to be minimized is not positive definite in general. We propose to utilize the geometric iteration methods for the local contact searching.

3. *To extend the bipenalty method for the solution of the contact-impact problems.*

   The explicit temporal integration schemes are known to be conditionally stable. Furthermore, it is well known that the penalty method has a negative effect on the critical time step. To address this issue, we propose to utilize the bipenalty method. Moreover, a critical time step estimate for the penalty and bipenalty method will be derived.

Chapter 4

Applied methods

This chapter describes the tools necessary to meet the objectives defined in the previous chapter. The basic building block of the isogeometric analysis is the geometry representation by different types of splines. Therefore, the first section will briefly describe the B-spline and NURBS basis functions. Next, the idea of geometric iteration method will be presented in Section 4.2. The first order method — the least square projection method — and two second order methods — sphere and torus approximation methods — are described in detail. The last section is devoted to the bipenalty method.

4.1 Isogeometric analysis

Isogeometric analysis is a new spatial discretization method [49] which utilizes as basis functions various types of splines. The fundamental idea is to accurately describe a physical domain of interest by a proper mathematical representation (e.g. B-spline, NURBS, etc.) and then by means of the isoparametric concept [48] to utilize the same basis for analysis.

4.1.1 Non-Uniform Rational B-Splines

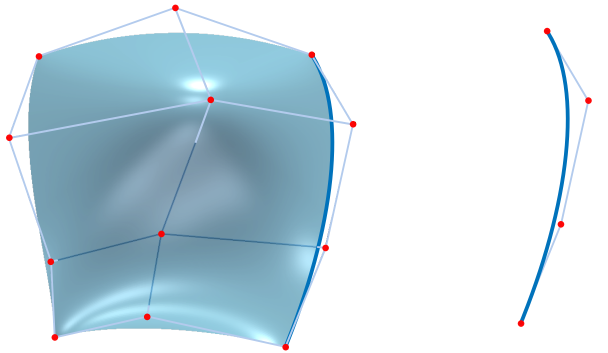
This section gives a brief overview of the main concerns of B-splines and NURBS. For more detailed description as well as efficient algorithms see, e.g. [166]. Throughout this section the polynomial degree will be indicated by p and the number of basis functions by n.

A B-spline or NURBS object is called patch. An example of a patch is depicted in Figure 4.1. The patch is parametrize by the linear combination of control points coordinates $\mathbf{P}_A \in \mathbb{R}^{n_{\mathrm{sd}}}$ and basis functions $N_A(\boldsymbol{\zeta}) : \mathbb{R}^{n_{\mathrm{pd}}} \mapsto [0, 1]$. A particular B-splines/NURBS basis function is defined with the aid of so called knot vector

$$\boldsymbol{\Xi}^i \quad := \quad \left\{ \underbrace{\zeta^i_1, \dots, \zeta^i_{p_i+1}}_{p_i+1 \text{ equal terms}}, \zeta^i_{p_i+2}, \dots, \zeta^i_{n_i}, \underbrace{\zeta^i_{n_i+1}, \dots, \zeta^i_{n_i+p_i+1}}_{p_i+1 \text{ equal terms}} \right\}, \quad i = 1, \dots n_{\mathrm{pd}}. \tag{4.1}$$

The knot vector is a non-decreasing sequence of parametric coordinates. The knot vector is said to be uniform if the knots are unequally spaced in the parametric space. If the first and the last knot value appears $p_i + 1$ times, the knot vector is called open. B-spline/NURBS object with open knot vectors are interpolatory at the corner of a patch. It means that the boundary of a

B-spline/NURBS object with n_{pd} parametric dimensions is itself a B-spline/NURBS object with $n_{\mathrm{pd}}-1$ dimensions.



(a) : NURBS surface **(b) :** NURBS curve

Figure 4.1: An example of NURBS surface and curve.

The B-spline basis functions are defined by the Cox-de Boor recursion formula. For $p=0$ it is

$$N_{j,0}(\zeta) = \begin{cases} 1 & \zeta \in [\zeta_j, \zeta_{j+1}) \,, j = 1 \dots n \\ 0 & \text{otherwise} \end{cases} , \tag{4.2}$$

and for $p>0$

$$N_{j,p}(\zeta) = \frac{\zeta - \zeta_j}{\zeta_{j+p} - \zeta_j} N_{j,p-1}(\zeta) + \frac{\zeta_{j+1+p} - \zeta}{\zeta_{j+1+p} - \zeta_{j+1}} N_{j+1,p-1}(\zeta). \tag{4.3}$$

B-splines are known to be unable to exactly describe some curves and surfaces. For instance circles or cylinders. NURBS was developed to extend interpolatory capability of the B-splines. While the B-spline basis functions are ordinary polynomials, the NURBS basis functions are rational polynomials. The extension originates from projection geometry of conic sections. More details can be found in [166]. The p-th degree NURBS basis function is defined as

$$R_j^p(\zeta) := \frac{N_{j,p}(\zeta) w_j}{\sum_{\hat{j}=1}^{n} N_{\hat{j},p}(\zeta) w_{\hat{j}}}, \tag{4.4}$$

where w_j is referred to as the j-th weight.

Multivariate NURBS objects can be constructed simply by tensor product of univariate NURBS basis functions (4.4). For $n_{\mathrm{pd}}=2$

$$R_{j_1,j_2}^{p_1,p_2}(\zeta^1,\zeta^2) \quad := \quad \frac{N_{j_1,p_1}(\zeta^1) N_{j_2,p_2}(\zeta^2) w_{j_1,j_2}}{\sum_{\hat{j}_1=1}^{n_1} \sum_{\hat{j}_2=1}^{n_2} N_{\hat{j}_1,p_1}(\zeta^1) N_{\hat{j}_2,p_2}(\zeta^2) w_{\hat{j}_1,\hat{j}_2}}, \tag{4.5}$$

and similarly for higher parametric dimensions.

With multivariate NURBS basis functions at hand we can introduce surface discretization by

$$\mathbf{x}(\zeta^1,\zeta^2) = \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} R_{j_1,j_2}^{p_1,p_2}(\zeta^1,\zeta^2) \mathbf{P}_{j_1,j_2}, \tag{4.6}$$

where $\mathbf{P}_{j_1,j_2} \in \mathbb{R}^{n_{\rm sd}}$ is the control net, i.e., array of coordinates of control points. Adopting the isogeometric concept, an analogous interpolation is used for unknown displacement field and its variation. Utilizing proper connectivity arrays one can write

$$\mathbf{x}(\boldsymbol{\zeta}) = \sum_{A=1}^{n_{\rm cp}} R_A(\boldsymbol{\zeta})\mathbf{x}_A, \tag{4.7}$$

$$\mathbf{u}(\boldsymbol{\zeta}) = \sum_{A=1}^{n_{\rm cp}} R_A(\boldsymbol{\zeta})\mathbf{d}_A, \tag{4.8}$$

$$\delta\mathbf{u}(\boldsymbol{\zeta}) = \sum_{A=1}^{n_{\rm cp}} R_A(\boldsymbol{\zeta})\mathbf{c}_A, \tag{4.9}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^{n_{\rm pd}}$ is the vector of isoparametric coordinates, A is the index of global basis function, which is related to indices of univariate basis function by the connectivity array $A = \texttt{INC}(j_1, j_2, \ldots, j_{n_{\rm pd}})$, and $n_{\rm cp}$ is the number of control points of the patch. It is also useful to consider local mappings defined over one individual knot span which can be interpreted as a finite element. Then one can recast (4.7)–(4.9) into the form

$$\mathbf{x}(\boldsymbol{\zeta}) = \sum_{a=1}^{n_{\rm ecp}} R_a(\boldsymbol{\zeta})\mathbf{x}_a, \tag{4.10}$$

$$\mathbf{u}(\boldsymbol{\zeta}) = \sum_{a=1}^{n_{\rm ecp}} R_a(\boldsymbol{\zeta})\mathbf{d}_a, \tag{4.11}$$

$$\delta\mathbf{u}(\boldsymbol{\zeta}) = \sum_{a=1}^{n_{\rm ecp}} R_a(\boldsymbol{\zeta})\mathbf{c}_a, \tag{4.12}$$

where a is the number of local basis function, and $n_{\rm ecp}$ is the number of element control points. The index of local and global control point are joined by connectivity array $A = \texttt{IEN}(e, a)$, where e is the index of element. Just described discretization was extended for multipatch geometries which was exploited for computation of eigenvalues in works [187, 188].

4.2 Geometric iteration methods for local contact searching

Geometry iteration methods attack the equation (2.170) by replacing the original parametrization of the surface, $\mathbf{x}^{(k)}(\boldsymbol{\xi})$, with a suitably chosen approximation. The aim is to simplify the calculation of the projection onto this approximate surface and recover the closest point projection onto the original surface in the iterative process. An example of such an approximation is the tangent plane which leads to the first order geometry iteration method, here called the least square projection method [189]. The projections on the sphere and torus will be also presented.

4.2.1 The least square projection method

The Newton-Raphson method was based on the linearization of the residual (2.8) whereas the least squares projection directly linearizes the parametrization (2.124). One has

$$\mathbf{x}^{(k)}(\boldsymbol{\xi}) \simeq \mathbf{x}^{(k)}\big|_{\boldsymbol{\xi}_k} + \left.\frac{\partial \mathbf{x}^{(k)}}{\partial \xi^{\alpha}}\right|_{\boldsymbol{\xi}_k} (\xi^{\alpha} - \xi^{\alpha}_k) \tag{4.13}$$

$$= \mathbf{x}^{(k)}\big|_{\boldsymbol{\xi}_k} + \boldsymbol{\tau}^{(k)}_{\alpha}\big|_{\boldsymbol{\xi}_k} (\xi^{\alpha} - \xi^{\alpha}_k)\,, \tag{4.14}$$

which defines a plane tangent to the surface $\Gamma^{(k)}_{\mathrm{c}}$ at a certain point $\mathbf{x}^{(k)}$. Now, $\mathbf{x}^{(i)}$ is projected onto this plane. Substituting into Equation (2.170), the problem becomes linear

$$\left[\mathbf{x}^{(i)} - \left(\mathbf{x}^{(k)}\big|_{\boldsymbol{\xi}_k} + \boldsymbol{\tau}^{(k)}_{\alpha}\big|_{\boldsymbol{\xi}_k} (\xi^{\alpha} - \xi^{\alpha}_k)\right)\right] \cdot \boldsymbol{\tau}^{(k)}_{\beta}\big|_{\boldsymbol{\xi}_k} = 0. \tag{4.15}$$

Using definition (2.170) in the latter expression, the iterative scheme can be devised as

$$\xi^{\alpha}_{k+1} = \xi^{\alpha}_k - \left[\boldsymbol{\tau}^{(k)}_{\alpha}\big|_{\boldsymbol{\xi}_k} \cdot \boldsymbol{\tau}^{(k)}_{\beta}\big|_{\boldsymbol{\xi}_k}\right]^{-1} \left.\frac{\partial d}{\partial \xi^{\beta}}\right|_{\boldsymbol{\xi}_k} \tag{4.16}$$

$$= \xi^{\alpha}_{k+1} = \xi^{\alpha}_k - \left.\left[m^{(k)}\right]^{-1}_{\alpha\beta}\right|_{\boldsymbol{\xi}_k} \left.\frac{\partial d}{\partial \xi^{\beta}}\right|_{\boldsymbol{\xi}_k}. \tag{4.17}$$

Benson et al. [88] employ three iterations of this method to generate an initial guess for the full Newton-Raphson procedure in their local contact treatment. However, the least square projection method may also be used as a fully-fledged algorithm.

4.2.2 The sphere approximation method

As it was mentioned, the least square projection method linearizes the parametrization (2.124). Therefore, it can be viewed as a first order method. The next logical step is to utilize the curvature, i.e. the second order information, to approximate Equation (2.124). This idea leads to the second order geometric methods. A representative of this approach is the sphere approximation method which was proposed in Reference [100].

Each step of the sphere approximation method starts with the calculation of the vector $\mathbf{s}_k = \boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}_k$ using the least square projection (4.17), i.e.

$$s^{\alpha}_k = - \left.\left[m^{(k)}\right]^{-1}_{\alpha\beta}\right|_{\boldsymbol{\xi}_k} \left.\frac{\partial d}{\partial \xi^{\beta}}\right|_{\boldsymbol{\xi}_k}. \tag{4.18}$$

Next, the normal curvature κ_{n} is calculated according to formula

$$\kappa_{\mathrm{n}} = \frac{s^{\alpha}_k \kappa^{(k)}_{\alpha\beta}\big|_{\boldsymbol{\xi}_k} s^{\beta}_k}{s^{\alpha}_k m^{(k)}_{\alpha\beta}\big|_{\boldsymbol{\xi}_k} s^{\beta}_k}, \tag{4.19}$$

where the metric tensor,(4.17), and the curvature tensor,(2.30), are evaluated at point ξ_k. An example of the osculating sphere and the osculating circle of the surface at point $\mathbf{x}^{(k)}_k$ is plotted in Figure (4.2). The osculating circle lies in the plane defined by the normal vector $\mathbf{n}^{(k)}$ and the auxiliary tangent vector $\mathbf{t}$ which is defined as

$$\mathbf{t} = \boldsymbol{\tau}^{(k)}_{\alpha}\big|_{\boldsymbol{\xi}_k} (\xi^{\alpha}_{k+1} - \xi^{\alpha}_k). \tag{4.20}$$

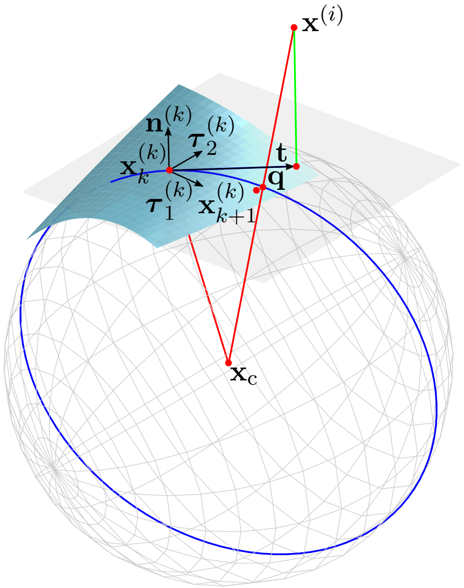


Figure 4.2: Osculating sphere of the surface at point $\mathbf{x}_k^{(k)}$.

Since the inverse value of the normal curvature is the radius of curvature, the centre of the osculation sphere can be easily obtained as

$$\mathbf{x}_\mathrm{c} = \mathbf{x}_k^{(k)} + \frac{\mathbf{n}^{(k)}}{\kappa_\mathrm{n}}. \tag{4.21}$$

The projection of the slave point $\mathbf{x}^{(i)}$ onto the osculation sphere yields

$$\mathbf{q} = \mathbf{x}_\mathrm{c} + \frac{\mathbf{x}^{(i)} - \mathbf{x}_\mathrm{c}}{|\kappa_\mathrm{n}| \left\|\mathbf{x}^{(i)} - \mathbf{x}_\mathrm{c}\right\|}. \tag{4.22}$$

Finally, the point $\mathbf{q}$ has to be mapped onto the original surface (2.124) to obtain new approximation of the parametric coordinates $\boldsymbol{\xi}_{k+1}$ by scaling the least square projection step

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + \theta_k \mathbf{s}_k, \tag{4.23}$$

where for the step-length parameter θ_k it holds

$$\theta_k = \operatorname{sign}(\mathbf{t} \cdot (\mathbf{q} - \mathbf{x}_k^{(k)})) \sqrt{\frac{2\left\|\mathbf{t} \times (\mathbf{q} - \mathbf{x}_k^{(k)})\right\|}{|\kappa_\mathrm{n}| \left\|\mathbf{t}\right\|^3}}. \tag{4.24}$$

4.2.3 The torus approximation method

The extension of the sphere approximation method was proposed in Reference [101] where, instead of the normal curvature, the principal curvatures at a given point of surface are utilized. The local shape of the surface is approximated by a torus patch as shown in Figure 4.3. The major and minor circles of the torus are constructed using the maximum and minimum principal curvatures κ_α yielded from the solution of the eigenvalue problem

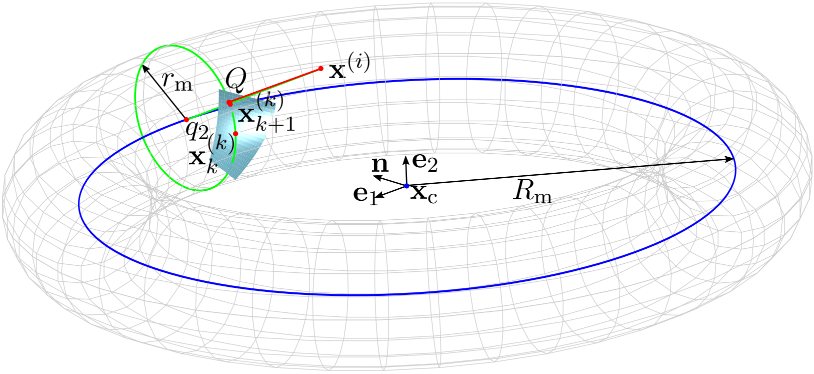


Figure 4.3: Approximation torus of the surface at point $\mathbf{x}_k^{(k)}$.

$$\left(\boldsymbol{\kappa}^{(k)} - \kappa_\alpha \mathbf{m}^{(k)}\right)\Big|_{\boldsymbol{\xi}_k} \mathbf{e}_\alpha = \mathbf{0}, \qquad \alpha = 1,2 \tag{4.25}$$

where the metric tensor $\mathbf{m}^{(k)}$ is given by (2.15) and the curvature tensor $\boldsymbol{\kappa}^{(k)}$ by (2.30). Note that the eigenvalues (principal curvatures) κ_α and eigenvectors (principal directions) $\mathbf{e}_\alpha$ are sorted in such a way that $|\kappa_1| < |\kappa_2|$. If $\kappa_1 > 0$ then κ_1,κ_2, and the normal vector $\mathbf{n}^{(k)}$ are multiplied by -1. In Figure 4.3 the blue circle represents the major circle and the green circle represents the minor one. Similarly to equation (4.21), the centre of the torus can be obtained as

$$\mathbf{x}_\mathrm{c} = \mathbf{x}_k^{(k)} + \frac{\mathbf{n}^{(k)}}{\kappa_1}, \tag{4.26}$$

and the major and minor radii of the torus are calculated from

$$R_\mathrm{m} = -\frac{1}{\kappa_1} + \frac{1}{\kappa_2}, \qquad \text{and} \qquad r_\mathrm{m} = \frac{1}{|\kappa_2|}, \tag{4.27}$$

respectively.

In each iteration, the centre point $\mathbf{x}_\mathrm{c}$ and the major R_m and minor r_m radii of the approximation torus are calculated at point $\mathbf{x}_k^{(k)}$. Next, the projections of the slave point $\mathbf{x}^{(i)}$ onto the torus are performed. Specifically, the slave point $\mathbf{x}^{(i)}$ is projected onto the major circle

$$\mathbf{q}_1 = [\mathbf{I} - \mathbf{e}_2 \otimes \mathbf{e}_2]\left(\mathbf{x}^{(i)} - \mathbf{x}_\mathrm{c}\right), \tag{4.28}$$

$$\mathbf{q}_2 = \mathbf{x}_\mathrm{c} + R_\mathrm{m}\frac{\mathbf{q}_1}{\|\mathbf{q}_1\|}, \tag{4.29}$$

where $\mathbf{I} \in \mathbb{R}^{n_\mathrm{sd} \times n_\mathrm{sd}}$ is the identity matrix and $\mathbf{q}_1$ is an auxiliary vector. The point $\mathbf{q}_2$ lying on the major circle of the torus is projected onto the minor circle

$$\mathbf{Q} = \mathbf{q}_2 + r\frac{(\mathbf{x}^{(i)} - \mathbf{q}_2)}{\|\mathbf{x}^{(i)} - \mathbf{q}_2\|}. \tag{4.30}$$

Finally, the torus point $\mathbf{Q}$ is mapped back onto the original surface (2.124) to obtain a new approximation of the parametric coordinates of $\mathbf{x}_{k+1}^{(k)}$ assuming that

$$\mathbf{x}^{(k)}(\boldsymbol{\xi}_{k+1}) = \mathbf{Q}. \tag{4.31}$$

The Taylor expansion of equation (4.31) about the point $\mathbf{x}_k^{(k)}$ yields

$$\mathbf{x}_k^{(k)} + \left.\frac{\partial \mathbf{x}^{(k)}}{\partial \xi^\alpha}\right|_{\boldsymbol{\xi}_k} s_k^\alpha + \frac{1}{2} s_k^\alpha \left.\frac{\partial \mathbf{x}^{(k)}}{\partial \xi^\alpha \partial \xi^\beta}\right|_{\boldsymbol{\xi}_k} s_k^\beta = \mathbf{Q}. \tag{4.32}$$

This overconstrained system of nonlinear algebraic equations is solved by the Newton-Raphson method for computation of vector $\mathbf{s}_k$. The new approximation of parametric coordinates $\boldsymbol{\xi}_{k+1}$ is obtained as

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + \mathbf{s}_k. \tag{4.33}$$

4.3 Bipenalty method

In dynamic transient analysis, the penalty method can also be applied to the mass matrix. This technique is known as the mass penalty or the inertia penalty method and it initially gained popularity for frequency domain problems, see [167, 168]. In contrast to the stiffness penalty approach, it significantly reduces one or more eigenfrequencies. In Reference [169] the bipenalty technique was introduced for time domain problems, where the both penalty formulations were used simultaneously. The goal of this method is to find the optimum of the so-called critical penalty ratio (CPR) defined as the ratio of stiffness and mass penalty parameters so that the maximum eigenfrequency and the critical time step are preserved. The calculation of CPR requires an analysis of the full bipenalised problem. Owing to mathematical difficulty, it limits the classes of elements that can be taken into account. In order to overcome this problem, a simple relationship between the CPR of an element and its maximum unpenalised eigenfrequency was derived in [170]. Thus, the multiple constraints and more complex element formulations can be directly accounted for [171].

In dynamics, the simultaneous use of the stiffness penalties and inertia/mass penalties, called the bipenalty method, was originally proposed in [169]. There was defined the penalty ratio as

$$R = \frac{\epsilon_\mathrm{s}}{\epsilon_\mathrm{m}}, \tag{4.34}$$

where ϵ_s and ϵ_m are the stiffness and mass penalty parameter, respectively. There were also derived optimum values of the penalty ratios — the so-called critical penalty ratios (CPR) — for a number of finite elements such that the critical time step of the penalised system remains unaffected. A new method of calculating the CPR associated with a finite element formulation was developed in [170]. Recently, this finding was extended to include systems with an arbitrary set of multipoint constraints [171] and to cohesive zone (surface) elements [172].

Now, a brief description of the bipenalty method in application to the contact-impact problem [192, 190] follows. For the sake of simplicity, only linear elastodynamic with small deformations will be considered. Let us assume that the contact boundary $\Gamma_\mathrm{c}^{(i)}$ is known. The standard stiffness penalty method adds an extra term to the strain energy to enforce the zero gap on the contact boundary

$$\mathcal{E}_\mathrm{pot}^{\mathrm{p}(i)}(\mathbf{u}^{(i)}) = \int_{\Omega_0^{(i)}} \frac{1}{2}\boldsymbol{\sigma}^{(i)} : \boldsymbol{\varepsilon}^{(i)} \,\mathrm{d}\Omega^{(i)} + \int_{\Gamma_\mathrm{c}^{(i)}} \frac{1}{2}\epsilon_\mathrm{s} \left(g_\mathrm{N}^{(i)}\right)^2 \mathrm{d}\Gamma^{(i)}, \tag{4.35}$$

where $\boldsymbol{\varepsilon}^{(i)}$ is the infinitesimal strain tensor defined as the symmetric part of the gradient of deformation. Further, the inertia penalty term can also be added to the kinetic energy to enforce the zero gap rate on the contact interface

$$\mathcal{E}_{\mathrm{kin}}^{\mathrm{p}(i)}(\dot{\mathbf{u}}^{(i)}) = \int_{\Omega_0^{(i)}} \frac{1}{2}\rho_0 \dot{\mathbf{u}}^{(i)} \cdot \dot{\mathbf{u}}^{(i)} \, \mathrm{d}\Omega^{(i)} + \int_{\Gamma_{\mathrm{c}}^{(i)}} \frac{1}{2}\epsilon_{\mathrm{m}} \left(\dot{g}_{\mathrm{N}}^{(i)}\right)^2 \mathrm{d}\Gamma^{(i)}. \tag{4.36}$$

Now, a new penalised Lagrangian functional can be defined as

$$\mathcal{L}_{\mathrm{p}}(\mathbf{u}^{(i)}, \dot{\mathbf{u}}^{(i)}) = \sum_{i=1}^{2} \left\{ \mathcal{T}_{\mathrm{p}}^{(i)}(\dot{\mathbf{u}}^{(i)}) - \left(\mathcal{U}_{\mathrm{p}}^{(i)}(\mathbf{u}^{(i)}) - \mathcal{W}^{(i)}(\mathbf{u}^{(i)}) \right) \right\}. \tag{4.37}$$

The unknown displacement field can be found as one which renders the penalised action functional stationary

$$\delta \int_0^T \mathcal{L}_{\mathrm{p}}(\mathbf{u}^{(i)}, \dot{\mathbf{u}}^{(i)}) \, \mathrm{d}t = 0, \tag{4.38}$$

where δ denotes the first variation or the directional derivative in the direction of virtual work $\delta\mathbf{u}^{(i)}$. Using the standard procedures one arrives to the principle of virtual displacements which serves the base for the finite element discretization

$$\sum_{i=1}^{2} \left(\delta\Pi_{\mathrm{int}}^{(i)} + \delta\Pi_{\mathrm{ext}}^{(i)} + \delta\Pi_{\mathrm{c}}^{(i)} \right) = 0. \tag{4.39}$$

where

$$\delta\Pi_{\mathrm{int}}^{(i)}(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}) := \int_{\Omega_0^{(i)}} \delta\mathbf{u}^{(i)} \cdot \rho_0^{(i)} \ddot{\mathbf{u}}^{(i)} \, \mathrm{d}\Omega^{(i)} + \int_{\Omega_0^{(i)}} \delta\boldsymbol{\varepsilon}^{(i)} : \boldsymbol{\sigma}^{(i)} \, \mathrm{d}\Omega^{(i)}, \tag{4.40}$$

$$\delta\Pi_{\mathrm{ext}}^{(i)}(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}) := -\int_{\Omega_0^{(i)}} \delta\mathbf{u}^{(i)} \cdot \mathbf{f}_0^{(i)} \, \mathrm{d}\Omega^{(i)} - \int_{\Gamma_\sigma^{(i)}} \delta\mathbf{u}^{(i)} \cdot \hat{\mathbf{T}}^{(i)} \, \mathrm{d}\Gamma^{(i)}, \tag{4.41}$$

$$\delta\Pi_{\mathrm{c}}^{(i)}(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}) := -\int_{\Gamma_{\mathrm{c}}^{(i)}} \delta g_{\mathrm{N}}^{(i)} \left(\epsilon_{\mathrm{m}} \ddot{g}_{\mathrm{N}}^{(i)} + \epsilon_{\mathrm{s}} g_{\mathrm{N}}^{(i)} \right) \mathrm{d}\Gamma^{(i)}, \tag{4.42}$$

represent the virtual work of the inertia forces and internal forces, external forces, and contact forces, respectively. Now the main feature of the bipenalty method is immediately visible from the expression for the virtual work of the contact forces (4.42), where in addition to the work of the stiffness penalty, there is also the term expressing the work due to inertia penalty. Tuning the ratio of penalty parameters, R, can control the stability of the explicit temporal integration, as will be shown on examples in Chapter 6.

Chapter 5

Results and Discussion

This chapter presents the main results demonstrating the fulfillment of the defined objectives of this dissertation. Namely, in the first section the newly developed symmetry preserving Gauss-point-to-segment isogeometric contact formulation is presented. Then, the proposed local contact search procedure utilizing the geometry iteration methods is compared with the other conventional techniques. In the last section of this chapter, a stable time-step size estimation for bipenalty method is derived.

5.1 Symmetry preserving Gauss-point-to-segment isogeometric contact formulation

In this section, the discretization of the contact residual term $\delta\Pi_{\mathrm{c}}^{(i)}$ by the isogeometric finite elements [191] will be described. By the substitution of (4.9) into (2.104)

$$\delta\Pi_{\mathrm{c}}^{h(i)} = -\int_{\Gamma_{\mathrm{c}}^{(i)}} \sum_{A=1}^{n_{\mathrm{cp}}} R_A \mathbf{c}_A^{(i)} \cdot \left(p_{\mathrm{c}}^{(i)} \bar{\mathbf{n}}^{(k)} + t_{\mathrm{T}\alpha}^{(i)} \bar{\boldsymbol{\tau}}^{(k)\alpha} \right) \mathrm{d}\Gamma^{(i)}. \tag{5.1}$$

Note that the superscript h indicating discretized quantities has been omitted to simplify the notation. Similarly to the bulk elements, $\Omega_0^{e(i)}$, also the boundary $\Gamma_{\mathrm{c}}^{(i)}$ is decomposed into finite elements, here called segments

$$\Gamma_{\mathrm{c}}^{(i)} = \bigcup_{s=1}^{n_{\mathrm{s}}} \Gamma_{\mathrm{c}}^{s(i)}. \tag{5.2}$$

Consequently, the contact residual can be assembled segment-wise

$$\delta\Pi_{\mathrm{c}}^{h(i)} = \bigcup_{s=1}^{n_{\mathrm{s}}} - \left(\mathbf{c}^{s(i)} \right)^{\mathrm{T}} \int_{\Gamma_{\mathrm{c}}^{s(i)}} \left(p_{\mathrm{c}}^{(i)} \mathbf{N}^{s(i)} + t_{\mathrm{T}\alpha}^{(i)} \mathbf{T}_\alpha^{s(i)} \right) \mathrm{d}\Gamma^{s(i)}, \tag{5.3}$$

where n_{s} indicates number of contact segments, $\mathbf{c}^{s(i)}$ is the segment matrix of control point weighted displacements, and $\mathbf{N}^{s(i)}, \mathbf{T}_\alpha^{s(i)}$ are the segment matrices defined as

$$\mathbf{N}^{s(i)} = \begin{bmatrix} R_1 \bar{\mathbf{n}}^{(k)} \\ \vdots \\ R_{n_{\mathrm{scp}}} \bar{\mathbf{n}}^{(k)} \end{bmatrix}, \qquad \mathbf{T}_\alpha^{s(i)} = \begin{bmatrix} R_1 \bar{\tau}^{(k)\alpha} \\ \vdots \\ R_{n_{\mathrm{scp}}} \bar{\tau}^{(k)\alpha} \end{bmatrix}, \tag{5.4}$$

where n_{scp} is the number of segment control points. Finally, one can introduce the vector of equivalent control point contact forces through the identity

$$\delta\Pi_{\mathrm{c}}^{h(i)} = \bigcup_{s=1}^{n_{\mathrm{s}}} \left(\mathbf{c}^{s(i)}\right)^{\mathrm{T}} \mathbf{F}_{\mathrm{c}}^{s(i)}, \tag{5.5}$$

where for $\mathbf{F}_{\mathrm{c}}^{s(i)}$ it holds

$$\mathbf{F}_{\mathrm{c}}^{s(i)} = -\int_{\Gamma_{\mathrm{c}}^{s(i)}} \left(p_{\mathrm{c}}^{(i)}\mathbf{N}^{s(i)} + t_{\mathrm{T}\alpha}^{(i)}\mathbf{T}_{\alpha}^{s(i)}\right) \mathrm{d}\Gamma^{s(i)}. \tag{5.6}$$

The global vector of control point contact forces is then assembled in a conventional manner

$$\mathbf{F}_{\mathrm{c}} = \sum_{i=1}^{2} \bigcup_{s=1}^{n_s} \mathbf{F}_{\mathrm{c}}^{s(i)}. \tag{5.7}$$

5.2 Assessment of methods for the local contact searching

In this section, different local contact searching methods will be assessed from both, the qualitative and the quantitative point of view. First of all a benchmark problem will be defined in Section 5.2.1. The actual comparison of the methods will be described in Section 5.2.2. The last section, Section 5.2.3, concentrates on the performance of the local contact search methods.

5.2.1 Benchmark problem

In order to assess the methods introduced in Sections 2.10.2 and 4.2, a suitable benchmark function is needed. Indeed, there are several popular test functions such as the Rosenbrock function or Himmelblau's function which are well-known by the optimization community [173]. However, since our motivation comes from computational contact mechanics, we have chosen a test function which arises in this field.

The benchmark test follows from an example which involves the bending of two rectangular plates over a cylinder [1, p. 2632]. During the iterative solution of this problem, the instability of the calculation of the normal vector by the Newton-Raphson method occurred for the specific surface $\Gamma_{\mathrm{c}}^{(k)}$ and the slave Gauss point $\mathbf{x}^{(i)}$. The geometry of this master segment of the second order serendipity element was used as the benchmark configuration for the comparison of described methods . Note that the complete numerical solution of this example will be presented in Section 6.1.6.

The input parameters for this benchmark problem are given in Tables 5.1–5.3. The shape functions of a quadrilateral element with eight nodes are summarized in Table 5.2. The coordinates of the nodal points $\mathbf{x}_a^{(k)}$ of the master segment $\Gamma_{\mathrm{c}}^{(k)}$ and slave points $\mathbf{x}^{(i)}$ are shown in Tables 5.3 and 5.1, respectively.

$x_1^{(i)}$	$x_2^{(i)}$	$x_3^{(i)}$
0.004676691973675341	0.0855075528940456	0.06290027930804223

Table 5.1: Coordinates of the point $\mathbf{x}^{(i)}$.

a	$N_a(\xi^1, \xi^2)$
1	$0.25(1-\xi^1)(1-\xi^2) - 0.5N_5 - 0.5N_8$
2	$0.25(1+\xi^1)(1-\xi^2) - 0.5N_5 - 0.5N_6$
3	$0.25(1+\xi^1)(1+\xi^2) - 0.5N_6 - 0.5N_7$
4	$0.25(1-\xi^1)(1+\xi^2) - 0.5N_7 - 0.5N_8$
5	$0.5(1-(\xi^1)^2)(1-\xi^2)$
6	$0.5(1-(\xi^2)^2)(1+\xi^1)$
7	$0.5(1-(\xi^1)^2)(1+\xi^2)$
8	$0.5(1-(\xi^2)^2)(1-\xi^1)$

Table 5.2: The shape functions of a quadrilateral element with eight nodes.

a	$x_{1a}^{(k)}$	$x_{2a}^{(k)}$	$x_{3a}^{(k)}$
1	0.0386717163175745	-0.0405563517184674	0.0381663218813485
2	0.0000000000000000	-0.0397596588441705	0.0391125455253157
3	0.0000000000000000	-0.0394782972838687	0.0763120389982561
4	0.0385870095736594	-0.0411702783414374	0.0757307152845918
5	0.0194722110243407	-0.0357571570173317	0.0387704050499758
6	0.0000000000000000	-0.0406822372736114	0.0581663708592389
7	0.0190469010033457	-0.0407176751343564	0.0759898919202766
8	0.0387103243000099	-0.0409359436667850	0.0569337608379921

Table 5.3: Coordinates of the nodal points $\mathbf{x}_a^{(k)}$ of the surface $\Gamma_c^{(k)}$.

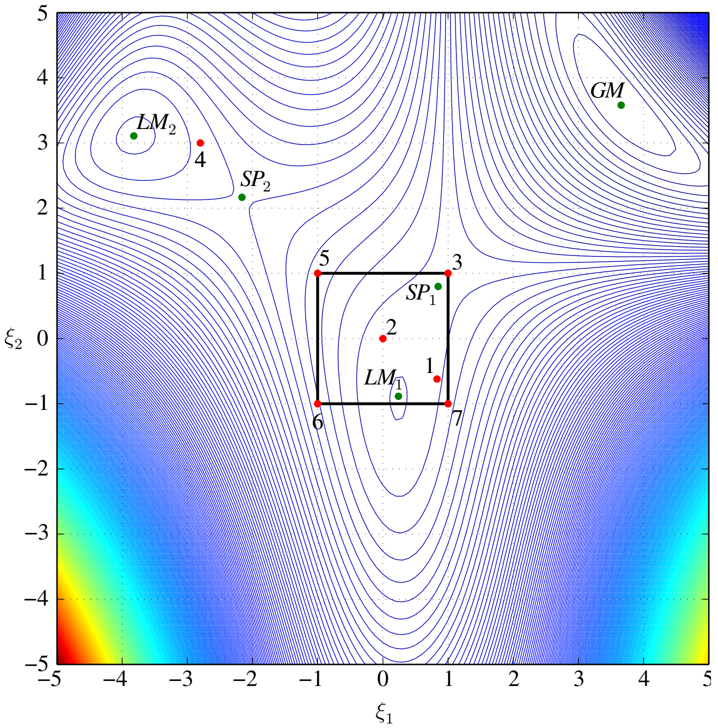


Figure 5.1: The contours of the square distance function.

The contours of the squared distance function (2.7) for this particular geometry are depicted in Figure 5.1. The master segment facet is outlined by the black square in the middle of the figure. Furthermore, there are positions of five stationary points of the squared distance function marked by green dots. The local minima are denoted by $(\mathrm{LM}_1, \mathrm{LM}_2)$, the global minimum by (GM), and remaining two are the saddle points $(\mathrm{SP}_1, \mathrm{SP}_2)$.

Initial guesses $\boldsymbol{\xi}_0$ have to be prescribed. Seven points have been chosen whose coordinates are listed in Table 5.4. These seven initial guesses are marked by red dots in Figure 5.1. The initial guess #1 corresponds to a linear estimate through one iteration of the least square projection method for $\boldsymbol{\xi}_0 = \{0, 0\}^{\mathrm{T}}$, see Reference [88]. The initial guess #2 placed into the origin of the parametric coordinates is rather the natural choice of the starting estimate. The initial guess #3 is located near the saddle point of the squared distance function. Similarly, the initial guess #4 is intentionally located in the vicinity of the local minimum (LM_2). The remaining initial estimates are considered only for performance testing purposes.

#	ξ^1	ξ^2
1	0.830817352545586	-0.621994609286856
2	0	0
3	1	1
4	-2.8	3
5	-1	1
6	-1	-1
7	1	-1

Table 5.4: Isoparametric coordinates of the initial guesses.

Finally, it is necessary to define a convergence criterion. The iterative process is controlled by the change of vector $\boldsymbol{\xi}_k$. For all the tested methods we have set the criterion

$$\|\boldsymbol{\xi}_k - \boldsymbol{\xi}_{k-1}\| \leq \mathtt{TOL} \tag{5.8}$$

with $\mathtt{TOL} = 10^{-10}$. In case of the simplex method there is a vector $\boldsymbol{\xi}_{ki}$ for each of three vertices. Therefore, we calculate the centroid of the simplex as the arithmetic average in the current iteration step

$$\boldsymbol{\xi}_k = \frac{1}{3} \sum_{i=1}^{3} \boldsymbol{\xi}_{ki} \tag{5.9}$$

and, as well as in the previous iteration step $\boldsymbol{\xi}_{k-1}$ and substitute them into the criterion (5.8). In the following section, the results for each of tested method will be discussed in detail.

5.2.2 Methods assessment

After introducing the benchmark problem in the previous section, the assessment of the methods follows. An overview of the results is presented in Table 5.5 for all the tested methods and initial estimates. For each of them the solution vector $\boldsymbol{\xi}_\infty$, the number of iterations (`NITER`), and the CPU time are shown in Table 5.5. There are also the values of principal minors of the Hessian matrix, $\det \mathbf{H}(\boldsymbol{\xi}_\infty)$ and $H_{11}(\boldsymbol{\xi}_\infty)$, for the determination of the type of stationary point. The error norm, $\|\boldsymbol{\xi}_k - \boldsymbol{\xi}_\infty\|$, during the iterative process is plotted in Figures 5.2a-5.2h. The trajectories of the iterative process for the Newton-Raphson method, the least square projection and the simplex method are plotted in Figures 5.3, 5.4, and 5.5, respectively.

The Newton-Raphson method

The inadvisability of the Newton-Raphson method is clearly demonstrated by the iterations for the initial guess #1, which is plotted in Figure 5.3. Due to the fact that the Hessian matrices $\mathbf{H}_0$ and $\mathbf{H}_{10}$ are indefinite, the approximation of the solution "rebounds" far from the previous iteration. It is also apparent in Figure 5.2a, where the error norm for the Newton-Raphson iterations is plotted. Finally, the method converges to the local minimum LM_1 in 28 iterations–see Table 5.6a.

A similar behaviour occurs for other initial estimates, except guesses #2 and #3. In the case of point #2, the Hessian matrix is positive definite in each iteration. Thus, the method converges quadratically to the local minimum LM_1 in 5 iterations. On the other hand, in the case of initial point #3, the Hessian matrix is indefinite in each iteration. Consequently, the method converges quadratically to the saddle point SP_1 also in 5 iterations. Thus, if the initial guess lies in the domain, where the distance function is convex, the method converges quadratically to the nearest local minimum. However, the convergence of the Newton-Raphson method to the nearest local minimum is difficult to achieve since the positive definiteness of the Hessian matrix is not guaranteed in general.

The least square projection method

Table 5.6b indicates the interesting fact that this method always converges to the global minimum GM. Figure 5.4, where the iterations are depicted, demonstrates this fact for the initial guess #2. Note that three iterations with a least square projection are recommended to generate an initial guess in Reference [88]. However, as is clear from the Figure 5.4 (number 3), such linear guess does not generate better initial estimation, at least for quadratic segments. Moreover, the global minimizer may be situated outside of the master segment face–see Figure 5.1. Thus, the least square projection is not a suitable method for local contact search.

The sphere approximation method

In contrast to the least square projection this method can converge to the closest local minimum as shown in Table 5.6c. However, this is not the case with all starting points. For example, the solution for the initial guess #3 was expected to converge to the global minimum GM rather than to the local minimum LM_1. Similarly, the sphere approximation method did not converge to the closest minimum for the initial estimates #4 and #7, comparing the type of stationary point in Table 5.6c with their initial position in Figure 5.1.

It should be noted that the convergence criterion had to be modified to achieve the convergence. This is due to the vector product in Equation (4.24). The vectors become collinear as the sphere approximation method converges. As a consequence, the method oscillates around the solution. In order to suppress this effect the convergence tolerance was adjust to a more free value $\texttt{TOL} = 10^{-7}$.

The torus approximation method

The torus approximation method converges to exactly the same results as the sphere approximation method, which is apparent from the comparison of Table 5.6d and Table 5.6c. In contrast to the sphere approximation method it shows better performance manifested by lower number of iterations and CPU times (see also Figure 5.2c and Figure 5.2d). However, the convergence to the nearest local minimum is also difficult to achieve in general. Therefore, the location of initial guesses has to be carefully selected in practical applications of this method.

The steepest descent method

In general, the steepest descent method, as well as the other gradient methods, converges to a saddle point much less than the quasi-Newton methods. It can be confirmed in Table 5.5e where no saddle point occurs in the solution of the benchmark problem. This is the consequence of the fact that the Hessian matrix is not employed in the computation.

The convergence to a saddle point is significantly influenced by the line search technique. Although the steepest descent method returns a descent search direction, the step length procedure may cause the solution to "wander" in the vicinity of the saddle point. Such behaviour is observed for the initial guess #3. In Figure 5.2e the error norm behaviour suffers from a certain fluctuation near the saddle point SP_1. The method finally converges to the global minimum GM. Note that the tested quasi-Newton methods did not share this feature.

Poor convergence was observed for the initial guess #5. There are difficulties with the step-length θ_0. Since the initial guess #5 is located close to the saddle point, the quadratic interpolation in the line search procedure returns a parabola with a small curvature. Consequently, the step-length causes a huge rebound in the next iteration. The method converges very slowly to the global minimum GM.

The Broyden method

The Broyden method is sensitive to the selection of the initial secant matrix $\mathbf{D}_0$. It plays a critical role in the computation of the initial search direction $\mathbf{p}_0$. A good idea is to select a positive definite matrix ensuring a descent direction. Using the multiple of the identity matrix $\mathbf{D}_0 = f'|_0 \,\mathbf{I}$ ensures this condition.

Based on the comparison of results of the steepest descent method (see Table 5.5e) and Broyden's method (see Table 5.5f) we can conclude that a better convergence rate is obtained by Broyden's method. Figure 5.2f shows that numbers of iterations are up to eleven for each initial estimate. The exception is the initial guess #5 for which the method converges into 15669 iterations. Just as in the preceding section, this is caused by the step-length procedure.

The BFGS method

The BFGS secant matrix is initialized identically to Broyden's method. Based on Table 5.5g it can be concluded that the best convergence rate is reached by this method. For example, the method converges in only eight iterations for the initial guess #1. It should be emphasized that the error norm for all initial guesses except for #5 is practically zero within four iterations (see Figure 5.2g).

The simplex method

At the beginning of the starting procedure, the length of the simplex edge has to be set. Since the dependence on the number of iterations of the simplex edge length is stochastic, the value was set to 0.43 for all initial guesses. This value was obtained via an ad hoc procedure.

Note that the numerical solution for all corner and central starting points, except for point #3, converge to the nearest local minimum LM_1. Figure 5.5 shows the typical transforming of the simplex toward the nearest local minimum. For the initial guess #3 the solution directs toward the global minimum GM since the initial position of the simplex is on the opposite sides of the saddle ridge.

The distribution of the error norm for the first fifty iterations is depicted in Figure 5.2h. The numbers of iterations are higher than one hundred (see Table 5.5h) even though a considerable decrease of error occurs within thirty iteration steps (see Figure 5.2h).

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	28	36.6
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	5	7.4
3	0.923	0.810	-1.76e-06	7.46e-04	SP_1	5	7.3
4	0.923	0.810	-1.76e-06	7.46e-04	SP_1	9	13.0
5	-2.167	2.133	-2.68e-06	-2.29e-04	SP_2	12	16.2
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	12	16.3
7	0.923	0.810	-1.76e-06	7.46e-04	SP_1	17	22.3

(a) : Newton-Raphson method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	3.649	3.624	8.03e-03	7.66e-06	GM	133	16.0
2	3.649	3.624	8.03e-03	7.66e-06	GM	134	16.5
3	3.649	3.624	8.03e-03	7.66e-06	GM	116	14.3
4	3.649	3.624	8.03e-03	7.66e-06	GM	124	16.3
5	3.649	3.624	8.03e-03	7.66e-06	GM	134	16.7
6	3.649	3.624	8.03e-03	7.66e-06	GM	137	16.8
7	3.649	3.624	8.03e-03	7.66e-06	GM	124	15.4

(b) : Least square projection method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	20	55.5
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	42	99.3
3	0.241	-0.926	1.24e-06	2.82e-03	LM_1	47	109.1
4	3.649	3.624	7.66e-06	6.33e-03	GM	15	33.9
5	0.241	-0.926	1.24e-06	2.82e-03	LM_1	19	43.3
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	49	113.7
7	-3.804	3.111	6.76e-06	2.53e-03	LM_2	38	88.9

(c) : Sphere approximation method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	9	46.4
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	8	24.4
3	0.241	-0.926	1.24e-06	2.82e-03	LM_1	8	21.5
4	3.649	3.624	7.66e-06	6.33e-03	GM	8	24.3
5	0.241	-0.926	1.24e-06	2.82e-03	LM_1	10	30.0
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	7	17.6
7	-3.804	3.111	6.76e-06	2.53e-03	LM_2	7	19.7

(d) : Torus approximation method

Table 5.5: Results of the benchmark test.

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	51	164.7
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	71	223.1
3	3.648	3.624	7.66e-06	6.33e-03	GM	29	95.1
4	-3.804	3.111	6.76e-06	2.53e-03	LM_2	13	41.4
5	3.648	3.624	7.66e-06	6.33e-03	GM	1543	5232.6
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	28	92.3
7	0.241	-0.926	1.24e-06	2.82e-03	LM_1	14	44.5

(e) : Steepest descent method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	9	17.6
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	11	21.2
3	0.923	0.810	-1.76e-06	7.46e-04	SP_1	11	20.3
4	-3.804	3.111	6.76e-06	2.53e-03	LM_2	10	19.3
5	0.241	-0.926	1.24e-06	2.82e-03	LM_1	15669	29981
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	10	18.7
7	0.241	-0.926	1.24e-06	2.82e-03	LM_1	9	16.9

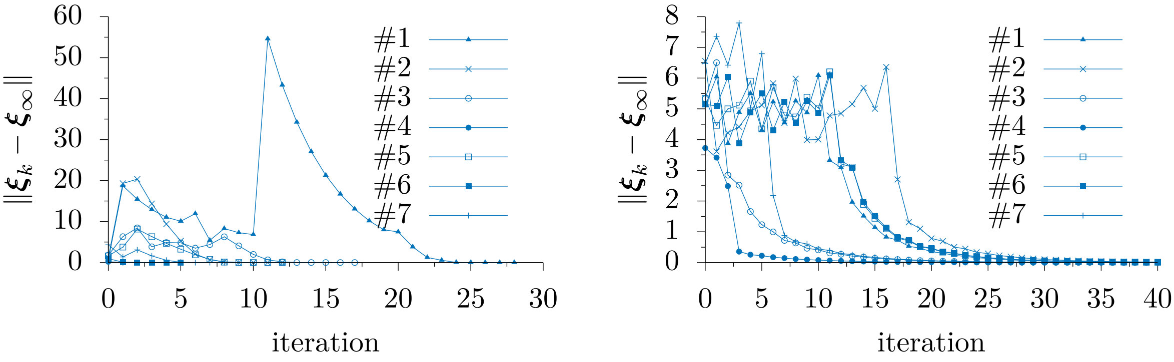
(f) : Broyden's method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	8	13.6
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	8	15.4
3	0.923	0.810	-1.76e-06	7.46e-04	SP_1	10	17.4
4	-3.804	3.111	6.76e-06	2.53e-03	LM_2	8	14.9
5	0.923	0.810	-1.76e-06	7.46e-04	SP_1	23	44.4
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	8	14.9
7	0.241	-0.926	1.24e-06	2.82e-03	LM_1	7	13.2

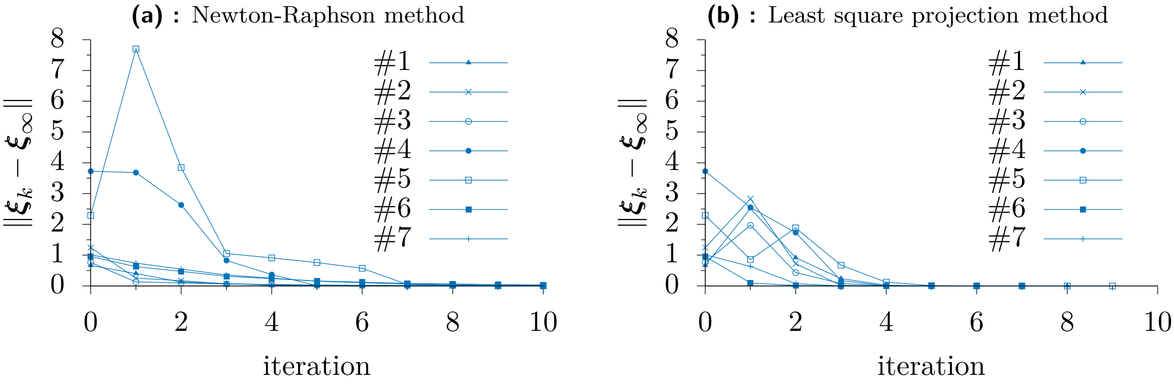
(g) : BFGS method

#	ξ^1_∞	ξ^2_∞	$\det \mathbf{H}(\boldsymbol{\xi}_\infty)$	$H_{11}(\boldsymbol{\xi}_\infty)$	Type	`NITER`	CPU time [ms]
1	0.241	-0.926	1.24e-06	2.82e-03	LM_1	190	247.2
2	0.241	-0.926	1.24e-06	2.82e-03	LM_1	200	260.9
3	3.649	3.624	7.66e-06	6.33e-03	GM	210	278.6
4	-3.804	3.111	6.76e-06	2.53e-03	LM_2	190	238.1
5	0.241	-0.926	1.24e-06	2.82e-03	LM_1	196	253.0
6	0.241	-0.926	1.24e-06	2.82e-03	LM_1	192	247.2
7	3.241	3.926	1.24e-06	2.82e-03	LM_1	184	250.4

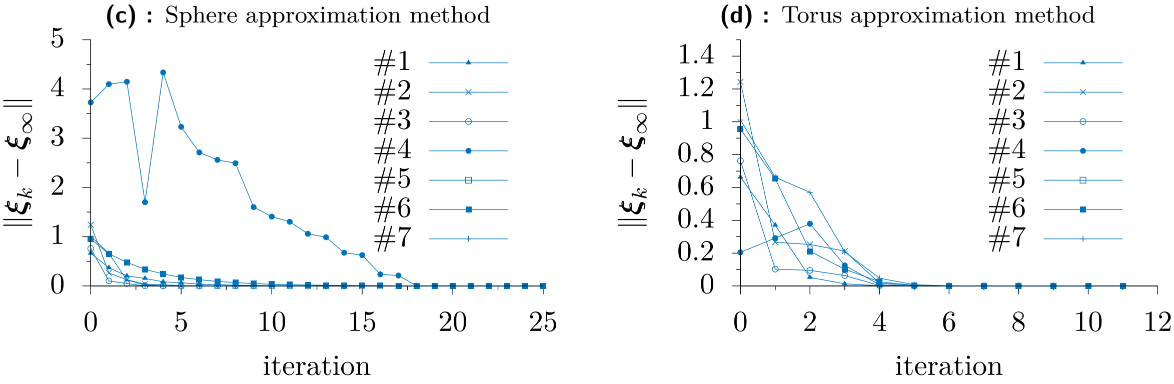
(h) : Simplex method



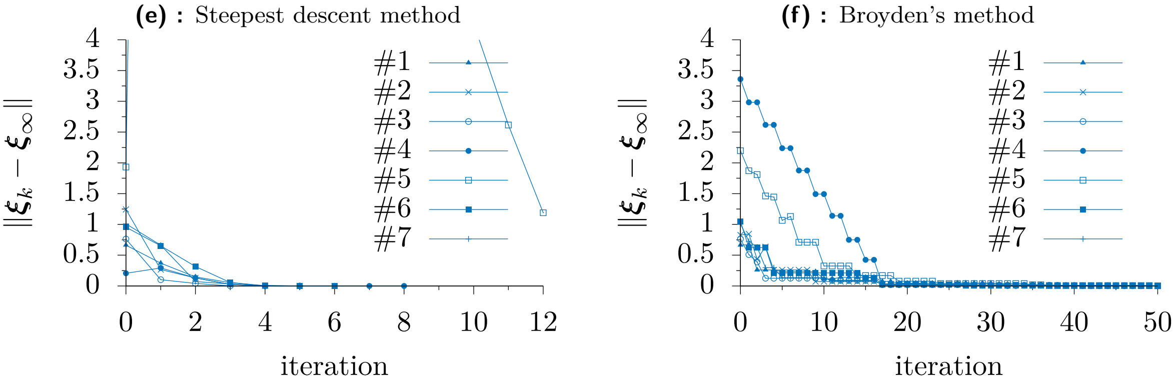
(a) : Newton-Raphson method **(b) :** Least square projection method



(c) : Sphere approximation method **(d) :** Torus approximation method



(e) : Steepest descent method **(f) :** Broyden's method



(g) : BFGS method **(h) :** Simplex method

Figure 5.2: Distribution of error norm.

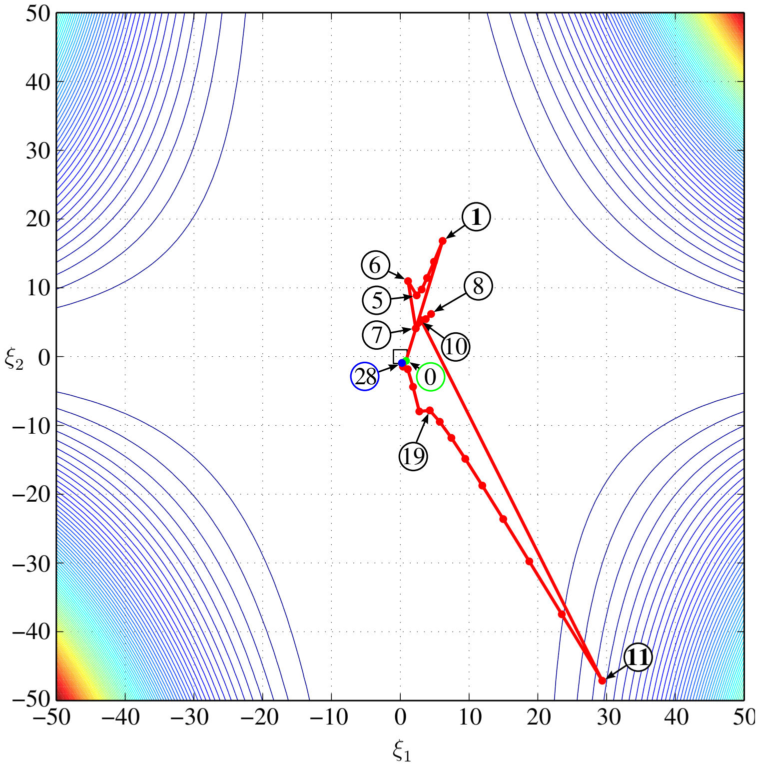


Figure 5.3: Trajectory of iterative process for the Newton-Raphson method.

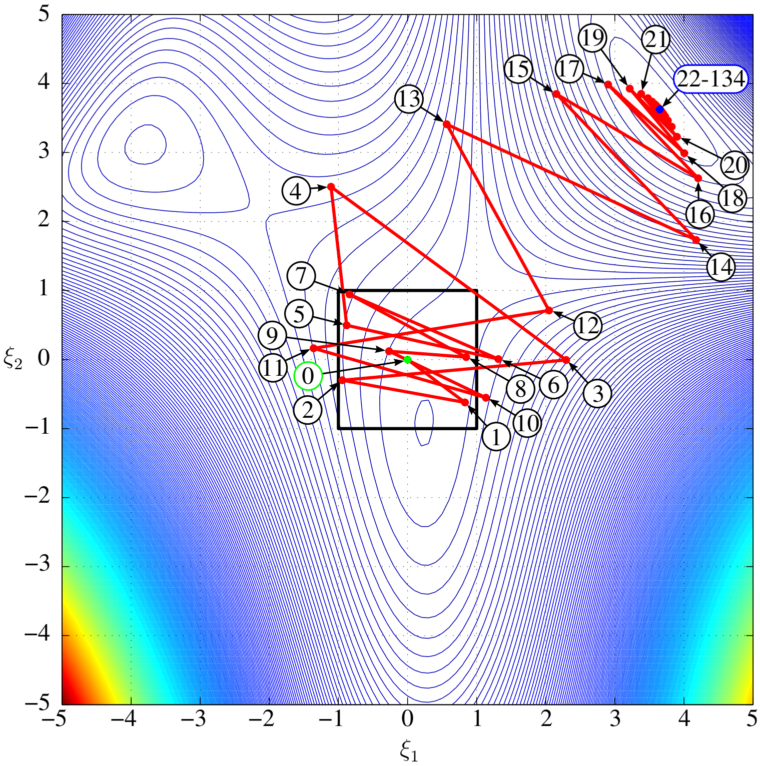


Figure 5.4: Trajectory of iterative process for the least square projection.

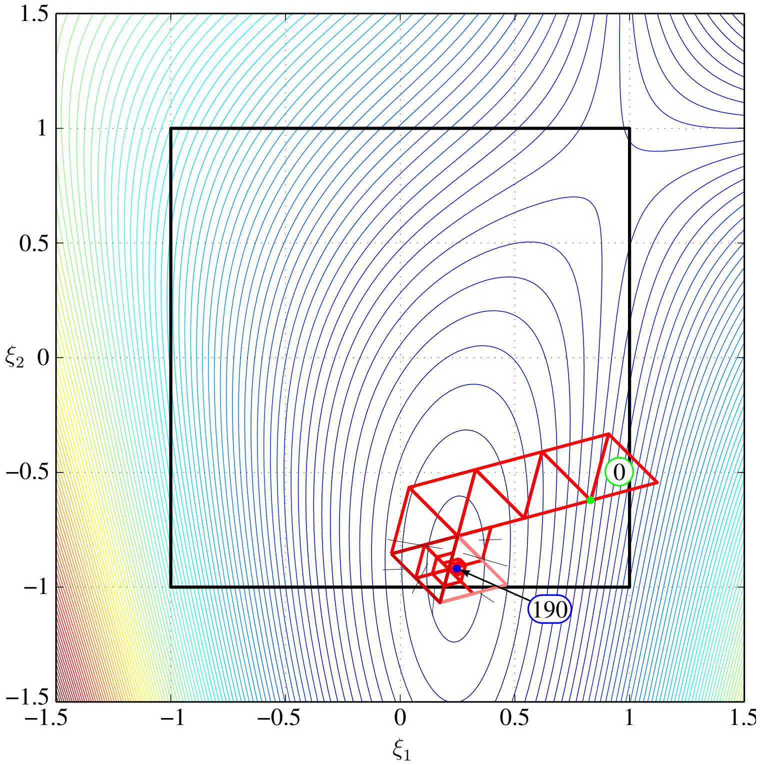


Figure 5.5: Trajectory of iterative process for the simplex method.

5.2.3 Performance test

In benchmark problem, the attention was focused on the qualitative assessment of methods for a given slave point and seven selected initial guesses. However, it is hardly possible to draw conclusions about robustness of these methods. Therefore, the previous benchmark test was extended to a performance test involving an equally spaced three-dimensional grid of slave points, which are projected onto the master segment. The grid of slave points is created by nine squares, which consist of five by five slave points. These square grids are placed parallel to the element facet (see Figure 5.6).

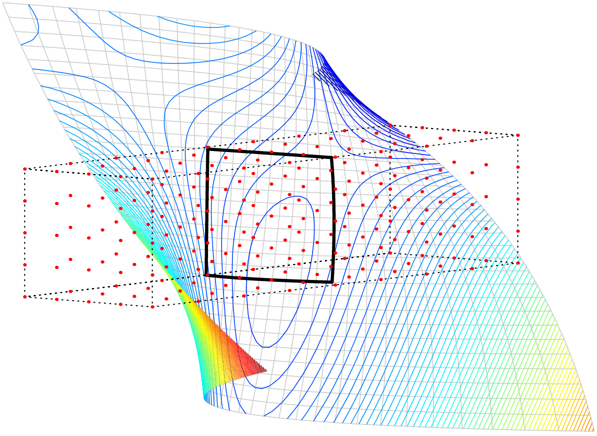


Figure 5.6: Grid of the slave points in the performance test.

For each of the slave points the closest point projection was calculated with 121 initial guesses of starting point defined by the bi-unit square $[-1,-0.8,\dots,1]\times[-1,-0.8,\dots,1]$. Thus, the total number of calculation of the closest point projection for each method was 27 225. The results of the performance test are summarized in Tables 5.5–5.7, where the average number of iterations (AVRG `NITER`), the maximum number of iterations (MAX `NITER`), the number of "failed" calculations (DNF `NITER`), and the average CPU time (AVRG CPU) are shown. Besides the average CPU time the percentage share of computational time consumption of the line search procedure is also displayed. Note that the iteration process is referred to be divergent if twenty thousand iterations are reached. The results of the performance test for all eight tested methods are shown in Table 5.5. The methods utilizing the line search technique employ quadratic interpolation to calculate the step-length parameter as in the previous benchmark test.

Method	AVRG `NITER`	MAX `NITER`	DNF `NITER`	AVRG CPU [ms]
Newton-Raphson	14	762	19	34
Least square projection	52	726	5427	76
Sphere approx. method	26	4189	0	75
Torus approx. method	7	15	0	20
Steepest descent	52	19076	183	137 (60.8 %)
Broyden	166	19940	922	442 (55.4 %)
BFGS	12	12175	8	48 (45.2 %)
Simplex	199	503	0	303

Table 5.5: Results of the performance test for eight tested methods.

The local contact search procedure has to return a correct value, otherwise the calculation is terminated. Therefore a robust method for the local contact searching should be able to converge in every situation. This requirement is satisfied only for both the second order algorithms and the simplex method as it is indicated by the zero value of DNF `NITER` in Table 5.5. As far as the convergence rate is concerned, the torus approximation method is the fastest one. It overcomes the sphere approximation method more than three times. It is even faster than the Newton-Raphson or the BFGS method as it is apparent from AVRG CPU numbers in Table 5.5. On the other hand, the average CPU time of the simplex method is about fifteen times higher in comparison with the torus approximation method.

The results of the performance test for the remaining methods, namely the least square projection method, the steepest descent method and Broyden's method, are comparable or even worse than for the benchmark problem. They suffer from complete lack of robustness and efficiency.

Method	AVRG `NITER`	MAX `NITER`	DNF `NITER`	AVRG CPU [ms]
Steepest descent	20	193	0	253 (91.2 %)
Broyden	8	27	0	579 (97.2 %)
BFGS	8	27	0	54 (78.3 %)

Table 5.6: Results of the performance test for the line search methods utilizing Illinois algorithm.

In order to remedy the robustness of the line search method, the Illinois algorithm [110] as a promising alternative to the quadratic step length procedure was considered. The results are shown in Table 5.6, where only the methods employing the line search technique are listed. Zero values of the DNF `NITER` confirm the correctness of this modification. As far as the efficiency is concerned, the average CPU time substantially rises for the steepest descent method by 84.7% and the Broyden method by 31%, comparing AVRG CPU numbers in Table 5.5 and Table 5.6. The most CPU time is spent in the line search procedure. The best results were obtained using the BFGS method, where the average CPU is only 54 [ms].

Method	AVRG `NITER`	MAX `NITER`	DNF `NITER`	AVRG CPU [ms]
N-R with modified Hessian	7	101	0	40 (61.1 %)
N-R with steepest descent	20	193	0	253 (91.2 %)
N-R with simplex	8	640	0	32

Table 5.7: Results of the performance test for three variants of the improvement of the Newton-Raphson method.

For the purpose of the improvement of the Newton-Raphson scheme, a variation of this method with Hessian modification and advantageous coupling with the steepest descent method or the simplex method were considered. As the step length procedure the Illinois algorithm was used again. The results for these three variants are shown in Table 5.7. The idea of Hessian modification is based on its eigenvalue modification when negative eigenvalue is replaced by a small positive number to ensure that the Hessian matrix is sufficiently positive definite [33]. Similarly, in the case of the combination of methods, the steepest descent method or simplex method is performed when the Hessian matrix is not positive definite. In all cases, the Hessian matrix was considered to be not positive definite if one of the eigenvalues was less than 10^{-5}. The Newton-Raphson method with Hessian modification and coupling with the simplex method seem to be competitive with the torus approximation method as the most effective methods for local contact search.

Although the presented results of the benchmark problem including performance test show certain features of the methods, one should not draw relevant conclusions unless the methods are used in a real contact problem. Therefore, each of the methods were implemented to the local contact search procedure of FEA software PMD [174] and tested by means of numerical examples. The results are discussed in the following section.

5.3 Stable time step size estimation for bipenalty method

Unfortunately, there are no stability theorems for contact-impact problems [40]. In this case the linear stability theory can be applied carefully. In practice, for example, the stability may be preserved by checking the energy balance during a nonlinear computation. In Reference [175, p. 558] an upper bound for the stiffness penalty was derived. Moreover, it was shown that the stiffness penalty always decreases the stable time step. In this work, we generalize this estimate for the bipenalty approach [192, 202, 190] following the Belytschko approach [175].

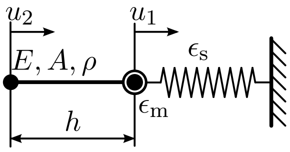


Figure 5.7: A simple dynamic system with two degrees-of-freedom — Signorini problem.

Let us consider a simple dynamic system, depicted in Figure (5.7), with two degrees-of-freedom. The system consists of one 1D constant strain truss element with lumped mass matrix. The active contact constraint is set in node 1. The aim is to determine the maximum eigenfrequency of this system to estimate the stable time step in the form (2.164). To this end, the eigenvalue problem can be formulated as

$$\frac{EA}{h}\begin{bmatrix} 1+\beta_{\mathrm{s}} & -1 \\ -1 & 1 \end{bmatrix}\mathbf{u} = \omega^2 \frac{\rho_0 A h}{2}\begin{bmatrix} 1+\beta_{\mathrm{m}} & 0 \\ 0 & 1 \end{bmatrix}\mathbf{u}, \tag{5.10}$$

where E is Young's modulus, A is the cross section area, h is the characteristic length of the element, $\mathbf{u}$ is the displacement vector which has the form $\mathbf{u} = [u_1, u_2]^{\mathrm{T}}$ and the dimensionless mass and stiffness penalty have been introduced as

$$\beta_{\mathrm{m}} = \frac{2}{\rho_0 A h}\epsilon_{\mathrm{m}}, \quad \beta_{\mathrm{s}} = \frac{h}{EA}\epsilon_{\mathrm{s}}. \tag{5.11}$$

Note that since a single body is in contact with a rigid obstacle, the body index, (i), is not necessary and is therefore omitted here, and also in the further similar cases. We define the maximum angular velocity of a separate linear finite element with the lumped mass matrix $\omega_{max} = 2c_0/h$, where c_0 marks the wave speed in a bar given by $c_0 = \sqrt{E/\rho_0}$. Corresponding critical time step size is given as

$$\Delta t_{\mathrm{c}} = \frac{2}{\omega_{\max}} = \frac{h}{c_0}. \tag{5.12}$$

The maximum eigenfrequency of the problem (5.10) is given by

$$\omega_{\max} = \frac{c_0}{h}\sqrt{1 + \frac{(1+\beta_{\mathrm{s}})}{(1+\beta_{\mathrm{m}})} + \sqrt{1 + \frac{(1+\beta_{\mathrm{s}})^2}{(1+\beta_{\mathrm{m}})^2} + \frac{2(1-\beta_{\mathrm{s}})}{(1+\beta_{\mathrm{m}})}}}. \tag{5.13}$$

Now, it is useful to introduce a new dimensionless penalty ratio r as

$$r = \frac{1}{2}\frac{\beta_{\mathrm{s}}}{\beta_{\mathrm{m}}} = \frac{h^2}{4c_0^2}R, \tag{5.14}$$

and the Courant dimensionless number C_{r} is defined as

$$C_{\mathrm{r}} = \frac{c_0 \Delta t}{h}. \tag{5.15}$$

It should be mentioned, the well-known CFL condition for the linear FE with lumped mass matrix in one-dimensional case reaches the value $C_{\mathrm{r}}^{\mathrm{crit}} = c_0 \Delta t_{\mathrm{c}}/h = 1$.

Substituting (5.13) into (5.12) using (5.15) the upper bound of the stable Courant number for the bipenalty method is obtained

$$\boxed{C_{\mathrm{r}} = \frac{c_0 \Delta t_c}{h} = \frac{2}{\sqrt{1 + \frac{(1+\beta_{\mathrm{s}})}{(1+\beta_{\mathrm{m}})} + \sqrt{1 + \frac{2(1-\beta_{\mathrm{s}})}{(1+\beta_{\mathrm{m}})} + \frac{(1+\beta_{\mathrm{s}})^2}{(1+\beta_{\mathrm{m}})^2}}}},} \tag{5.16}$$

with the property $\omega_{\max}|_{r=1} = 2$ and $C_{\mathrm{r}}|_{r=1} = 1$ independently of β_{s}. This is the main advantage of the bipenalty method.

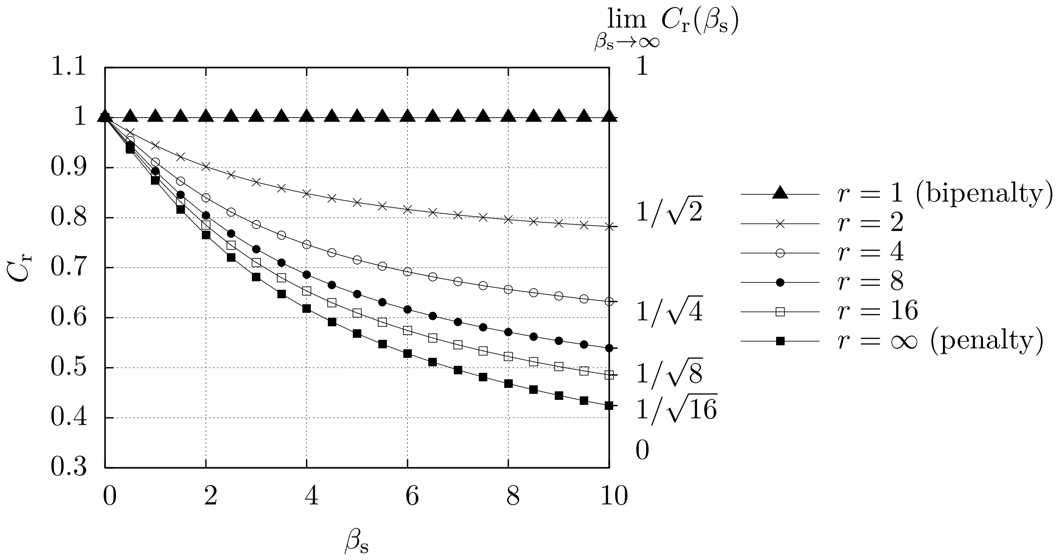


Figure 5.8: The dependence of the Courant number C_r on the dimensionless stiffness penalty β_s for selected dimensionless penalty ratios r.

The dependence of the Courant number C_r on the dimensionless stiffness penalty β_s is plotted in Figure 5.8, where the dimensionless penalty ratio r is employed as the parameter. The curve for $r \to \infty$ (i.e. $\beta_m \to 0$) corresponds to the standard stiffness penalty method. It illustrates the main disadvantages of the standard stiffness penalty method: the Courant number C_r rapidly decrease with increasing dimensionless stiffness penalty β_s. On the other hand, the curve for $r = 1$ confirms the existence of the CPR, for which the stable time step remains unchanged for an arbitrary value of the dimensionless stiffness penalty β_s. In addition, there are more curves in Figure 5.8 for dimensionless penalty ratios $r = 1, 2, 4, 8, 16$ and ∞. For each of them, there are limits of the Courant number for $\beta_s \to \infty$ on the right edge of the picture. Taking dimensionless ratios $r < 1$ does not decrease the critical time step below the level of the unconstrained system. It is clear that the bipenalty method with the penalty ratio equal to the CPR is superior over the standard stiffness penalty method.

Chapter 6

Numerical examples

The performance of the newly developed method will be demonstrated by means of several numerical examples involving static and dynamic problem. Both the 2D and the 3D cases are considered. Moreover, an engineering application is presented in Section 6.3, where the contact between the upper and lower part of the high pressure inner turbine casing is analyzed.

6.1 Static contact problems

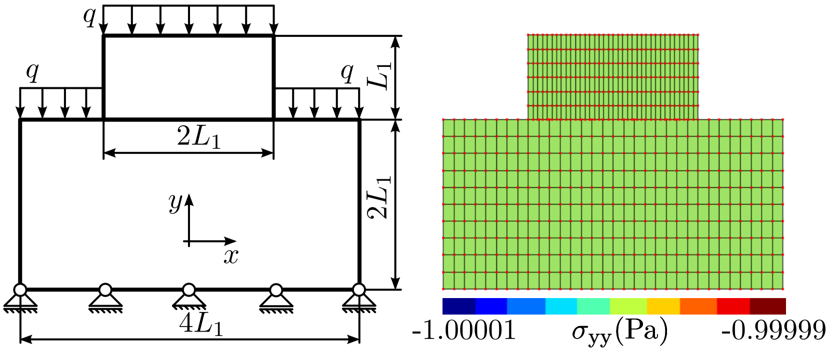
Seven static contact problems are solved in this section. Four two-dimensional and three three-dimensional. The friction is considered in two cases. Namely in the examples called: ”2D shallow ironing problem” and ”3D sliding of a cube on a rigid plate”.

6.1.1 2D contact patch test

In the first example, the contact patch test according to Taylor and Papadopoulos [56] is presented [193, 191]. The purpose of this example is to test the ability of contact discretization to accurately transfer constant pressure loads from one body to another. Two blocks of different size are pressed against each other. Dimensions are depicted in the Figure 6.1a, where $L_1 = 1\,\mathrm{m}$. Both blocks are subjected to a pressure $q = 1\,\mathrm{Pa}$. The same material with Poisson's ratio $\nu = 0.3$ and Young's modulus $E = 1000\,\mathrm{Pa}$ is used for both blocks. The plane strain state is considered. The analytical solution is $\sigma_{\mathrm{yy}} = -1\,\mathrm{Pa}$.

Either of the blocks is discretized by one bivariate NURBS patch of order $p^{(i)} = 1$ in each parametric dimension. The lower block is discretized by $n_1^{(1)} = 33$ times $n_2^{(1)} = 11$ control points and the upper block by $n_1^{(2)} = 37$ times $n_2^{(2)} = 7$ control points. Corresponding uniform knot vectors were considered.

In Figure 6.1b, there are contours of σ_{yy} stress component, which perfectly match the analytical solution. In fact, the maximum error of the numerical solution is in the order of machine precision. Note that the same results were obtained for bi-quadratic and bi-cubic NURBS patches.



(a) : Dimensions and boundary conditions.

(b) : Contour of stress component σ_{yy}.

Figure 6.1: 2D contact patch test.

6.1.2 2D Hertz-Signorini problem

In the second example, the classic Hertzian problem is investigated [191, 194]. In particular, the plane strain contact of cylinder with rigid plane is considered. The material parameters, geometry, and boundary conditions as well as IGA mesh parametrization are taken from the work of Temizer et al. [50, p. 1104].

The cylinder with the outer radius $L_1 = 1\,\mathrm{m}$ is made of a linear elastic material characterized by the Poisson's ratio $\nu = 0.3$ and Young's modulus $E = 1\,\mathrm{Pa}$. The vertical load per unit length of the cylinder is $q_1 = 10^{-3}\,\mathrm{Pa}$, so that the total vertical force is $F_\mathrm{c} = 2L_1 q_1 = 2 \times 10^{-3}\,\mathrm{N}$. In the work of Temizer et al. [50, p. 1105] one can find analytical solution for the contact area radius $r_\mathrm{c} = \sqrt{4F_\mathrm{c}L_1(1-\nu^2)/(\pi E)}$, and maximum contact pressure $p_\mathrm{c}^{\max} = 2F_\mathrm{c}/(\pi r_\mathrm{c})$. The contact pressure distribution is $p_\mathrm{c}(x) = p_\mathrm{c}^{\max}\sqrt{1-(x/r_\mathrm{c})^2}$. As in Reference [50], the cylinder is modelled with a non-zero inner radius. The corresponding vertical load on the inner radius is $q_2(\vartheta) = q_1 \cos(\vartheta)$.

The entries of the knot vector are located in such a way that approximately 75 % of the elements in the radial direction are distributed in the outer circular ring (see Figure 6.2a) of radius $L_3 = 0.1L_2$. Likewise, approximately 75 % of the elements in the angular direction are distributed in the region defined by the angle $\vartheta_1 = 0.1\pi/2$. Particularly, 48 linear elements along the radial direction and 81 linear elements along the angular direction are considered. The r-refinement is employed during the increasing the polynomial order of the basis functions, i.e. the control points are relocated but their number is kept constant.

For illustration, the contours of the normalized vertical stress component $\sigma^*_{\mathrm{yy}} = \sigma_{\mathrm{yy}}/p_\mathrm{c}^{\max}$ are showed in Figure 6.2b. The detail study of contact pressure distribution in dependence on the contact area radius, r_c, is presented in Figures 6.3 and 6.4. In these figures, the plots in the left column were obtained for the newly developed symmetry preserving GPTS isogeometric contact formulation, whereas the plots in the right column were obtained for the mortar isogeometric contact formulation according to Temizer et al. [50]. Rows of Figures 6.3 and 6.4 differs in the polynomial order, p, of NURBS basis functions. In each sub-plot of Figures 6.3 and 6.4, there is analytical solution, the contact pressure at Gauss points $p_{\mathrm{c}_g} = \epsilon_\mathrm{N} g_{\mathrm{N}_g}$, and a post-processed contact pressure denoted in legend as NURBS.

Here employed post-processing scheme for the contact pressure was originally proposed by

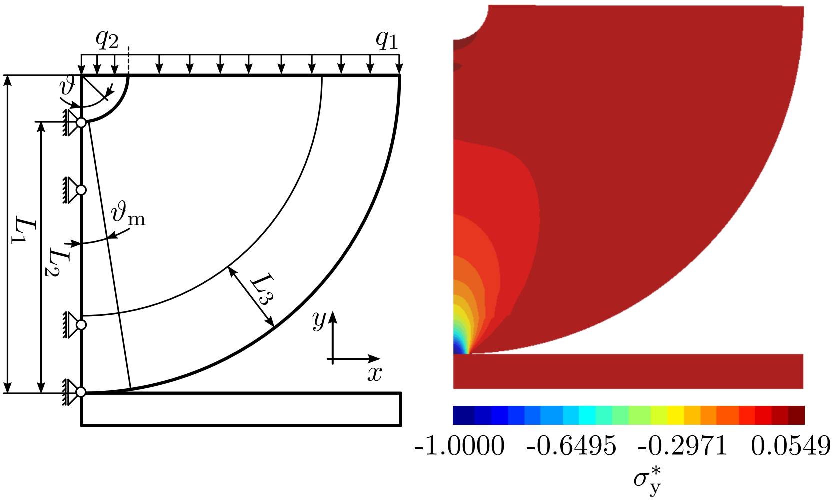
Sauer [132, p. 306]. The idea is to calculate control point contact pressures p_{c_A} to get post-processed contact pressure in the form

$$p_{\mathrm{c}}^{\mathrm{p}} = \sum_{A=1}^{n_{\mathrm{scp}}} R_A p_{\mathrm{c}_A}. \tag{6.1}$$

The calculation of the control point values of the contact pressure is inspirited by the mortar method [50]. In particular, control point pressures are evaluated as a weighted average

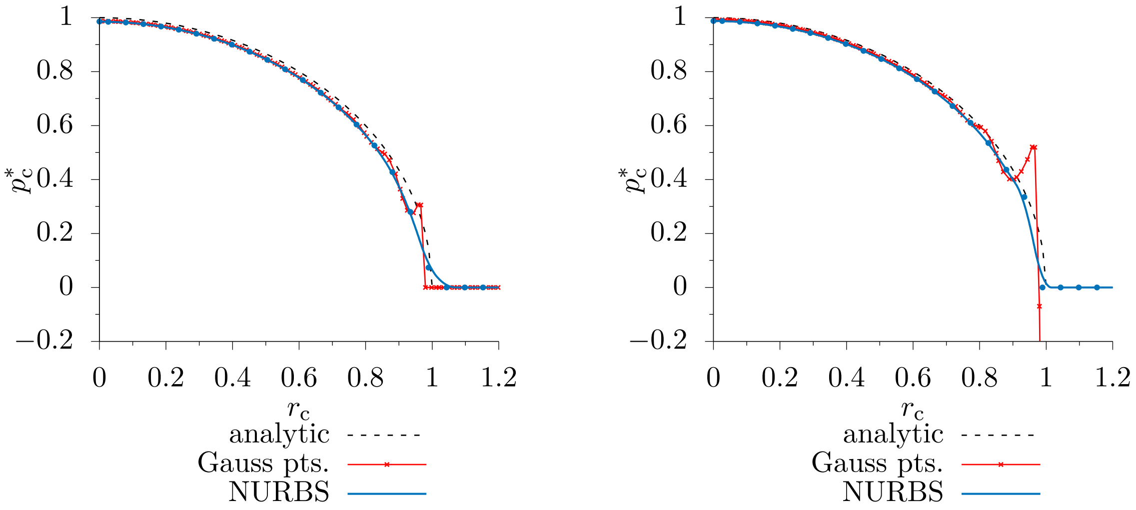
$$p_{\mathrm{c}_A} = \frac{\int_{\Gamma_\mathrm{c}} R_A \epsilon_\mathrm{N} \langle g_\mathrm{N} \rangle \, \mathrm{d}\Gamma}{\int_{\Gamma_\mathrm{c}} R_A \, \mathrm{d}\Gamma}. \tag{6.2}$$

One can see in Figures 6.3 and 6.4 that post-processed contact pressure shows very good agreement with the analytical solution. On the other hand, the contact pressure at Gauss points exhibits oscillations that increase with the polynomial order of the basis functions. This is the expected result because the GPTS formulation leads to the over-constrained system of equations. Note that calculation of the weighted average quantities is the bottom line of the mortar-KTS algorithm [50]. While Equation (6.2) was used only once in a post-processing solution, the mortar FE algorithm uses this formula in each iteration of the non-linear solution to obtain the correct number of contact constraints. It can be concluded that the oscillations are smaller for the newly developed algorithm than for the KTS-mortar algorithm.



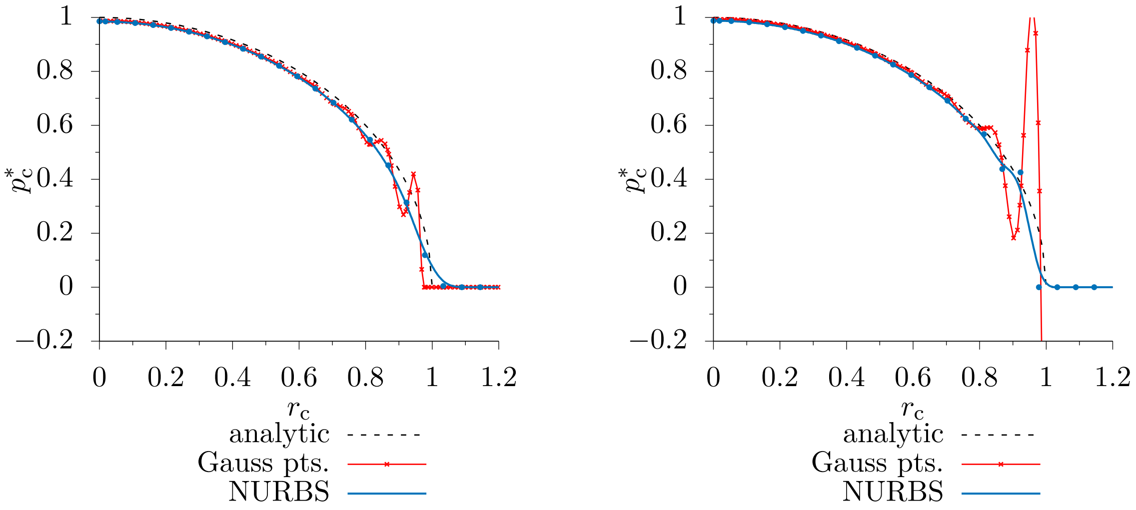
(a) : Parameters of the IGA mesh. **(b) :** Contour of stress component σ_y^*.

Figure 6.2: Hertzian contact problem.



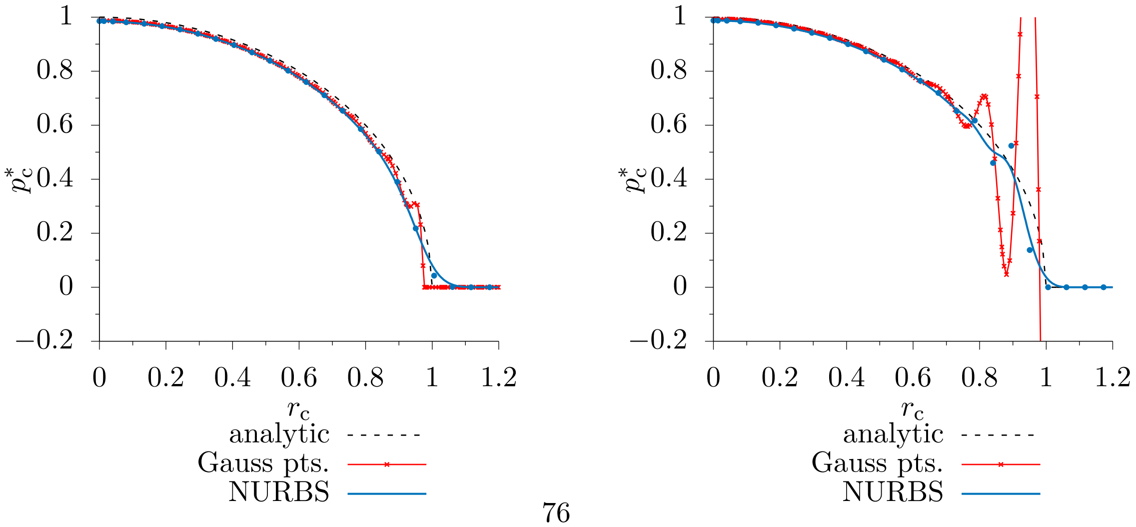
(a) : GPTS: $p = 2$, $n_{\text{gp}} = 4$.

(b) : Mortar-based: $p = 2$, $n_{\text{gp}} = 4$.



(c) : GPTS: $p = 3$, $n_{\text{gp}} = 6$.

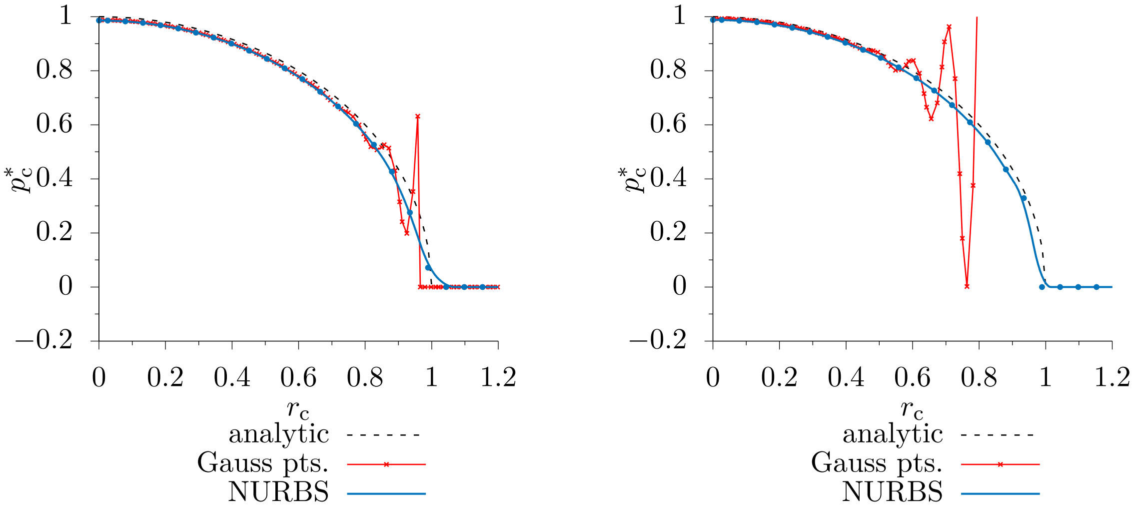
(d) : Mortar-based: $p = 3$, $n_{\text{gp}} = 6$.



(e) : GPTS: $p = 4$, $n_{\text{gp}} = 8$.

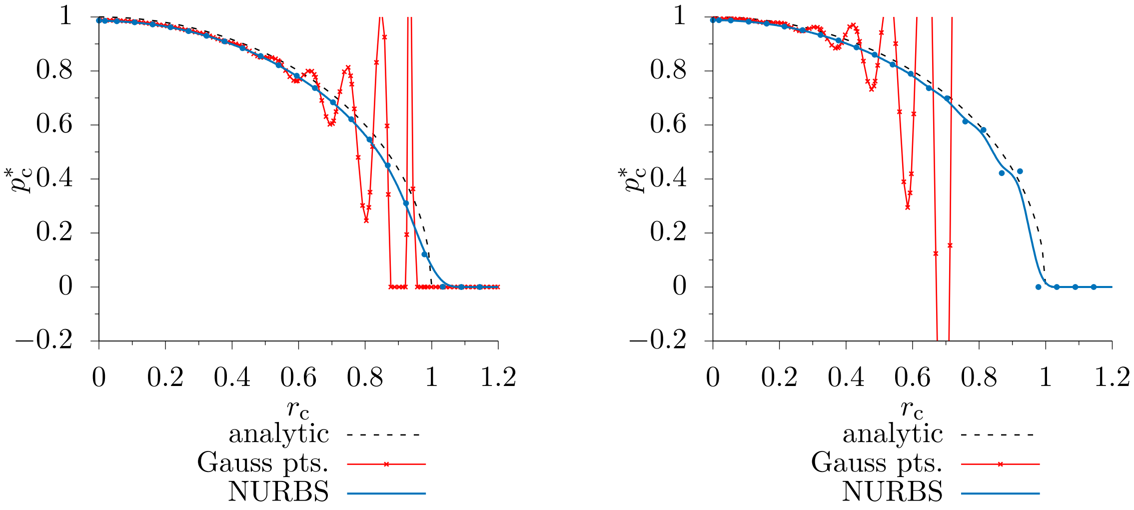
(f) : Mortar-based: $p = 4$, $n_{\text{gp}} = 8$.

Figure 6.3: Contact pressure distribution for $\epsilon_{\text{N}} = 10^4$.



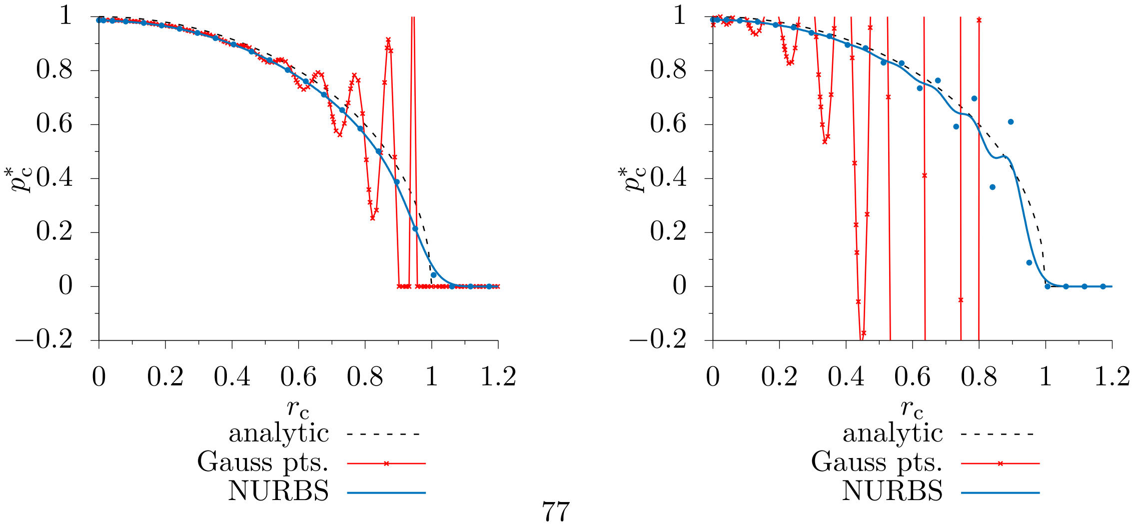
(a) : GPTS: $p = 2$, $n_{gp} = 4$.

(b) : Mortar-based: $p = 2$, $n_{gp} = 4$.



(c) : GPTS: $p = 3$, $n_{gp} = 6$.

(d) : Mortar-based: $p = 3$, $n_{gp} = 6$.



(e) : GPTS: $p = 4$, $n_{gp} = 8$.

(f) : Mortar-based: $p = 4$, $n_{gp} = 8$.

Figure 6.4: Contact pressure distribution for $\epsilon_N = 10^6$.

6.1.3 2D bending of two cantilever beams

This example was inspired by the problem solved in a[140, p. 37]. 2D plane stress cantilever beams, depicted in Figure 6.5, are placed in a way that one is just above the other so they do not touch but there is an initial gap between them $L_3 = 0.01\,\mathrm{m}$. The thickness of both beams is $L_2 = 2\,\mathrm{m}$. The length of the bottom beam is $L_1 = 28\,\mathrm{m}$. The length of the upper one is $L_1 + L_2 = 30\,\mathrm{m}$. The upper beam is subjected to the death load $q = 0.175\,\mathrm{Nm}^{-1}$. The Saint Venant-Kirchhoff material model [176, sec. 5.3.1] is considered for both beams with Young's modulus $E = 200\,\mathrm{Pa}$ and Poisson's ratio $\nu = 0.3$.

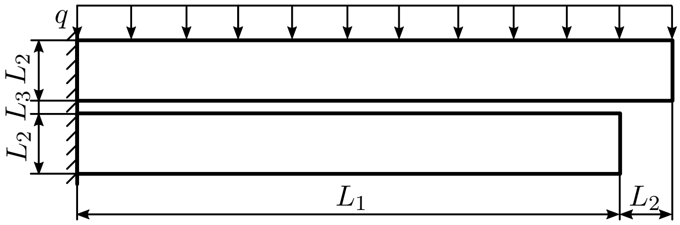


Figure 6.5: Bending of two cantilever beams with contact.

The beams are discretized by quadratic B-spline IGA elements. The upper beam consists of 15 elements and 51 control points and the lower one consists of 14 elements and 48 control points. The control net can be seen in Figure 6.6, where several deformed configurations are plotted. There is also final solution with contours of the HMH reduced stress. During the bending the upper beam slides (without friction) over the lower one. The penalty parameter was set to $\epsilon_{\mathrm{N}} = 100E\,\mathrm{Nm}^{-2}$ and the contact residual is integrated with 8 Gaussian quadrature points per each contact segment. The solution was obtained within 10 sub-increments indicating the good robustness of the newly proposed contact algorithm.

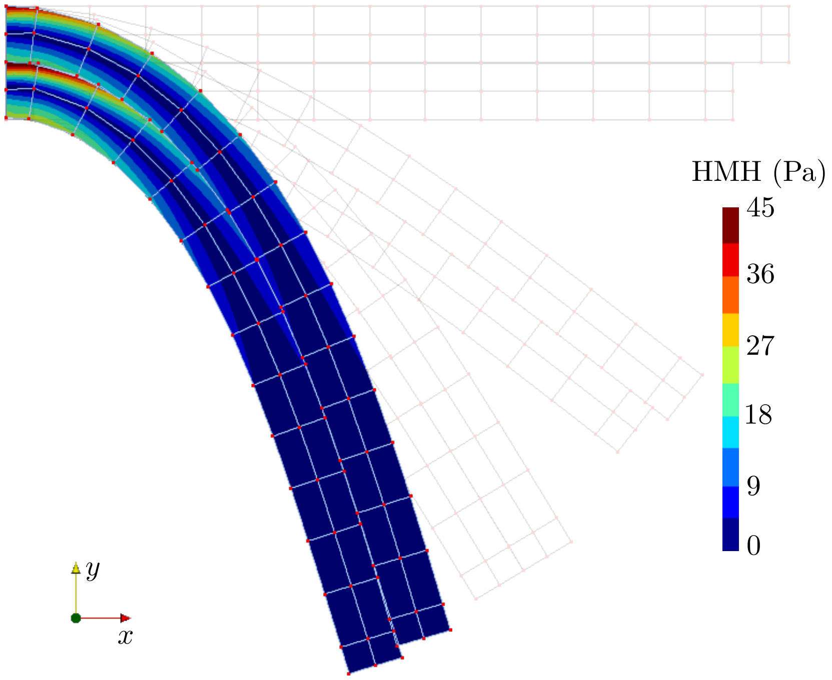


Figure 6.6: Several intermediate deformed configurations and final solution with HMH stress contours.

6.1.4 2D shallow ironing problem with friction

The shallow ironing problem becomes a popular benchmark to demonstrate the performance of a finite deformation frictional contact formulation [84, 177, 178]. It was shown by Fischer and Wriggers [84, p. 5035] that conventional NTS contact discretization fails for this problem.

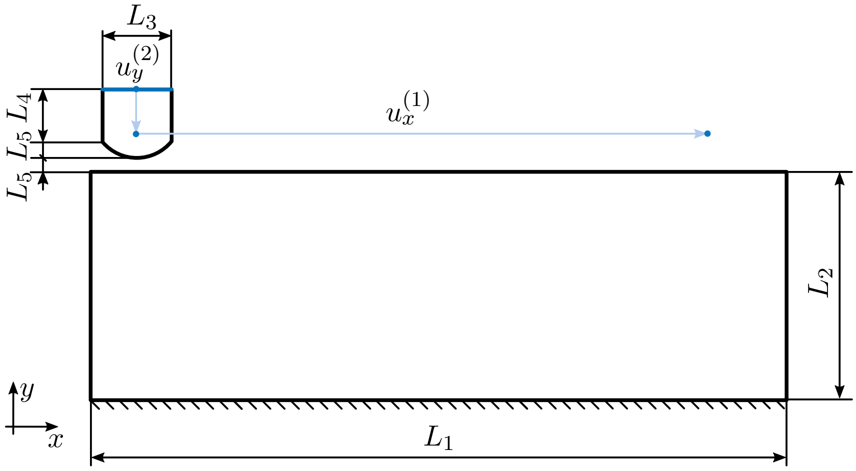
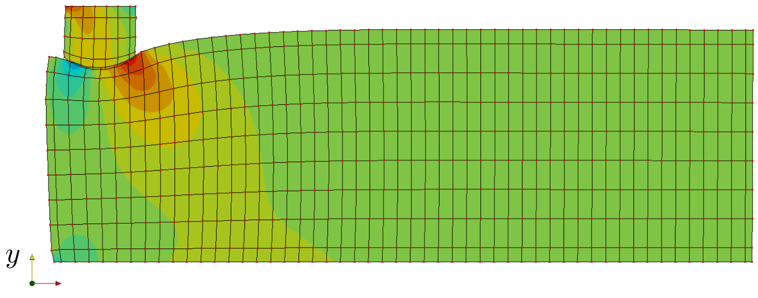
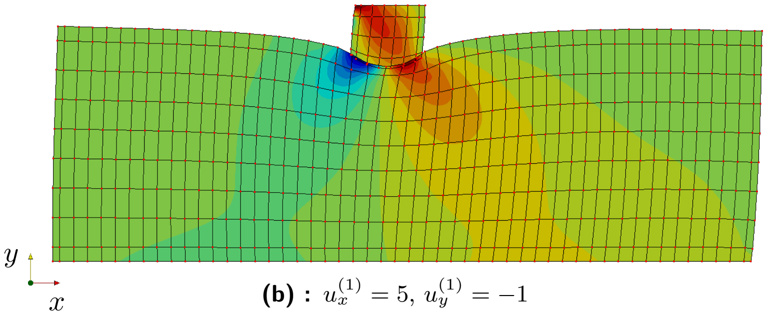


Figure 6.7: Geometry setting of the 2D shallow ironing problem.

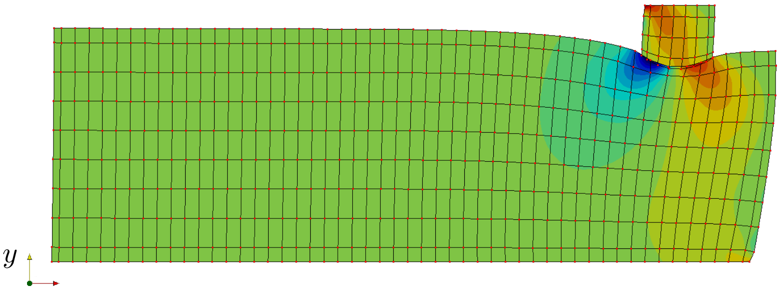
The geometry setting of the problem is shown in Figure 6.7, where the geometry parameters of the block are: $L_3 = 1.2\,\mathrm{m}$, $L_4 = 0.9\,\mathrm{m}$, $L_5 = 0.3\,\mathrm{m}$, and dimensions of the slab are $L_1 = 12\,\mathrm{m}$ and $L_2 = 4\,\mathrm{m}$. For both the block and the slab, the Neo-Hookean material is considered with Young's modulus $E = 68.96 \times 10^8\,\mathrm{Pa}$ and $E = 68.96 \times 10^7\,\mathrm{Pa}$, respectively. Both bodies has the same Poisson's ratio $\nu = 0.32$. The displacement-driven loading is applied on the upper edge of the block, which is depicted by the blue line in Figure 6.7. The displacement is prescribed in two steps. In the first step, the block is pressed against the slab with displacement $u_y^{(1)} = -0.05\,\mathrm{m}$. This movement is realized in 10 sub-increments from $t = 0\,\mathrm{s}$ to $t = 1\,\mathrm{s}$. After that, the block is forced to slip by prescribing the horizontal displacement $u_x^{(1)} = 10.0\,\mathrm{m}$ from time $t = 1\,\mathrm{s}$ to time $t = 2\,\mathrm{s}$ in 500 sub-increments. Both the block and the slab are discretized by quadratic NURBS elements. The slab consists of 50 by 8 elements in the horizontal and vertical direction, respectively. Similarly the block consists of 6 by 3 quadratic NURBS elements. The control net is apparent form the Figure 6.8. The slab is discretized by 520 and the block by 40 control points. The contact residual is integrated with 20 Gaussian quadrature points per each contact segment. The penalty parametrs was set to $\epsilon_\mathrm{N} = \epsilon_\mathrm{T} = 10^{10}\,\mathrm{Nm}^{-3}$ and the coefficient of friction was $\mu = 0.3$.



(a) : $u_x^{(1)} = 0$, $u_y^{(1)} = -1$



(b) : $u_x^{(1)} = 5$, $u_y^{(1)} = -1$



(c) : $u_x^{(1)} = 10$, $u_y^{(1)} = -1$

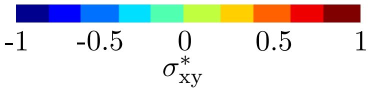


Figure 6.8: Contours of the shear stress component σ^*_{xy} for three selected time instances.

In Figure 6.8, there are three deformed configurations at time $t = 1\,\text{s}$ ($u_x^{(1)} = 0\,\text{m}$, $u_y^{(1)} - 1\,\text{m}$), $t = 1.5\,\text{s}$ ($u_x^{(1)} = 5\,\text{m}$, $u_y^{(1)} - 1\,\text{m}$), and $t = 2\,\text{s}$ ($u_x^{(1)} = 10\,\text{m}$, $u_y^{(1)} - 1\,\text{m}$). Each of the configurations is coloured by contours of normalized shear stress component σ^*_{xy}. One can notice the penetration of control nets. Its due to the fact that control points and control net of a NURBS curve do not interpolate the curve. Therefore, the real penetration caused by the low value of the penalty parameter is smaller than it appears.

In Figure 6.9, the time evolution of the normal and tangential components of the contact forces are plotted. The obtained results are smoother than those presented in [84, 177, 178]. In these references, although quadratic elements were used the contact force oscillates during the sliding over the edge between two adjacent elements. This is because of C^0 continuity of classic FEA. On the other hand, in the case of quadratic NURBS elements, the continuity is C^1 which results in the smoother contact forces.

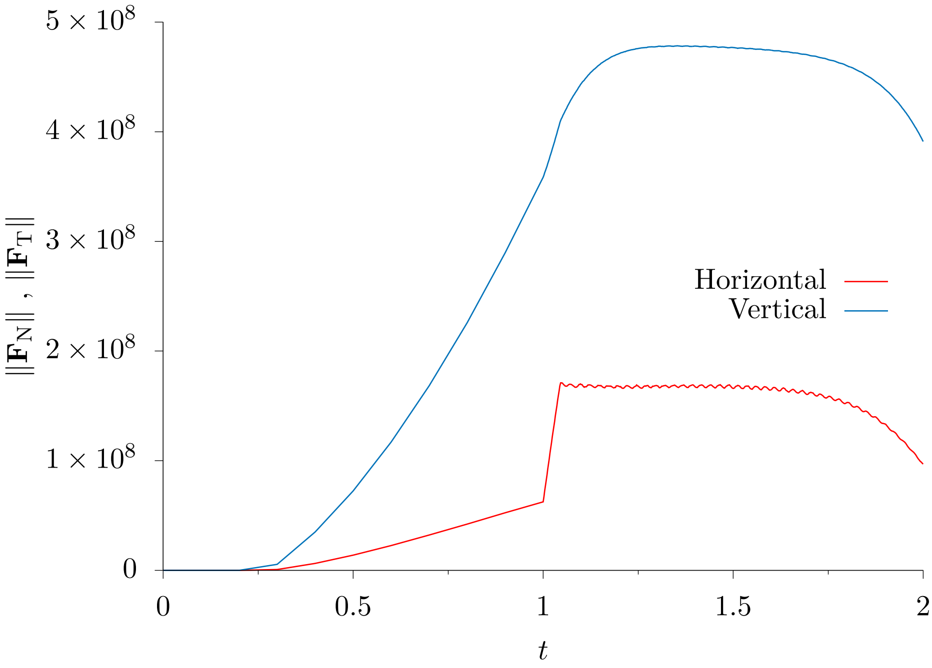


Figure 6.9: Magnitude of the normal and tangential components of the contact force.

6.1.5 3D frictional sliding of a cube on a rigid plane

As a representative example of the 3D friction problem was chosen an example according to Yastrebov [178, p. 292]. It consists of the frictional sliding of an elastic block on a rigid plate. The geometry is presented in Figure 6.10, where the size parameter L_1 is set to 1 m. The material parameters of the elastic block are Young's modulus $E = 210\,\mathrm{Pa}$ and Poisson's ratio $\nu = 0.3$. The displacement-driven loading is also denoted in Figure 6.10. The displacement of the upper side of the block is prescribed in two steps. In the first step the block is pressed against the rigid plate by moving $u_z^{(1)} = -0.05\,\mathrm{m}$ in the z-axis direction. In the second step the block is forced to slip by prescribing the horizontal displacement $u_x^{(1)} = 1/3\,\mathrm{m}$ in the x-axis direction. The vertical displacement is applied during 1 s and the horizontal displacement is performed in 2 s. Three coefficients of friction are considered: $\mu = 0.2$, 0.5, and 0.8.

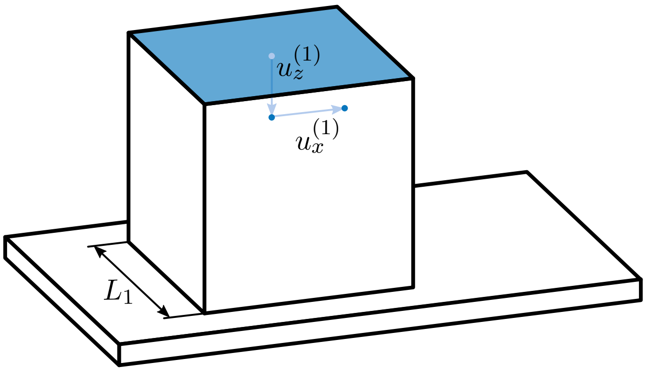


Figure 6.10: Geometry setting of the 3D frictional sliding of a cube on a rigid plane.

The elastic block is discretized by the trivariate quadratic B-spline basis functions and five isogeometric elements in each of the parametric directions. The knots in the knot vector are placed in a way that the linear parametrization is obtained. The control net can be seen from Figure 6.12. The total number of quadratic isogeometric elements is 125 and the total number of control points is 343.

For each of three considered frictional coefficients different number of load sub-increments is considered. For the frictional coefficient $\mu = 0.2$, the vertical displacement is applied in 12 and the horizontal displacement in 36 load sub-increments. For the frictional coefficient $\mu = 0.5$, the vertical displacement is applied in 10 and the horizontal displacement in 30 load sub-increments. And finally for the frictional coefficient $\mu = 0.8$, the vertical displacement is applied in 5 and the horizontal displacement in 25 load sub-increments.

The magnitude of the tangent reaction component is plotted in Figure 6.11 together with the normal reaction component scaled by the frictional coefficient. One can observe approaching of the two curves after switching to the slip state.

Deformed geometries and the corresponding contours of the shear stress σ_{xz} are presented in Figure 6.12 for three considered deformation states. Note that in accordance with Yastrebov's solution [178, p. 295], in the case of $\mu = 0.8$ contact detachment occurs.

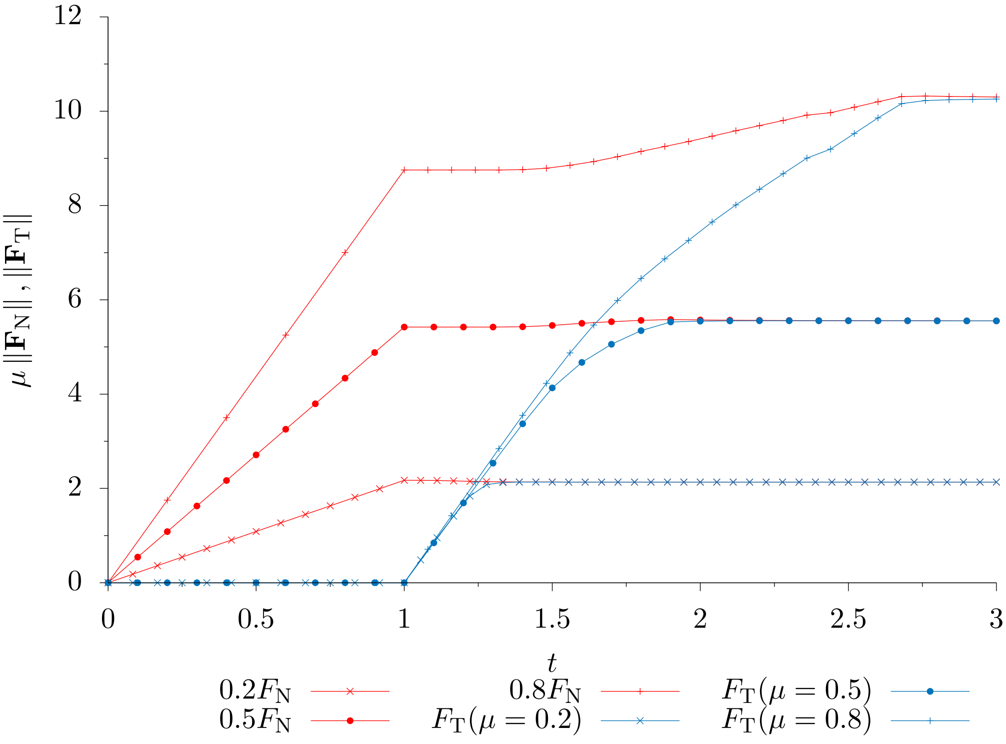
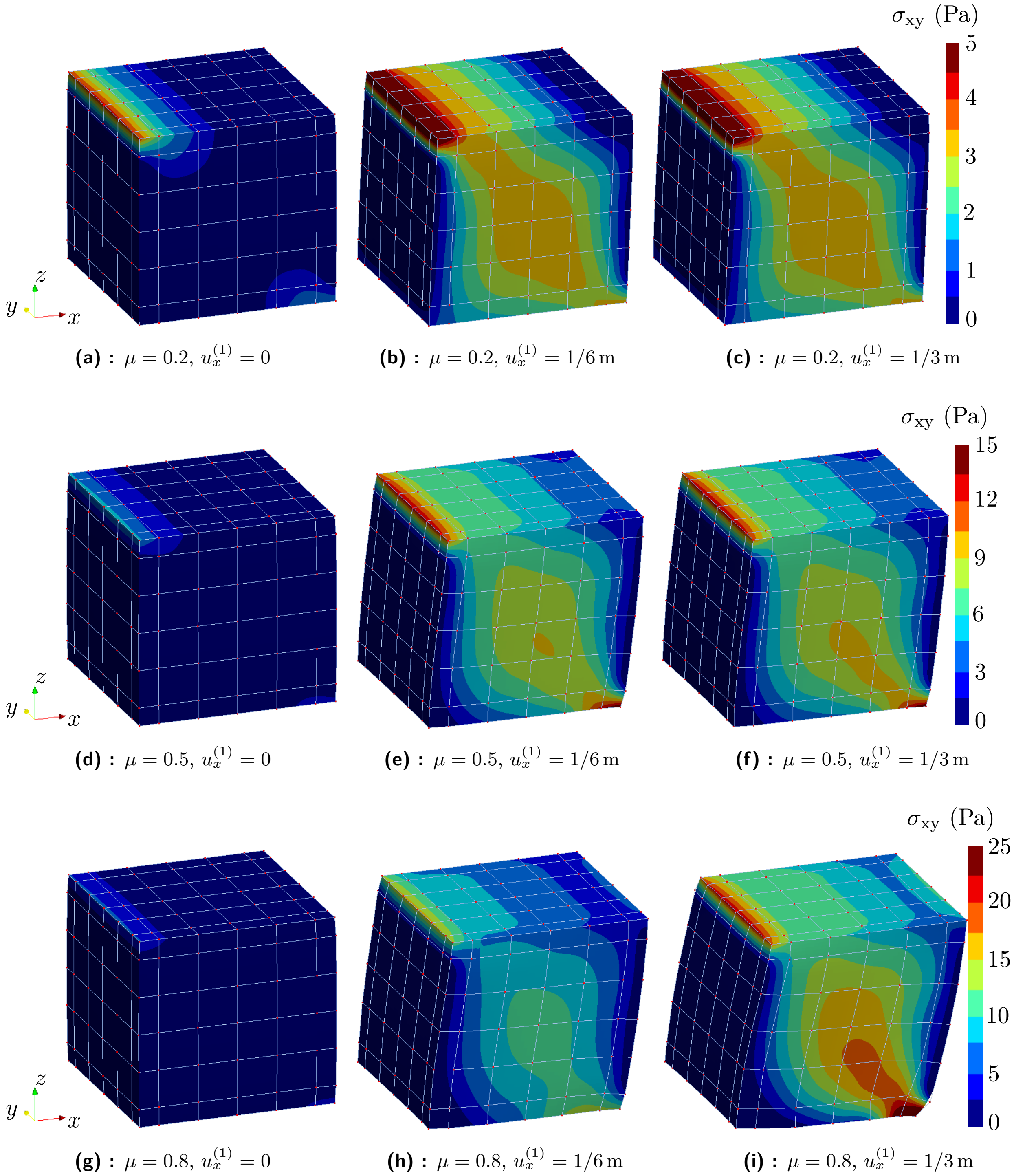


Figure 6.11: Evolution of the magnitude of the tangent contact reaction for three values of coefficient of friction ($\mu = 0.2$, $\mu = 0.5$, and $\mu = 0.8$).



(a) : $\mu = 0.2,\ u_x^{(1)} = 0$ **(b) :** $\mu = 0.2,\ u_x^{(1)} = 1/6\,\mathrm{m}$ **(c) :** $\mu = 0.2,\ u_x^{(1)} = 1/3\,\mathrm{m}$

(d) : $\mu = 0.5,\ u_x^{(1)} = 0$ **(e) :** $\mu = 0.5,\ u_x^{(1)} = 1/6\,\mathrm{m}$ **(f) :** $\mu = 0.5,\ u_x^{(1)} = 1/3\,\mathrm{m}$

(g) : $\mu = 0.8,\ u_x^{(1)} = 0$ **(h) :** $\mu = 0.8,\ u_x^{(1)} = 1/6\,\mathrm{m}$ **(i) :** $\mu = 0.8,\ u_x^{(1)} = 1/3\,\mathrm{m}$

Figure 6.12: Contours of the shear stress component σ_{xy} for three selected time instances.

6.1.6 3D bending of two rectangular plates over a cylinder

As was mentioned in Section 5.2.1, the benchmark problem arises from a numerical example that was originally introduced in [1] and resolved in [210, 189]. It involves the bending of two elastic rectangular plates over an elastic cylinder (see Figure 6.13). The radius of the cylinder is $L_4 = 0.4\,\mathrm{m}$, the dimensions of the plates are: length $L_1 = 2\,\mathrm{m}$, width $L_2 = 0.6\,\mathrm{m}$ and thickness $L_3 = 0.08\,\mathrm{m}$. The material properties are: Young's modulus $E = 2.1 \times 10^5\,\mathrm{MPa}$ and Poisson's ratio $\nu = 0.36$. The plates were loaded with uniformly distributed surface traction $q = 22.5\,\mathrm{MPa}$. Due to the triple, symmetry only one-eighth was modelled using 1168 20-node hexahedral classic FEA quadratic serendipity elements. Since the applied loading caused a significant deflection of the plate, the effect of large displacements and rotations in total Lagrangian formulation was considered.

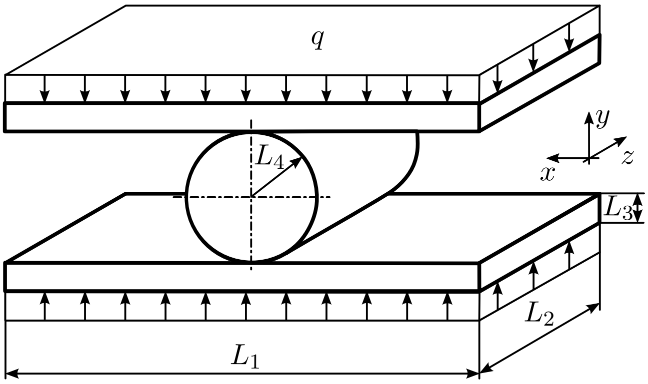


Figure 6.13: Bending of two rectangular plates over a cylinder.

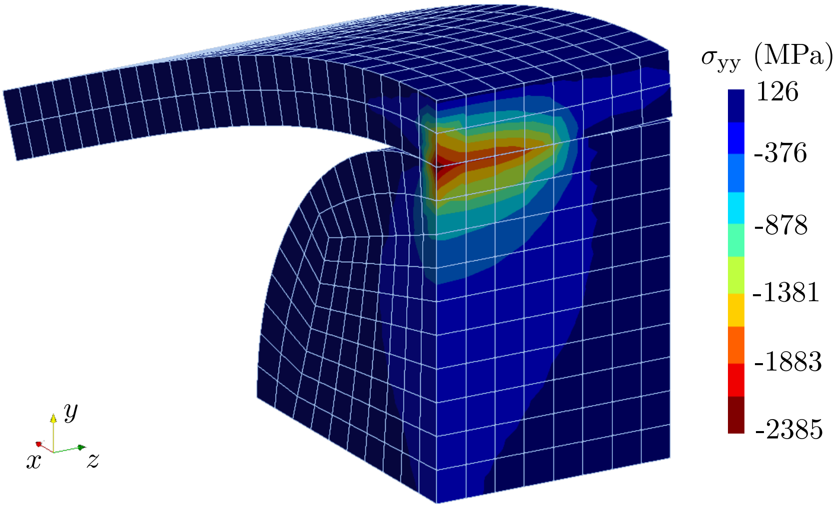
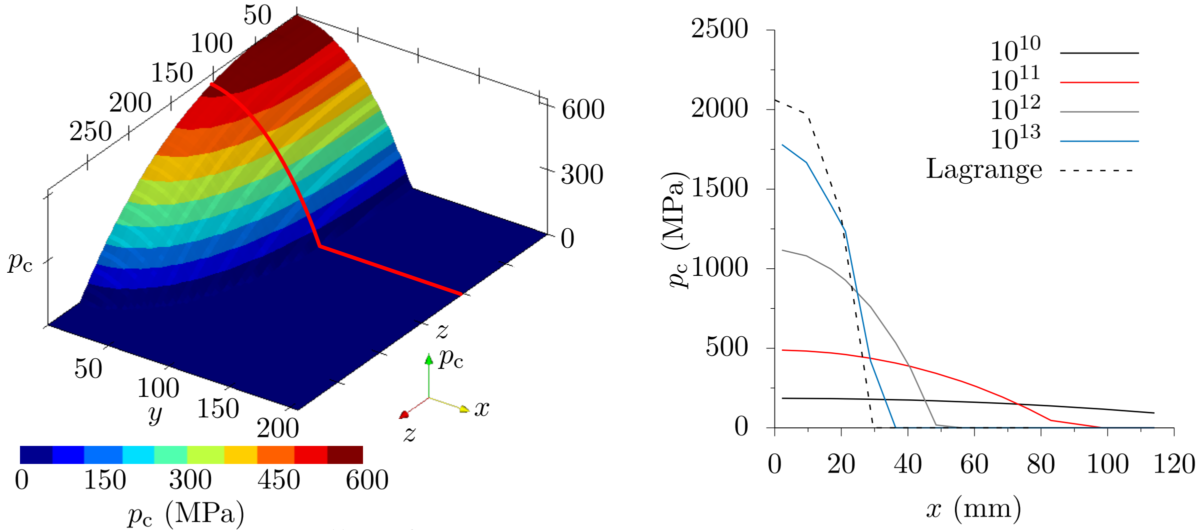


Figure 6.14: Distribution of σ_{yy}-stress colour contours in deformed configuration of the model.

The numerical solution obtained by a penalty-based contact algorithm [1] is illustrated in Figure 6.14. In this figure the distribution of σ_{yy} stress colour contours in the deformed configuration of the model for the penalty parameter $\epsilon_{\mathrm{N}} = 10^{13}\,\mathrm{Nm}^{-3}$ is plotted. The effect known from the theory of beams on elastic foundation can be observed when the plate disengages from the cylinder in the vicinity of the edges.



(a) : $\epsilon_N = 10^{11}\,\mathrm{Nm}^{-3}$ **(b):**

Figure 6.15: Contact pressure distribution on the plate.

Since the analytical solution is not available, the ANSYS solution by the Lagrange multipliers method was taken as a reference solution. The mortar segment-to-segment contact formulation [81] with symmetrically treated contact surfaces (two-pass) was applied. In Figure 6.15a, the distribution of the contact pressure p_c on the plate is plotted for the penalty parameter $\epsilon_N = 10^{11}\,\mathrm{Nm}^{-3}$. For a chosen section $z = 102.07\,\mathrm{mm}$ drawn in Figure 6.15a by the red curve, the distribution of the contact pressure evaluated at the surface Gaussian points is drawn in Figure 6.15b. It can be observed that the numerical solution obtained by the penalty contact algorithm [1] gradually drifts with increasing value of the penalty parameter ϵ_N to the reference calculation denoted as Lagrange in the legend of Figure 6.15b.

Finally, attention was focused on the crucial assessment of different methods in the local contact search procedure. A series of calculations was performed with the aim to monitor the effectiveness of all methods. Note that the Illinois algorithm was used as the step length procedure for the line search methods. The results are summarized in Table 6.1 for different values of penalty parameters ϵ_N. The cost of analysis was measured by the number of iterations `NITER` and total CPU time. In order to achieve convergence for higher values of the penalty parameter, it was necessary to introduce the load subincrementation controlled by user-defined number `NSUBI`. In Table 6.1, parameter `NSUBI` indicates a minimum value of the load subincrement for which the numerical solution was obtained. If a method did not converge even for `NSUBI` = 1000, the calculation was considered to be divergent.

It is clear that increasing the values of the penalty parameter ϵ_N places higher requirements on the robustness of the local contact search procedure. Table 6.1 shows that only the simplex method and the combination of the Newton-Raphson method with the simplex method converge to the solution even for the penalty parameter $\epsilon_N = 10^{14}\,\mathrm{Nm}^{-3}$. Furthermore, both of these methods need only a small amount of load subincrementation in contrast to other methods with the exception of the BFGS, the Newton-Raphson with modified Hessian and the torus approximation method.

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	5
10^{11}	1	15	9
10^{12}	2	46	32
10^{13}	8	232	253
10^{14}		***	

(a) : Newton-Raphson method

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	5
10^{11}	2	24	16
10^{12}	10	151	101
10^{13}	64	1442	1090
10^{14}		***	

(b) : Least square projection method

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	5
10^{11}	1	16	11
10^{12}	1	29	23
10^{13}	4	189	172
10^{14}		***	

(c) : Sphere approximation method

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	5
10^{11}	1	14	9
10^{12}	1	23	24
10^{13}	3	42	56
10^{14}		***	

(d) : Torus approximation method

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	6
10^{11}	2	24	16
10^{12}	10	151	108
10^{13}	69	1715	1291
10^{14}		***	

(e) : Steepest descent method

ϵ_N (Nm^{-3})	NSUBI	NITER	CPU time (s)
10^{10}	1	9	5
10^{11}	1	15	9
10^{12}	3	63	43
10^{13}	82	1676	1270
10^{14}		***	

(f) : Broyden's method

ϵ_N [N/m^3]	NSUBI	NITER	CPU time [s]
10^{10}	1	9	5
10^{11}	1	15	9
10^{12}	1	26	19
10^{13}	2	90	127
10^{14}		***	

(g) : BFGS method

ϵ_N [N/m^3]	NSUBI	NITER	CPU time [s]
10^{10}	1	9	6
10^{11}	1	15	11
10^{12}	1	26	23
10^{13}	2	90	156
10^{14}	7	1151	3345

(h) : Simplex method

ϵ_N [N/m^3]	NSUBI	NITER	CPU time [s]
10^{10}	1	9	6
10^{11}	1	15	11
10^{12}	1	25	23
10^{13}	2	88	143
10^{14}	7	1458	2955

(i) : Newton-Raphson + Simplex method

ϵ_N [N/m^3]	NSUBI	NITER	CPU time [s]
10^{10}	1	9	5
10^{11}	1	15	10
10^{12}	1	26	35
10^{13}	3	127	207
10^{14}		***	

(j) : Newton-Raphson with modified Hessian

Table 6.1: Effectiveness of methods in local contact search.

As far as the influence of the local contact search procedure on the convergence rate of the contact algorithm is concerned, the torus approximation method and the quasi-Newton BFGS method remain the fastest ones. The combination of the Newton-Raphson method with the simplex method is slightly slower. For lower penalty values the method preserves the quadratic convergence properties of the pure Newton-Raphson method. On the other hand, for higher penalty parameters, the simplex modification helps to increase the robustness of the local contact search and hence the effectiveness of the complete contact algorithm. The convergence rate of remaining methods is significantly worse.

6.1.7 3D ironing problem

In the last static numerical example, the finite deformation contact of a cylindrical elastic die with a rubber block is analyzed. This problem was first formulated in the work of Puso and Laursen [44, p. 614] and similar investigations were made by Popp et al. [38, p. 1448].

The Poisson's ratio and Young's modulus of the elastic die are $E = 1000\,\mathrm{Pa}$, $\nu = 0.3$ and of the rubber block $E = 1\,\mathrm{Pa}$, $\nu = 0.3$. The conventional incompressible neo-Hookean material model is considered [179, p. 183]. The dimensions of the block are $L_1 = 9\,\mathrm{m}$, $L_2 = 4\,\mathrm{m}$, and $L_3 = 3\,\mathrm{m}$. The cylindrical die has outer radius $L_4 = 3\,\mathrm{m}$, width $L_5 = 5.2\,\mathrm{m}$, and wall thickness $L_6 = 0.2\,\mathrm{m}$. Note that only 90 deg sector of the cylinder is considered, in order to represent the die only by single patch.

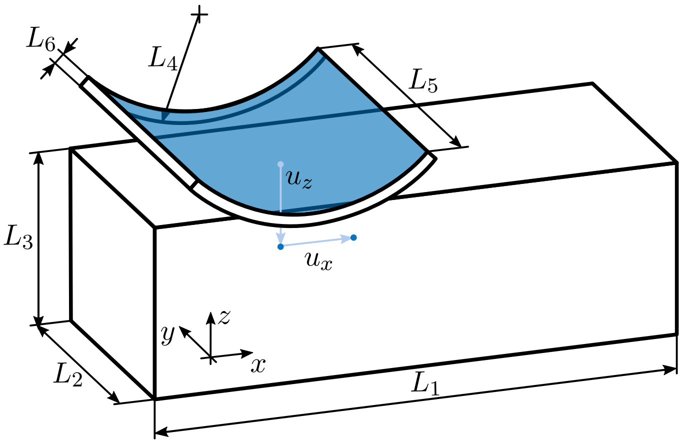
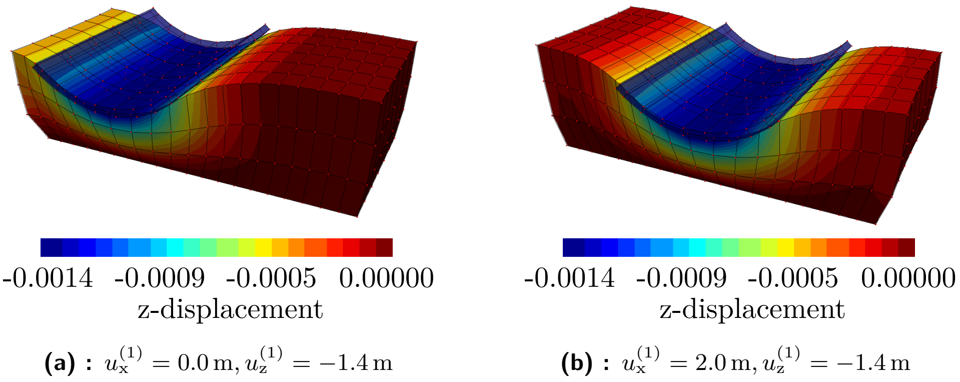


Figure 6.16: Geometry setting of the 3D ironing problem.

Both the die and the block are discretized by one trivariate NURBS patch of first order in each parametric dimension. The exception is the rounded direction where second order is considered in order to exactly describe the geometry. The die consists of $n_1^{(1)} = 10$, $n_2^{(1)} = 6$, and $n_3^{(1)} = 2$ control points in each parametric dimension. Similarly the block is discretized by $n_1^{(2)} = 21$, $n_2^{(2)} = 7$, and $n_3^{(2)} = 4$ control points. Uniform knot vectors are considered.

The initial position of the die centre is $2.5\,\mathrm{m}$ from the left edge of the block and $3.0\,\mathrm{m}$ from the top surface of the block. The die is first pressed into the block by prescribing a vertical displacement $u_\mathrm{z}^{(1)} = -1.4\,\mathrm{m}$. This stage is recorded in Figure 6.17a. Then it slides along the block until a horizontal displacement $u_\mathrm{x}^{(1)} = 4.0\,\mathrm{m}$ in the x-direction is reached. The moment when the die is in the middle of the block is illustrated in Figure 6.17b. It can be concluded that presented contact formulation is capable to solve finite sliding contact problems.



(a) : $u_{\mathrm{x}}^{(1)} = 0.0\,\mathrm{m}, u_{\mathrm{z}}^{(1)} = -1.4\,\mathrm{m}$ **(b) :** $u_{\mathrm{x}}^{(1)} = 2.0\,\mathrm{m}, u_{\mathrm{z}}^{(1)} = -1.4\,\mathrm{m}$

Figure 6.17: Two characteristic stages of deformation of the ironing problem.

6.2 Dynamic contact problems

Two 1D impact problems are presented in this section to demonstrate the critical time-step estimate derived in Section 5.3. Next, the study of influence of mass matrix lumping is performed by means of 2D dynamic Hertz problem in Section 6.2.3. Finally, the 3D impact of two tubes in Section 6.2.4 demonstrates the performance of the new local contact search procedure on the contact-impact problem.

6.2.1 1D impact of a bar with a rigid obstacle

In this section, the stability of explicit contact-impact algorithm using bipenalty technique was studied on the dynamic Signorini problem, which was represented by the motion of a bar that comes into contact with a rigid obstacle (see Figure 6.18). The bar of length $L_1 = 1\,\mathrm{m}$ with the initial velocity $V_{0x}^{(1)} = -1\,\mathrm{m}\cdot\mathrm{s}^{-1}$ is situated at distance of $g_{\mathrm{N}}^{0} = 0\,\mathrm{m}$ in front of the obstacle. The area of the bar section $A\,(\mathrm{m}^2)$, Young's modulus $E\,(\mathrm{MPa})$ and density $\rho_0\,(\mathrm{kg}\cdot\mathrm{m}^{-3})$ were chosen to be unit.

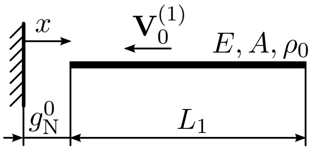


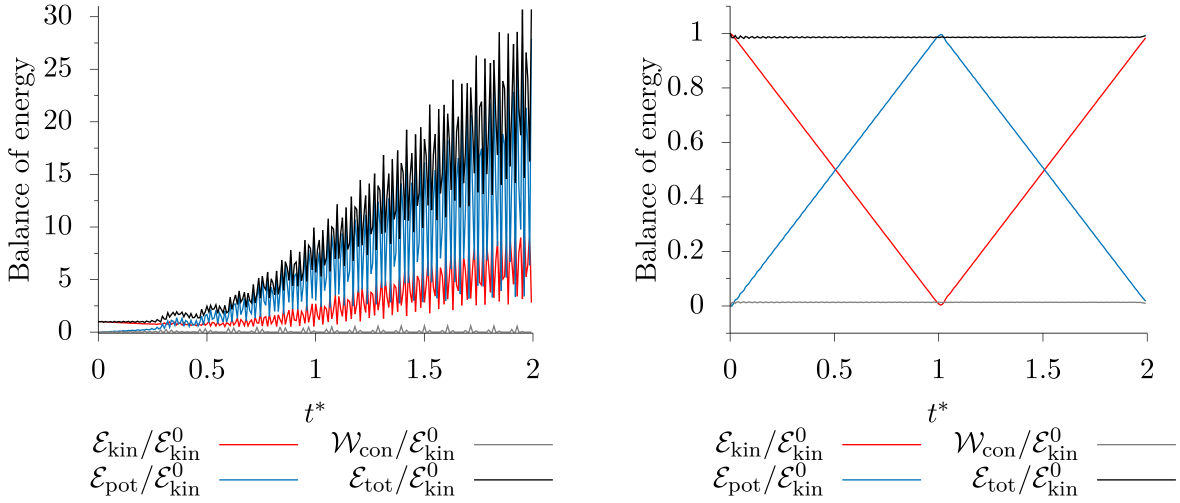
Figure 6.18: 1D dynamic Signorini problem.

The bar was discretized by a regular finite element mesh containing one hundred 1D constant strain truss elements. For the effective integration of equilibrium equations by the CDM the consistent mass matrix was diagonalized by the row sum technique. The maximum eigenfrequency was $\omega_{\max} = 200\,\mathrm{s}^{-1}$.

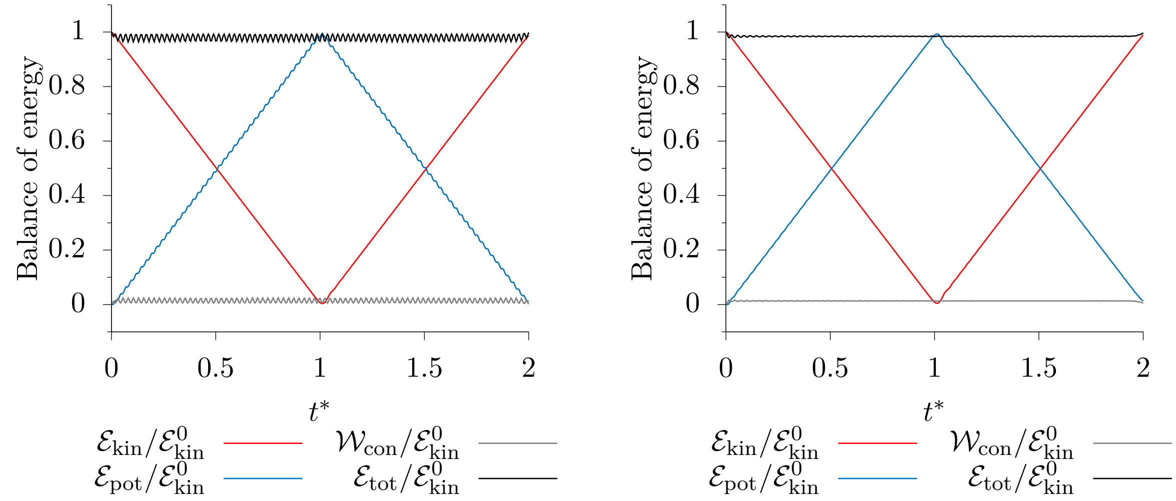
Let us introduce following dimensionless quantities

$$t^* = \frac{c_0 t}{L_1}, \qquad x^* = \frac{x}{L_1}, \qquad u^* = \frac{u|_{x=0}}{L_1}, \qquad F_{\mathrm{c}}^* = \frac{c_0 F_{\mathrm{c}}}{V_{0x} E A}, \qquad \sigma^* = \frac{\sigma A}{F_{\mathrm{c}}}, \tag{6.3}$$

where $t^*, x^*, u^*, F_\mathrm{c}^*, \sigma^*$ are the dimensionless time, coordinate, contact displacement, contact force, stress, respectively. In the following figures, the results for the standard penalty method (left) and the bipenalty method (right) are plotted.



(a) : $C_\mathrm{r} = 0.82$; $r = \infty$ (penalty) **(b) :** $C_\mathrm{r} = 0.82$; $r = 1$ (bipenalty)

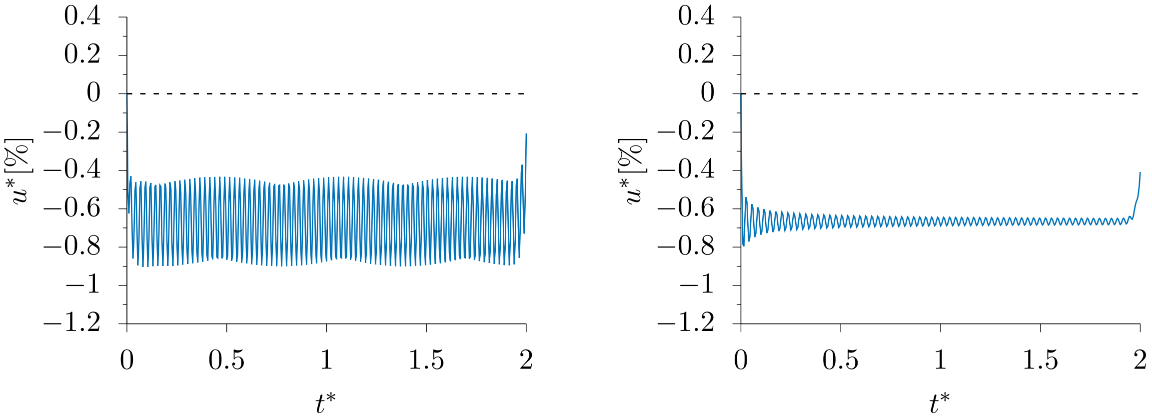


(c) : $C_\mathrm{r} = 0.5$; $r = \infty$ (penalty) **(d) :** $C_\mathrm{r} = 0.5$; $r = 1$ (bipenalty)

Figure 6.19: Time distribution of the balance of energy for $\beta_\mathrm{s} = 1.5$.

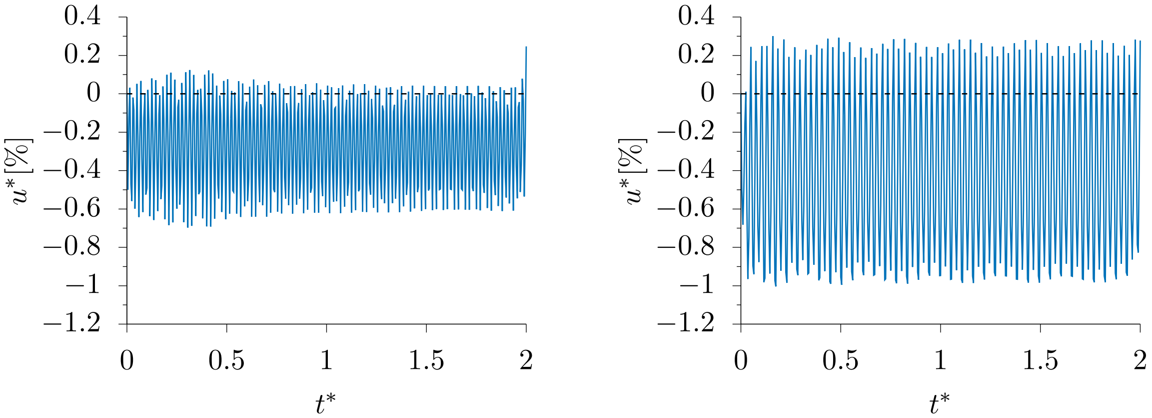
The dimensionless stiffness penalty β_s was chosen equal to 1.5. In order to verify derived formula of the stability (5.16) the Courant number C_r was set to 0.82, which was slightly higher than the critical value $C_\mathrm{r} = 0.81649658$ for the penalty method. The results are shown in Figure 6.19, where time distributions of the kinetic energy, $\mathcal{E}_\mathrm{kin}$, the potential energy, $\mathcal{E}_\mathrm{pot}$, the total energy, $\mathcal{E}_\mathrm{tot}$, and the work done by contact forces, $\mathcal{W}_\mathrm{con}$, are plotted. It was confirmed that the stability of the CDM was lost for the penalty method (see Figure 6.19a), whereas the solution obtained by the bipenalty method still perfectly conserved the total energy, as is shown in Figure 6.19b. When the Courant number C_r was set to 0.5 both methods were stable (see Figures 6.19c and 6.19d).

Note that the work of contact force is almost zero. In fact, it should be exactly zero because the displacement of the contact force was restricted by the rigid obstacle. However, in the penalty-like methods, contact forces perform a spurious work on penetrations. This work converges to zero as β_s tends to infinity. Nevertheless, a finite value of the stiffness penalty parameter always results in a non-zero work of contact force. One can also notice the presence of oscillations in the distributions of the potential energy and the work of contact forces, which result in oscillations in the distributions of the total energy. This phenomenon is primarily caused by the oscillations in the gap function, which will be discussed further.



(a) : $\beta_\mathrm{s} = 1.5$; $r = \infty$ (penalty)

(b) : $\beta_\mathrm{s} = 1.5$; $r = 1$ (bipenalty)



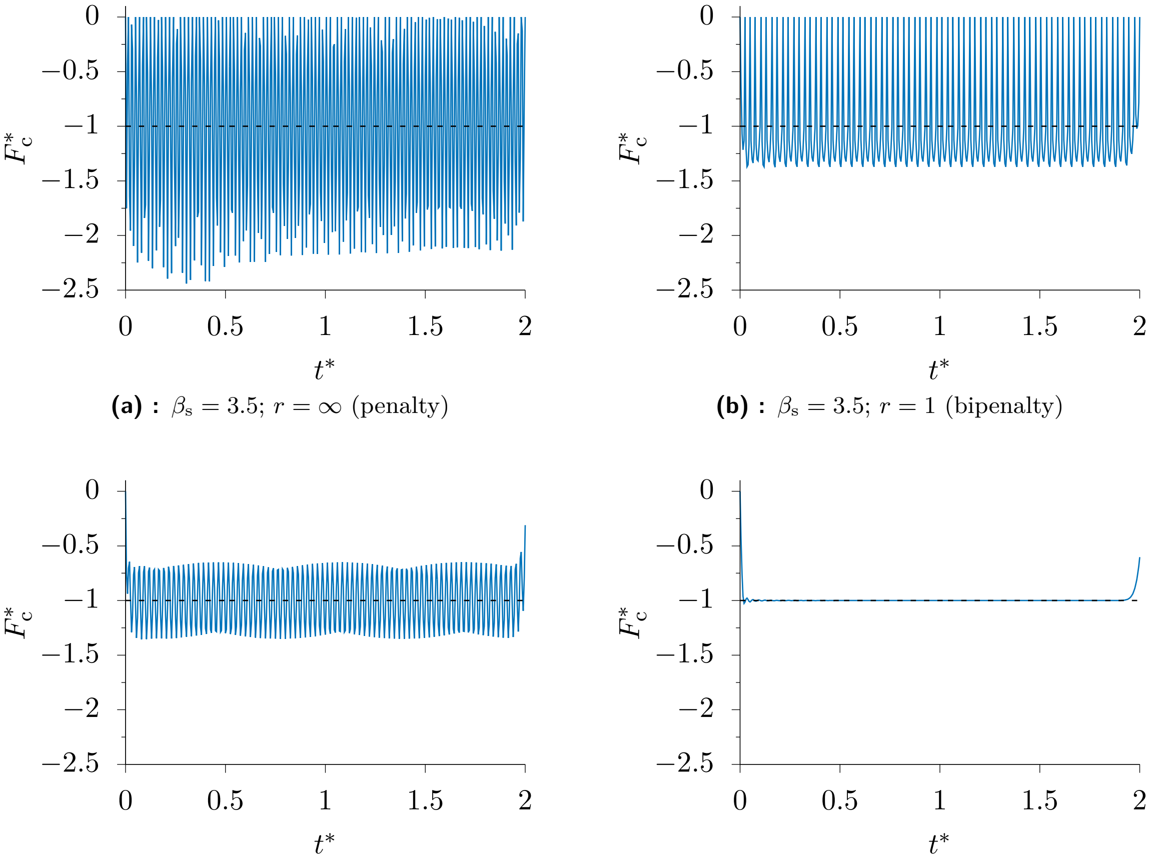
(c) : $\beta_\mathrm{s} = 3.5$; $r = \infty$ (penalty)

(d) : $\beta_\mathrm{s} = 3.5$; $r = 1$ (bipenalty)

Figure 6.20: Time distribution of the dimensionless contact displacement for $C_\mathrm{r} = 0.5$.

Figure 6.20 shows time distribution of the dimensionless contact displacement for $\beta_\mathrm{s} = 1.5$ and $C_\mathrm{r} = 0.5$. The gap should be equal to zero during the impact, which is indicated by the dashed line in the Figure 6.20. It is well known that penalty-like methods allow certain penetration of contact interfaces. As a result, the oscillations of kinematic and stress quantities can occur in impact problems. Figure 6.20b displays an attenuation of the oscillations for the bipenalty approach in comparison with the penalty method in Figure 6.20a. However, from a certain value of the dimensionless stiffness penalty β_s it was observed that the amplitude of oscillations were even higher for the bipenalty method than for the penalty method. An example is shown in Figures 6.20c and 6.20c, where $\beta_s = 3.5$ was considered. The reason probably is that the oscillation of the contact displacement overshot zero value. Thus, the contact constraint was deactivated and the contact force disappeared. In consequent iterations, the contact constraint was again activated. Therefore, the system was switching between two states which generated the oscillations.

This phenomenon can also be observed in Figures 6.21a and 6.21b, where time distribution of the dimensionless contact force is plotted. Both distributions are bounded by zero value. On the other hand, for the previously chosen value of dimensionless stiffness penalty $\beta_s = 1.5$ time dependence of of the dimensionless contact force oscillated around the exact solution as indicated in Figures 6.21c and 6.21d. Similarly to distribution of the contact displacement in Figure 6.20b the bipenalty method dumped oscillations in the distribution of contact force depicted in Figure 6.21d.



(a) : $\beta_s = 3.5$; $r = \infty$ (penalty)

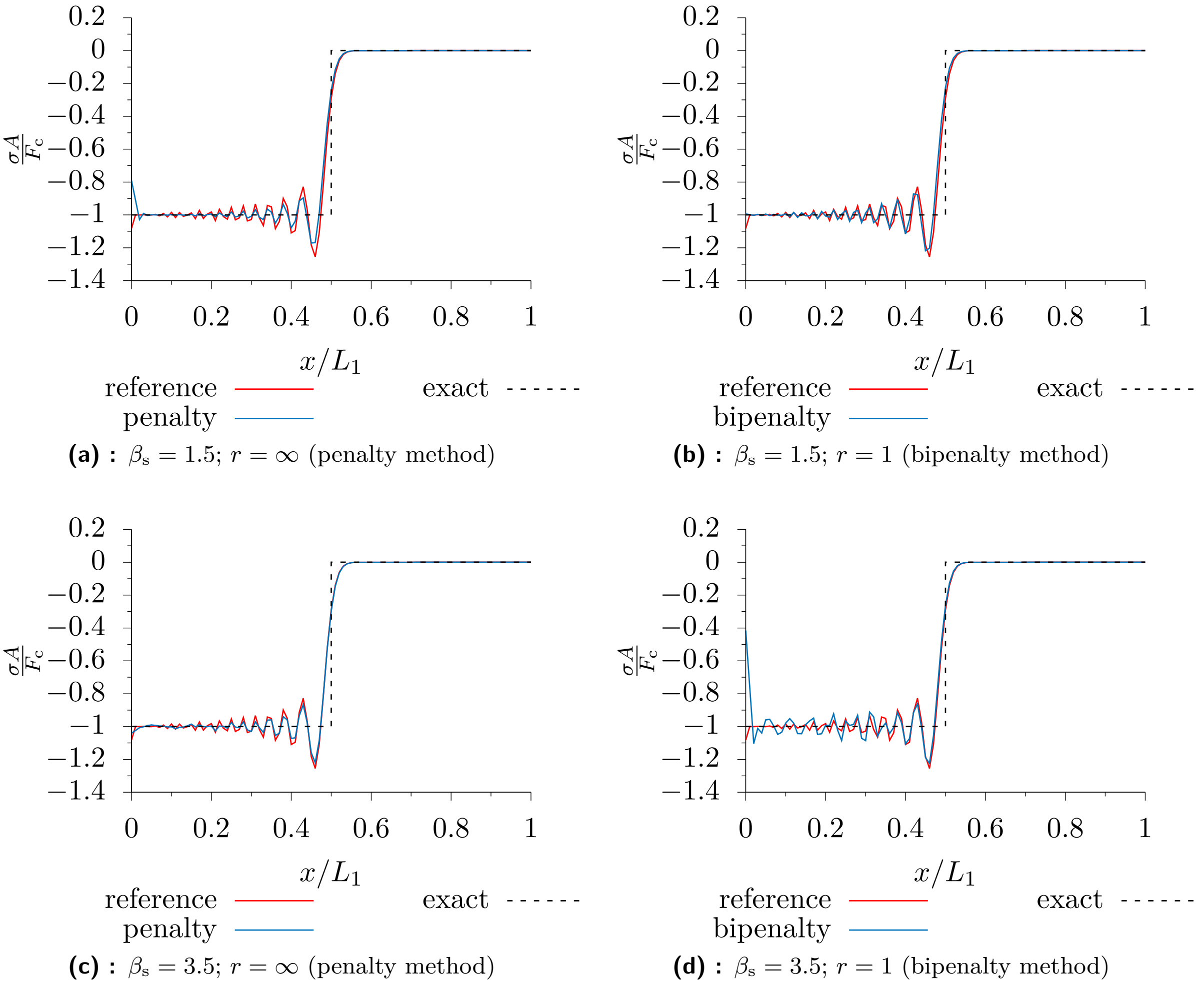
(b) : $\beta_s = 3.5$; $r = 1$ (bipenalty)

(c) : $\beta_s = 1.5$; $r = \infty$ (penalty)

(d) : $\beta_s = 1.5$; $r = 1$ (bipenalty)

Figure 6.21: Time distribution of the dimensionless contact force for and $C_r = 0.5$.

Figure 6.22 shows spatial distribution stress along the bar when the wavefront reached a half of the bar. In addition to contact analysis, a reference calculation was performed, where the axial displacement of the contact node was fixed. Stress distributions for both the penalty and the bipenalty method was in a good agreement with the reference solution. The reason probably is that the finite element mesh behaves as a low-pass filter [180] and therefore high frequency oscillations introduced by the penalty-like methods do not affect the solution.



(a) : $\beta_s = 1.5$; $r = \infty$ (penalty method)

(b) : $\beta_s = 1.5$; $r = 1$ (bipenalty method)

(c) : $\beta_s = 3.5$; $r = \infty$ (penalty method)

(d) : $\beta_s = 3.5$; $r = 1$ (bipenalty method)

Figure 6.22: Spatial distribution of the dimensionless stress for $C_r = 0.5$.

6.2.2 1D impact of two bars with different lengths

In this example, we follow the work by Huněk [181], where the contact-impact problem of two bars of identical materials and with different lengths are studied. A scheme of this test is depicted in Figure 6.23. The left bar is moving to the right with a constant velocity $V_{0x}^{(1)}$. The right bar with fixed end is at rest. The geometrical, material and numerical parameters were adopted from [181] as follows: the lengths $L_1 = 10\,\mathrm{m}$ and $L_2 = 20\,\mathrm{m}$, the Young's modulus $E^{(1)} = E^{(2)} = 100\,\mathrm{Pa}$, the mass density $\rho_0^{(1)} = \rho_0^{(2)} = 0.01\,\mathrm{kg}\cdot\mathrm{m}^{-3}$, the cross-sectional area $A^{(1)} = A^{(2)} = 1\,\mathrm{m}^2$, the initial impact velocities $V_{0x}^{(1)} = 0.1\,\mathrm{m/s}$ and $V_{0x}^{(1)} = 0\,\mathrm{m/s}$, the number of elements $n_{\mathrm{nel}}^{(1)} = 50$, $n_{\mathrm{nel}}^{(2)} = 100$, thus the finite element lengths are set up as $h^{(1)} = h^{(2)} = 0.2\,\mathrm{m}$, the initial contact gap $g_{\mathrm{N}}^{0} = 0\,\mathrm{m}$, the duration time $T = 0.7\,\mathrm{s}$. The analytical solution for the contact force is $F_{\mathrm{c}} = 0.05\,\mathrm{N}$ for $t \in (0; 0.2)$ and $t \in (0.4; 0.6)$ and zero otherwise.

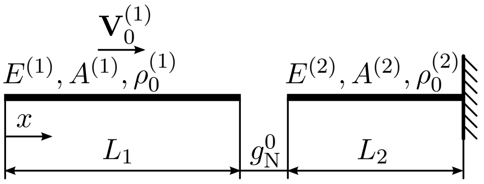
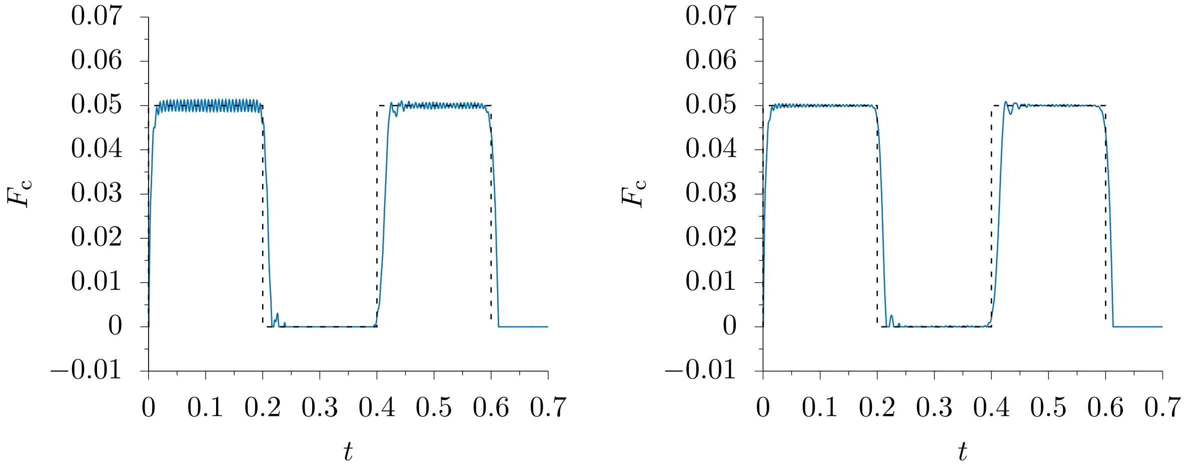


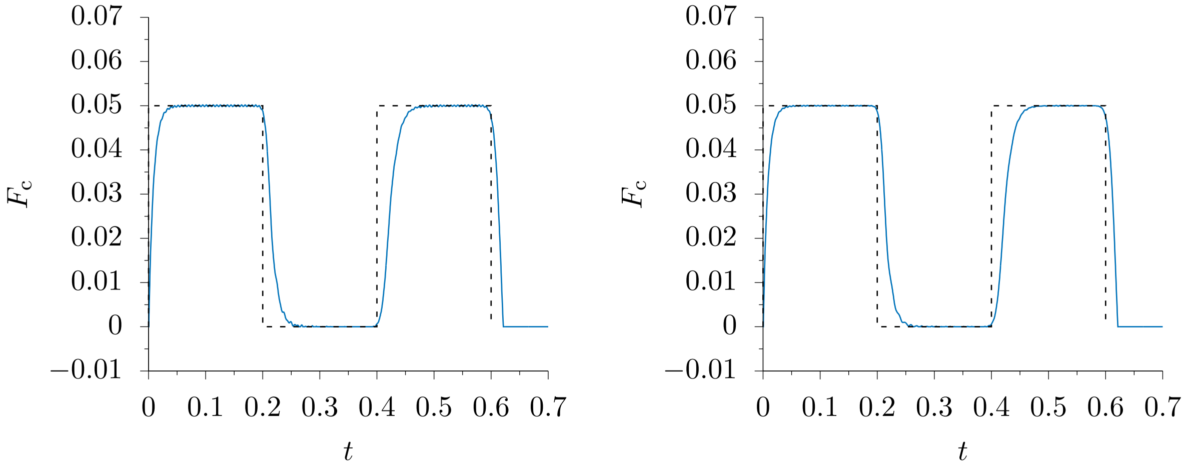
Figure 6.23: A scheme of the Huněk test.

Figures 6.24 and 6.25 show time distribution of the contact force on the contacting node of the left bar for the Courant numbers $C_{\mathrm{r}} = 0.2$ and $C_{\mathrm{r}} = 0.999$, respectively. The influence of the stiffness penalty parameters $\beta_s = 0.125, 0.25$, and 1 together with the corresponding mass penalty parameters for the penalty method ($r = \infty$) and the bipenalty method given by the CPR the value ($r = 1$) are investigated. For $C_{\mathrm{r}} = 0.2$ (see Figure 6.24) the results exhibit a stable character. For the value of the stiffness parameter $\beta_s = 1$ both penalty and bipenalty method suffer from oscillations. On the other hand, the results show a delay effect in contact force distribution for smaller values of the stiffness parameters $\beta_s = 0.124$ and $\beta_s = 0.25$, respectively. This effect could be theoretically explained based on the reflection-transmission analysis [190, p. 10–13]. For $C_{\mathrm{r}} = 0.999$ (see Figure 6.25) the results for the penalty method ($r = \infty$) show unstable character while for the bipenalty method with the CPR exhibit stable behaviour, regardless of the choice of stiffness parameters. However, the distribution of the contact forces indicate large oscillations.



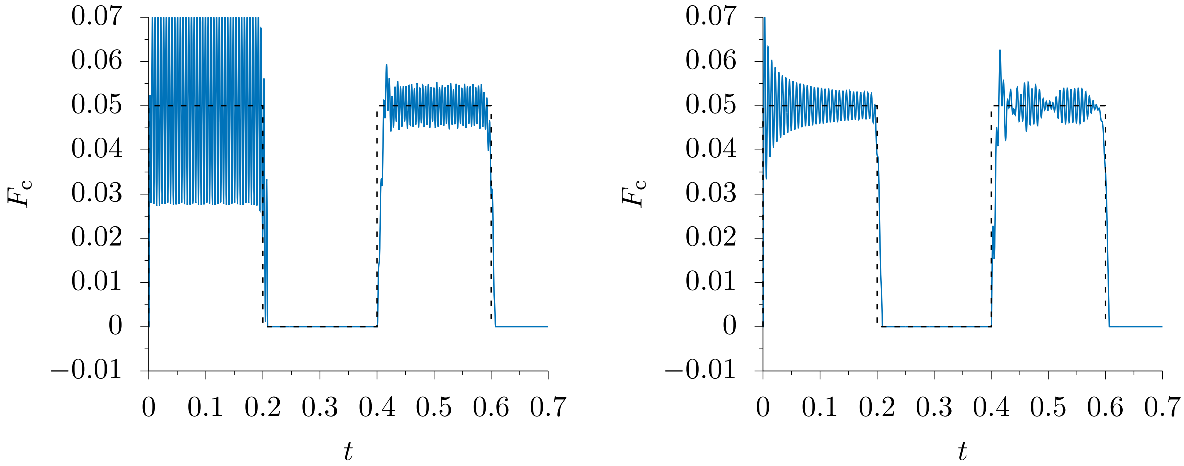
(a) : $\beta_s = 0.125, \beta_m = 0$ (penalty)

(b) : $\beta_s = 0.125, \beta_m = 0.0625$ (bipenalty)



(c) : $\beta_s = 0.25, \beta_m = 0$ (penalty)

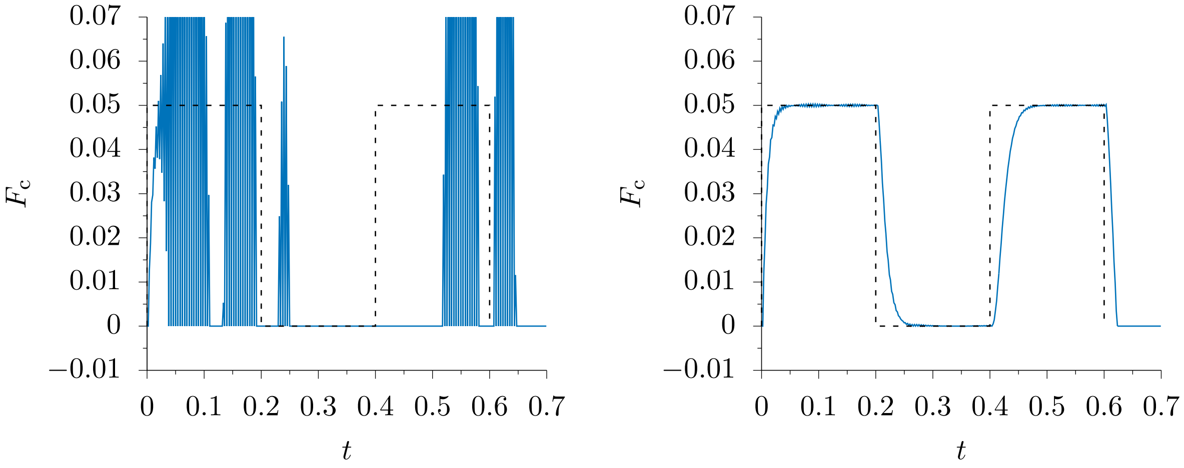
(d) : $\beta_s = 0.25, \beta_m = 0.125$ (bipenalty)



(e) : $\beta_s = 1, \beta_m = 0$ (penalty)

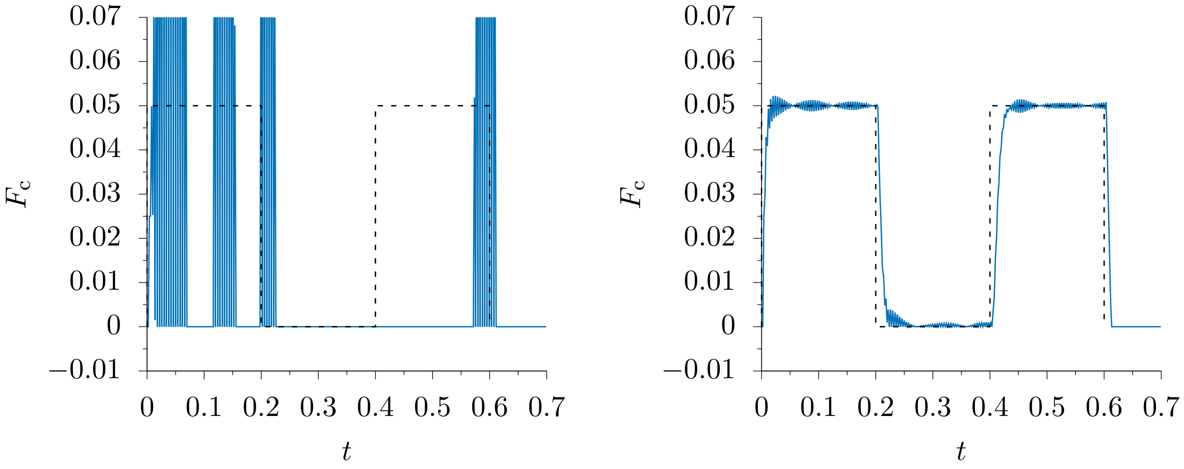
(f) : $\beta_s = 1, \beta_m = 0.5$ (bipenalty)

Figure 6.24: Contact force with respect to time for Huněk test for the Courant number $C_r = 0.2$.



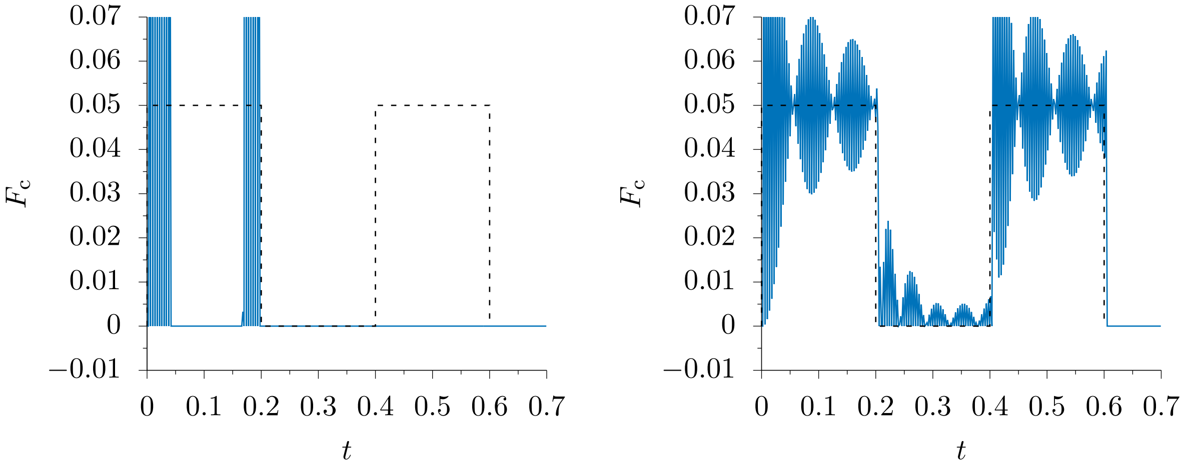
(a) : $\beta_s = 0.125, \beta_m = 0$ (penalty)

(b) : $\beta_s = 0.125, \beta_m = 0.0625$ (bipenalty)



(c) : $\beta_s = 0.25, \beta_m = 0$ (penalty)

(d) : $\beta_s = 0.25, \beta_m = 0.125$ (bipenalty)



(e) : $\beta_s = 1, \beta_m = 0$ (penalty)

(f) : $\beta_s = 1, \beta_m = 0.5$ (bipenalty)

Figure 6.25: Contact force with respect to time for Huněk test for the Courant number $C_{\mathrm{r}} = 0.999$.

6.2.3 2D Hertz dynamic problem

The purpose of this example is to illustrate the performance of the classic FEA and IGA contact-impact explicit algorithm [203, 195, 211, 196]. The example deals with Hertz dynamic problem, a classical benchmark due to the availability of an analytical solution [182, p. 117]. In the example, the effect of mass lumping is investigated. The comparisons in this section are limited to the second order elements. In particular, quadratic serendipity eight-node finite elements are used in the case of FEA, and second order basis function for IGA.

The presented numerical example deal with frictionless impact of two cylinders of radius $L_1 = 4\,\mathrm{m}$, see Figure 6.26. The material of each of the cylinders is linearly elastic with Young's modulus $E = 1000\,\mathrm{MPa}$, Poisson's ratio $\nu = 0.2$, and density $\rho_0 = 1\,\mathrm{kg}\cdot\mathrm{m}^3$. The non-zero components of the initial velocity vectors are $V_{0y}^{(1)} = 2\,\mathrm{m}\cdot\mathrm{s}^{-1}$ and $V_{0y}^{(2)} = -2\,\mathrm{m}\cdot\mathrm{s}^{-1}$. In the initial configuration the cylinders just touches each other in a point. Due to symmetry, only the half of each cylinder is considered. The penalty parameter is $\epsilon_\mathrm{N} = 1 \times 10^5\,\mathrm{N}\cdot\mathrm{m}^{-2}$.

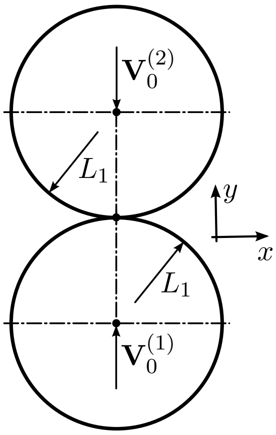
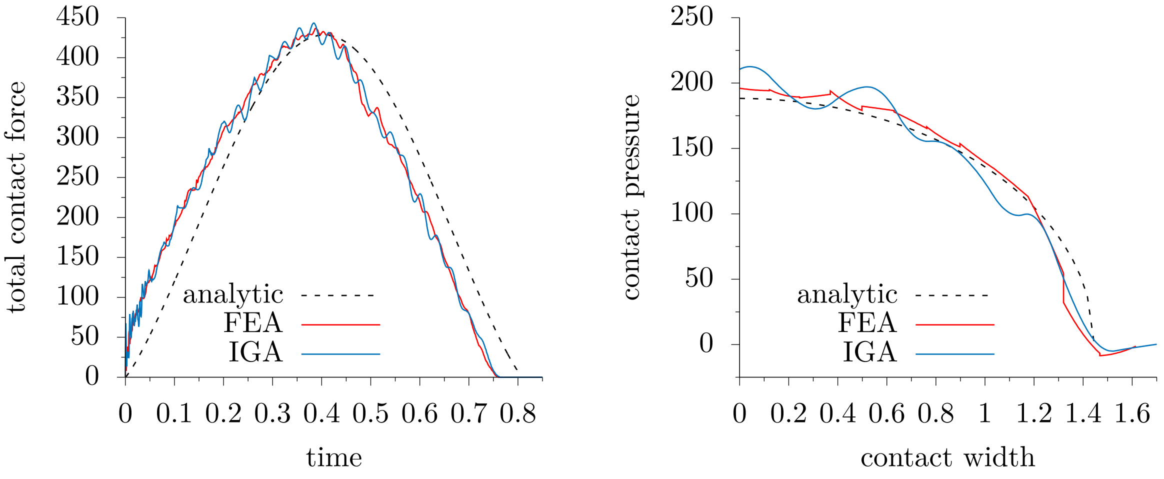


Figure 6.26: Dynamic Hertz problem.

The explicit time integration by CDM is performed for $T = 0.9\,\mathrm{s}$ with the time step $\Delta t = 5 \times 10^{-4}\,\mathrm{s}$. It should be noted that only one mesh is considered. The effect of mesh refinement will be studied in the further work.

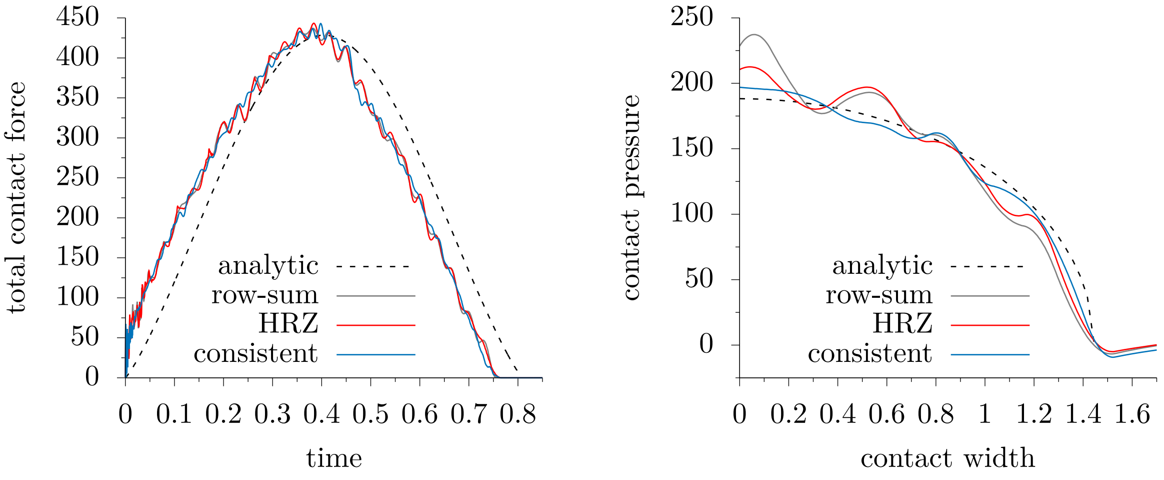
Figure 6.27 shows the contact force and the maximal contact pressure obtained for both FEA and IGA with HRZ mass lumping technique [204, 197]. The HRZ method was chosen because in case of second order Lagrange elements the row sum method leads to negative mass on diagonal, which is not admissible. Results show a satisfactory agreement with the analytical solution. One can notice that the FEA solution in comparison with IGA solution exhibits lower oscillations.

In order to evaluate the effect of mass lumping techniques on the oscillations of the contact forces and contact pressure distribution in IGA, further analyses are performed by using consistent mass matrix and mass matrix lumped by the row sum method. Figure 6.28 illustrates that consistent mass matrix delivers a more accurate contact pressure distribution than row sum and HRZ mass lumping techniques.



(a) : contact force **(b) :** contact pressure

Figure 6.27: Comparison of classic FEA and IGA solution of contact forces and contact pressures for HRZ mass lumping method.



(a) : contact force **(b) :** contact pressure

Figure 6.28: Influence of mass lumping techniques on contact forces and contact pressures for IGA.

6.2.4 3D contact-impact between two tubes

In order to demonstrate the robustness of the new local contact search procedure, impact between two tubes was studied [212, 189]. It was a challenging problem since the contact conditions are enforced on the three-dimensional surface which rapidly changes its curvature during iterative solution. The example was originally presented in [183]. Two perpendicular tubes shown in Figure 6.29 are in contact with each other and the non-zero components of the initial velocity vectors are prescribed as $V_{0x}^{(1)} = 35\,\mathrm{m\cdot s^{-1}}$ and $V_{0x}^{(2)} = -35\,\mathrm{m\cdot s^{-1}}$. The length, radius and thickness of the tubes are $L_1 = 0.46\,\mathrm{m}$, $L_2 = 0.1\,\mathrm{m}$ and $L_3 = 0.003\,\mathrm{m}$, respectively. The tubes consist of elastic-perfectly plastic material with Young's modulus $E = 2 \times 10^5\,\mathrm{MPa}$, Poisson's ratio $\nu = 0.3$ and yield stress $\sigma_\mathrm{Y} = 200\,\mathrm{MPa}$. The density is $\rho_0 = 7\,840\,\mathrm{kg\cdot m^{-3}}$.

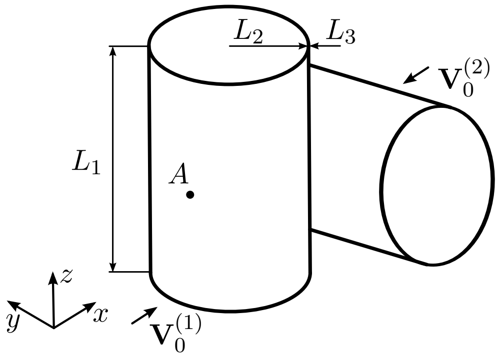
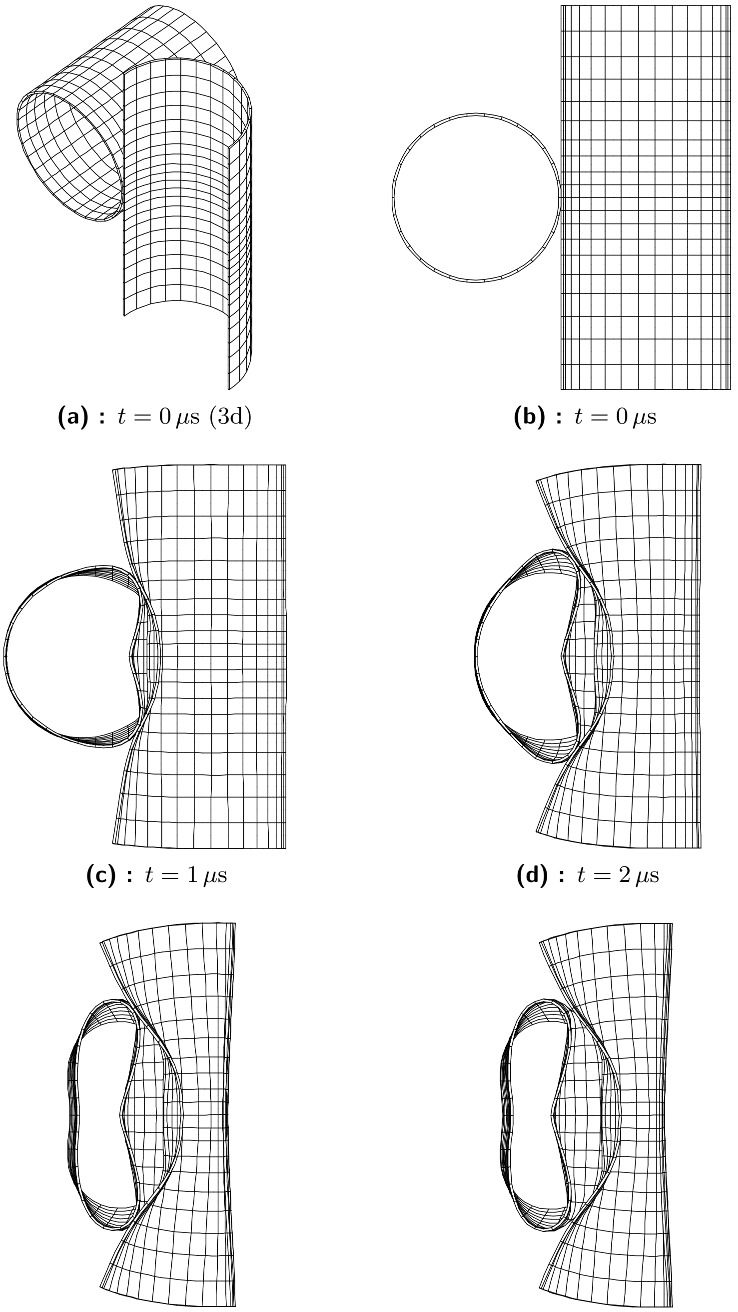


Figure 6.29: Initial configuration of impacting tubes.

Based on the recommendation of a choice of permissible dimensionless wavelengths for quadratic serendipity finite element meshes to suppress the dispersion errors [184], each of the tubes was discretized by 600 quadratic twenty-node elements. The central difference method with the diagonal mass matrix was used to solve the problem. The diagonal mass matrix was assembled by HRZ (Hinton-Rock-Zienkiewicz) lumping scheme [156]. The value of the penalty parameter was set to $\epsilon_\mathrm{N} = 10^{12}\,\mathrm{N/m^3}$. It is well known that the penalty method tends to decrease the critical time step in conditionally stable explicit time integration schemes. The time step was set carefully to $\Delta t = 0.005\,\mu\mathrm{s}$ and the stability of the integration process was ensured by energy balance monitoring. The impact response of the tubes was calculated for $T = 8\,\mu\mathrm{s}$.

Several distorted configurations in selected times are shown in Figures 6.30b–6.30f depicted for the central section of one of the tubes, which is perpendicular to the axis of the other tube simultaneously (see Figure 6.30a). One can observe how the contact zone changes its size and position during the impact. As the starting approximation, the contact is realized by a single point as is shown in Figure 6.30b. For a short time after impact, the centre of contact area comes unstuck. As the deformation continues, the contact zone moves near to the edge of the tubes, so that finite sliding is involved. Finally, the plastically deformed tubes rebound as Figure 6.30f shows.

By means of this example, it was confirmed that the proposed local contact search procedure significantly increased the effectiveness of the numerical solution of contact-impact problem between two tubes.



(a) : $t = 0\,\mu\text{s}$ (3d)

(b) : $t = 0\,\mu\text{s}$

(c) : $t = 1\,\mu\text{s}$

(d) : $t = 2\,\mu\text{s}$

(e) : $t = 6\,\mu\text{s}$

(f) : $t = 8\,\mu\text{s}$

Figure 6.30: Distorted configurations of impact between two tubes in selected times.

6.3 An engineering application

The chapter about numerical examples is closed by a presentation of a real-world engineering problem which involved a creep analysis of a turbine casing. Only a brief description of the task is provided. An emphasis is placed on the contact analysis of the upper and lower parts at the dividing plane.

6.3.1 High pressure steam turbine inner casing

Methods developed in this dissertation were implemented in the FEM software PMD [2] and used to solve a real-world engineering problem which involved the thermal, static, and creep analysis of the high pressure (HP) steam turbine inner casing. Namely static heat conduction, elastic, elasto-plastic, and creep problems were solved using a complex creep model [185]. The aim of the work was to evaluate residual deformations after 10 000 and 200 000 operation hours in selected control point of the dividing plane after relieving, cooling and dismantling the turbine inner casing. Further details on the load and material parameters can be found in the report [200].

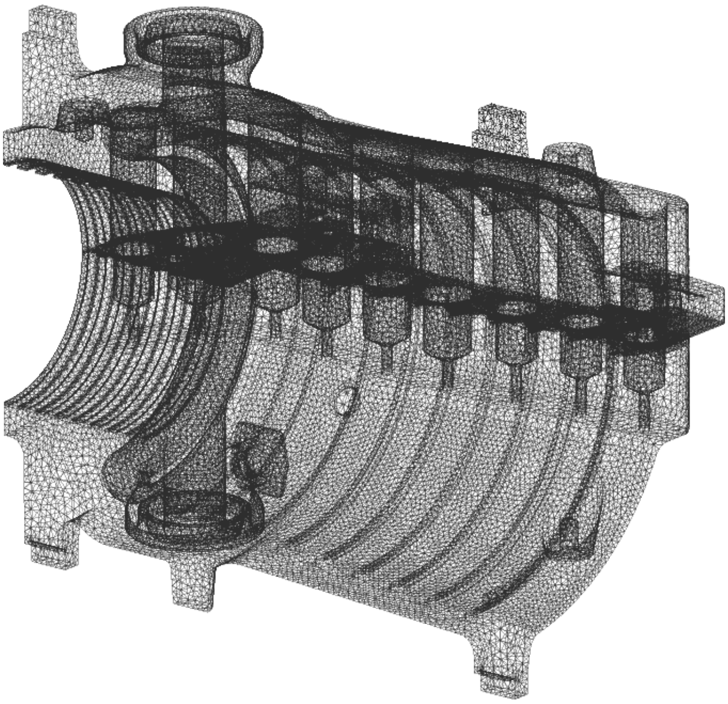


Figure 6.31: Finite element mesh of the upper and lower part of the turbine casing.

A finite element mesh of 400 869 quadratic 10-node tetrahedral elements with 671 212 nodes in total, which was taken over from the assignor — Doosan Škoda Power, is shown in Figure 6.31. The total number of degrees of freedom was 2 006 153. The contact interface was discretized by 20 149 quadratic 6-node triangle elements. The 7-point Gauss quadrature rule [13, p. 467] was employed to integrate the contact virtual work term. As a consequence, the contact detection had to be perform among 141 043 points and 20 149 contact segments, i.e. 2 841 875 407 combinations would be necessary to evaluate if an all-to-all contact detection algorithm was used. If we further realize that contact detection is repeated for each integration step of the temporal integration scheme [186] of the creep model and each iteration of the non-linear solver, the calculation could not be completed in a satisfactorily short time. One execution of the all-to-all contact detection algorithm for so many combinations takes a few hours. Unfortunately, one iteration of the all-to-all algorithm is performed in the initialization phase of the current contact algorithm [1, p. 2626]. Therefore, the global contact search algorithm [91] was implemented which uses a linked-list data structure to effectively sort and search for contact pairs. Contact detection with this type of global contact search algorithm takes only a few seconds. Another improvement of the contact algorithm was achieved by considering the inexact linearization of the contact residual. Figure 6.32 shows the positive effect of the approximative contact tangent matrix on the number of iteration of the non-linear solver.

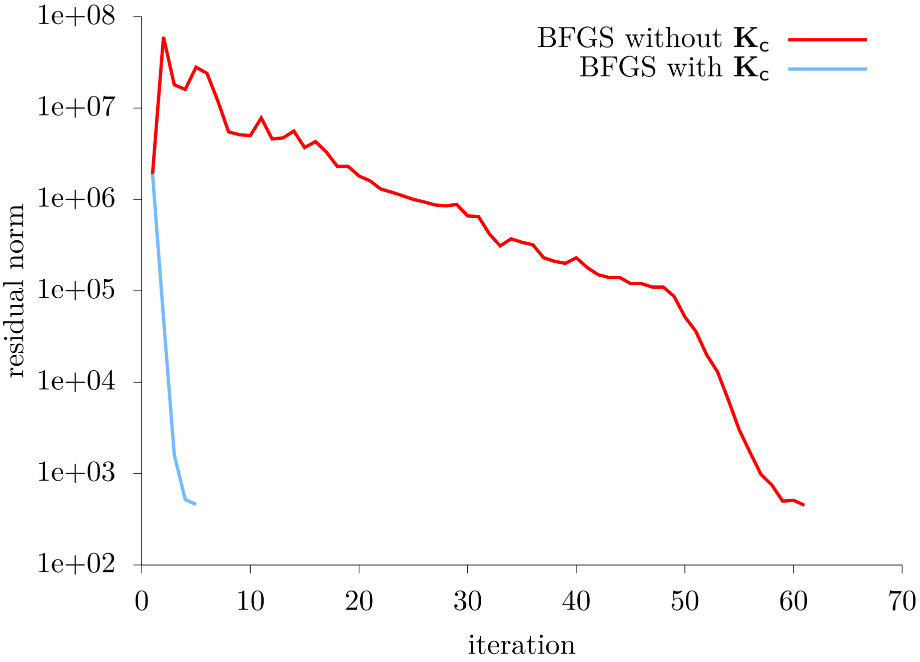
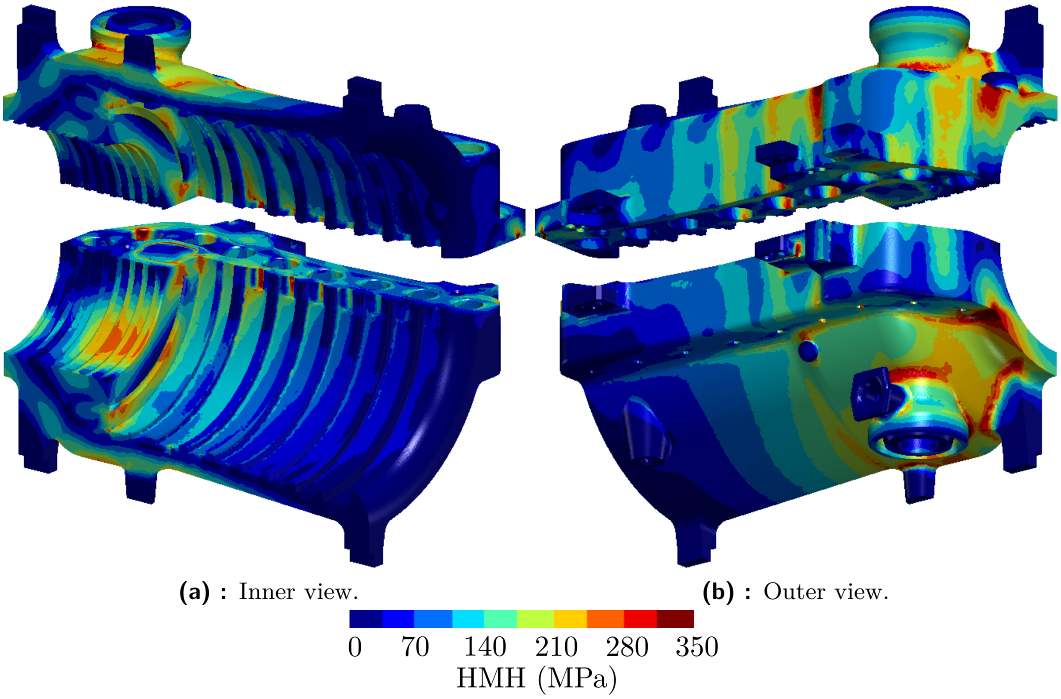


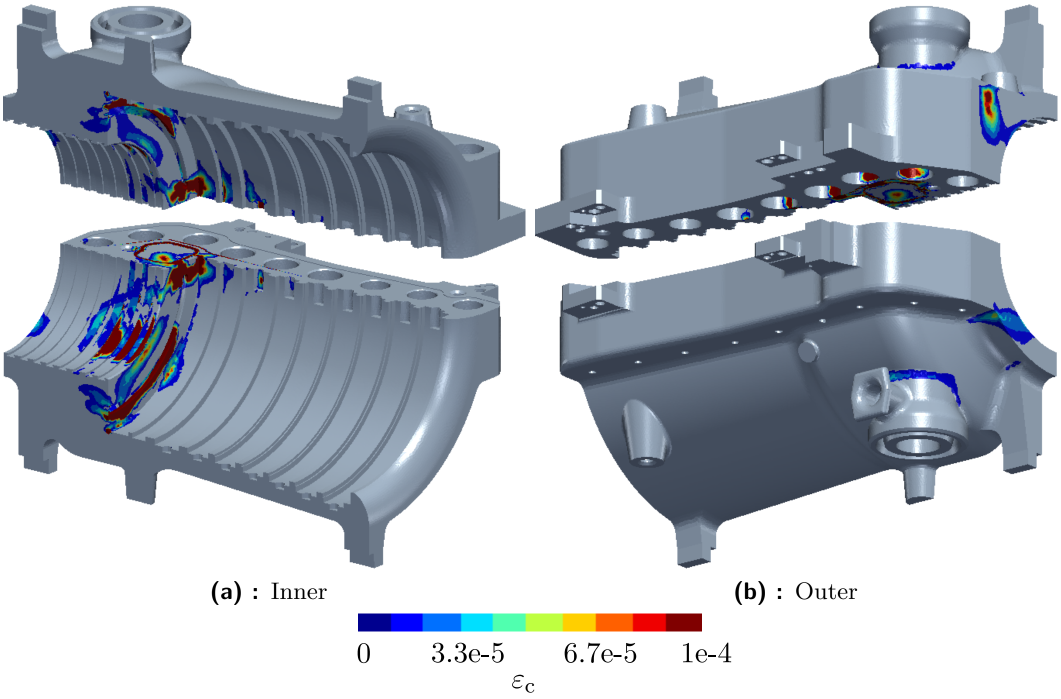
Figure 6.32: Dependence of the residual norm convergence on the number of iterations obtained for the therom-elastic calculation.

Further in this section, several selected results will be presented. In Figure 6.33, there are contours of Huber-von Mises-Hencky (HMH) reduced stress after thermo-elasto-plastic calculation. One may notice the values of yield stress, especially on the outer and inner parts of the inlet where the part is exposed to high temperatures and pressures. In these places, permanent deformations occur due to creep after 10 000 operation hours. Specific locations and creep strain values are shown in Figure 6.34. In this state, pressure unloading, cooling and subsequent disassembly are performed. As the result of this, the contact surface of the the upper and lower parts are permanently deformed. The contours of z-displacements on the contact interface are depicted in Figure 6.35. It should be noted that the results obtained by the FEA software PMD were validated with a measurement and an alternative computation in FEA software ANSYS with a different creep model. Both the measurement and the alternative computation were done by the assignor. All three approaches were in a very good agreement and confirmed the hypothesis, that the real component was permanently distorted due to creep deformations.



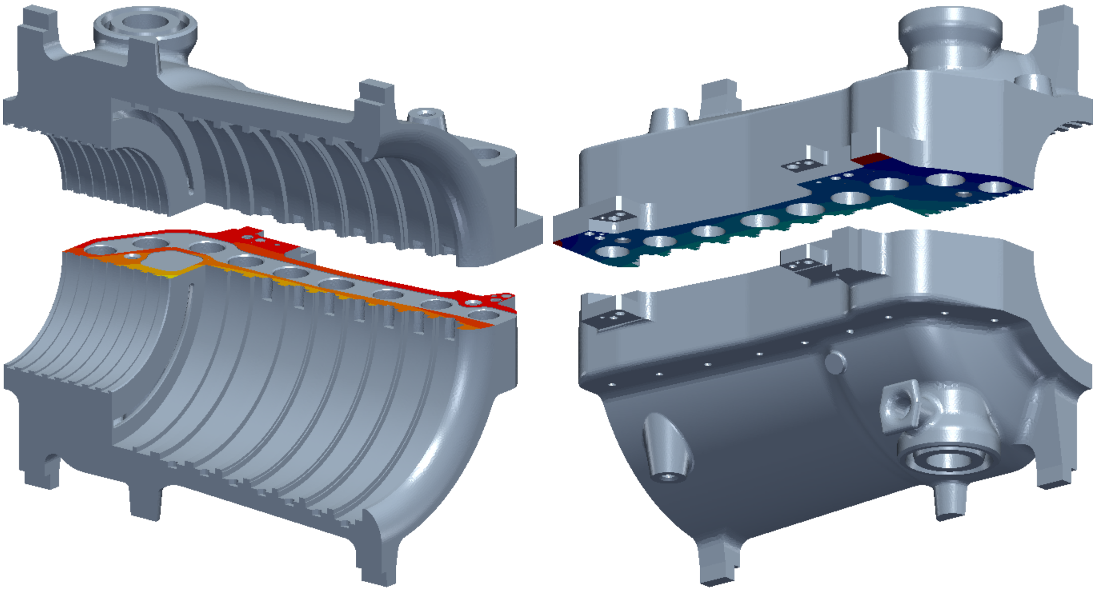
(a) : Inner view. **(b) :** Outer view.

Figure 6.33: Contours of HMH reduced stress on the inner and outer surface of the turbine casing.

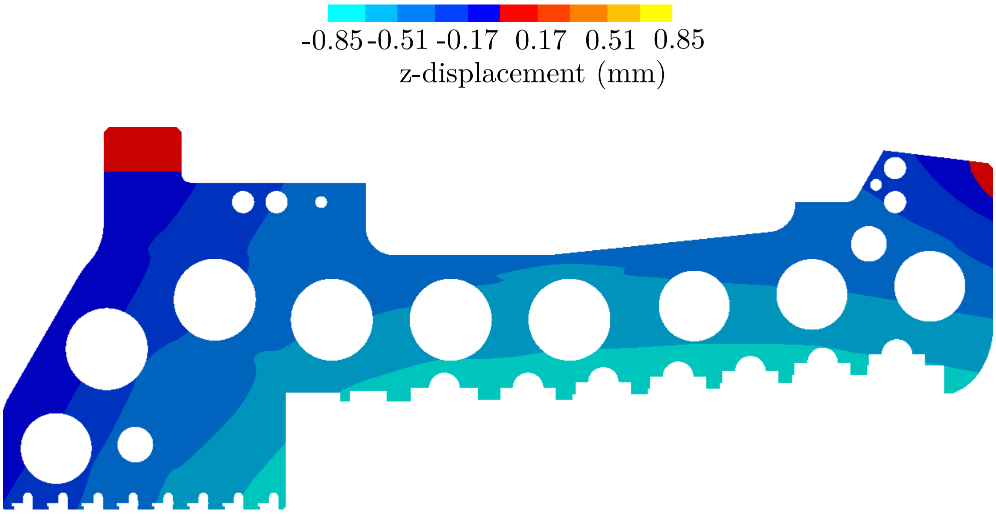


(a) : Inner **(b) :** Outer

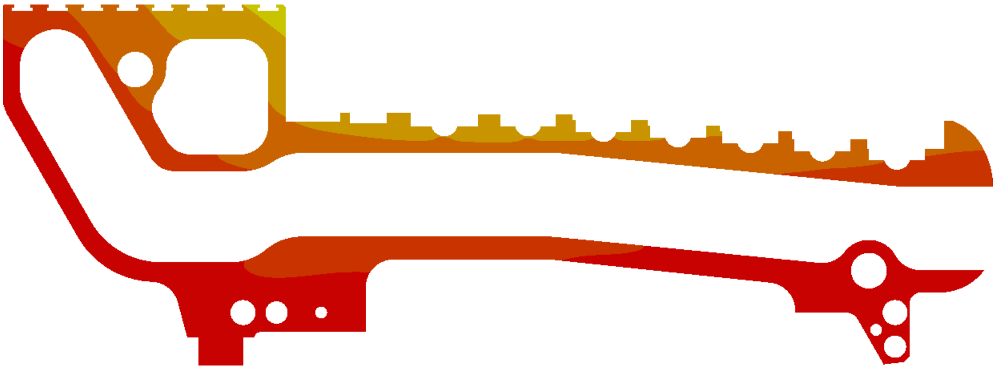
Figure 6.34: Contours of creep strain, ε_C, on the inner and outer surface of the turbine casing.



(a) : Inner view. **(b) :** Outer view.



(c) : Detail of the upper dividing plane.



(d) : Detail of the lower dividing plane.

Figure 6.35: Geometry of the upper and lower part of the turbine casing with contours of the z-displacement at the dividing plane.

Chapter 7

Theoretical and Practical Outcomes of the Thesis

Regarding the **theoretical outcomes of the thesis**, the greatest contribution is the development of an isogeometric finite element explicit contact-impact algorithm. Motivated by the desire for an accurate numerical solution of impact problems and stress wave propagation in materials, attention was paid to the study of numerical properties of the new spatial discretization method — the isogeometric analysis. For this purpose, a completely new finite element software was developed. Its general design enabled to solve even complex geometries consisting of multiple patches, which were exploited in the works [187, 188]. Comparison of frictionless contact treatment in classical finite element analysis and isogeometric analysis in static regime was presented in [205, 193, 206]. The algorithm was further extended for dynamic problems with particular attention paid to the accuracy of contact forces and contact pressures [203, 211, 196, 212], mass matrix lumping [204, 197] and contact detection [195].

Next theoretical outcome of the thesis is the proposal of a new local contact search procedure [189]. The design of a new method was preceded by a critical assessment of a set of unconstrained optimization methods [198, 207, 208]. The assessment included not only static, but also dynamic contact problems [209, 199, 210]. The fundamental purpose of the local contact search procedure is the fast and accurate resolution of the closest point projection (CPP) problem. The CPP is a general problem which can be encountered across the various technological applications; computer graphics, CAD, CAM, robotics, to name some. Therefore, it is believed that the newly proposed method can find application also in other fields.

The third contribution of this work is the utilization of the bipenalty method for contact-impact problems [190]. It is well known that the pure penalty method negatively influences the critical time step of an explicit temporal integration scheme. The bipenalty method was originally proposed for the time domain computational dynamics and for arbitrary multipoint constraints. The theoretical contribution of this work lies in the derivation of a critical time step estimate for the penalty and bipenalty method first published in [192] and applied for explicit IGA [202] contact-impact analysis. The stability and reflection-transmission properties of the bipenalty method for contact-impact problems in 1D homogeneous case was studied in [190].

Regarding the **practical outcomes of the thesis**, the stability of the BFGS algorithm for a constrained system was improved by considering a new initial tangent matrix enriched by approximative linearization of the contact residual term. Further, the new local contact search procedure was implemented into the FE system PMD [2]. The resulting adjustment of the BFGS algorithm an the local contact search procedure were successfully tested and applied to an engineering problem of high pressure steam turbine inner casing [200, 201].

Chapter 8

Conclusions and Future Prospectives

The presented dissertation aimed to increase the efficiency and robustness of the finite element contact algorithm for resolution of the general static and dynamic large deformation contact problems. Three objectives were defined to achieve this goal. First, develop an original isogeometric contact formulation. Second, propose a new method for local contact searching. And third, expand the bipenalty method for the solution of the contact-impact problems.

Regarding the first objective, the originality consists in connecting the symmetry preserving contact formulation and isogeometric analysis. By the symmetry preserving formulation is meant an unbiased treatment of the contact residual, thanks to which there is no need to distinguish between the master and slave bodies. The isogeometric analysis is a modern method of spatial discretization, which utilizes as basis functions various types of splines, most frequently NURBS. The most attractive feature of the isogeometric contact analysis is the ability to control the geometric continuity on the interface of finite elements. While the basis functions of the classical FEA are only C^0 continuous at the interface of elements, the continuity of IGA basis functions can be up to C^{p-1}, where p is the order. Consequently, the isogeometric contact analysis in comparison with the conventional finite element method gives smoother contact forces and pressures. This is mainly due to an unambiguous definition of the normal vector field at the contact interface. The performance of the newly developed isogeometric contact formulation was tested by mean of several numerical examples involving 2D contact patch test, 2D Hertz-Signorini problem, 2D/3D large deformation bending problems, 2D frictional/3D frictionless ironing problems and 3D frictional sliding of a cube on a rigid plate.

The newly developed isogeometric contact formulation was also extended into the explicit contact-impact algorithm for the numerical solution of dynamic and wave propagation problems. A particular attention was paid to the influence of the mass matrix lumping on the contact pressure oscillations. Two main conclusions for explicit dynamic application may be drawn: First, for higher order elements and mass matrix lumped by the HRZ method, IGA in comparison with classic FEA leads to a more oscillatory contact force and consequently also contact pressure. Second, the oscillations of the contact forces in IGA are minimal for consistent mass matrix.

As far as the second objective is concerned, today's implementations of the local contact search procedure most frequently employs the Newton-Raphson method. Although this method works passably for linear elements, it suffers from convergence difficulties for higher-order elements. This is due to the fact that the Hessian matrix of the square distance function to be minimized is not positive definite in general. Therefore, various alternative methods were tested. Beside the geometric iteration methods, namely the least square projection, the sphere and torus approximation methods, and the standard gradient methods such as the steepest descent method were considered. Next, two quasi-Newton methods — the Broyden and the BFGS method — were examined. Finally, the simplex method was chosen as a representative of the derivative-free methods.

The assessment of the methods was performed in a benchmark problem based on the geometry of a real distorted master segment of the second order serendipity element and the position of a slave point. The detailed behaviour of each method was monitored for seven chosen initial estimates. In order to increase the validity of obtained results, the benchmark problem was extended to a performance test involving an equally spaced three-dimensional grid of slave points, which were projected onto this master segment. Again, the behaviour of each method was thoroughly tested. It was shown that only the second order geometric iteration methods and the simplex method were robust because only these three methods converged in all 27 225 cases. Rectifying this shortcoming for the line search methods was solved using the Illinois algorithm as an alternative to the quadratic interpolation technique. Next, different possibilities to improve the Newton-Raphson method were considered, taking into account the Hessian modification or coupling with the steepest descent method or the simplex method. Based on the results of the performance test, the torus approximation method, the combination of the Newton-Raphson method with the simplex method, the Newton-Raphson method with Hessian modification and the BFGS method utilizing Illinois line search were the most cost-effective methods for local contact search.

Finally, the methods were implemented to the local contact search procedure of the FEA software PMD and tested through the numerical example which involved the bending of two elastic rectangular plates over an elastic cylinder. For lower values of the penalty parameter the torus approximation method seemed to be the most efficient ones. Note that the Newton-Raphson method with modified Hessian was not very efficient. This is probably due to the low size of the Hessian matrix. Once the Hessian modification has to be carried out, there is left only one search direction given by the remaining eigenvector, which need not be effective. It can be concluded, based on overall results, that the torus approximation method is the most effective method for the local contact searching.

Regarding the thirds objective of this dissertation, the stability of an explicit contact-impact algorithm using the existing bipenalty approach in finite element analysis has been studied and analyzed for one-dimensional problems. The main attention has been paid on an upper bound estimation of the stable Courant number for the bipenalty method with respect to stiffness penalty and mass penalty parameters. It was shown that the critical Courant number tends towards zero for the stiffness penalty parameter approaching infinity whereas the mass penalty parameter is considered to be zero, i.e. when only a pure penalty formulation is considered. On the other hand, setting the penalty ratio between mass and stiffness penalty parameters to the critical penalty ratio preserves the stable Courant number at the level of the unpenalized system for an arbitrary value of the stiffness penalty parameter. The original method for computation of the critical penalty ratio is based on the highest frequency of a one-element subproblem with one active constraint with a rigid body. It considers cases for two-node bar, Euler-Bernoulli beam and two-dimensional solid elements with lumped and consistent mass matrix approximations.

The theoretical derivations are verified via numerical examples. The stability estimate gives correct critical Courant number for a simple 1D dynamic Signorini problem and the Huněk test. Predicted dependencies of the critical time step are confirmed: pure penalty method decreases the critical time step size and the bipenalty method with critical penalty ratio preserves the critical time step. It is an excellent property of the bipenalty method, which can be employed in real numerical solutions of contact-impact problems. The two numerical contact-impact examples also revealed that both (penalty and bipenalty) methods caused spurious oscillations in the distributions of displacement and the contact force. This effect was especially obvious for higher values of stiffness penalty parameters, where contact damping is often employed. Solutions with bipenalty method tend to have less spurious oscillations.

The estimation of the penalty parameter ensuring non-oscillating behavior of both methods will be investigated in further work together with implementation and testing the bipenalty method in multi-dimensional contact-impact problems. Further, the analysis of the bipenalty method in contact-impact problems for one- and multi-dimensional heterogeneous bodies will be investigated in detail.

Let us recall that the theoretical and practical outcomes of the thesis were highlighted in the previous chapter. At the end, it is concluded that **all aims of the thesis have been fulfilled.**

References

[1] D. Gabriel, J. Plešek, and M. Ulbin. "Symmetry preserving algorithm for large displacement frictionless contact by the pre-discetization penalty method". In: *International Journal for Numerical Methods in Engineering* 61(15) (Dec. 2004), pp. 2615–2638. DOI: `10.1002/nme.1173`.

[2] Collective of Authors. *PMD f77.11: Příručka uživatele, referenční příručka.* Dolejškova 5, Praha 8: Institute of Thermomechanics AS CR, v.v.i., 2013.

[3] H. R. Hertz. "Über die Berüehrung fester elastischer Körper (On the Contact of Solid Elastic Bodies)". In: *Journal für die reine und angewandte Mathematik (Journal for the pure and applied mathematics)* (92 (in German)) (1881), pp. 156–171.

[4] A. Signorini. "Sopra alcune questioni di elastostatica (On some problems of elastostatics)". In: *Atti della Società Italiana per il Progresso delle Scienze (Proceedings of the Italian Society for the Advancement of Science)* (11 (in Italian)) (1933), pp. 143–148.

[5] A. Signorini. "Questioni di elasticità non linearizzata e semilinearizzata (Topics in non linear and semilinear elasticity)". In: *Rendiconti di Matematica e delle sue Applicazioni (Reports on Mathematics and its applications)* 18 (1959), pp. 95–139.

[6] G. Fichera. "Problemi elastostatici con vincoli unilaterali: il problema di Signorini con ambigue condizioni al contorno (Elastostatic problems with unilateral constraints: the Signorini problem with ambiguous boundary conditions)". In: *Memorie della Accademia Nazionale dei Lincei, Classe di Scienze Fisiche, Matematiche e Naturali (Memoirs of the National Academy of Lincei, Class of Physical, Mathematical and Natural Sciences)* 7(2) (1964), pp. 91–140.

[7] J. L. Lions and G. Stampacchia. "Variational inequalities". In: *Pure and Applied Mathematics* 20(3) (Aug. 1967), pp. 493–519. DOI: `10.1002/cpa.3160200302`.

[8] G. Duvaut and J. L. Lions. *Inequalities in Mechanics and Physics.* A Series of Comprehensive Studies in Mathematics. Springer-Verlag Berlin Heidelberg, 1976. DOI: `10.1007/978-3-642-66165-5`.

[9] D. Kinderlehrer and G. Stampacchia. *An Introduction to Variational Inequalities and Their Applications.* Ed. by J. Robert E. O'Malley. Classics in Applied Mathematics. SIAM, 2000. DOI: `10.1137/1.9780898719451`.

[10] N. Kikuchi and J. T. Oden. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods.* Vol. 8. Studies in Applied and Numerical Mathematics. SIAM, 1988.

[11] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. *The Finite Element Method: Its Basis and Fundamentals*. 6th ed. Vol. 1. Butterworth-Heinemann, 2005.

[12] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publication, 2000.

[13] K. J. Bathe. *Finite Element Procedures*. Prentice-Hall, 1996, p. 1037.

[14] J. Fish and T. Belytschko. *A First Course in Finite Elements*. John Wiley & Sons Ltd, 2007.

[15] J. Geng, W. Yan, and W. Xu, eds. *Application of the Finite Element Method in Implant Dentistry*. Advanced Topics in Science and Technology in China. Springer Berlin Heidelberg, 2008. DOI: `10.1007/978-3-540-73764-3`.

[16] S. K. Chan and I. S. Tuba. "A finite element method for contact problems of solid bodies–Part I. Theory and validation". In: *International Journal of Mechanical Sciences* 13(7) (1971), pp. 615–625. DOI: `10.1016/0020-7403(71)90032-4`.

[17] S. K. Chan and I. S. Tuba. "A finite element method for contact problems of solid bodies–Part II. Application to turbine blade fastenings". In: *International Journal of Mechanical Sciences* 13(7) (July 1971), 627 (in IN7), 637–636-IN8-639. DOI: `10.1016/0020-7403(71)90033-6`.

[18] A. Francavilla and O. C. Zienkiewicz. "A note on numerical computation of elastic contact problems". In: *International Journal for Numerical Methods in Engineering* 9 (1975), pp. 913–924. DOI: `10.1002/nme.1620090410`.

[19] T. J. R. Hughes, R. L. Taylor, J. L. Sackman, A. Curnier, and W. Kanoknukulchai. "A finite element method for a class of contact-impact problems". In: *Computer Methods in Applied Mechanics and Engineering* 8 (1976), pp. 249–276. DOI: `10.1016/0045-7825(76)90018-9`.

[20] J. T. Stadter and R. O. Weiss. "Analysis of contact through finite element gaps". In: *Computers & Structures* 10(6) (1979), pp. 867–873. DOI: `10.1016/0045-7949(79)90055-5`.

[21] J. T. Oden. "Exterior Penalty Methods for Contact Problems in Elasticity". In: *Nonlinear Finite Element Analysis in Structural Mechanics: Proceedings of the Europe-U.S. Workshop*. Ed. by W. Wunderlich, E. Stein, and K. J. Bathe. Vol. 5. Ruhr-Universität Bochum, Germany: Springer Berlin Heidelberg, July 1980, pp. 655–665. DOI: `10.1007/978-3-642-81589-8_33`.

[22] K. J. Bathe and A. Chaudhary. "A solution method for planar and axisymmetric contact problems". In: *International Journal for Numeric* 21(1) (Jan. 1985), pp. 65–88. DOI: `10.1002/nme.1620210107`.

[23] J. O. Hallquist, G. L. Goudreau, and D. J. Benson. "Sliding interfaces with contact-impact in large-scale Lagrangian computations". In: *Computer Methods in Applied Mechanics and Engineering* 51(1–3) (Sept. 1985), pp. 107–137. DOI: `10.1016/0045-7825(85)90030-1`.

[24] J. C. Simo, P. Wriggers, and R. L. Taylor. "A perturbed Lagrangian formulation for the finite element solution of contact problems". In: *Computational Methods in Applied Mechanics and Engineering* 50 (1985), pp. 163–180. DOI: `10.1016/0045-7825(85)90088-X`.

[25] M. E. Gurtin, E. Fried, and L. Anand. *The Mechanics and Thermomechanics of Continua*. Cambridge University Press, 2010, p. 694. DOI: `10.1017/CBO9780511762956`.

[26] C. Truesdell and R. A. Toupin. *Principles of Classical Mechanics and Field Theory / Prinzipien der Klassischen Mechanik und Feldtheorie.* Ed. by S. Flügge. 1st ed. Vol. III/1. Encyclopedia of Physics / Handbuch der Physik. Springer-Verlag Berlin Heidelberg, 1960. ISBN: 978-3-642-45943-6. DOI: `10.1007/978-3-642-45943-6`.

[27] R. W. Ogden. *Non-Linear Elastic Deformations.* Dover Publications, 1997. ISBN: 0486696480.

[28] R. E. Jones and P. Papadopoulos. "A yield-limited Lagrange multiplier formulation for frictional contact". In: *International Journal for Numerical Methods in Engineering* 48(8) (2000), pp. 1127–1149. ISSN: 00295981. DOI: `10.1002/(SICI)1097-0207(20000720)48:8<1127::AID-NME937>3.0.CO;2-C`.

[29] T. A. Laursen. *Computational Contact and Impact Mechanics.* Springer, 2002. DOI: `10.1007/978-3-662-04864-1`.

[30] R. de Borst, M. A. Crisfield, J. J. C. Remmers, and C. V. Verhoosel. *Nonlinear Finite Element Analysis of Solids and Structures.* 2nd ed. John Wiley & Sons, 2012.

[31] T. A. Laursen. "The convected description in large deformation frictional contact problems". In: *International Journal of Solids and Structures* 31(5) (1994), pp. 669–681. ISSN: 00207683. DOI: `10.1016/0020-7683(94)90145-7`.

[32] P. Wriggers, T. V. Van, and E. Stein. "Finite element formulation of large deformation impact-contact problems with friction". In: *Computers & Structures* 37(3) (1990), pp. 319–331. DOI: `10.1016/0045-7949(90)90324-U`.

[33] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, 1999, p. 634.

[34] T. A. Laursen and J. C. Simo. "A continuum-based finite element formulation for the implicit solution of multibody, large deformation frictional contact problems". In: *International Journal for Numerical Methods in Engineering* 36 (1993), pp. 3451–3485. DOI: `10.1002/nme.1620362005`.

[35] P. Wriggers and J. C. Simo. "A general procedure for the direct computation of turning and bifurcation points". In: *International Journal for Numerical Methods in Engineering* 30 (1990), pp. 155–176. DOI: `10.1002/nme.1620300110`.

[36] J. Lubliner. "A maximum-dissipation principle in generalized plasticity". In: *Acta Mechanica* 52(3–4) (Sept. 1984), pp. 225–237. DOI: `10.1007/BF01179618`.

[37] A. Popp, M. W. Gee, and W. A. Wall. "A finite deformation mortar contact formulation using a primal-dual active set strategy". In: *International Journal for Numerical Methods in Engineering* 79(11) (Sept. 2009), pp. 1354–1391. DOI: `10.1002/nme.2614`.

[38] A. Popp, M. Gitterle, M. W. Gee, and W. A. Wall. "A dual mortar approach for 3D finite deformation contact with consistent linearization". In: *International Journal for Numerical Methods in Engineering* 83(11) (Sept. 2010), pp. 1428–1465. DOI: `10.1002/nme.2866`.

[39] A. Curnier and P. Alart. "A generalized newton method for contact problems with friction". In: *Journal de Mecanique Theorique et Appliquee* 7 (1988), pp. 67–82.

[40] T. Belytschko, W. K. Liu, and B. Moran. *Nonlinear Finite Element for Continua and Structures.* John Wiley & Sons, Ltd, 2000.

[41] M. Gitterle, A. Popp, M. W. Gee, and W. A. Wall. “Finite deformation frictional mortar contact using a semi-smooth Newton method with consistent linearization”. In: *International Journal for Numerical Methods in Engineering* 84 (2010), pp. 543–571. DOI: `10.1002/nme.2907`.

[42] L. Debnath and P. Mikusinski. *Introduction to Hilbert Spaces with Applications*. Elsevier, 2005.

[43] M. W. Heinstein and T. A. Laursen. “A three dimensional surface-to-surface projection algorithm for non-coincident domains”. In: *Communications in Numerical Methods in Engineering* 19 (2003), pp. 421–432. DOI: `10.1002/cnm.601`.

[44] M. A. Puso and T. A. Laursen. “A mortar segment-to-segment contact method for large deformation solid mechanics”. In: *Computer Methods in Applied Mechanics and Engineering* 193(6–8) (Feb. 2004), pp. 601–629. DOI: `10.1016/j.cma.2003.10.010`.

[45] G. Rebel, K. C. Park, and C. A. Felippa. “A contact formulation based on localized Lagrange multipliers: formulation and application to two-dimensional problems”. In: *International Journal for Numerical Methods in Engineering* 54 (2002), pp. 263–297. DOI: `10.1002/nme.426`.

[46] P. Wriggers. *Nonlinear Finite Element Methods*. Springer-Verlag Berlin Heidelberg, 2008, p. 560. ISBN: 978-3-540-71000-4. DOI: `10.1007/978-3-540-71001-1`.

[47] P. Wriggers. *Computational Contact Mechanics*. 2nd. Springer, 2006.

[48] B. M. Irons. “Engineering applications of numerical integration in stiffness methods”. In: *AIAA Journal* 4(11) (1966), pp. 2035–2037. DOI: `10.2514/3.3836`.

[49] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs. “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement”. In: *Computer Methods in Applied Mechanics and Engineering* 194(39–41) (2005), pp. 4135–4195. DOI: `10.1016/j.cma.2004.10.008`.

[50] İ. Temizer, P. Wriggers, and T. J. R. Hughes. “Contact treatment in isogeometric analysis with NURBS”. In: *Computational Methods in Applied Mechanics and Engineering* 200 (2011), pp. 1100–1112. DOI: `10.1016/j.cma.2010.11.020`.

[51] L. De Lorenzis, I. Temizer, P. Wriggers, and G. Zavarise. “A large deformation frictional contact formulation using NURBS-based isogeometric analysis”. In: *International Journal for Numerical Methods in Engineering* 87 (2011), pp. 1278–1300. DOI: `10.1002/nme.3159`.

[52] İ. Temizer. “A mixed formulation of mortar-based frictionless contact”. In: *Computational Methods in Applied Mechanics and Engineering* 223–224 (2012), pp. 173–185. DOI: `10.1016/j.cma.2012.02.017`.

[53] L. De Lorenzis, P. Wriggers, and G. Zavarise. “A mortar formulation for 3D large deformation contact using NURBS-based isogeometric analysis and the augmented Lagrangian method”. In: *Computational Mechanics* 49 (2012), pp. 1–20. DOI: `10.1007/s00466-011-0623-4`.

[54] İ. Temizer, P. Wriggers, and T. J. R. Hughes. “Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS”. In: *Computational Methods in Applied Mechanics and Engineering* 209–2012 (2012), pp. 115–128. DOI: `10.1016/j.cma.2011.10.014`.

[55] A. Iserles. *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, 1996.

[56] R. L. Taylor and P. Papadopoulos. “On a Patch Test for Contact Problems in Two Dimensions”. In: *Nonlinear Computational Mechanics*. Ed. by P. Wriggers and W. Wagner. Springer-Verlag, Berlin, 1991, pp. 690–702.

[57] M. A. Crisfield. “Re-visiting the contact patch test”. In: *International Journal for Numerical Methods in Engineering* 48 (2000), pp. 435–449. DOI: `10.1002/(SICI)1097-0207(20000530)48:3<435::AID-NME891>3.0.CO;2-V`.

[58] T. J. R. Hughes, R. L. Taylor, J. L. Sackman, and W. Kanoknukulchai. *Finite Element Formulation and Solution of Contact-Impact Problems in Continuum Mechanics-IV*. Tech. rep. UC SESM 76-4. University of California, Berkeley, July 1976.

[59] P. Wriggers and J. C. Simo. “A note on tangent stiffness for fully nonlinear contact problems”. In: *Communications in Applied Numerical Methods* 1(5) (1985), pp. 199–203. DOI: `10.1002/cnm.1630010503`.

[60] P. Wriggers and C. Miehe. “Contact constraints within coupled thermomechanical analysis – A finite element model”. In: *Computational Methods in Applied Mechanics and Engineering* 113 (1994), pp. 301–319. DOI: `10.1016/0045-7825(94)90051-5`.

[61] G. Zavarise and P. Wriggers. “A Segment-to-Segment Contact Strategy”. In: *Mathematical and Computer Modelling* 28(4–8) (1998), pp. 497–515. DOI: `10.1016/S0895-7177(98)00138-1`.

[62] G. Zavarise and L. D. Lorenzis. “A modified node-to-segment algorithm passing the contact patch test”. In: *International Journal for Numerical Methods in Engineering* 79 (2009), pp. 379–416. DOI: `10.1002/nme.2559`.

[63] N. El-Abbasi and K.-J. Bathe. “Stability and patch test performance of contact discretizations and a new solution algorithm”. In: *Computers* 79 (2001), pp. 1473–1486. DOI: `10.1016/S0045-7949(01)00048-7`.

[64] P. Papadopoulos and R. L. Taylor. “A mixed formulation for the finite element solution of contact problems”. In: *Computer Methods in Applied Mechanics and Engineering* 94-3 (1992), pp. 373–389. DOI: `10.1016/0045-7825(92)90061-N`.

[65] Y. Maday, C. Mavriplis, and A. T. Patera. “Nonconforming mortar element methods: application to spatial discretization”. In: *Domain Decomposition Methods for Partial Differential Equations: 3rd: International Symposium Proceedings*. Houston, Texas: SIAM, Mar. 1989, pp. 392–418.

[66] G. Anagnostou, Y. Maday, C. Mavriplis, and A. T. Patera. “On the mortar element method: Generalizations and implementation”. In: *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*. Phila- Phila- Philadelphia, PA: SIAM, 1990, pp. 157–173.

[67] C. Bernardi, Y. Maday, and A. Patera. “A new nonconforming approach to domain decomposition: the mortar element method”. In: *Nonlinear Partial Differential Equations and their Applications*. Ed. by H. Brezis and J. Lions. London/New York: Pitman/Wiley, 1994, pp. 13–51.

[68] F. B. Belgacem, P. Hild, and L. Patrick. “Approximation of the unilateral contact problem by the mortar finite element method”. In: *Comptes Rendus de l’Académie des Sciences - Series I - Mathematics* 324(1) (Jan. 1997), pp. 123–127. DOI: `10.1016/S0764-4442(97)80115-2`.

[69] F. B. Belgacem, P. Hild, and L. Patrick. “The mortar finite element method for contact problems”. In: *Mathematical and Computer Modelling* 28(4–8) (Aug. 1998), pp. 263–271. DOI: `10.1016/S0895-7177(98)00121-6`.

[70] B. I. Wohlmuth. “A Mortar Finite Element Method Using Dual Spaces for the Lagrange Multiplier”. In: *SIAM Journal on Numerical Analysis* 38(3) (2000), pp. 989–1012. DOI: `10.1137/S0036142999350929`.

[71] B. I. Wohlmuth. *Discretization Methods and Iterative Solvers Based on Domain Decomposition.* Lecture Notes in Computational Science and Engineering. Berlin Heidelberg: Springer-Verlag, 2001.

[72] P. Hild. “Numerical implementation of two nonconforming finite element methods for unilateral contact”. In: *Computer Methods in Applied Mechanics and Engineering* 184(1) (Mar. 2000), pp. 99–123. DOI: `10.1016/S0045-7825(99)00096-1`.

[73] T. W. McDevitt and T. A. Laursen. “A mortar-fnite element formulation for frictional contact problems”. In: *International Journal for Numerical Methods in Engineering* 48(10) (Aug. 2000), pp. 1525–1547. DOI: `10.1002/1097-0207(20000810)48:10$<$1525::AID-NME953$>$3.0.CO;2-Y`.

[74] T. Cichosz and M. Bischoff. “Consistent treatment of boundaries with mortar contact formulations using dual Lagrange multipliers”. In: *Computational Methods in Applied Mechanics and Engineering* 200 (2011), pp. 1317–1332. DOI: `10.1016/j.cma.2010.11.004`.

[75] B. Yang, T. A. Laursen, and X. Meng. “Two dimensional mortar contact methods for large deformation frictional sliding”. In: *International Journal for Numerical Methods in Engineering* 62(9) (Mar. 2005), pp. 1183–1225. DOI: `10.1002/nme.1222`.

[76] S. Hüeber, A. Matei, and B. I. Wohlmuth. “Efficient Algorithms for Problems with Friction”. In: *SIAM Journal on Scientific Computing* 29(1) (2007), pp. 70–92. DOI: `10.1137/050634141`.

[77] M. A. Puso and T. A. Laursen. “Mesh tying on curved interfaces in 3D”. In: *Engineering Computations* 20(3) (2003), pp. 305–321. DOI: `10.1108/02644400310467225`.

[78] M. A. Puso. “A 3D mortar method for solid mechanics”. In: *International Journal for Numerical Methods in Engineering* 59 (2004), pp. 315–336. DOI: `10.1002/nme.865`.

[79] S. Hartmann and E. Ramm. “A mortar based contact formulation for non-linear dynamics using dual Lagrange multipliers”. In: *Finite Elements in Analysis and Design* 44(5) (Mar. 2008), pp. 245–258. DOI: `10.1016/j.finel.2007.11.018`.

[80] A. Popp, B. I. Wohlmuth, M. W. Gee, and W. A. Wall. “Dual Quadratic Mortar Finite Element Methods for 3D Finite Deformation Contact”. In: *SIAM Journal on Scientific Computing* 34(4) (2012), B421–B446. DOI: `10.1137/110848190`.

[81] M. A. Puso and T. A. Laursen. “A mortar segment-to-segment frictional contact method for large deformations”. In: *Computer Methods in Applied Mechanics and Engineering* 193(45–47) (Nov. 2004), pp. 4891–4913. DOI: `10.1016/j.cma.2004.06.001`.

[82] M. A. Puso, T. A. Laursen, and J. Solberg. “A segment-to-segment mortar contact method for quadratic elements and large deformations”. In: *Computational Methods in Applied Mechanics and Engineering* 197 (2008), pp. 555–566. DOI: `10.1016/j.cma.2007.08.009`.

[83] K. A. Fischer and P. Wriggers. “Frictionless 2D contact formulations for finite deformations based on the mortar method”. In: *Computational Mechanics* 36 (2005), pp. 226–244. DOI: `10.1007/s00466-005-0660-y`.

[84] K. A. Fischer and P. Wriggers. “Mortar based frictional contact formulation for higher order interpolations using the moving friction cone”. In: *Computational Methods in Applied Mechanics and Engineering* 195 (2006), pp. 5020–5036. DOI: `10.1016/j.cma.2005.09.025`.

[85] M. Tur, F. J. Fuenmayor, and P. Wriggers. “A mortar-based frictional contact formulation for large deformations using Lagrange multipliers”. In: *Computational Methods in Applied Mechanics and Engineering* 198(37–40) (Aug. 2009), pp. 2860–2873. DOI: `10.1016/j.cma.2009.04.007`.

[86] H. M. Hilber, T. J. R. Hughes, and R. L. Taylor. “Improved numerical dissipation for time integration algorithms in structural dynamics”. In: *Earth Engineering Structural Dynamics* 5(3) (July 1977), pp. 283–292. DOI: `10.1002/eqe.4290050306`.

[87] N. M. Newmark. “A method of computation for structural dynamics”. In: *Journal of the Engineering Mechanics Division* 85(3) (1959), pp. 67–94.

[88] D. J. Benson and J. O. Hallquist. “A Single Surface Contact Algorithm for the Post-buckling Analysis of Shell Structures”. In: *Computer Methods in Applied Mechanics and Engineering* 78(2) (Jan. 1990), pp. 141–163. DOI: `10.1016/0045-7825(90)90098-7`.

[89] M. Oldenburg and L. Nilsson. “The position code algorithm for contact searching”. In: *International Journal for Numerical Methods in Engineering* 37(3) (Feb. 1994), pp. 359–386. DOI: `10.1002/nme.1620370302`.

[90] A. Munjiza and K. R. F. Andrews. “NBS contact detection algorithm for bodies of similar size”. In: *International Journal for Numerical Methods in Engineering* 43(1) (Sept. 1998), pp. 131–149. DOI: `10.1002/(SICI)1097-0207(19980915)43:1<131::AID-NME447>3.0.CO;2-S`.

[91] W. Fujun, C. Jiangang, and Y. Zhenhan. “A contact searching algorithm for contact-impact problems”. In: *Acta Mechanica Sinica* 16 (2000). DOI: `10.1007/BF02487690`.

[92] H. Chen, Z. Lei, and M. Zang. “LC-Grid: a linear global contact search algorithm for finite element analysis”. In: *Computational Mechanics* 54(5) (Nov. 2014), pp. 1285–1301. DOI: `10.1007/s00466-014-1058-5`.

[93] B. Yang and T. A. Laursen. “A contact searching algorithm including bounding volume trees applied to finite sliding mortar formulations”. In: *Computational Mechanics* 41(2) (Jan. 2008), pp. 189–205. DOI: `10.1007/s00466-006-0116-z`.

[94] A. Konyukhov and K. Schweizerhof. “On the solvability of closest point projection procedures in contact analysis: Analysis andsolution strategy for surfaces of arbitrary geometry”. In: *Comput. Methods Appl. Mech. Engrg.* 197(33–40) (June 2008), pp. 3045–3056. DOI: `10.1016/j.cma.2008.02.009`.

[95] A. B. Chaudhary and K. J. Bathe. “A solution method for static and dynamic analysis of three-dimensional contact problems with friction”. In: *Computers and Structures* 24(6) (1986), pp. 855–873. DOI: `10.1016/0045-7949(86)90294-4`.

[96] Y. T. Keum, E. Nakamachi, R. H. Wagoner, and J. K. Lee. “Compatible description of tool surfaces and FEM meshes for analysing sheet forming operations”. In: *International Journal for Numerical Methods in Engineering* 30(8) (Dec. 1990), pp. 1471–1502. DOI: `10.1002/nme.1620300810`.

[97] S. P. Wang and E. Nakamachi. “The inside-outside contact search algorithm for finite element analysis”. In: *International Journal for Numerical Methods in Engineering* 40(19) (Oct. 1997), pp. 3665–3685. DOI: `10.1002/(SICI)1097-0207(19971015)40:19<3665::AID-NME234>3.0.CO;2-K`.

[98] W. Fujun, C. Jiangang, and Y. Zhenhan. “FFS contact searching algorithm for dynamic finite element analysis”. In: *International Journal for Numerical Methods in Engineering* 52(7) (Nov. 2001), pp. 655–672. DOI: `10.1002/nme.221`.

[99] K. Ko and T. Sakkalis. “Orthogonal projection of points in CAD/CAM applications: an overview”. In: *Journal of Computational Design and Engineering* 1(2) (Feb. 2014), pp. 116–127. DOI: `10.7315/JCDE.2014.012`.

[100] S.-M. Hu and J. Wallner. “A second order algorithm for orthogonal projection onto curves and surfaces”. In: *Computer Aided Geometric Design* 22(3) (Mar. 2005), pp. 251–260. DOI: `10.1016/j.cagd.2004.12.001`.

[101] X.-M. Liu, L. Yang, J.-H. Yong, H.-J. Gu, and J.-G. Sun. “A torus patch approximation approach for point projection on surfaces”. In: *Computer Aided Geometric Design* 26(5) (June 2009), pp. 593–598. DOI: `10.1016/j.cagd.2009.01.004`.

[102] L. A. Piegel and W. Tiller. “Parametrization for surface fitting in reverse engineering”. In: *Computer-Aided Design* 33(8) (July 2001), pp. 593–603. DOI: `10.1016/S0010-4485(00)00103-2`.

[103] Y. L. Ma and W. Hewitt. “Point inversion and projection for NURBS curve and surface: Control polygon approach”. In: *Computer Aided Geometric Design* 20(2) (May 2003), pp. 79–99. DOI: `10.1016/S0167-8396(03)00021-9`.

[104] I. Selimovic. “Improved algorithms for the projection of points on NURBS curves and surfaces”. In: *Computer Aided Geometric Design* 23 (July 2006), pp. 439–445. DOI: `10.1016/j.cagd.2006.01.007`.

[105] X.-D. Chen, J.-H. Yong, G. Wang, J.-C. Paul, and G. Xu. “Computing the minimum distance between a point and a NURBS curve”. In: *Computer-Aided Design* 40 (Oct. 2008), pp. 1051–1054. DOI: `10.1016/j.cad.2008.06.008`.

[106] Y.-T. Oh, Y.-J. Kim, J. Lee, M.-S. Kim, and G. Elber. “Efficient point-projection to freeform curves and surfaces”. In: *Computer Aided Geometric Design* 29 (Apr. 2012), pp. 242–254. DOI: `10.1016/j.cagd.2011.04.002`.

[107] L. De Lorenzis, P. Wriggers, and T. J. R. Hughes. “Isogeometric contact: a review”. In: *GAMM-Mitteilungen* 37(1) (July 2014), pp. 85–123. DOI: `10.1002/gamm.201410005`.

[108] S. Zhuang, M. G. Lee, Y. T. Keum, J. H. Kim, and R. H. Wagoner. “Improved contact procedure for implicit finite element sheet forming simulation”. In: *International Journal for Numerical Methods in Engineering* 83(13) (Sept. 2010), pp. 1759–1779. DOI: `10.1002/nme.2890`.

[109] A. M. Aragón, V. A. Yastrebov, and J.-F. Molinari. "A constrained-optimization methodology for the detection phase in contact mechanics simulations". In: *International Journal for Numerical Methods in Engineering* 96 (2013), pp. 323–338. DOI: `10.1002/nme.4561`.

[110] M. Dowell and P. Jarratt. "A modified regula falsi method for computing the root of an equation". In: *BIT Numerical Mathematics* 11(2) (1971), pp. 168–174. DOI: `10.1007/BF01934364`.

[111] A. L. Peressini, F. E. Sulliven, and J. J. Uhl. *The Mathematics in Nonlinear Programming.* Spinger, 1988.

[112] W. Spendley, G. R. Hext, and F. R. Himsworth. "Sequential application of simplex designs in optimisation and evolutionary operation". In: *Technometrics* 4(4) (1962), pp. 441–461. DOI: `10.1080/00401706.1962.10490033`.

[113] H. Matthies and G. Strang. "The solution of nonlinear finite element equations". In: *International Journal for Numerical Methods in Engineering* 14(11) (1979), pp. 1613–1626. DOI: `10.1002/nme.1620141104`.

[114] T. A. Laursen and B. N. Maker. "An augmented Lagrangian quasi-Newton solver for constrained nonlinear finite element applications". In: *International Journal for Numerical Methods in Engineering* 38(21) (Nov. 1995), pp. 3571–3590. DOI: `10.1002/nme.1620382103`.

[115] M. L. Wilkins. "Calculation of Elastic-Plastic Flow". In: *Methods of Computational Physics.* Ed. by B. Alder, S. Fernback, and M. Rotenberg. Vol. 3. Academic Press, 1964.

[116] A. E. Giannakopoulos. "The return mapping method for the integration of friction constitutive relations". In: *Computers & Structures* 32(1) (1989), pp. 157–167. DOI: `10.1016/0045-7949(89)90081-3`.

[117] A. L. Eterovic and K. J. Bathe. "An interface interpolation scheme for quadratic convergence in the finite element analysis of contact problems". In: *Nonlinear Computational Mechanics: State of the Art.* Ed. by P. Wriggers and W. Wagner. 1st ed. Springer-Verlag Berlin, Heidelberg, Sept. 1991, pp. 703–715.

[118] V. Padmanabhan and T. A. Laursen. "A framework for development of surface smoothing procedures in large deformation frictional contact analysis". In: *Finite Elements in Analysis and Design* 37(3) (Mar. 2001), pp. 173–198. DOI: `10.1016/S0168-874X(00)00029-9`.

[119] P. Wriggers, L. Krstulović-Opara, and J. Korelc. "Smooth C^1-interpolations for two-dimensional frictional contact problems". In: *International Journal for Numerical Methods in Engineering* 51(12) (May 2001), pp. 1469–1495. DOI: `10.1002/nme.227`.

[120] M. Tur, E. Giner, F. J. Fuenmayor, and P. Wriggers. "2D contact smooth formulation based on the mortar method". In: *Computational Methods in Applied Mechanics and Engineering* 247–248 (Nov. 2012), pp. 1–14. DOI: `10.1016/j.cma.2012.08.002`.

[121] N. El-Abbasi, S. A. Meguid, and A. Czekanski. "On the modelling of smooth contact surfaces using cubic splines". In: *International Journal for Numerical Methods in Engineering* 50(4) (Feb. 2001), pp. 953–967. DOI: `10.1002/1097-0207(20010210)50:4<953::AID-NME64>3.0.CO;2-P`.

[122] M. Stadlar, G. A. Holzapfel, and J. Korelc. "C^n continuous modelling of smooth contact surfaces using NURBS and application to 2D problems". In: *International Journal for Numerical Methods in Engineering* 57(15) (Aug. 2003), pp. 2177–2203. DOI: `10.1002/nme.776`.

[123] D. Chamoreta, P. Saillardb, A. Rassineuxc, and J. M. Bergheaua. “New smoothing procedures in contact mechanics”. In: *Journal of Computational and Applied Mathematics* 168(1–2) (July 2004), pp. 107–116. DOI: `10.1016/j.cam.2003.06.007`.

[124] J. J. Muñoz. “Modelling unilateral frictionless contact using the null-space method and cubic B-Spline interpolation”. In: *Computer Methods in Applied Mechanics and Engineering* 197(9–12) (Feb. 2008), pp. 979–993. DOI: `10.1016/j.cma.2007.09.022`.

[125] S. Stupkiewicz, J. Lengiewicz, and J. Korelc. “Sensitivity analysis for frictional contact problems in the augmented Lagrangian formulation”. In: *Computer Methods in Applied Mechanics and Engineering* 199(33–36) (July 2010), pp. 2165–2176. DOI: `10.1016/j.cma.2010.03.021`.

[126] J. Lengiewicz, J. Korelc, and S. Stupkiewicz. “Automation of finite element formulations for large deformation contact problems”. In: *International Journal for Numerical Methods in Engineering* 85(10) (Mar. 2011), pp. 1252–1279. DOI: `10.1002/nme.3009`.

[127] M. Stadlar and G. A. Holzapfel. “Subdivision schemes for smooth contact surfaces of arbitrary mesh topology in 3D”. In: *International Journal for Numerical Methods in Engineering* 60(7) (June 2004), pp. 1161–1195. DOI: `10.1002/nme.1001`.

[128] M. A. Puso and T. A. Laursen. “A 3D contact smoothing method using Gregory patches”. In: *International Journal for Numerical Methods in Engineering* 54(8) (July 2002), pp. 1161–1194. DOI: `10.1002/nme.466`.

[129] D. M. Neto, M. C. Oliveira, L. F. Menezes, and J. L. Alves. “A contact smoothing method for arbitrary surface meshes using Nagata patches”. In: *Computer Methods in Applied Mechanics and Engineering* 299 (Feb. 2016), pp. 283–315. DOI: `10.1016/j.cma.2015.11.011`.

[130] T. Belytschko, W. J. T. Daniel, and G. Ventura. “A monolithic smoothing-gap algorithm for contact-impact based on the signed distance function”. In: *International Journal for Numerical Methods in Engineering* 55(1) (Sept. 2002), pp. 101–125. DOI: `10.1002/nme.568`.

[131] A. Konyukhov and K. Schweizerhof. “Incorporation of contact for high-order finite elements in covariant form”. In: *Computer Methods in Applied Mechanics and Engineering* 198(13–14) (Mar. 2009), pp. 1213–1223. DOI: `10.1016/j.cma.2008.04.023`.

[132] R. A. Sauer. “Local finite element enrichment strategies for 2D contact computations and a corresponding post-processing scheme”. In: *Computational Mechanics* 52 (2013), pp. 301–319. DOI: `10.1007/s00466-012-0813-8`.

[133] C. J. Corbett and R. A. Sauer. “NURBS-enriched contact finite elements”. In: *Computer Methods in Applied Mechanics and Engineering* 275 (June 2014), pp. 55–75. DOI: `10.1016/j.cma.2014.02.019`.

[134] C. J. Corbett and R. A. Sauer. “Three-dimensional isogeometrically enriched finite elements for frictional contact and mixed-mode debonding”. In: *Computer Methods in Applied Mechanics and Engineering* 284 (Feb. 2015), pp. 781–806. DOI: `10.1016/j.cma.2014.10.025`.

[135] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, 2009.

[136] J. Lu. “Isogeometric contact analysis: Geometric basis and formulation for frictionless contact”. In: *Comput. Methods Appl. Mech. Engrg.* 200 (2010), pp. 726–741. DOI: `10.1016/j.cma.2010.10.001`.

[137] İ. Temizer and P. Wriggers. "Inelastic analysis of granular interfaces via computational contact homogenization". In: *International Journal for Numerical Methods in Engineering* 84 (2010), pp. 883–915. DOI: `10.1002/nme.2921`.

[138] İ. Temizer, M. M. Abdalla, and Z. Gürdal. "An interior point method for isogeometric contact". In: *Computer Methods in Applied Mechanics and Engineering* 276 (July 2014), pp. 589–611. DOI: `10.1016/j.cma.2014.03.018`.

[139] J.-Y. Kim and S.-K. Youn. "Isogeometric contact analysis using mortar method". In: *International Journal for Numerical Methods in Engineering* 89 (2011), pp. 1559–1581. DOI: `10.1002/nme.3300`.

[140] M. E. Matzen, T. Cichosz, and M. Bischoff. "A point to segment contact formulation for isogeometric, NURBS based finite elements". In: *Computer Methods in Applied Mechanics and Engineering* 255 (Mar. 2013), pp. 27–39. DOI: `10.1016/j.cma.2012.11.011`.

[141] M. E. Matzen and M. Bischoff. "A weighted point-based formulation for isogeometric contact". In: *Computer Methods in Applied Mechanics and Engineering* 308 (Aug. 2016), pp. 73–95. DOI: `10.1016/j.cma.2016.04.010`.

[142] M. Dittmann, M. Franke, İ. Temizer, and C. Hesch. "Isogeometric Analysis and thermomechanical Mortar contact problems". In: *Computer Methods in Applied Mechanics and Engineering* 274 (June 2014), pp. 192–212. DOI: `10.1016/j.cma.2014.02.012`.

[143] A. Seitz et al. "Isogeometric dual mortar methods for computational contact mechanics". In: *Computer Methods in Applied Mechanics and Engineering* 301 (Apr. 2016), pp. 259–280. DOI: `10.1016/j.cma.2015.12.018`.

[144] F. Auricchio, L. B. a. da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli. "Isogeometric collocation methods". In: *Mathematical Models and Methods in Applied Sciences* 20(11) (Nov. 2010), pp. 2075–2107. DOI: `10.1142/S0218202510004878`.

[145] F. Auricchio, L. Beirão da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli. "Isogeometric collocation for elastostatics and explicit dynamics". In: *Computational Methods in Applied Mechanics and Engineering* 249–252 (Dec. 2012), pp. 2–14. DOI: `10.1016/j.cma.2012.03.026`.

[146] L. De Lorenzis, J. A. Evans, T. J. R. Hughes, and A. Rea. "Isogeometric collocation: Neumann boundary conditions and contact". In: *Computer Methods in Applied Mechanics and Engineering* 284 (Feb. 2015), pp. 21–54. DOI: `10.1016/j.cma.2014.06.037`.

[147] R. Kruse, N. Nguyen-Thanh, L. De Lorenzis, and T. J. R. Hughes. "Isogeometric collocation for large deformation elasticity and frictional contact problems". In: *Computer Methods in Applied Mechanics and Engineering* 296 (Nov. 2015), pp. 73–112. DOI: `10.1016/j.cma.2015.07.022`.

[148] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. "T-splines and T-NURCCs". In: *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2003* 22(3) (2003), pp. 477–484. DOI: `10.1145/882262.882295`.

[149] T. W. Sederberg et al. "T-spline simplification and local refinement". In: *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2004* 23(3) (Aug. 2004), pp. 276–283. DOI: `10.1145/1015706.1015715`.

[150] R. Dimitri et al. “Isogeometric large deformation frictionless contact using T-splines”. In: *Computer Methods in Applied Mechanics and Engineering* 269 (Feb. 2014), pp. 394–414. DOI: `10.1016/j.cma.2013.11.002`.

[151] A. V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon. “A hierarchical approach to adaptive local refinement in isogeometric analysis”. In: *Computer Methods in Applied Mechanics and Engineering* 200(49–52) (Dec. 2011), pp. 3554–3567. DOI: `10.1016/j.cma.2011.09.004`.

[152] İ. Temizer and C. Hesch. “Hierarchical NURBS in frictionless contact”. In: *Computer Methods in Applied Mechanics and Engineering* 299 (Feb. 2016), pp. 161–186. DOI: `10.1016/j.cma.2015.11.006`.

[153] C. Hesch, M. Franke, M. Dittmann, and İ. Temizer. “Hierarchical NURBS and a higher-order phase-field approach to fracture for finite-deformation contact problems”. In: *Computer Methods in Applied Mechanics and Engineering* 301 (Apr. 2016), pp. 242–258. DOI: `10.1016/j.cma.2015.12.011`.

[154] S. Hartmann and D. J. Benson. “Mass scaling and stable time step estimates for isogeometric analysis”. In: *International Journal for Numerical Methods in Engineering* 102(3–4) (Apr. 2015), pp. 671–687. DOI: `10.1002/nme.4719`.

[155] M. Pagani, S. Reesea, and U. Peregob. “Computationally efficient explicit nonlinear analyses using reduced integration-based solid-shell finite elements”. In: *Computer Methods in Applied Mechanics and Engineering* 268 (Jan. 2014), pp. 141–159. DOI: `10.1016/j.cma.2013.09.005`.

[156] E. Hinton, T. Rock, and O. C. Zienkiewicz. “A note on mass lumping and related processes in the finite element method”. In: *Earthquake Engineering & Structural Dynamics* 4(3) (Jan. 1976), pp. 245–249. DOI: `10.1002/eqe.4290040305`.

[157] C. Adam, S. Bouabdallah, M. Zarroug, and H. Maitournam. “Stable time step estimates for NURBS-based explicit dynamics”. In: *Computational Methods in Applied Mechanics and Engineering* 295 (Oct. 2015), pp. 581–605. DOI: `10.1016/j.cma.2015.03.017`.

[158] D. Wang, W. Liu, and H. Zhang. “Novel higher order mass matrices for isogeometric structural vibration analysis”. In: *Computer Methods in Applied Mechanics and Engineering* 260 (June 2013), pp. 92–108. DOI: `10.1016/j.cma.2013.03.011`.

[159] L. Olovsson, M. Unosson, and K. Simonsson. “Selective mass scaling for thin walled structures modeled with tri-linear solid elements”. In: *Computational Mechanics* 34(2) (July 2004), pp. 134–136. DOI: `10.1007/s00466-004-0560-6`.

[160] L. Olovsson and K. Simonsson. “Selective mass scaling for explicit finite element analyses”. In: *International Journal for Numerical Methods in Engineering* 63(10) (July 2005), pp. 1436–1445. DOI: `10.1002/nme.1293`.

[161] L. Olovsson and K. Simonsson. “Iterative solution technique in selective mass scaling”. In: *Communications in Numerical Methods in Engineering* 22(1) (Jan. 2006), pp. 77–82. DOI: `10.1002/cnm.806`.

[162] A. Tkachuk and M. Bischoff. “Variational methods for selective mass scaling”. In: *Computational Mechanics* 52 (Sept. 2013), pp. 563–570. DOI: `10.1007/s00466-013-0832-0`.

[163] G. Cocchetti, M. Pagani, and U. Perego. “Selective mass scaling and critical time-step estimate for explicit dynamics analyses with solid-shell elements”. In: *Computers & Structures* 127 (Oct. 2013), pp. 39–52. DOI: `10.1016/j.compstruc.2012.10.021`.

[164] A. Tkachuk and M. Bischoff. “Direct and sparse construction of consistent inverse mass matrices: general variational formulation and application to selective mass scaling”. In: *International Journal for Numerical Methods in Engineering* 101(6) (Feb. 2015), pp. 435–469. DOI: `10.1002/nme.4805`.

[165] A. Tkachuk and M. Bischoff. “Local and global strategies for optimal selective mass scaling”. In: *Computational Mechanics* 53 (June 2014), pp. 1197–1207. DOI: `10.1007/s00466-013-0961-5`.

[166] L. A. Piegel and W. Tiller. *The NURBS Book (Monographs in Visual Communication)*. 2nd ed. Springer-Verlag, 1997.

[167] S. Ilanko. “Introducing the use of positive and negative inertial functions in asymptotic modelling”. In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. Vol. 461. The Royal Society. 2005, pp. 2545–2562. DOI: `10.1098/rspa.2005.1472`.

[168] S. Ilanko and L. E. Monterrubio. “Bipenalty method from a frequency domain perspective”. In: *International Journal for Numerical Methods in Engineering* 90(10) (2012), pp. 1278–1291. DOI: `10.1002/nme.4266`.

[169] H. Askes, M. Caramés-Saddler, and A. Rodríguez-Ferran. “Bipenalty method for time domain computational dynamics”. In: *Proceedings of the Royal Society A* 466 (2010), pp. 1389–1408. DOI: `10.1098/rspa.2009.0350`.

[170] J. Hetherington, A. Rodríguez-Ferran, and H. Askes. “A new bipenalty formulation for ensuring time step stability in time domain computational dynamics”. In: *International Journal for Numerical Methods in Engineering* 90 (2012), pp. 269–286. DOI: `10.1002/nme.3314`.

[171] J. Hetherington, A. Rodríguez-Ferran, and H. Askes. “The bipenalty method for arbitrary multipoint constraints”. In: *International Journal for Numerical Methods in Engineering* 93 (2013), pp. 465–482. DOI: `10.1002/nme.4389`.

[172] J. Hetherington and H. Askes. “A mass matrix formulation for cohesive surface elements”. In: *Theoretical and Applied Fracture Mechanics* 69 (Feb. 2014), pp. 110–117. DOI: `10.1016/j.tafmec.2013.11.011`.

[173] H. H. Rosenbrock. “An automatic method for finding the greatest or least value of a function”. In: *The Computer Journal* 3(3) (1960), pp. 175–184. DOI: `10.1093/comjnl/3.3.175`.

[174] *PMD version f77.10, VAMET/Institute of Thermomechanics*. 2011.

[175] T. Belytschko and M. O. Neal. “Contact-impact by the pinball algorithm with penalty and Lagrangian methods”. In: *International Journal for Numerical Methods in Engineering* 31 (1991), pp. 547–572. DOI: `10.1002/nme.1620310309`.

[176] J. Bonet and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Ed. by 1. Cambridge University Press, 1997.

[177] S. Hartmann, J. Oliver, R. Weyler, J. C. Cante, and J. A. Hernández. “A contact domain method for large deformation frictional contact problems. Part 2: Numerical aspects”. In: *Computer Methods in Applied Mechanics and Engineering* 198(33–36) (July 2009), pp. 2607–2631. DOI: `10.1016/j.cma.2009.03.009`.

[178] V. A. Yastrebov. *Numerical Methods in Contact Mechanics*. Ed. by B. Piotr. London (UK) and Hoboken (USA): ISTE Ltd and John Wiley & Sons, Inc., 2013, p. 391. ISBN: 9781118647974. DOI: `10.1002/9781118647974`.

[179] O. C. Zienkiewicz, R. L. Taylor, and D. Fox. *The Finite Element Method for Solid and Structural Mechanics*. 7th ed. Butterworth-Heinemann, 2014, p. 624. ISBN: 978-1-85617-634-7. DOI: `10.1016/B978-1-85617-634-7.00016-8`.

[180] L. Jiang and R. J. Rogers. "Effects of spatial discretization on dispersion and spurious oscillations in elastic wave propagation". In: *International Journal for Numerical Methods in Engineering* 29(6) (May 2005), pp. 1205–1218. DOI: `10.1002/nme.1620290607`.

[181] I. Huněk. "On a penalty formulation for contact-impact problems". In: *Computers & Structures* 48(2) (July 1993), pp. 193–203. ISSN: 0045-7949. DOI: `http://dx.doi.org/10.1016/0045-7949(93)90412-7`.

[182] Y. Miyazaki and K. C. Park. "A formulation of conserving impact system based on localized Lagrange multipliers". In: *International Journal for Numerical Methods in Engineering* 68 (2006), pp. 98–124. DOI: `10.1002/nme.1703`.

[183] Z.-H. Zhong. *Finite element procedures for contact-impact problems*. Oxford University Press, 1993.

[184] R. Kolman, J. Plešek, M. Okrouhlík, and D. Gabriel. "Grid dispersion analysis of plane square biquadratic serendipity finite elements in transient elastodynamics". In: *International Journal for Numerical Methods in Engineering* 96(1) (Oct. 2013), pp. 1–28. DOI: `10.1002/nme.4539`.

[185] V. Bína and R. Pech. "Probabilistic approach to description of the creep strain characteristic and prediction for long life-time." In: *Materials for Advanced Power Engineering 1994*. Ed. by D. Coustouradis et al. Boston: Springer Netherlands, 1994, pp. 601–610.

[186] J. Plešek and J. Korouš. "Explicit integration method with time step control for viscoplasticity and creep". In: *Advances in Engineering Software* 33(7–10) (July 2002), pp. 621–630. DOI: `10.1016/S0965-9978(02)00071-6`.

Author's peer-reviewed journal papers

[187] R. Kolman, S. Sorokin, B. Bastl, J. Kopačka, and J. Plešek. "Isogeometric analysis of free vibration of simple shaped elastic samples". In: *The Journal of the Acoustical Society of America* 137(4) (2015), pp. 2089–2100. ISSN: 0001-4966. DOI: `10.1121/1.4916199`.

[188] S. Sorokin, R. Kolman, and J. Kopačka. "The boundary integral equations method for analysis of high-frequency vibrations of an elastic layer". In: *Archive of Applied Mechanics* 87(4) (2016), pp. 1–14. DOI: `10.1007/s00419-016-1220-y`.

[189] J. Kopačka, D. Gabriel, J. Plešek, and M. Ulbin. "Assessment of methods for computing the closest point projection, penetration, and gap functions in contact searching problems". In: *International Journal for Numerical Methods in Engineering* 105(11) (2016), pp. 803–833. DOI: `10.1002/nme.4994`.

[190] J. Kopačka et al. “On stability and reflection-transmission analysis of the bipenalty method in contact-impact problems: a one-dimensional, homogeneous case study”. In: *International Journal for Numerical Methods in Engineering* (Oct. 2017), pp. 1–30. ISSN: 00295981. DOI: `10.1002/nme.5712`.

Author's book chapters

[191] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “A large deformation frictionless contact treatment in NURBS-based isogeometric analysis”. In: *Advances in Computational Coupling and Contact Mechanics*. Ed. by L. Rodríguez-Tembleque and M. H. F. Aliabadi. London, UK: World Scientific Publishing, 2018, pp. 1–35. ISBN: 978-1-78634-477-9.

Author's conference papers

[192] J. Kopačka, D. Gabriel, R. Kolman, J. Plešek, and M. Ulbin. “Studies in numerical stability of explicit contact-impact algorithm to the finite element solution of wave propagation problems”. In: *ECCOMAS Thematic Conference - COMPDYN 2013: 4th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering, Proceedings - An IACM Special Interest Conference*. Kos Island, Greece, 2013, pp. 787–800.

[193] J. Kopačka, R. Kolman, D. Gabriel, and J. Plešek. “Frictionless contact of elastic bodies: comparison of treatment in finite element analysis and isogeometric analysis”. In: *Engineering Mechanics 2012*. Ed. by J. Náprstek and C. Fischer. Svratka, Czech Republic: Institute of Theoretical and Applied Mechanics, Academy of Sciences of the Czech Republic, v.v.i., 2012, pp. 665–671.

[194] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “A symmetry preserving contact treatment in isogeometric analysis”. In: *Proceedings of the 7th GACM Colloquium on Computational Mechanics for Young Scientists from Academia and Industry*. Ed. by M. von Scheven, M.-A. Keip, and N. Karajan. Stuttgart, Germany: Institute for Structural Mechanics, University of Stuttgart, 2017, pp. 194–198. DOI: `10.18419/opus-9334`.

[195] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “Isogeometric contact analysis: a study of an explicit dynamic contact algorithm”. In: *Výpočty konstrukcí metodou konečných prvků 2014*. Ed. by V. Laš and J. Krystek. Pilsen, Czech Republiv: University of West Bohemia, Faculty of Applied Sciences, Department of Mechanics, 2014, pp. 1–6.

[196] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “On the contact pressure oscillations of an isogeometric contact-impact algorithm”. In: *YIC GACM 2015 - Book of Abstracts*. Aachen, Germany: RWTH Aachen University, 2015, pp. 1–4.

[197] J. Kopačka, D. Gabriel, J. Plešek, and R. Kolman. “Influence of mass lumping techniques on contact pressure oscillations in explicit finite element contact-impact algorithm based on isogeometric analysis with NURBS”. In: *Stability, Vibration, and Control of Machines and Structures*. Ed. by A. Guran and J. Gwinner. Lecture Notes in Mechanical Engineering. Springer, 2014, pp. 1–13.

[198] J. Kopačka, D. Gabriel, J. Plešek, and M. Ulbin. “Utilising optimization methods for computing of normal vector to contact surface”. In: *Engineering Mechanics 2010*. Svratka, Czech Republic, 2010, pp. 1–13.

[199] D. Gabriel, J. Kopačka, J. Plešek, and M. Ulbin. “Searching for local contact constraints by the quasi-newton methods in the finite element procedures for contact-impact problems”. In: *ECCOMAS Thematic Conference - COMPDYN 2011: 3rd International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering: An IACM Special Interest Conference, Programme*. Ed. by M. Papadrakakis, N. D. Lagaros, and M. Fragiadakis. Corfu, Greece, 2011, pp. 1–9.

Author’s technical reports

[200] Z. Hrubý, D. Gabriel, J. Kopačka, and J. Plešek. *Creepová analýza VT vnirřního tělesa DSPWR: shrnutí za rok 2014*. Tech. rep. rev. 01. Prague, Czech Republic: Institute of Thermomechanics of the CAS, v. v. i., Mar. 2015, pp. 1–54.

[201] J. Kopačka, J. Masák, D. Gabriel, and J. Plešek. *Creepová analýza VT vnitřního tělesa DSPWR – kontaktní úloha*. Tech. rep. Prague, Czech Republic: Institute of Thermomechanics of the CAS, v. v. i., 2017, pp. 1–13.

Author’s conference talks

[202] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “Contact-impact treatment based on the bipenalty technique in explicit transient dynamics using isogeometric analysis with NURBS”. In: *IV. International Conference on Computational Contact Mechanics*. Hannover, Germany, 2015, pp. 1–2.

[203] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “Convergence study of an explicit FE contact-impact algorithm based on isogeometric analysis with NURBS”. In: *Computational Mechanics 2013*. Špičák, Czech Republic, 2013, pp. 1–2.

[204] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. “Influence of mass lumping techniques on contact pressure oscillations in explicit finite element contact-impact algorithm based on isogeometric analysis with NURBS”. In: *Modelling 2014*. Rožnov pod Radhoštěm, Czech Republic, 2014, p. 1.

[205] J. Kopačka. “Contact treatment in isogeometric analysis”. In: *Splines and IsoGeometric Analysis 2012*. Prague, Czech Republic, 2012, pp. 1–2.

[206] J. Kopačka, R. Kolman, D. Gabriel, and J. Plešek. “Contact treatment in isogeometric analysis”. In: *Computational mechanics 2012*. Špičák, Czech Republic, 2012, pp. 1–2.

[207] J. Kopačka, D. Gabriel, J. Plešek, and M. Ulbin. “Application of Methods for Unconstrained Optimization in Computation of Normal Contact Vector”. In: *Sixth International PhD & DLA Symposium*. Pecs, Hungaria, 2010, p. 1.

[208] D. Gabriel, J. Kopačka, J. Plešek, and M. Ulbin. “Assesment of methods for calculating the normal contact vector in local search”. In: *ECCM 2010 - IV European Congress on Computational Mechanics: Solids, Structures and Coupled Problems in Engineering*. Paris, France, 2010, pp. 1–2.

[209] J. Kopačka, D. Gabriel, J. Plešek, and M. Ulbin. "Local contact search by unconstrained optimization methods in the FE procedures for contact-impact problems". In: *4th GACM Colloquium on Computational Mechanics*. Dresden, Germany: Technische Universität Dresden, 2011, p. 1.

Author's conference posters

[210] J. Kopačka, D. Gabriel, J. Plešek, and M. Ulbin. "Computation of the minimum distance between a point and a quadratic surface with application in contact-impact mechanics". In: *EUROMECH Colloquium 540: Advanced Modelling of Wave Propagation in Solids*. Prague, Czech Republic, 2012.

[211] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. "On the contact pressure oscillations of an isogeometric contact-impact algorithm". In: *3rd Aachen Conference on Computational Engineering Science (AC.CES)*. Aachen, Germany, 2015, p. 1.

[212] J. Kopačka, D. Gabriel, R. Kolman, and J. Plešek. "Performance and implementation aspects of an isogeometric mortar-based contact algorithm". In: *Výpočty konstrukcí metodou konečných prvků 2015*. Prague, Czech Republic: Institute of Thermomechanics AS CR, v.v.i., 2015, p. 1.

Appendices

Appendix A

Derivation of contact kinematic quantities

A.1 Minimization of the squared distance function

One can obtain stationary point of the squared distance function (2.7) by putting all its partial derivatives with respect to convective coordinates equal to zero

$$\begin{aligned}
\frac{\partial d(\boldsymbol{\xi})}{\partial \xi^\alpha} &= \frac{\partial}{\partial \xi^\alpha}\left[\frac{1}{2}\left\|\mathbf{x}^{(i)}(\mathbf{X}^{(i)},t) - \mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\boldsymbol{\xi}),t)\right\|^2\right] && \text{(A.1)}\\
&= \frac{\partial}{\partial \xi^\alpha}\left[\frac{1}{2}\left(\mathbf{x}^{(i)} - \mathbf{x}^{(k)}\right)\cdot\left(\mathbf{x}^{(i)} - \mathbf{x}^{(k)}\right)\right] && \text{(A.2)}\\
&= \frac{1}{2}\left(-\frac{\partial \mathbf{x}^{(k)}}{\partial \mathbf{X}^{(k)}}\frac{\partial \mathbf{X}^{(k)}}{\partial \xi^\alpha}\right)\cdot\left(\mathbf{x}^{(i)} - \mathbf{x}^{(k)}\right) && \text{(A.3)}\\
&+ \frac{1}{2}\left(\mathbf{x}^{(i)} - \mathbf{x}^{(k)}\right)\cdot\left(-\frac{\partial \mathbf{x}^{(k)}}{\partial \mathbf{X}^{(k)}}\frac{\partial \mathbf{X}^{(k)}}{\partial \xi^\alpha}\right) && \text{(A.4)}\\
&= -\left(\mathbf{x}^{(i)} - \mathbf{x}^{(k)}\right)\cdot\frac{\partial \mathbf{x}^{(k)}}{\partial \mathbf{X}^{(k)}}\frac{\partial \mathbf{X}^{(k)}}{\partial \xi^\alpha} = 0, && \text{(A.5)}
\end{aligned}$$

which yields minimizer, $\bar{\boldsymbol{\xi}}$, as the solution of the implicit term

$$\boxed{\left[\mathbf{x}^{(i)}(\mathbf{X}^{(i)},t) - \mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}}),t)\right]\cdot\frac{\partial \mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}}),t)}{\partial \mathbf{X}^{(k)}}\frac{\partial \mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}})}{\partial \xi^\alpha} = 0.} \tag{A.6}$$

A.2 The time derivative of convective coordinates

The unknown velocity $\dot{\bar{\xi}}^\alpha$ can be obtained by differentiating the orthogonality condition (2.12) with respect to time

$$\frac{\partial}{\partial t}\left[\left(\mathbf{x}^{(i)}(\mathbf{X}^{(i)},t)-\mathbf{x}^{(k)}(\mathbf{X}^{(k)}(\bar{\boldsymbol{\xi}}),t)\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right]=0, \tag{A.7}$$

$$\left(\frac{\partial\mathbf{x}^{(i)}}{\partial t}-\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial t}-\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\mathbf{X}^{(k)}}\frac{\partial\bar{\mathbf{X}}^{(k)}}{\partial\xi^{\beta}}\frac{\partial\bar{\xi}^{\beta}}{\partial t}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}+\left(\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(k)}\right)\cdot\frac{\partial\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}}{\partial t}=0, \tag{A.8}$$

$$\left(\frac{\partial\mathbf{x}^{(i)}}{\partial t}-\frac{\partial\bar{\mathbf{x}}^{(i)}}{\partial t}-\bar{\mathbf{F}}^{(k)}\bar{\mathcal{T}}_{\beta}^{(k)}\frac{\partial\bar{\xi}^{\beta}}{\partial t}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}+\left(\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(k)}\right)\cdot\dot{\bar{\boldsymbol{\tau}}}_{\alpha}^{(k)}=0, \tag{A.9}$$

$$\left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}-\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\dot{\bar{\xi}}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}+\left(\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(k)}\right)\cdot\dot{\bar{\boldsymbol{\tau}}}_{\alpha}^{(k)}=0, \tag{A.10}$$

$$\left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}-\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\dot{\bar{\xi}}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\dot{\bar{\boldsymbol{\tau}}}_{\alpha}^{(k)}=0, \tag{A.11}$$

where the identity $\left(\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(k)}\right)=-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}$ has been exploited. The derivative of the tangent vector with respect to time is

$$\begin{aligned}
\dot{\bar{\boldsymbol{\tau}}}_{\alpha}^{(k)} &= \frac{\partial}{\partial t}\left[\frac{\partial\mathbf{x}^{(k)}(\bar{\boldsymbol{\xi}},t)}{\partial\xi^{\alpha}}\right] && \text{(A.12)}\\
&= \frac{\partial\mathbf{x}^{(k)}(\bar{\boldsymbol{\xi}},t)}{\partial\xi^{\alpha}\partial t}+\frac{\partial\mathbf{x}^{(k)}(\bar{\boldsymbol{\xi}})}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta} && \text{(A.13)}\\
&= \frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}+\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta}. && \text{(A.14)}
\end{aligned}$$

Substitution of the expression for $\dot{\bar{\boldsymbol{\tau}}}_{\alpha}$ back to A.11 yields

$$\begin{aligned}
0 &= \left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}-\dot{\bar{\xi}}^{\beta}\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}+\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta}\right) && \text{(A.15)}\\
&= \left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-\dot{\bar{\xi}}^{\beta}\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}+\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta}\right) && \text{(A.16)}\\
&= \left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-\dot{\bar{\xi}}^{\beta}\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}+\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta}\right) && \text{(A.17)}\\
&= \left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-\bar{m}_{\alpha\beta}^{(k)}\dot{\bar{\xi}}^{\beta}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\frac{\partial\bar{\mathbf{x}}^{(k)}}{\partial\xi^{\alpha}\partial\xi^{\beta}}\dot{\bar{\xi}}^{\beta} && \text{(A.18)}\\
&= \left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-\bar{m}_{\alpha\beta}^{(k)}\dot{\bar{\xi}}^{\beta}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}-g_{\mathrm{N}}^{(i)}\bar{\kappa}_{\alpha\beta}^{(k)}\dot{\bar{\xi}}^{\beta}, && \text{(A.19)}
\end{aligned}$$

where (2.15) and (2.30) have been employed. From the last equation one finally obtains

$$\boxed{\dot{\bar{\xi}}^{\beta}=\left(\bar{m}_{\alpha\beta}^{(k)}+g_{\mathrm{N}}^{(i)}\bar{\kappa}_{\alpha\beta}^{(k)}\right)^{-1}\left[\left(\mathbf{V}^{(i)}-\bar{\mathbf{V}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}-g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\frac{\partial\bar{\mathbf{V}}^{(k)}}{\partial\xi^{\alpha}}\right].} \tag{A.20}$$

Appendix B

Variations of contact quantities

During the linearization of the contact residual term (2.111) the variation of several contact quantities has appears. The first variation of a functional F at point $\mathbf{u}^{(i)}$ in the direction of $\delta\mathbf{u}^{(i)}$ is defined as

$$\delta_{\delta\mathbf{u}^{(i)}}\left[F\right](\mathbf{u}^{(i)}) := \frac{\mathrm{d}}{\mathrm{d}\theta}F(\mathbf{u}^{(i)}+\theta\delta\mathbf{u}^{(i)})\Big|_{\theta=0}. \tag{B.1}$$

In what fallow a shorter notation will be employed. Namely, only δ symbol before varied quantities will indicate the variation

$$\delta\bullet := \delta_{\delta\mathbf{u}^{(i)}}\left[\bullet\right](\mathbf{u}^{(i)}). \tag{B.2}$$

Moreover, partial derivatives with respect to convective coordinates will be indicated as a subscript α, β, γ after the comma

$$\frac{\partial\bullet}{\partial\xi^{\alpha}} = \bullet_{,\alpha}. \tag{B.3}$$

The tangent vector

The first variation of the tangent vector (2.14) in the direction of $\delta\mathbf{u}^{(k)}$ is

$$\begin{aligned} \delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha} &= \delta\bar{\mathbf{x}}^{(k)}_{,\alpha} && \text{(B.4)}\\ &= \delta\bar{\mathbf{u}}^{(\mathrm{k})}_{,\alpha} + \bar{\mathbf{x}}^{(\mathrm{k})}_{,\alpha\beta}\delta\bar{\xi}^{\beta}. && \text{(B.5)}\end{aligned}$$

$$\boxed{\delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha} = \delta\bar{\mathbf{u}}^{(\mathrm{k})}_{,\alpha} + \bar{\mathbf{x}}^{(\mathrm{k})}_{,\alpha\beta}\delta\bar{\xi}^{\beta}.} \tag{B.6}$$

The normal vector

An elegant way how to derive the first variation of the normal vector (2.19) in the direction of $\delta\mathbf{u}^{(k)}$ is to come out from the orthogonality condition for normal and tangent vector

$$\bar{\mathbf{n}}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} = 0, \tag{B.7}$$

$$\delta\left(\bar{\mathbf{n}}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right) = 0, \tag{B.8}$$

$$\delta\bar{\mathbf{n}}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} + \bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} = 0, \tag{B.9}$$

$$\delta\bar{\mathbf{n}}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} = -\bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}. \tag{B.10}$$

In the last term the dot product on the left hand side expresses covariant components of the vector $\delta\bar{\mathbf{n}}^{(k)}$. So if the whole equation is premultiplied by the vector $\bar{\boldsymbol{\tau}}^{(k)\alpha}$, one gets the vector $\delta\bar{\mathbf{n}}^{(k)}$ expressed in the contravariant basis

$$\left(\delta\bar{\mathbf{n}}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right)\bar{\boldsymbol{\tau}}^{(k)\alpha} = \left(-\bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right)\bar{\boldsymbol{\tau}}^{(k)\alpha}, \tag{B.11}$$

$$\delta\bar{\mathbf{n}}^{(k)} = \left(-\bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right)\bar{\boldsymbol{\tau}}^{(k)\alpha}, \tag{B.12}$$

and after substitution from (B.5)

$$\delta\bar{\mathbf{n}}^{(k)} = \left[-\bar{\mathbf{n}}^{(k)} \cdot \left(\delta\bar{\mathbf{u}}_{,\alpha}^{(\mathrm{k})} + \bar{\mathbf{x}}_{,\alpha\beta}^{(\mathrm{k})}\delta\bar{\xi}^{\beta}\right)\right]\bar{\boldsymbol{\tau}}^{(k)\alpha}. \tag{B.13}$$

Recall that the contravariant basis can be calculated using the metric tensor (2.16) as

$$\bar{\boldsymbol{\tau}}^{(k)\alpha} = \left(\bar{\mathbf{m}}^{-1}\right)_{\alpha\beta}^{(k)}\bar{\boldsymbol{\tau}}_{\beta}^{(k)}, \tag{B.14}$$

the first variation of the normal vector is

$$\boxed{\delta\bar{\mathbf{n}}^{(k)} = \left[-\bar{\mathbf{n}}^{(k)} \cdot \left(\delta\bar{\mathbf{u}}_{,\alpha}^{(\mathrm{k})} + \bar{\mathbf{x}}_{,\alpha\beta}^{(\mathrm{k})}\delta\bar{\xi}^{\beta}\right)\right]\left(\bar{\mathbf{m}}^{-1}\right)_{\alpha\gamma}^{(k)}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}.} \tag{B.15}$$

The normal gap function

The first variation of the normal gap function (2.21) in the direction of $\delta\mathbf{u}^{(i)}$ is

$$\delta g_{\mathrm{N}}^{(i)} = \delta\left[-\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \bar{\mathbf{n}}^{(k)}\right] \tag{B.16}$$

$$= -\left(\delta\mathbf{x}^{(i)} - \delta\bar{\mathbf{x}}^{(k)} - \bar{\mathbf{x}}_{,\alpha}^{(k)}\delta\bar{\xi}^{\alpha}\right) \cdot \bar{\mathbf{n}}^{(k)} - \left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right) \cdot \delta\bar{\mathbf{n}}^{(k)} \tag{B.17}$$

$$= -\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\delta\bar{\xi}^{\alpha}\right) \cdot \bar{\mathbf{n}}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\mathbf{n}}^{(k)}. \tag{B.18}$$

If two kinds of orthogonality are further exploited. Namely, the obvious orthogonality of the normal and tangent vector and a little more complicated orthogonality of the normal vector and its variations

$$\delta\bar{\mathbf{n}}^{(k)} \cdot \bar{\mathbf{n}}^{(k)} = \left(-\bar{\mathbf{n}}^{(k)} \cdot \delta\bar{\boldsymbol{\tau}}_{\alpha}\right)\bar{\boldsymbol{\tau}}^{\alpha} \cdot \bar{\mathbf{n}} = 0, \tag{B.19}$$

which follows after substituting (B.12) from the orthogonality of the normal and contravariant tangent vector. For the first variation of the normal gap function one can finally write

$$\boxed{\delta g_{\mathrm{N}}^{(i)} = -\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right) \cdot \bar{\mathbf{n}}^{(k)}.} \tag{B.20}$$

The convective coordinates

The first variation of the convective coordinate in the direction of $\delta\mathbf{u}^{(i)}$ will be derived from the orthogonality condition (2.12) as

$$\begin{aligned}
\delta\left[\left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right] &= 0, &\text{(B.21)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\mathbf{x}}_{,\beta}^{(k)}\delta\bar{\xi}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} + \left(\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(k)}\right)\cdot\delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} &= 0, &\text{(B.22)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} &= 0, &\text{(B.23)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\beta}^{(\mathrm{k})}\delta\bar{\xi}^{\beta}\right) &= 0, &\text{(B.24)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - \left(\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\right)\delta\bar{\xi}^{\beta} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\beta}^{(\mathrm{k})}\delta\bar{\xi}^{\beta}\right) &= 0, &\text{(B.25)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - \bar{m}_{\alpha\beta}^{(k)}\delta\bar{\xi}^{\beta} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{k}\cdot\bar{\mathbf{x}}_{,\alpha\beta}^{(k)}\delta\bar{\xi}^{\beta} &= 0, &\text{(B.26)}\\
\left[\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}_{,\alpha}^{(k)}\right] - \left(\bar{m}_{\alpha\beta}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\kappa}_{\alpha\beta}^{(k)}\right)\delta\bar{\xi}^{\beta} &= 0. &\text{(B.27)}
\end{aligned}$$

Further, $\delta\bar{\xi}^{\beta}$ can be expressed as

$$\delta\bar{\xi}^{\beta} = \left(\bar{m}_{\alpha\beta}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\kappa}_{\alpha\beta}^{(k)}\right)^{-1}\left[\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - g_{\mathrm{N}}^{(k)}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}_{,\alpha}^{(k)}\right]. \tag{B.28}$$

The normal gap function is equal to zero on the contact boundary $\Gamma_{\mathrm{c}}^{(i)}$, therefore, the last relation can be further simplified

$$\delta\bar{\xi}^{\beta} = \left(\bar{\mathbf{m}}^{-1}\right)_{\alpha\beta}^{(k)}\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}, \tag{B.29}$$

or employing (2.16)

$$\boxed{\delta\bar{\xi}^{\beta} = \left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}^{(k)\beta}.} \tag{B.30}$$

Note that one more useful relation follows from the expression of the components (B.23) in the contravariant basis

$$\begin{aligned}
\left[\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta}\right)\cdot\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right]\bar{\boldsymbol{\tau}}^{(k)\alpha} - g_{\mathrm{N}}^{(i)}\left(\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\right)\bar{\boldsymbol{\tau}}^{(k)\alpha} &= 0, &\text{(B.31)}\\
\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta}\right) + g_{\mathrm{N}}^{(i)}\delta\bar{\mathbf{n}}^{(k)} &= 0, &\text{(B.32)}
\end{aligned}$$

where in the second addend the relation (B.12) can be recognized. Finally, one gets an useful relation

$$g_{\mathrm{N}}^{(i)}\delta\bar{\mathbf{n}}^{(k)} = -\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)} - \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta}\right). \tag{B.33}$$

Appendix C

Linearization of the contact quantities

Since the linearization has the same mathematical structure as variation, the linearization of some quantities can be obtained simply just by replacement of the symbol δ by Δ. Similar to the previous section a shorter notation for directional derivative will be occasionally used

$$\Delta \bullet := D_{\Delta \mathbf{u}^{(i)}} \left[\bullet \right] (\mathbf{u}^{(i)}). \tag{C.1}$$

The normal gap function

The directional derivative of the normal gap function in the direction of $\Delta \mathbf{u}^{(i)}$ is according to (B.20)

$$\boxed{D_{\Delta \mathbf{u}^{(i)}} \left[g_{\mathrm{N}}^{(i)} \right] = - \left(\Delta \mathbf{u}^{(i)} - \Delta \bar{\mathbf{u}}^{(k)} \right) \cdot \bar{\mathbf{n}}^{(k)}.} \tag{C.2}$$

The convective coordinates

The directional derivative of the convective coordinates in the direction of $\Delta \mathbf{u}^{(k)}$

$$\boxed{D_{\Delta \mathbf{u}^{(k)}} \left[\bar{\xi}^{\beta} \right] = \left(\bar{m}_{\alpha\beta}^{(k)} + g_{\mathrm{N}}^{(i)} \bar{\kappa}_{\alpha\beta}^{(k)} \right)^{-1} \left[\left(\Delta \mathbf{u}^{(i)} - \Delta \bar{\mathbf{u}}^{(k)} \right) \cdot \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} - g_{\mathrm{N}}^{(i)} \bar{\mathbf{n}}^{(k)} \cdot \Delta \bar{\mathbf{u}}_{,\alpha}^{(k)} \right].} \tag{C.3}$$

The tangent vector

The directional derivative of the *covariant* components of the tangent vector in the direction of $\Delta \mathbf{u}^{(k)}$ is

$$\boxed{D_{\Delta \mathbf{u}^{(k)}} \left[\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} \right] = \Delta \bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\beta}^{(k)} \Delta \bar{\xi}^{\beta}.} \tag{C.4}$$

The directional derivative of the *contravariant* components of the tangent vector in the direction of $\Delta \mathbf{u}^{(k)}$ is

$$\begin{aligned} D_{\Delta \mathbf{u}^{(k)}} \left[\bar{\boldsymbol{\tau}}^{(k)\alpha} \right] &= \Delta \left(\bar{m}^{(k)\beta\gamma} \bar{\boldsymbol{\tau}}_{\gamma}^{(k)} \right) && \text{(C.5)} \\ &= \Delta \bar{m}^{(k)\beta\gamma} \bar{\boldsymbol{\tau}}_{\gamma}^{(k)} + \bar{m}^{(k)\beta\gamma} \Delta \bar{\boldsymbol{\tau}}_{\gamma}^{(k)}. && \text{(C.6)} \end{aligned}$$

The normal vector

The directional derivative of the normal vector in the direction of $\Delta\mathbf{u}^{(k)}$ is

$$\begin{aligned} D_{\Delta\mathbf{u}^{(k)}}\left[\bar{\mathbf{n}}^{(k)}\right] &= \left(-\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\right)\bar{\boldsymbol{\tau}}^{(k)\alpha} && \text{(C.7)}\\ &= \left[-\bar{\mathbf{n}}^{(k)}\cdot\left(\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\beta}\Delta\bar{\xi}^{\beta}\right)\right]\bar{\boldsymbol{\tau}}^{(k)\alpha}. && \text{(C.8)} \end{aligned}$$

$$\boxed{D_{\Delta\mathbf{u}^{(k)}}\left[\bar{\mathbf{n}}^{(k)}\right] = \left[-\bar{\mathbf{n}}^{(k)}\cdot\left(\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\beta}\Delta\bar{\xi}^{\beta}\right)\right]\bar{\boldsymbol{\tau}}^{(k)\alpha}.} \qquad \text{(C.9)}$$

It turns out to be useful to calculate also the directional derivative of the dot product of the normal vector and its variations in the direction of $\Delta\mathbf{u}^{(k)}$ which, as we have shown in (B.19), is equal to zero

$$\begin{aligned} \bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)} &= 0, && \text{(C.10)}\\ D_{\Delta\mathbf{u}^{(k)}}\left[\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}\right] &= 0, && \text{(C.11)}\\ \Delta\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}+\bar{\mathbf{n}}^{(k)}\cdot\Delta\delta\bar{\mathbf{n}}^{(k)} &= 0, && \text{(C.12)} \end{aligned}$$

and from the last equality a useful identity can be expressed

$$\boxed{\bar{\mathbf{n}}^{(k)}\cdot\Delta\delta\bar{\mathbf{n}}^{(k)} = -\Delta\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}.} \qquad \text{(C.13)}$$

The variation of the normal gap function

When calculating the directional derivative of the variation of the normal gap function $\delta g^{(i)}_{\mathrm{N}}$ in the direction of $\Delta\mathbf{u}^{(i)}$, it is necessary to come out from equation (B.18), because the linearization will no longer apply the orthogonality, which has been exploited to simplify (B.20). Thus, the linearization of $\delta g^{(i)}_{\mathrm{N}}$ will be

$$\begin{aligned} D_{\Delta\mathbf{u}^{(i)}}\left[\delta g^{(i)}_{\mathrm{N}}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[-\left(\delta\mathbf{u}^{(i)}-\delta\bar{\mathbf{u}}^{(k)}-\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\delta\bar{\xi}^{\alpha}\right)\cdot\bar{\mathbf{n}}^{(k)}+g^{(i)}_{\mathrm{N}}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}\right] && \text{(C.14)}\\ &= -\left(-\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}\Delta\bar{\xi}^{\alpha}-\Delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\delta\bar{\xi}^{\alpha}-\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\Delta\delta\bar{\xi}^{\alpha}\right)\cdot\bar{\mathbf{n}}^{(k)} && \text{(C.15)}\\ &-\left(\delta\mathbf{u}^{(i)}-\delta\bar{\mathbf{u}}^{(k)}-\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\delta\bar{\xi}^{\alpha}\right)\cdot\Delta\bar{\mathbf{n}}^{(k)} && \text{(C.16)}\\ &+\Delta g^{(i)}_{\mathrm{N}}\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}+g^{(i)}_{\mathrm{N}}\left(\Delta\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}+\bar{\mathbf{n}}^{(k)}\cdot\Delta\delta\bar{\mathbf{n}}^{(k)}\right). && \text{(C.17)} \end{aligned}$$

Excluding the scalar products $\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\cdot\bar{\mathbf{n}}^{(k)}$ and $\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{n}}^{(k)}$, which are equal to zero, and if the identities (C.13) and (B.33) are invoked

$$\begin{aligned} D_{\Delta\mathbf{u}^{(i)}}\left[\delta g^{(i)}_{\mathrm{N}}\right] &= \left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}\Delta\bar{\xi}^{\alpha}+\Delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\delta\bar{\xi}^{\alpha}\right)\cdot\bar{\mathbf{n}}^{(k)}+g^{(i)}_{\mathrm{N}}\delta\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\mathbf{n}}^{(k)} && \text{(C.18)}\\ &= \left[\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}\Delta\bar{\xi}^{\alpha}+\left(\Delta\bar{\mathbf{u}}^{(\mathrm{k})}_{,\alpha}+\bar{\mathbf{x}}^{(\mathrm{k})}_{,\alpha\beta}\Delta\bar{\xi}^{\beta}\right)\delta\bar{\xi}^{\alpha}\right]\cdot\bar{\mathbf{n}}^{(k)}+g^{(i)}_{\mathrm{N}}\delta\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\mathbf{n}}^{(k)} && \text{(C.19)}\\ &= \bar{\mathbf{n}}^{(k)}\cdot\left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}\Delta\bar{\xi}^{\alpha}+\delta\bar{\xi}^{\alpha}\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha}\right)+\delta\bar{\xi}^{\alpha}\bar{\kappa}^{(k)}_{\alpha\beta}\Delta\bar{\xi}^{\beta}+g^{(i)}_{\mathrm{N}}\delta\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\mathbf{n}}^{(k)}, && \text{(C.20)} \end{aligned}$$

and by substitution $\delta\bar{\mathbf{n}}^{(k)}$ from (B.13) and $\Delta\bar{\mathbf{n}}^{(k)}$ from (C.7) one finally gets

$$\boxed{\begin{aligned} D_{\Delta\mathbf{u}^{(i)}}\left[\delta g_{\mathrm{N}}^{(i)}\right] \quad &= \quad \left(\delta\bar{\mathbf{u}}_{,\alpha}^{(k)}\Delta\bar{\xi}^{\alpha} + \delta\bar{\xi}^{\alpha}\Delta\bar{\mathbf{u}}_{,\alpha}^{(\mathrm{k})}\right)\cdot\bar{\mathbf{n}}^{(k)} + \delta\bar{\xi}^{\alpha}\bar{\kappa}_{\alpha\beta}^{(k)}\Delta\bar{\xi}^{\beta} \\ &+ \quad g_{\mathrm{N}}^{(i)}\left(\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\kappa}_{\alpha\gamma}^{(k)}\delta\bar{\xi}^{\gamma}\right)\bar{m}^{(k)\alpha\beta}\left(\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\mathbf{u}}_{,\alpha}^{(\mathrm{k})} + \bar{\kappa}_{\beta\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right). \end{aligned}} \tag{C.21}$$

The variation of the convective coordinates

The directional derivative of the convective coordinates $\delta\bar{\xi}^{\beta}$ can be obtained by derivation of (B.28)

$$\boxed{\begin{aligned} \left(\bar{m}_{\alpha\beta}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\kappa}_{\alpha\beta}^{(k)}\right)\Delta\delta\bar{\xi}^{\beta} \quad &= \quad -\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\left(\Delta\bar{\mathbf{u}}_{,\beta}^{(k)}\delta\bar{\xi}^{\beta} + \delta\bar{\mathbf{u}}_{,\beta}^{(k)}\Delta\bar{\xi}^{\beta}\right) \\ &- \quad \left(\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\bar{\mathbf{x}}_{,\beta\gamma}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\bar{\mathbf{x}}_{,\alpha\beta\gamma}^{(k)}\right)\Delta\bar{\xi}^{\gamma}\delta\bar{\xi}^{\beta} \\ &- \quad g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)}\cdot\left(\delta\bar{\mathbf{u}}_{,\alpha\beta}^{(k)}\Delta\bar{\xi}^{\beta} + \Delta\bar{\mathbf{u}}_{,\alpha\beta}^{(k)}\delta\bar{\xi}^{\beta}\right) \\ &- \quad \left(\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\delta\bar{\xi}^{\gamma}\right)\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\Delta\bar{\xi}^{\beta} - \left(\Delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right)\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\delta\bar{\xi}^{\beta} \\ &+ \quad \left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right)\cdot\left(\Delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right) \\ &+ \quad \left(\Delta\mathbf{u}^{(i)} - \Delta\bar{\mathbf{u}}^{(k)}\right)\cdot\left(\delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\delta\bar{\xi}^{\gamma}\right). \end{aligned}} \tag{C.22}$$

The relative slip rate

The directional derivative of components of the relative slip rate vector in the direction of $\Delta\mathbf{u}^{(i)}$ are

$$D_{\Delta\mathbf{u}^{(i)}}\left[\dot{g}_{\mathrm{T}\beta}^{(i)}\right] \quad = \quad \Delta\left(\dot{\bar{\xi}}^{\alpha}\bar{m}_{\alpha\beta}^{(k)}\right) \tag{C.23}$$

$$= \quad \Delta\dot{\bar{\xi}}^{\alpha}\bar{m}_{\alpha\beta}^{(k)} + \dot{\bar{\xi}}^{\alpha}\Delta\bar{m}_{\alpha\beta}^{(k)}. \tag{C.24}$$

$$\boxed{D_{\Delta\mathbf{u}^{(i)}}\left[\dot{g}_{\mathrm{T}\beta}^{(i)}\right] = \Delta\dot{\bar{\xi}}^{\alpha}\bar{m}_{\alpha\beta} + \dot{\bar{\xi}}^{\alpha}\Delta\bar{m}_{\alpha\beta}^{(k)}.} \tag{C.25}$$

The metric tensor

The directional derivative of the components of the metric tensor in the direction of $\Delta\mathbf{u}^{(k)}$ is

$$D_{\Delta\mathbf{u}^{(k)}}\left[\bar{m}_{\alpha\beta}^{(k)}\right] = \Delta\left(\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\right) \tag{C.26}$$

$$= \Delta\bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)} + \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\Delta\bar{\boldsymbol{\tau}}_{\beta}^{(k)} \tag{C.27}$$

$$= \left(\Delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right)\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)} + \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\left(\Delta\bar{\mathbf{u}}_{,\beta}^{(k)} + \bar{\mathbf{x}}_{,\beta\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right). \tag{C.28}$$

$$\boxed{D_{\Delta\mathbf{u}^{(k)}}\left[\bar{m}_{\alpha\beta}^{(k)}\right] = \left(\Delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right)\cdot\bar{\boldsymbol{\tau}}_{\beta}^{(k)} + \bar{\boldsymbol{\tau}}_{\alpha}^{(k)}\cdot\left(\Delta\bar{\mathbf{u}}_{,\beta}^{(k)} + \bar{\mathbf{x}}_{,\beta\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right).} \tag{C.29}$$

The unit tangent traction vector

The directional derivative of components of the unit tangent traction vector in the direction of $\Delta\mathbf{u}^{(i)}$ is

$$
\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{(i)}\left(t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}\right)^{-\frac{1}{2}}\right] && \text{(C.30)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} - \frac{1}{2}t_{\mathrm{T}\alpha}^{(i)}\left(t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}\right)^{-\frac{3}{2}}\Delta\left(t_{\mathrm{T}\delta}^{(i)}\bar{m}^{(k)\delta\eta}t_{\mathrm{T}\eta}^{(i)}\right) && \text{(C.31)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} - \frac{1}{2}t_{\mathrm{T}\alpha}^{(i)}\frac{\Delta\left(t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}\right)}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|\left\|\mathbf{t}_{\mathrm{T}}^{(k)}\right\|^{2}} && \text{(C.32)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} - \frac{1}{2}\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{\Delta t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)} + t_{\mathrm{T}\beta}^{(i)}\Delta\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)} + t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}\Delta t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|^{2}} && \text{(C.33)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} - \frac{1}{2}\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{2\Delta t_{\mathrm{T}\beta}^{(i)}\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)} + t_{\mathrm{T}\beta}^{(i)}\Delta\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|^{2}} && \text{(C.34)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{\Delta t_{\mathrm{T}\beta}^{(i)}t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|^{2}} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}\Delta\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}}{2\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|^{2}} && \text{(C.35)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right) - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}\Delta\bar{m}^{(k)\beta\gamma}t_{\mathrm{T}\gamma}^{(i)}}{2\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|^{2}} && \text{(C.36)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right) - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{\Delta\bar{m}^{(k)\beta\gamma}}{2}\frac{t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} && \text{(C.37)}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right) - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\Delta\bar{\boldsymbol{\tau}}^{(k)\beta}\cdot\bar{\boldsymbol{\tau}}^{(k)\gamma}\frac{t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}. && \text{(C.38)}
\end{aligned}
$$

$$
\boxed{D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right] = \frac{\Delta t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha} - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right) - \frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\Delta\bar{\boldsymbol{\tau}}^{(k)\beta}\cdot\bar{\boldsymbol{\tau}}^{(k)\gamma}\frac{t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}.} \quad \text{(C.39)}
$$

The contact pressure

The directional derivative of the contact pressure in the direction of $\Delta\mathbf{u}^{(i)}$ is

$$\begin{aligned} D_{\Delta \mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{(i)}\right] &= D_{\Delta \mathbf{u}^{(i)}}\left[\epsilon_{\mathrm{N}}\left\langle g_{\mathrm{N}}^{(i)}\right\rangle\right] &\text{(C.40)}\\ &= \epsilon_{\mathrm{N}} \frac{\partial\left\langle g_{\mathrm{N}}^{(i)}\right\rangle}{\partial g_{\mathrm{N}}^{(i)}} D_{\Delta \mathbf{u}^{(i)}}\left[\Delta g_{\mathrm{N}}^{(i)}\right] &\text{(C.41)}\\ &= \epsilon_{\mathrm{N}} H\left(g_{\mathrm{N}}^{(i)}\right) D_{\Delta \mathbf{u}^{(i)}}\left[\Delta g_{\mathrm{N}}^{(i)}\right]. &\text{(C.42)} \end{aligned}$$

where H stands for Heaviside function.

$$\boxed{D_{\Delta \mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{(i)}\right] = \epsilon_{\mathrm{N}} H\left(g_{\mathrm{N}}^{(i)}\right) D_{\Delta \mathbf{u}^{(i)}}\left[\Delta g_{\mathrm{N}}^{(i)}\right].} \qquad \text{(C.43)}$$

The tangential components of the contact traction vector

For the directional derivative of tangential components of the contact traction vector in the direction of $\Delta \mathbf{u}^{(i)}$ it is necessary to distinguish between the state of sticking and slipping. It is advantageous to come from Equation (2.169). In the case of sticking, for which $\Delta\lambda^{n+1} = 0$

$$\begin{aligned} D_{\Delta \mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right] &= D_{\Delta \mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n(i)} - \epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.44)}\\ &= D_{\Delta \mathbf{u}^{(i)}}\left[-\epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.45)}\\ &= D_{\Delta \mathbf{u}^{(i)}}\left[-\epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)}\right]\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right) - \epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)} D_{\Delta \mathbf{u}^{(i)}}\left[\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.46)}\\ &= -\epsilon_{\mathrm{T}} \Delta\bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right) - \epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)} \Delta\bar{\xi}^{n+1\beta}. &\text{(C.47)} \end{aligned}$$

$$\boxed{D_{\Delta \mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right] = -\epsilon_{\mathrm{T}} \Delta\bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right) - \epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)} \Delta\bar{\xi}^{n+1\beta}.} \qquad \text{(C.48)}$$

In the case of sliding, it is also advantageous to start with Equation (2.169) where for the Coulomb friction law it holds $\Delta\lambda^{n+1(i)} = -\frac{\Phi_{\mathrm{trial}}^{n+1(i)}}{\epsilon_{\mathrm{T}}}$

$$\begin{aligned} t_{\mathrm{T}\alpha}^{n+1(i)} &= t_{\mathrm{T}\alpha}^{n(i)} + \epsilon_{\mathrm{T}}\left[-\frac{\Phi_{\mathrm{trial}}^{n+1(i)}}{\epsilon_{\mathrm{T}}} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} - \bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.49)}\\ &= t_{\mathrm{T}\alpha}^{n} + \epsilon_{\mathrm{T}}\left[\frac{-\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\| + \mu p_{\mathrm{c}}^{n+1(i)}}{\epsilon_{\mathrm{T}}} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} - \bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.50)}\\ &= t_{\mathrm{T}\alpha}^{n} + \left[\frac{-\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\| t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} + \mu p_{\mathrm{c}}^{n+1(i)} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} - \epsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)}\left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right)\right] &\text{(C.51)}\\ &= t_{\mathrm{T}\alpha}^{n} - t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)} + \mu p_{\mathrm{c}}^{n+1(i)} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} + t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)} &\text{(C.52)}\\ &= t_{\mathrm{T}\alpha}^{n} + \mu p_{\mathrm{c}}^{n+1(i)} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}. &\text{(C.53)} \end{aligned}$$

So that

$$D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right] = D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n} + \mu p_{\mathrm{c}}^{n+1(i)} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}\right] \tag{C.54}$$

$$= D_{\Delta\mathbf{u}^{(i)}}\left[\mu p_{\mathrm{c}}^{n+1(i)} \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}\right] \tag{C.55}$$

$$= D_{\Delta\mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{n+1(i)}\right] \mu \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} + \mu p_{\mathrm{c}}^{n+1(i)} D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}\right]. \tag{C.56}$$

where the terms $D_{\Delta\mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{n+1(i)}\right]$ and $D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}\right]$ have been derived in the previous text. Note that for the trial solution it holds

$$t_{\mathrm{T}\alpha}^{\mathrm{trial}} = t_{\mathrm{T}\alpha}^{n} + \varepsilon_{\mathrm{T}} \bar{m}_{\alpha\beta}^{n(k)} \left(\bar{\xi}^{n+1\beta} - \bar{\xi}^{n\beta}\right). \tag{C.57}$$

$$\boxed{D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right] = D_{\Delta\mathbf{u}^{(i)}}\left[p_{\mathrm{c}}^{n+1(i)}\right] \mu \frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|} + \mu p_{\mathrm{c}}^{n+1(i)} D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha_{\mathrm{trial}}}^{n+1(i)}}{\left\|\mathbf{t}_{\mathrm{T}_{\mathrm{trial}}}^{n+1(i)}\right\|}\right].} \tag{C.58}$$

Appendix D

Matrix notation of the contact kinematic quantities

The gap function

The gap function (2.21) can be discretized as

$$\boxed{g_{\mathrm{N}}^{(i)} = -\mathbf{N}^{\mathrm{T}}\mathbf{x}.} \tag{D.1}$$

The variation of the gap function

The variation of the gap function (B.20) can be written in the discrete form as

$$\boxed{\delta g_{\mathrm{N}}^{(i)} = -\mathbf{c}^{\mathrm{T}}\mathbf{N}.} \tag{D.2}$$

The linearization of the gap function

The linearization of the gap function (C.2) can be discretized as

$$\boxed{\Delta g_{\mathrm{N}}^{(i)} = -\mathbf{N}^{\mathrm{T}}\Delta\mathbf{d}.} \tag{D.3}$$

The curvature tensor

The components of the curvature tensor (2.30) can be written in the discrete form as

$$\boxed{\bar{\kappa}_{\alpha\beta}^{(k)} = -\mathbf{N}_{\alpha\beta}^{\mathrm{T}}\mathbf{x}.} \tag{D.4}$$

The variation of convective coordinates

The variation of convective coordinates (B.28) can be discretized as

$$\delta\bar{\xi}^{\alpha} = \left(\mathbf{A}^{-1}\right)_{\alpha\beta}\left(\mathbf{c}^{\mathrm{T}}\mathbf{T}_{\beta} - g_{\mathrm{N}}\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\beta}\right), \tag{D.5}$$

where auxiliary matrix expressions have been introduced as

$$A_{\alpha\beta} = \bar{m}^{(k)}_{\alpha\beta} + g^{(i)}_{\mathrm{N}} \bar{\kappa}^{(k)}_{\alpha\beta}, \tag{D.6}$$

$$\left(\delta\mathbf{u}^{(i)} - \delta\bar{\mathbf{u}}^{(k)}\right) \cdot \bar{\boldsymbol{\tau}}^{(k)}_{\alpha} = \mathbf{c}^{\mathrm{T}}\mathbf{T}_{\alpha}, \tag{D.7}$$

$$\delta\bar{\mathbf{u}}^{(k)}_{,\alpha} \cdot \bar{\mathbf{n}}^{(k)} = -\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha}, \tag{D.8}$$

$$\boldsymbol{\Delta}\bar{\mathbf{u}}^{(k)}_{,\alpha} \cdot \bar{\mathbf{n}}^{(k)} = -\mathbf{N}^{\mathrm{T}}_{\alpha}\boldsymbol{\Delta}\mathbf{d}, \tag{D.9}$$

The term (2.30) can be further simplified by introducing the matrix $\mathbf{D}_{\alpha}$ as

$$\mathbf{D}_{\alpha} = \left(A_{\alpha\beta}\right)^{-1}\left(\mathbf{T}_{\beta} - g_{\mathrm{N}}\mathbf{N}_{\beta}\right). \tag{D.10}$$

Finally, one can write

$$\boxed{\delta\bar{\xi}^{\alpha} = \mathbf{c}^{\mathrm{T}}\mathbf{D}_{\alpha}.} \tag{D.11}$$

The linearization of the convective coordinates

The linearization of the convective coordinates (C.3) can be expressed analogously to (D.11)

$$\boxed{\Delta\bar{\xi}^{\alpha} = \mathbf{D}^{\mathrm{T}}_{\alpha}\Delta\mathbf{d}.} \tag{D.12}$$

The linearization of the variation of the normal gap function

The linearization of the variation of the normal gap function (C.21) can be discretized as

$$\begin{aligned}
\Delta\delta g^{h(i)}_{\mathrm{N}} &= \left(-\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha}\mathbf{D}^{\mathrm{T}}_{\alpha}\Delta\mathbf{d} - \mathbf{c}^{\mathrm{T}}\mathbf{D}_{\alpha}\mathbf{N}^{\mathrm{T}}_{\alpha}\boldsymbol{\Delta}\mathbf{d}\right) + \mathbf{c}^{\mathrm{T}}\mathbf{D}_{\alpha}\bar{\kappa}^{(k)}_{\alpha\beta}\mathbf{D}^{\mathrm{T}}_{\beta}\Delta\mathbf{d} && \text{(D.13)}\\
&+ g^{(i)}_{\mathrm{N}}\left(-\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha} + \bar{\kappa}^{(k)}_{\alpha\gamma}\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\gamma}\right)\bar{m}^{(k)\alpha\beta}\left(-\mathbf{N}^{\mathrm{T}}_{\beta}\boldsymbol{\Delta}\mathbf{d} + \bar{\kappa}^{(k)}_{\beta\gamma}\mathbf{D}^{\mathrm{T}}_{\gamma}\Delta\mathbf{d}\right) && \text{(D.14)}\\
&= \mathbf{c}^{\mathrm{T}}\left[\left(-\mathbf{N}_{\alpha}\mathbf{D}^{\mathrm{T}}_{\alpha} - \mathbf{D}_{\alpha}\mathbf{N}^{\mathrm{T}}_{\alpha}\right) + \mathbf{D}_{\alpha}\bar{\kappa}^{(k)}_{\alpha\beta}\mathbf{D}^{\mathrm{T}}_{\beta}\right. && \text{(D.15)}\\
&\left.+ g^{(i)}_{\mathrm{N}}\left(-\mathbf{N}_{\alpha} + \bar{\kappa}^{(k)}_{\alpha\gamma}\mathbf{D}_{\gamma}\right)\bar{m}^{(k)\alpha\beta}\left(-\mathbf{N}^{\mathrm{T}}_{\beta} + \bar{\kappa}^{(k)}_{\beta\gamma}\mathbf{D}^{\mathrm{T}}_{\gamma}\right)\right]\Delta\mathbf{d}. && \text{(D.16)}
\end{aligned}$$

If another auxiliary matrix will be introduced as

$$\bar{\mathbf{N}}_{\alpha} = \mathbf{N}_{\alpha} - \bar{\kappa}_{\alpha\beta}\mathbf{D}_{\beta}, \tag{D.17}$$

one can finally write

$$\boxed{\Delta\delta g^{h(i)}_{\mathrm{N}} = \mathbf{c}^{\mathrm{T}}\left[\left(-\mathbf{N}_{\alpha}\mathbf{D}^{\mathrm{T}}_{\alpha} - \mathbf{D}_{\alpha}\mathbf{N}^{\mathrm{T}}_{\alpha}\right) + \left(\mathbf{D}_{\alpha}\bar{\kappa}^{(k)}_{\alpha\beta}\mathbf{D}^{\mathrm{T}}_{\beta}\right) + \left(g_{\mathrm{N}}\bar{m}^{(k)\alpha\beta}\bar{\mathbf{N}}_{\alpha}\bar{\mathbf{N}}_{\beta}\right)\right]\Delta\mathbf{d}.} \tag{D.18}$$

The linearization of the variation of convective coordinate

Before the linearization of the variations of convective coordinate (C.22) will be written in the matrix notation, the attention will be first focused on its last addend. Here the subtraction of linearized displacement vectors will be useful to express using the dot product of the covariant components with its contravariant basis

$$\begin{aligned}
\left(\Delta\mathbf{u}^{(i)}-\Delta\bar{\mathbf{u}}^{(k)}\right)\cdot\left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\delta\bar{\xi}^{\gamma}\right) &= \left\{\left(-\Delta g^{(i)}_{\mathrm{N}}\right)\bar{\mathbf{n}}^{(k)}+\left[\left(\Delta\mathbf{u}^{(i)}-\Delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}^{(k)}_{\beta}\right]\bar{\boldsymbol{\tau}}^{(k)\beta}\right\} \\
&\quad\cdot\left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\delta\bar{\xi}^{\gamma}\right) \qquad\qquad \text{(D.19)} \\
&= -\Delta g^{(i)}_{\mathrm{N}}\left(\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{n}}^{(k)}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\delta\bar{\xi}^{\gamma}\right) \\
&\quad+\left[\left(\Delta\mathbf{u}^{(i)}-\Delta\bar{\mathbf{u}}^{(k)}\right)\cdot\bar{\boldsymbol{\tau}}^{(k)}_{\beta}\right]\bar{m}^{(k)\beta\delta}\bar{\boldsymbol{\tau}}^{(k)}_{\delta} \\
&\quad\cdot\left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\delta\bar{\xi}^{\gamma}\right) \qquad\qquad \text{(D.20)} \\
&= \mathbf{N}^{\mathrm{T}}\boldsymbol{\Delta}\mathbf{d}\left(-\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha}+\bar{\kappa}^{(k)}_{\alpha\beta}\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\alpha}\right) \\
&\quad+\mathbf{T}^{\mathrm{T}}_{\beta}\boldsymbol{\Delta}\mathbf{d}\bar{m}^{(k)\beta\delta}\left(-\mathbf{c}^{\mathrm{T}}\mathbf{T}_{\delta\alpha}+\bar{\boldsymbol{\tau}}^{(k)}_{\delta}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\gamma}\right) \qquad\qquad \text{(D.21)} \\
&= \mathbf{c}^{\mathrm{T}}\left[-\left(\mathbf{N}_{\alpha}+\bar{\kappa}^{(k)}_{\alpha\beta}\mathbf{D}_{\alpha}\right)\mathbf{N}^{\mathrm{T}}\right. \\
&\quad\left.-\bar{m}^{(k)\beta\delta}\left(\mathbf{T}_{\delta\alpha}-\bar{\boldsymbol{\tau}}^{(k)}_{\delta}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\mathbf{D}_{\gamma}\right)\mathbf{T}^{\mathrm{T}}_{\beta}\right]\boldsymbol{\Delta}\mathbf{d}, \qquad\qquad \text{(D.22)}
\end{aligned}$$

$$\left(\Delta\mathbf{u}^{(i)}-\Delta\bar{\mathbf{u}}^{(k)}\right)\cdot\left(\delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\delta\bar{\xi}^{\gamma}\right)=\mathbf{c}^{\mathrm{T}}\left[-\bar{\mathbf{N}}_{\alpha}\mathbf{N}^{\mathrm{T}}-\bar{m}^{(k)\beta\delta}\bar{\mathbf{T}}_{\delta\alpha}\mathbf{T}^{\mathrm{T}}_{\beta}\right]\boldsymbol{\Delta}\mathbf{d}, \tag{D.23}$$

where at the last line a new substitution has been introduced

$$\bar{\mathbf{T}}_{\delta\alpha}=\mathbf{T}_{\delta\alpha}-\left(\bar{\boldsymbol{\tau}}^{(k)}_{\delta}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\right)\mathbf{D}_{\gamma}. \tag{D.24}$$

Analogy can be followed also for the penultimate addend in the equation (C.22)

$$\left(\delta\mathbf{u}^{(i)}-\delta\bar{\mathbf{u}}^{(k)}\right)\cdot\left(\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\Delta\bar{\xi}^{\gamma}\right)=\mathbf{c}^{\mathrm{T}}\left[-\mathbf{N}\bar{\mathbf{N}}^{\mathrm{T}}_{\alpha}-\bar{m}^{(k)\beta\delta}\mathbf{T}_{\beta}\bar{\mathbf{T}}^{\mathrm{T}}_{\delta\alpha}\right]\boldsymbol{\Delta}\mathbf{d}. \tag{D.25}$$

Before expressing the final relation (C.22) in the matrix notation, some auxiliary matrices for several entities will be introduced

$$\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\cdot\Delta\bar{\mathbf{u}}^{(k)}_{,\beta}=-\mathbf{T}^{\mathrm{T}}_{\alpha\beta}\boldsymbol{\Delta}\mathbf{d}, \tag{D.26}$$

$$\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\cdot\delta\bar{\mathbf{u}}^{(k)}_{,\beta}=-\mathbf{c}^{\mathrm{T}}\mathbf{T}_{\alpha\beta}, \tag{D.27}$$

$$\bar{\mathbf{n}}^{(k)}\cdot\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha\beta}=-\mathbf{N}_{\alpha\beta}\boldsymbol{\Delta}\mathbf{d}, \tag{D.28}$$

$$\bar{\mathbf{n}}^{(k)}\cdot\delta\bar{\mathbf{u}}^{(k)}_{,\alpha\beta}=-\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha\beta}. \tag{D.29}$$

With their help and with the aid of previously established substitutions, the equation (C.22) can be written as

$$\begin{aligned}
(A_{\alpha\beta})\,\Delta\delta\bar{\xi}^{\alpha} = {} & \left(\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\beta}\mathbf{T}_{\alpha\beta}^{\mathrm{T}}\Delta\mathbf{d} + \mathbf{c}^{\mathrm{T}}\mathbf{T}_{\alpha\beta}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d}\right) \\
& - \left(\bar{\boldsymbol{\tau}}_{\alpha}^{(k)} \cdot \bar{\mathbf{x}}_{,\beta\gamma}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)} \cdot \bar{\mathbf{x}}_{,\alpha\beta\gamma}^{(k)}\right)\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\beta}\mathbf{D}_{\gamma}^{\mathrm{T}}\Delta\mathbf{d} \\
& + g_{\mathrm{N}}^{(i)}\left(\mathbf{c}^{\mathrm{T}}\mathbf{N}_{\alpha\beta}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d} + \mathbf{c}^{\mathrm{T}}\mathbf{D}_{\beta}\mathbf{N}_{\alpha\beta}^{\mathrm{T}}\Delta\mathbf{d}\right) \\
& + \left[\mathbf{c}^{\mathrm{T}}\mathbf{T}_{\beta\alpha}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d} - \mathbf{c}^{\mathrm{T}}\left(\bar{\mathbf{x}}_{,\alpha\gamma}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\right)\mathbf{D}_{\gamma}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d}\right] \\
& + \left[\mathbf{c}^{\mathrm{T}}\mathbf{D}_{\beta}\mathbf{T}_{\beta\alpha}^{\mathrm{T}}\Delta\mathbf{d} - \mathbf{c}^{\mathrm{T}}\left(\bar{\mathbf{x}}_{,\alpha\gamma}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\beta}^{(k)}\right)\mathbf{D}_{\beta}\mathbf{D}_{\gamma}^{\mathrm{T}}\Delta\mathbf{d}\right] \\
& + \mathbf{c}^{\mathrm{T}}\left(-\mathbf{N}\bar{\mathbf{N}}_{\alpha}^{\mathrm{T}} - \bar{m}^{(k)\beta\delta}\mathbf{T}_{\beta}\bar{\mathbf{T}}_{\delta\alpha}^{\mathrm{T}}\right)\boldsymbol{\Delta}\mathbf{d} \\
& + \mathbf{c}^{\mathrm{T}}\left(-\bar{\mathbf{N}}_{\alpha}\mathbf{N}^{\mathrm{T}} - \bar{m}^{(k)\beta\delta}\bar{\mathbf{T}}_{\delta\alpha}\mathbf{T}_{\beta}^{\mathrm{T}}\right)\boldsymbol{\Delta}\mathbf{d},
\end{aligned} \tag{D.30}$$

where in the fourth row one can recognize the already established substitution

$$\mathbf{c}^{\mathrm{T}}\left[\mathbf{T}_{\beta\alpha} - \left(\bar{\mathbf{x}}_{,\alpha\gamma}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\beta}\right)\mathbf{D}_{\gamma}\right]\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d} \;=\; \mathbf{c}^{\mathrm{T}}\bar{\mathbf{T}}_{\beta\alpha}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d}, \tag{D.31}$$

$$\mathbf{c}^{\mathrm{T}}\left[\mathbf{D}_{\beta}\mathbf{T}_{\beta\alpha}^{\mathrm{T}} - \left(\bar{\mathbf{x}}_{,\alpha\gamma}^{(k)} \cdot \bar{\boldsymbol{\tau}}_{\beta}\right)\mathbf{D}_{\beta}\mathbf{D}_{\gamma}^{\mathrm{T}}\right]\Delta\mathbf{d} \;=\; \mathbf{c}^{\mathrm{T}}\mathbf{D}_{\beta}\bar{\mathbf{T}}_{\beta\alpha}^{\mathrm{T}}\Delta\mathbf{d}, \tag{D.32}$$

so the linearization of the variation of convective coordinates (C.22) can be written in matrix notation as

$$\boxed{\begin{aligned}
\Delta\delta\bar{\xi}^{\alpha} \;=\; & \mathbf{c}^{\mathrm{T}}A^{\alpha\beta}\Big[\left(\mathbf{D}_{\beta}\mathbf{T}_{\alpha\beta}^{\mathrm{T}} + \mathbf{T}_{\alpha\beta}\mathbf{D}_{\beta}^{\mathrm{T}}\right) - \left(\bar{\boldsymbol{\tau}}_{\alpha} \cdot \bar{\mathbf{x}}_{,\beta\gamma}^{(k)} + g_{\mathrm{N}}^{(i)}\bar{\mathbf{n}}^{(k)} \cdot \bar{\mathbf{x}}_{,\alpha\beta\gamma}^{(k)}\right)\mathbf{D}_{\beta}\mathbf{D}_{\gamma}^{\mathrm{T}} \\
& + g_{\mathrm{N}}^{(i)}\left(\mathbf{N}_{\alpha\beta}\mathbf{D}_{\beta}^{\mathrm{T}} + \mathbf{D}_{\beta}\mathbf{N}_{\alpha\beta}^{\mathrm{T}}\right) + \left(\bar{\mathbf{T}}_{\beta\alpha}\mathbf{D}_{\beta}^{\mathrm{T}} + \mathbf{D}_{\beta}\bar{\mathbf{T}}_{\beta\alpha}^{\mathrm{T}}\right) \\
& + \left(-\mathbf{N}\bar{\mathbf{N}}_{\alpha}^{\mathrm{T}} - \bar{m}^{(k)\beta\delta}\mathbf{T}_{\beta}\bar{\mathbf{T}}_{\delta\alpha}^{\mathrm{T}}\right) + \left(-\bar{\mathbf{N}}_{\alpha}\mathbf{N}^{\mathrm{T}} - \bar{m}^{(k)\beta\delta}\bar{\mathbf{T}}_{\delta\alpha}\mathbf{T}_{\beta}^{\mathrm{T}}\right)\Big]\Delta\mathbf{d}.
\end{aligned}} \tag{D.33}$$

The linearization of the contact pressure

The linearization of the contact pressure (C.43) in matrix form is

$$\begin{aligned}
\Delta p_{\mathrm{c}}^{(i)} &= \epsilon_{\mathrm{N}}H\left(g_{\mathrm{N}}^{(i)}\right)\Delta g_{\mathrm{N}}^{(i)}, && \text{(D.34)} \\
&= \epsilon_{\mathrm{N}}H\left(g_{\mathrm{N}}^{(i)}\right)\left(-\mathbf{N}^{\mathrm{T}}\Delta\mathbf{d}\right). && \text{(D.35)}
\end{aligned}$$

The linearization of the metric tensor

The linearization of the metric tensor (C.29) in matrix form is

$$\begin{aligned}
\Delta m_{\alpha\beta}^{(k)} &= \left(\Delta\bar{\mathbf{u}}_{,\alpha}^{(k)} + \bar{\mathbf{x}}_{,\alpha\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right) \cdot \bar{\boldsymbol{\tau}}_{\beta}^{(k)} + \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} \cdot \left(\Delta\bar{\mathbf{u}}_{,\beta}^{(k)} + \bar{\mathbf{x}}_{,\beta\gamma}^{(k)}\Delta\bar{\xi}^{\gamma}\right) && \text{(D.36)} \\
&= \left(-\mathbf{T}_{\alpha\beta}^{\mathrm{T}} + \bar{\boldsymbol{\tau}}_{\beta}^{(k)} \cdot \bar{x}_{,\alpha\gamma}^{(k)}\mathbf{D}_{\gamma}^{\mathrm{T}}\right)\Delta\mathbf{d} + \left(-\mathbf{T}_{\beta\alpha}^{\mathrm{T}} + \bar{\boldsymbol{\tau}}_{\alpha}^{(k)} \cdot \bar{x}_{,\beta\gamma}^{(k)}\mathbf{D}_{\gamma}^{\mathrm{T}}\right)\Delta\mathbf{d}, && \text{(D.37)} \\
&= -\left(\bar{\mathbf{T}}_{\alpha\beta} + \bar{\mathbf{T}}_{\beta\alpha}\right)\Delta\mathbf{d}. && \text{(D.38)}
\end{aligned}$$

$$\boxed{\Delta m_{\alpha\beta}^{(k)} = -\left(\bar{\mathbf{T}}_{\alpha\beta} + \bar{\mathbf{T}}_{\beta\alpha}\right)\Delta\mathbf{d}.} \tag{D.39}$$

The linearization of the unit tangent traction vector

The linearization of the unit tangent traction vector (C.39) in matrix form is

$$\begin{align}
D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right] &= \frac{\Delta t_{\mathrm{T}\alpha}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}}^{(i)\beta}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right)-\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\frac{t_{\mathrm{T}\beta}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\Delta\bar{\boldsymbol{\tau}}^{(k)\beta}\cdot\bar{\boldsymbol{\tau}}^{(k)\gamma}\frac{t_{\mathrm{T}\gamma}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} \tag{D.40}\\
&= \frac{\Delta t_{\mathrm{T}\alpha}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\Delta\bar{\boldsymbol{\tau}}^{(k)\beta}\cdot\mathbf{p}_{\mathrm{T}}^{(i)}, \tag{D.41}
\end{align}$$

where substitution

$$\mathbf{p}_{\mathrm{T}}^{(i)}=\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\bar{\boldsymbol{\tau}}^{(k)\alpha}, \tag{D.42}$$

for unit tangent traction vector has been introduced.

$$\begin{align}
D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t_{\mathrm{T}\alpha}^{(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\right] &= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\Delta\bar{\boldsymbol{\tau}}^{(k)\beta}\cdot\mathbf{p}_{\mathrm{T}}^{(i)} \notag\\
&= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\left(\Delta m^{(k)\beta\gamma}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}+m^{(k)\beta\gamma}\Delta\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\right)\cdot\mathbf{p}_{\mathrm{T}}^{(i)} \notag\\
&= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right) \notag\\
&- \left(p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\Delta m^{(k)\beta\gamma}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}+p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}m^{(k)\beta\gamma}\Delta\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)}\right) \notag\\
&= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\Delta\bar{\boldsymbol{\tau}}_{\beta}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)} \notag\\
&- p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\Delta m^{(k)\beta\gamma}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)} \notag\\
&= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\left(-p_{\mathrm{T}}^{(i)\gamma}\bar{\mathbf{P}}_{\gamma}\Delta\mathbf{u}\right) \notag\\
&- p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\left[-\left(\bar{\mathbf{T}}_{\beta\gamma}+\bar{\mathbf{T}}_{\gamma\beta}\right)\Delta\mathbf{u}\right]\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)} \notag\\
&= \frac{\Delta t_{\mathrm{T}\beta}^{\mathrm{trial}(i)}}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|}\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)+p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\bar{\mathbf{P}}_{\beta}\Delta\mathbf{u} \tag{D.43}\\
&+ p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\left(\bar{\mathbf{T}}_{\beta\gamma}+\bar{\mathbf{T}}_{\gamma\beta}\right)\Delta\mathbf{u}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)}, \tag{D.44}
\end{align}$$

where the term

$$\begin{aligned}
\Delta\bar{\boldsymbol{\tau}}^{(k)}_{\alpha}\cdot\mathbf{p}_{\mathrm{T}} &= \left(\Delta\bar{\mathbf{u}}^{(k)}_{,\alpha}+\bar{\mathbf{x}}^{(k)}_{,\alpha\beta}\Delta\xi^{\beta}\right)\cdot\mathbf{p}^{(i)}_{\mathrm{T}} && \text{(D.45)}\\
&= \left[\mathbf{P}^{\mathrm{T}}_{\alpha}-\left(\mathbf{p}^{(i)}_{\mathrm{T}}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\beta}\right)\mathbf{D}^{\mathrm{T}}_{\beta}\right]\Delta\mathbf{d} && \text{(D.46)}\\
&= \bar{\mathbf{P}}^{\mathrm{T}}_{\alpha}\Delta\mathbf{d}, && \text{(D.47)}
\end{aligned}$$

was described with the aid of matrices $\mathbf{P}_{\alpha}$ and $\bar{\mathbf{P}}_{\alpha}$

$$\mathbf{P}_{\alpha} := \begin{bmatrix} 0\\ \vdots\\ 0\\ -N_{1,\alpha}\mathbf{p}^{(i)}_{\mathrm{T}}\\ \vdots\\ -N_{n^{(2)}_{sn},\alpha}\mathbf{p}^{(i)}_{\mathrm{T}} \end{bmatrix}, \tag{D.48}$$

$$\bar{\mathbf{P}}_{\alpha} := \mathbf{P}_{\alpha}-\left(\mathbf{p}^{(i)}_{\mathrm{T}}\cdot\bar{\mathbf{x}}^{(k)}_{,\alpha\gamma}\right)\mathbf{D}_{\gamma}. \tag{D.49}$$

So that

$$\boxed{\begin{aligned} D_{\Delta\mathbf{u}^{(i)}}\left[\frac{t^{(i)}_{\mathrm{T}\alpha}}{\left\|\mathbf{t}^{(i)}_{\mathrm{T}}\right\|}\right] &= \frac{\Delta t^{\mathrm{trial}(i)}_{\mathrm{T}\beta}}{\left\|\mathbf{t}^{(i)}_{\mathrm{T}}\right\|}\left(\delta^{\alpha}_{\beta}-p^{(i)}_{\mathrm{T}\alpha}p^{(i)\beta}_{\mathrm{T}}\right)+p^{(i)}_{\mathrm{T}\alpha}p^{(i)\beta}_{\mathrm{T}}\bar{\mathbf{P}}_{\beta}\Delta\mathbf{d}\\ &+\ p^{(i)}_{\mathrm{T}\alpha}p^{(i)}_{\mathrm{T}\beta}\left(\bar{\mathbf{T}}_{\beta\gamma}+\bar{\mathbf{T}}_{\gamma\beta}\right)\Delta\mathbf{d}\bar{\boldsymbol{\tau}}^{(k)}_{\gamma}\cdot\mathbf{p}^{(i)}_{\mathrm{T}}. \end{aligned}} \tag{D.50}$$

The linearization of the tangential components of the contact traction vector

Again, it is necessary to distinguish between the state of sticking and slipping. In the case of *sticking*

$$\begin{aligned}
D_{\Delta\mathbf{u}^{(i)}}\left[t^{n+1(i)}_{\mathrm{T}\alpha}\right] &= D_{\Delta\mathbf{u}^{(i)}}\left[t^{n+1(i)}_{\mathrm{T}\alpha_{\mathrm{trial}}}\right] && \text{(D.51)}\\
&= -\epsilon_{\mathrm{T}}\bar{m}^{n(k)}_{\alpha\beta}\Delta\bar{\xi}^{n+1\beta}-\epsilon_{\mathrm{T}}\Delta\bar{m}^{n(k)}_{\alpha\beta}\left(\bar{\xi}^{n+1\beta}-\bar{\xi}^{n\beta}\right) && \text{(D.52)}\\
&= -\epsilon_{\mathrm{T}}\left[\bar{m}^{n(k)}_{\alpha\beta}\mathbf{D}^{\mathrm{T}}_{\beta}\Delta\mathbf{d}-\left(\bar{\mathbf{T}}_{\alpha\beta}+\bar{\mathbf{T}}_{\beta\alpha}\right)\Delta\mathbf{d}\left(\bar{\xi}^{n+1\beta}-\bar{\xi}^{n\beta}\right)\right], && \text{(D.53)}
\end{aligned}$$

$$\boxed{D_{\Delta\mathbf{u}^{(i)}}\left[t^{n+1(i)}_{\mathrm{T}\alpha_{\mathrm{trial}}}\right] = \epsilon_{\mathrm{T}}\left(\bar{\mathbf{T}}_{\alpha\beta}+\bar{\mathbf{T}}_{\beta\alpha}\right)\Delta\mathbf{d}\left(\bar{\xi}^{n+1\beta}-\bar{\xi}^{n\beta}\right)-\epsilon_{\mathrm{T}}\bar{m}_{\alpha\beta}\mathbf{D}^{\mathrm{T}}_{\beta}\Delta\mathbf{d},} \tag{D.54}$$

and for the case of *slipping*

$$D_{\Delta\mathbf{u}^{(i)}}\left[t^{n+1(i)}_{\mathrm{T}\alpha}\right] = D_{\Delta\mathbf{u}^{(i)}}\left[p^{n+1(i)}_{\mathrm{c}}\right]\mu p^{(i)}_{\mathrm{T}\alpha}+\mu p^{n+1(i)}_{\mathrm{c}}D_{\Delta\mathbf{u}^{(i)}}\left[p^{(i)}_{\mathrm{T}\alpha}\right]. \tag{D.55}$$

$$\boxed{\begin{aligned} D_{\Delta\mathbf{u}^{(i)}}\left[t_{\mathrm{T}\alpha}^{n+1(i)}\right] &= -\epsilon_{\mathrm{N}}\mu p_{\mathrm{T}\alpha}^{(i)} H\left(g_{\mathrm{N}}^{(i)}\right)\mathbf{N}^{\mathrm{T}}\Delta\mathbf{d} \\ &+\mu p_{\mathrm{c}}^{n+1(i)}\left[\epsilon_{\mathrm{T}}\left(\bar{\mathbf{T}}_{\alpha\beta}+\bar{\mathbf{T}}_{\beta\alpha}\right)\Delta\mathbf{d}\left(\bar{\xi}^{n+1\beta}-\bar{\xi}^{n\beta}\right)-\epsilon_{\mathrm{T}}\bar{m}_{\alpha\beta}\mathbf{D}_{\beta}^{\mathrm{T}}\Delta\mathbf{d}\right]\frac{\left(\delta_{\beta}^{\alpha}-p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\right)}{\left\|\mathbf{t}_{\mathrm{T}}^{(i)}\right\|} \\ &+\mu p_{\mathrm{c}}^{n+1(i)}p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}}^{(i)\beta}\bar{\mathbf{P}}_{\beta}\Delta\mathbf{d} \\ &+\mu p_{\mathrm{c}}^{n+1(i)}p_{\mathrm{T}\alpha}^{(i)}p_{\mathrm{T}\beta}^{(i)}\left(\bar{\mathbf{T}}_{\beta\gamma}+\bar{\mathbf{T}}_{\gamma\beta}\right)\Delta\mathbf{d}\bar{\boldsymbol{\tau}}_{\gamma}^{(k)}\cdot\mathbf{p}_{\mathrm{T}}^{(i)}. \end{aligned}} \tag{D.56}$$