Technische Universität Graz

DISSERTATION

Frictional mortar contact for finite deformation problems with synthetic contact kinematics

---

ausgeführt zum Zwecke der
Erlangung des akademischen Grades
eines
Doktors der technischen Wissenschaften

eingereicht an der
Technischen Universität Graz
Fakultät für Maschinenbau und Wirtschaftswissenschaften

von

Dipl.-Ing. Michael E. Hammer

Betreuer: Prof. Christian Celigoj   Technische Universität Graz
Gutachter: Prof. Alois Steindl   Technische Universität Wien

Institut für Festigkeitslehre der Technischen Universität Graz
Graz, im Mai 2012

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am ……………………………   13. Mai 2012   ………………………………………………..
                                                                                    (Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

  13. Mai 2012   ………………………………………………..
…………………………   (signature)
        date

Abstract

This thesis treats the numerical modelling of structural contact problems. Beside the normal contact we will implement Coulomb's friction law in tangential direction. A special focus is given on the handling of non matching mesh interfaces. In comparison to the classical node-on-segment approach we will apply a rather new segment-to-segment method, namely the mortar method. It is well known that the node-on-segment methods suffer deficiencies of locking or suboptimal convergence behavior. In contrast the mortar method shows optimal convergence behavior. The advantages of this theory have been proven by various works on this topic. The mortar method was implemented as mixed method with displacements and Lagrange multipliers as degrees of freedom. For the mortar method the contact constraints are fulfilled in weak manner. In this thesis will concentrate on the contact kinematics and compare different formulations for the surface normal field, which is vital for any contact formulation. Further a special kind of numerical integration scheme is applied. The presented theory is then applied to various numerical examples to show the performance of our algorithm.

Kurzfassung

Diese Arbeit behandelt die numerische Modellierung des Kontakts zweier verformbarer Körper. Neben dem Kontakt in Richtung der Oberflächennormalen wurde auch der Kontakt in Tangentialrichtung als Coulombsche Reibung implementiert. Spezielles Augenmerk wurde dabei auf die Behandlung nicht kompatibler Netzdiskretisierung gelegt. Im Vergleich zu den klassischen "node-on-segment" Methoden, wurden die neuere "segment-to-segment" Methode, die so genannte Mortar Methode, verwendet. Es ist bekannt, dass "node-on-segment" Methoden diverse Schwierigkeit wie "Locking"- Effekte und suboptimales Konvergenzverhalten zeigen. Im Gegensatz dazu zeigt die Mortar Methode optimales Konvergenzverhalten. Die Vorteile dieser Methode wurden bereits durch mehrere Arbeiten zu diesem Thema bewiesen. Die Mortar Methode wurde als gemischtes Verfahren, mit Verschiebungen und den Lagrange Multiplikatoren als Unbekannte, realisiert. Die Kontaktbedingungen werden hier nur mehr in schwacher Form erfüllt. In dieser Arbeit werden wir uns auf die Formulierung der Kontaktkinematik konzentrieren, um einen Vergleich zwischen verschiedenen Oberflächennormalen zu ermöglichen. Darüber hinaus kommt ein spezielles numerisches Integrationsschema zur Anwendung. Die dargestellte Theorie wird auf verschiedene numerische Beispiele angewandt, um die Leistungsfähigkeit des Algorithmus zu zeigen.

Acknowledgements

Contents

Contents

Contents

List of Figures

List of Figures

List of Algorithms

List of Algorithms

1. Introduction

Contact mechanics is part of a wide range of engineering problems. The interaction between different bodies, in particular, support of structures, is always connected with contact analysis. There have been numerical contact algorithms for quite a while now. But for real world calculations, still many challenges remain.

There are different aspects of contact mechanics which can be reviewed. The problem of finding a suitable constitutive law for the tangential contact part is one of them. In this work we will use the classical Coulomb friction law. Beside the various friction laws, wearing and anisotropy is part of the topological analysis which is also not part of this thesis. For structural computations, the thermomechanical contact problem is part of a complete analysis. A suitable heat transfer equation over the contact surface has to be formulated for this kind of problems. However, heat transfer problems are not considered here.

In this work we concentrate on the numerical modelling of the mechanical contact problem and not on the description of the physical incidents. Because of its simplicity and efficiency the most frequently used method for numerical contact analysis on non matching mesh interfaces is the node-on-segment approach (see Hallquist et al. [13] or Laursen and Simo [23] for extended treatment). It is well known, that these methods suffer deficiencies of locking or suboptimal convergence behavior. On the other hand it is known (Wohlmuth [37]), that the mortar method shows optimal convergence behavior. Various engineering approaches to the mortar method have been realized, for instance Puso and Laursen [29], Puso and Laursen [30], Fischer and Wriggers [6], Hartmann and Ramm [15]. All have shown the performance and advantages of this algorithm.

1.1. Current state-of-the-art of contact mechanics

Various methods for numerical structural contact mechanics are known. For a better classification of the mortar method we will sketch the other methods and show the differences and the main properties.

1.1.1. The node-to-node contact method

The node-to-node contact method might be seen as the most simple method to treat structural contact numerically. The situation is illustrated in Figure 1.1. The mathematics is quite simple as we only add nodal forces to the system. The implementation might be tricky as activation and deactivation of linkages might be a difficult algorithmic problem. It is also important that the mesh discretization of the two bodies in contact have to be

compatible. It is difficult to connect a lot of nodes of one side to a few nodes on the other one.
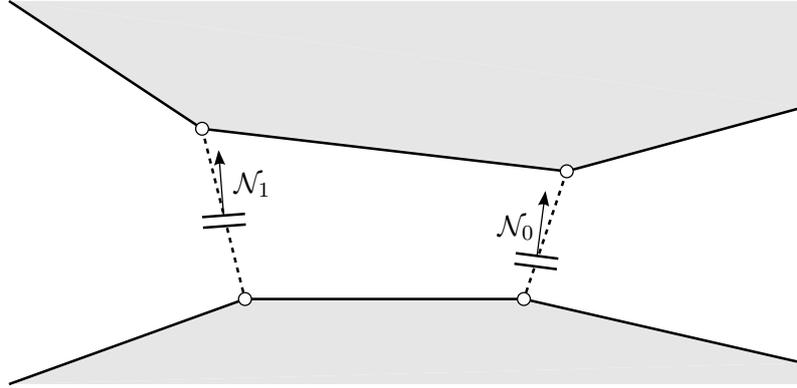
Figure 1.1.: The node-to-node contact method as a simple contact method. The nodes of two bodies in contact are brought into direct connection. The link between the two nodes does not have to be fixed. To simulate a Coulomb friction a sliding element is embedded in the linkage. Normally the normal vector $\mathcal{N}$ of this sliding element can be choosen freely. The model of the linkage can be extended easily, e.g. by adding a damper in series.

1.1.2. The node-to-segment or segment-to-segment contact method

In this method no direct relation between nodes is formulated. We map points on one side to segments of the other. For this method we need to define a slave and a master surface. This is necessary due to the algorithm and has no physical background. In real world both contact partners are equally prioritized. This is one drawback of those methods which is also the case for the mortar method.

We formulate a virtual work of the surface traction (to simplify the situation we treat the normal part only, a similar procedure is applied for tangential part, as we will see in this thesis) as follows

$$\delta\Pi_N = \int_{\gamma_C} t_N \delta g_N \mathrm{d}\gamma \tag{1.1}$$

which can be extracted from the weak formulation for our boundary value problem (see Section 5.2). This virtual contact work will lead to a non-symmetric equation system.

We do not know the surface traction t_N a priori. There are different possibilities to overcome this problem.

Penalty regularization We define the normal traction to depend on the gap function

$$t_N = \varepsilon g_N \tag{1.2}$$

with the parameter ε (the penalty parameter). The virtual contact work is then given by

$$\delta^P \Pi_N = \int_{\gamma_C} \varepsilon g_N \delta g_N \mathrm{d}\gamma \tag{1.3}$$

and has to be zero for equilibrium. A graphical interpretation of this situation is given in Figure 1.2. To carry out the given quadrature one has to make the decision on which side the integration is done. This is the first reason why we have to define a master and a slave side. The second issue is the calculation of corresponding master contact points for a given slave point. Later on, the considered slave point will be the quadrature point and the projection point on the master side, will have to be determined by a nearest point projection algorithm.
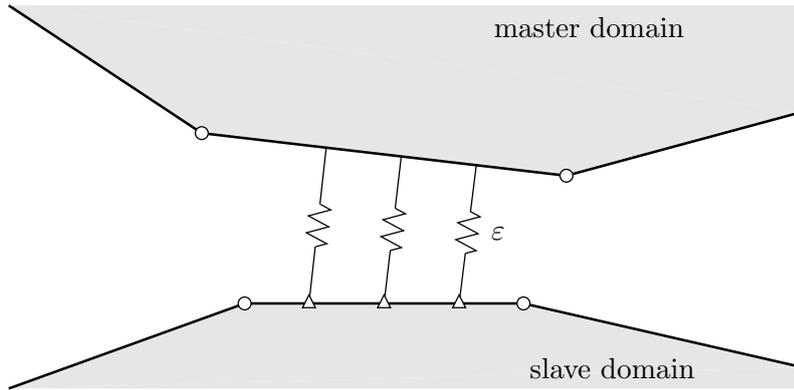
Figure 1.2.: The penalty regularized segment-to-segment contact method. One can interpret the definition of the contact traction as springs mounted between the integration points on the slave side and its corresponding projection point on the master side. The springs are "activated" if the master point penetrates into the slave surface and "deactivated" else. As we need a penetration to activate the spring and to produce a reaction force, it is impossible to get zero penetration, if the normal traction $t_N \neq 0$.

Lagrange parameters By introducing Lagrange parameters for the normal traction as additional unknowns, it is also possible to incorporate the virtual contact work into our equation system. The drawback is, that we have additional unknowns and zero entries on the main diagonal of the stiffness matrix. This can be avoided by using an augmented Lagrangian method. The virtual contact work is given by

$$\delta^L \Pi_N = \int_{\gamma_C} \lambda_N \delta g_N \mathrm{d}\gamma \tag{1.4}$$

and has to be zero for equilibrium. This method already allows zero penetration for certain situations. A visualization is given in Figure 1.3. The problem of detecting active segments (or nodes) has to be solved by a suitable active set strategy. This decision is often taken point-wise by checking the traction or the penetration in the nodes.
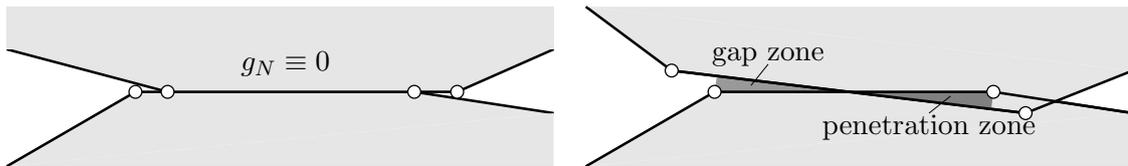
Figure 1.3.: The segment-to-segment contact method using Lagrange multipliers.

1.1.3. The mortar contact method

The mortar method was developed as a framework for coupling non-matching discretizations. The first application of this framework on linear contact problems is well presented in Belgacem et al. [3].

> **Remark:** "mortar" is the translation of the original french word "joint" into English ("Mauerfuge", "Mörtel" in German). The expression should illustrate the agglutinative character of this method although the meshes (the bricks of the wall) do not coincide.

As pointed out in this work, the main property of the mortar method is the incorporation of the contact inequalities (the Karush-Kuhn-Tucker conditions) in a weak form. This is the extension to earlier contact methods like the node-on-node or the segment-to-segment contact method. For the mortar method we incorporate not only the virtual work of the contact tractions t_N (that is the surface traction in normal direction on the contact surface) but also the variational form of the contact condition and obtain

$$\delta\Pi_N = \int_{\gamma_C} t_N \delta g_N \mathrm{d}\gamma + \int_{\gamma_C} \delta t_N g_N \mathrm{d}\gamma \tag{1.5}$$

where it is already observable that this method might lead to a symmetric equation system.

The mortar method can be realized as a mixed method with displacements and Lagrange multipliers as unknowns. But there exist various formulations, including regularized ones, with penalty or augmented Lagrangian methods.

As the kinetic and the kinematic contact constraints are only fulfilled in weak form (in an integral formulation of segments γ_C^h) there might occur regions with gaps or regions of penetration in the final solution. The situation is equal to the one presented in Figure 1.3.

The Lagrange multipliers have to be constructed (interpolated) on one contact partner. This contact partner is called the non-mortar or slave domain. The integration of the virtual contact work is realized on this side. The other side is called mortar or master domain. This also means, the mortar method is variant with respect to the choice, which contact partner is mortar and which is non-mortar. It is important to choose the finer discretization as non-mortar domain (see Wolmuth [38]).

It also part of a mortar method to define a more sophisticated active set strategy which is directly related to the incorporated weak non-penetration condition. The given inequalities

$$g_N \geq 0 \qquad \text{(no penetration)} \tag{1.6}$$
$$t_n \leq 0 \qquad \text{(no addhesion)} \tag{1.7}$$

are transferred into a weak form too

$$\int_{\gamma_C} g_N \, \delta t_N \mathrm{d}\gamma \geq 0 \qquad \text{(no penetration)} \tag{1.8}$$

$$\int_{\gamma_C} t_N \, \delta g_N \mathrm{d}\gamma \leq 0 \qquad \text{(no addhesion)} \tag{1.9}$$

There is still the challenge of handling these non-linear inequalities. We present a solution in this work which is suitable for a Newton-Raphson procedure. It is also possible (but not done in this thesis) to include the active set conditions into the set of linear equations. This leads to the so called semi-smooth Newton-Rapshon procedure (see Hüeber and Wohlmuth [17] or Popp et al. [27]).

By using the so called dual mortar methods (see Wolmuth [38]) it is possible to eliminate the Lagrange multiplier from the set of linear equations. This has been successfully applied for non-linear engineering problems by Hartmann and Ramm [15] and Popp et al. [27] and successive works. We will omit the condensation of Lagrange parameters in this work.

1.2. Aim and motivation

For this thesis we try to merge the various engineering approaches for finite deformation, large sliding contact algorithms based on the mortar method. The variational formulation may be based on the weak formulation of the boundary value problem, adding a weak formulation of the contact constraint. It is also possible to define a contact potential (see Fischer and Wriggers [6]). We will show the connection of the two approaches.

In many publications on finite deformation problems the contact conditions have been regularized. We will circumvent the regularization procedure and solve the saddle point problem as done by Tur et al. [35], Popp et al. [27] or Hartmann and Ramm [15].

To formulate the contact constraints one has to parametrize and describe the contact surface. Based on this description the kinematic properties, like the gap function, can be calculated. Schweizerhof and Konyukhov [32] showed a fully covariant description of the contact kinematics which is given based on the non-discretized continuous surface. A more classical formulation (used by most people) can be found in Laursen and Simo [23]. A challenge for numerical contact algorithms is, that the surface in general is only C^0 continuous after finite element discretization. This discontinuity leads to different problems which will be discussed throughout this work. Common to most algorithms is, to average the normal field. This was first proposed by Yang et al. [44]. We will analyze the influence of this averaging procedure on the quality of the solution. Therefore we formulate the contact kinematics in a rather synthetic way and try to conserve as much of the discrete surface structure as possible. This leads to a pretty new way of formulating the contact kinematics.

A crucial part of the mortar method is the solution of the integral arising by weakly imposing the contact constraints. There are two possible approaches for doing so. One can segment the integration domain at each corner node of the boundary. This is very popular as it represents the exact quadrature of the mortar integrals for two dimensional problems. Fischer and Wriggers [6] have already shown in their work, that a concentrated integration

method is suitable too. For this method we evaluate the integrand on the integration points without detecting the C^1 discontinuities on the boundary. We will show by numerical experiments the convergence performance of this approximative but fast algorithm.

To formulate the tangential contact we need an objective velocity measure for the relative slip. As Yang et al. [44] already showed, this is not self-evident. We will transfer their procedure of creating an objective velocity on our integration scheme and interpret the arising variables.

It is possible to develop a combined algorithm which solves the boundary value problem and the active set at once, the so called semi-smooth Newton-Raphson procedure. This is shown e.g. in Popp et al. [27]. We will apply a fixed point Newton-Raphson procedure, where we search for the displacements first and actualize the active set afterwards.

1.3. Outline of the work

We start by giving a short introduction into continuum mechanics. We only add those special quantities, which arise in our contact problem. In particular, this means an extension of the boundary value problem on two (or multiple) distinct bodies.

Next the contact kinematics for the continuous non discretized contact surface is discussed. Here we try to formulate synthetic kinematics, which is able to deal with different formulations for the normal field. Therefore we have to anticipate and consider some problems which arise later through discretization. However, primarily we treat the continuous surface at this point.

Now we transfer the continuous contact kinematics onto the discretized surface. Here we will distinguish between two normal fields, which we will discuss:

- A non-continuous mortar side normal field,

- and an averaged non-mortar side normal field.

At this point we have to deal with the first special cases and argue over the solvability of the nearest point projection procedure.

With the help of contact kinematics it is now possible to formulate the contact kinetics and define the contact constraints. This means we specify the Karush-Kuhn-Tucker conditions for the structural contact problem. We also show the incorporation of these strong inequalities in the variational formulation. Here the character of the mortar method becomes obvious. We end up with the formulation of the weak boundary value problem.

We will solve this problem via the finite element method which is then introduced. Since our contact conditions, the strain measure and the constitutive law are non-linear, we apply a linearization procedure to solve the non-linear system (Newton-Raphson procedure). For the contact constraint enforcement we will present two methods. First, a penalty regularization is described, which is not the classical mortar method, but allows a more compact formulation. Second, we introduce the Lagrange multiplier method which extends the weak problem to a mixed formulation. We also define a finite mortar element and discuss

two integration schemes. To incorporate the contact conditions formulated as inequalities we develop an active set strategy.

Until now we did not mention how to implement the presented mathematical model. A brief introduction of the global solution algorithm and the extensions of a ordinary (non-contact) Newton-Raphson procedure is given. The nearest point projection is an essential part of every contact algorithm and for the discrete surface we have to deal with various special cases. It is also required to find the possible contact partners as they are not known a priori. We will sketch the algorithm here. We close the chapter with the implementation of the active set strategy.

The performance of the presented algorithm is shown via numerical experiments. We have chosen different examples with varying amount of unknowns and different challenges to illustrate the robustness and characteristics of our implementation. The examples are based on problems well known from literature.

We will end this work with a short conclusion and an outlook of future challenges.

1. Introduction

2. Continuum mechanics

In this chapter a brief introduction to continuum mechanics is given. There is well known literature on elasticity theory, the book of Bonet and Wood [5] as excellent introduction, Marsden and Hughes [25] describe the mathematical foundations of elasticity theory on manifolds. The basics of continuum mechanics are described in the famous work of Truesdell and Noll [34]. Only the essentials needed for structural contact mechanics are treated here.

We start with kinematics, used in elasticity theory, to describe the bodies, their deformation and motion. To formulate the equilibrium state a strain measure and the definition of a stress measure is required. Those properties are used to define the boundary value problem for deformable bodies. This strong formulation is then transferred with the help of Galerkin's method into a weak formulation suitable for the finite element method. Our differential equation requires a constitutive law connecting the kinematic strain measure with the kinetic stress tensor.

During this chapter no special attention on contact mechanics is given. All the presented theory is applicable on classical structural problems. We just extend the theory to multibody problems and introduce a contact surface.

2.1. Finite kinematics

In elasticity theory the motion and deformation of bodies are described with the help of two configurations (see Figure 2.1). The initial or material configuration (also Lagrangian manifold) $\mathbf{X} \in \Omega$ at the time $t = 0$ and the running or spatial configuration (also Eulerian manifold) denoted as $\mathbf{x} \in \omega$ at the time t. We will denote points and properties in the material configuration with upper case letters (like $\mathbf{P}$, $\mathbf{X}$) and on the material configuration with lower case letters (like $\mathbf{p}$, $\mathbf{x}$). Furthermore we define the mapping of $\mathbf{X}$ onto $\mathbf{x}$ by $\varphi(\mathbf{X}, t)$

$$\mathbf{x} = \varphi\left(\mathbf{X}, t\right) \tag{2.1}$$

and the inverse mapping by $\varphi^{-1}(\mathbf{x}, t)$

$$\mathbf{X} = \varphi^{-1}\left(\mathbf{x}, t\right) \tag{2.2}$$

2.1.1. Deformation gradient

A fundamental quantity for large deformation analysis is the deformation gradient. This is defined as

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial \varphi(\mathbf{X}, t)}{\partial \mathbf{X}} \tag{2.3}$$
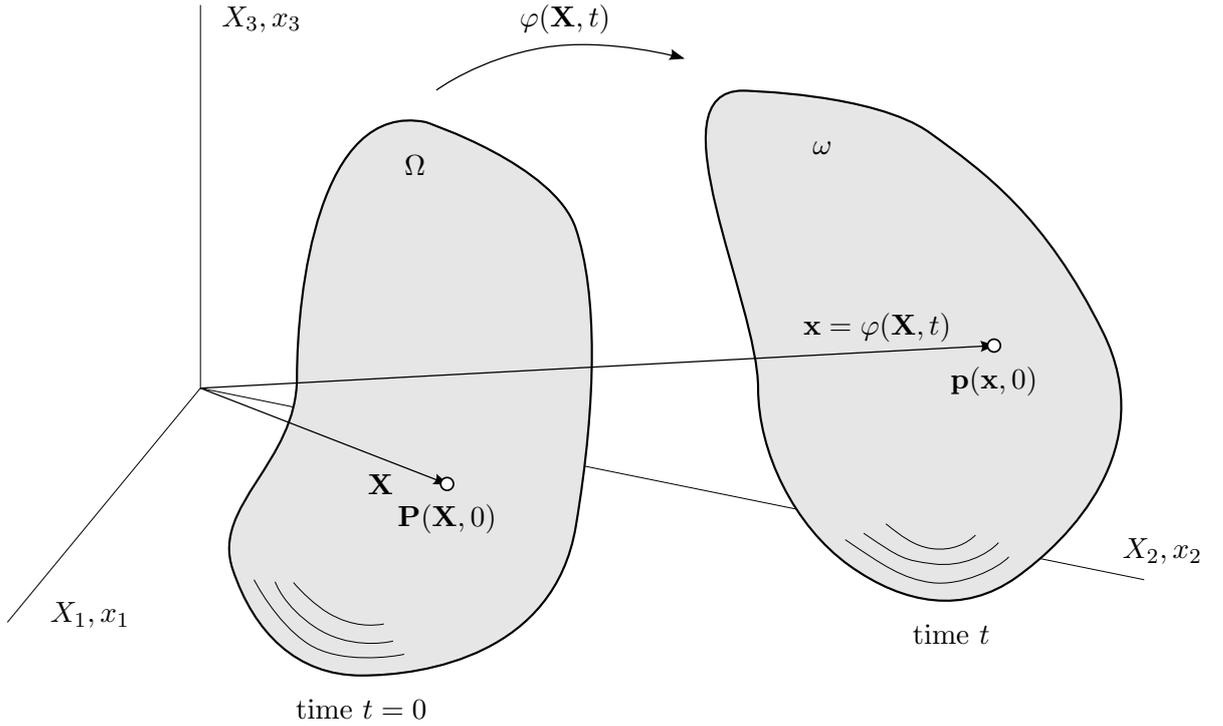
Figure 2.1.: Kinematic configurations of continuum mechanics for a single body containing the material configuration Ω and the spatial configuration ω. A point $\mathbf{P}$ is shown in the material configuration and its counterpart $\mathbf{p}$ on the current configuration. Their position vectors are $\mathbf{X}$ and $\mathbf{x}$, respectively. The kinematic mapping from tge material configuration to the spatial configuration is described with φ. This is done in the coordinate system $\mathbf{X}_I$, $\mathbf{x}_i$ which coincide for both configurations.

which allows a tangential mapping

$$\mathrm{d}\mathbf{x} = \mathbf{F}\,\mathrm{d}\mathbf{X} \tag{2.4}$$

The components of the deformation gradient are

$$F_{iI} = \frac{\partial x_i}{\partial X_I} \tag{2.5}$$

The deformation gradient $\mathbf{F}$ defines the tangential mappings from the material to the spatial frame. This *push forward* operation $\Phi_*[]$ is given in the following for the infinitesimal material vector $\mathrm{d}\mathbf{X}$

$$\mathrm{d}\mathbf{x} = \Phi_*\left[\mathrm{d}\mathbf{X}\right] = \mathbf{F}\mathrm{d}\mathbf{X} \tag{2.6}$$

The same can be done for the inverse direction by doing a *pull back* operation $\Phi_*^{-1}[]$ of the spatial vector $\mathrm{d}\mathbf{x}$ onto the material frame via

$$\mathrm{d}\mathbf{X} = \Phi_*^{-1}\left[\mathrm{d}\mathbf{x}\right] = \mathbf{F}^{-1}\mathrm{d}\mathbf{x} \tag{2.7}$$

2.1.2. Velocity

For contact mechanics the relative slip between the mortar and the non-mortar body is vital. Therefore we analyze the rates of change of kinematic quantities - especially the velocities. The definition of the velocity vector given at a material point is straight forward

$$\mathbf{V}(\mathbf{X}, t) = \frac{\partial}{\partial t}[\varphi(\mathbf{X}, t)] \tag{2.8}$$

For the contact definition we need the definition of the spatial velocity which can be given as

$$\mathbf{v}(\mathbf{x}, t) = \mathbf{V}\left(\varphi^{-1}(\mathbf{x}, t), t\right) \tag{2.9}$$

2.1.3. Objectivity (frame indifference)

To formulate constitutive laws (like the tangential part of the frictional formulation) the important concept of "objectivity" or "frame indifference" has to be considered.

The kinematic mapping is given with $\varphi(\mathbf{X}, t)$. Now we assume that we view the same motion defined in the reference frame $\mathbf{x}$ from a different reference frame $\mathbf{x}^*$. We define the transformation between the two reference frames as

$$\mathbf{x}^* = \mathbf{c}(t) + \mathbf{Q}(t) \cdot \mathbf{x} \tag{2.10}$$

with $\mathbf{Q}(t)$ being an orthogonal rotation matrix ($\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$) and $\mathbf{c}(t)$ being a translational vector. In the second reference frame the motion appears as

$$\mathbf{x}^* = \varphi^*(\mathbf{X}, t) = \mathbf{c}(t) + \mathbf{Q}(t)\,\varphi(\mathbf{X}, t) \tag{2.11}$$

An objective tensor $\mathbf{b}$ must be invariant with respect to the chosen reference frame, $\mathbf{x}^*$ or $\mathbf{x}$ which means for a tensor of second order

$$\mathbf{b}^* \equiv \mathbf{Q}^T\,\mathbf{b}\,\mathbf{Q} \tag{2.12}$$

This is important to take into account, if it comes to the definition of the relative velocity of two contact surfaces (see Section 3.3.1.1).

2.1.3.1. Lie derivative

A generalization of the procedure, of generating objective rates, is given with the Lie derivative (see e.g. Bonet and Wood [5], Laursen [24]). Let us consider a given spatial tensor $\mathbf{b}$, the Lie derivative of $\mathbf{b}$ is defined by

$$\mathcal{L}_\Phi\{\mathbf{b}\} = \Phi_*\left[\frac{\mathrm{d}}{\mathrm{d}t}\left(\Phi_*^{-1}[\mathbf{b}]\right)\right] \tag{2.13}$$

This means that the time derivation is done on the material configuration with a fixed reference frame. The pull-back to the material configuration $\Phi_*^{-1}[]$ and the push-forward operation $\Phi_*[]$ is frame indifferent per definition. According to Marsden and Hughes [25] (p.99 ff) all so-called objective rates my be obtained by application of the Lie derivative.

2.2. Strain

Our calculations are based on an updated Lagrangian formulation. Therefore wee need the right Cauchy-Green deformation tensor $\mathbf{C}$ which is defined as

$$\mathrm{d}\mathbf{x}_1 \cdot \mathrm{d}\mathbf{x}_2 = \mathrm{d}\mathbf{X}_1 \cdot \mathbf{F}^T\mathbf{F} \cdot \mathrm{d}\mathbf{X}_2 = \mathrm{d}\mathbf{X}_1 \cdot \mathbf{C} \cdot \mathrm{d}\mathbf{X}_2$$
$$\mathbf{C} = \mathbf{F}^T\mathbf{F} \tag{2.14}$$

The right Cauchy-Green tensor is a material tensor. Based on $\mathbf{C}$ we define a deformation measure on the material frame by

$$\frac{1}{2}\left(\mathrm{d}\mathbf{x}_1 \cdot \mathrm{d}\mathbf{x}_2 - \mathrm{d}\mathbf{X}_1 \cdot \mathrm{d}\mathbf{X}_2\right) = \mathrm{d}\mathbf{X}_1 \cdot \mathbf{E} \cdot \mathrm{d}\mathbf{X}_2 \tag{2.15}$$

With this procedure we obtain the Green-Lagrange strain tensor $\mathbf{E}$

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{C} - \mathbf{I}\right) \tag{2.16}$$

with $\mathbf{I}$ being the second order unit tensor. Of course there are various other strain measures imaginable. These can be found in, e.g. Bonet and Wood [5] or Truesdell and Noll [34].

2.3. Stress

In the first step the definition of a stress tensor is done on the spatial frame. The Cauchy-stress is the "physical" stress tensor. Let us define a traction vector $\mathbf{t}(\mathbf{n})$ at a point point $\mathbf{p}(\mathbf{x})$ in the cross section with the normal $\mathbf{n}$ by

$$\mathbf{t}(\mathbf{n}) = \lim_{\Delta a \to 0} \frac{\Delta \mathcal{P}}{\Delta a} \tag{2.17}$$

Based on the idea of doing three linearly independent cross sections a stress tensor $\boldsymbol{\sigma}$ can be defined. For one cross section the situation is sketched in Figure 2.2. There are well suitable derivations in various books like Bonet and Wood [5]. Translational equilibrium on the Cauchy tetraeder gives

$$\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n} \tag{2.18}$$

The rotational equilibrium conditions, as we will see later on, render $\boldsymbol{\sigma}$ to be symmetric.

Further, let us consider the traction on the surface γ. If one knows the Cauchy stress on the boundary at the surface point ($\mathbf{p} \in \gamma$), the surface traction $\mathbf{t}_\gamma$ is given by

$$\mathbf{t}_\gamma = \boldsymbol{\sigma}\big|_\gamma \cdot \mathbf{n}\big|_\gamma \tag{2.19}$$

We will see later on (see Figure 5.3) that the surface traction inside the contact surface γ_C is very crucial for the contact formulation. Variational contact formulations due to Nitsche (see, e.g. Wriggers and Zavarise [39]) rely on Cauchy's equation from Eq. (2.18) to extend the body stresses $\boldsymbol{\sigma}$ onto the boundary γ_C.
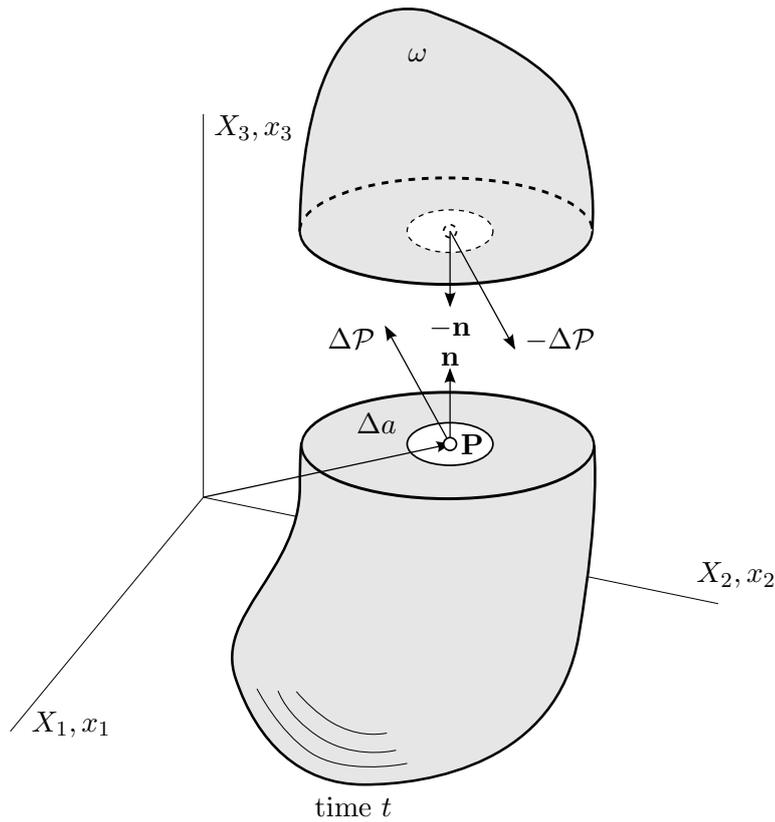
Figure 2.2.: The definition of the Cauchy stress tensor is based on a local force $\Delta\mathcal{P}$ in the spatial frame. This force is defined on the cross section Δa through the point $\mathbf{p}$ given through the normal vector $\mathbf{n}$. By taking the limit $\Delta a \to 0$ the traction $\mathbf{t}(\mathbf{n})$ is created.

2.3.1. First Piola-Kirchhoff

We formulate the weak form of the equilibrium in the material frame. Therefore we want to find the work conjugated stress measure which allows us to formulate the equilibrium with material frame values only. $\mathbf{l} = \dot{\mathbf{F}}\mathbf{F}^{-1}$ is the spatial velocity gradient and its symmetric part $\mathbf{d} = \frac{1}{2}\left(\mathbf{l} + \mathbf{l}^{T}\right)$ is called rate-of-deformation tensor. $\delta\mathbf{l}$ and $\delta\mathbf{d}$ are the virtual quantities, respectively.

We start with the virtual work rate $\delta\dot{\Pi}_{int}$ which is given by

$$\delta\dot{\Pi}_{int} = \int_{\omega} \boldsymbol{\sigma} : \delta\mathbf{d} \, \mathrm{d}v \tag{2.20}$$

We will see in Eq. (2.28) that the Cauchy stress is symmetric and therfore we can replace the symmetric virtual rate-of-deformation tensor $\delta\mathbf{d}$ with the virtual spatial velocity gradient

$\delta\mathbf{l}$

$$\delta\dot{\Pi}_{int} = \int_\omega \boldsymbol{\sigma} : \delta\mathbf{l} \, dv$$

$$= \int_\Omega J\boldsymbol{\sigma} : \delta\dot{\mathbf{F}}\,\mathbf{F}^{-1} \, dV$$

$$= \int_\Omega J\boldsymbol{\sigma}\,\mathbf{F}^{-T} : \delta\dot{\mathbf{F}} \, dV$$

$$= \int_\Omega \mathbf{P} : \delta\dot{\mathbf{F}} \, dV \tag{2.21}$$

$\mathbf{P}$ is called first Piola-Kirchhoff stress tensor and is given by

$$\mathbf{P} = J\boldsymbol{\sigma}\,\mathbf{F}^{-T} \tag{2.22}$$

2.3.2. Second Piola-Kirchhoff

Taking again

$$\delta\dot{\Pi}_{int} = \int_\omega \boldsymbol{\sigma} : \delta\mathbf{d} \, dv \tag{2.23}$$

and considering, that $\delta\mathbf{d}$ is the push-forward of $\delta\dot{\mathbf{E}}$

$$\delta\dot{\Pi}_{int} = \int_\Omega J\boldsymbol{\sigma} : \mathbf{F}^{-T}\delta\dot{\mathbf{E}}\,\mathbf{F}^{-1} \, dV$$

$$= \int_\Omega J\mathbf{F}^{-1}\boldsymbol{\sigma}\,\mathbf{F}^{-T} : \delta\dot{\mathbf{E}} \, dV$$

$$= \int_\Omega \mathbf{S} : \delta\dot{\mathbf{E}} \, dV \tag{2.24}$$

$\mathbf{S}$ is called second Piola-Kirchhoff stress tensor

$$\mathbf{S} = J\mathbf{F}^{-1}\boldsymbol{\sigma}\,\mathbf{F}^{-T} = \mathbf{F}^{-1}\mathbf{P} \tag{2.25}$$

2.4. Balance law

The local translational equilibrium (for statics only) in the spatial frame is given as

$$\mathrm{div}\,(\boldsymbol{\sigma}) + \mathbf{f}_B = \mathbf{0} \tag{2.26}$$

with $\mathbf{f}_B$ being the body force per spatial volume unit. A suitable derivation can be found in Bonet and Wood [5]. This strong form of the spatial equilibrium is fulfilled for each point in the domain ω.

If we formulate the global rotational equilibrium

$$\int\limits_{\gamma} \mathbf{x} \times \mathbf{t} \, \mathrm{d}a + \int\limits_{\omega} \mathbf{x} \times \mathbf{f}_B \, \mathrm{d}v = \mathbf{0} \tag{2.27}$$

insert $\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}$ and apply Gauss theorem, we end up with

$$\begin{bmatrix} \sigma_{12} - \sigma_{21} \\ \sigma_{23} - \sigma_{32} \\ \sigma_{31} - \sigma_{13} \end{bmatrix} = \mathbf{0} \tag{2.28}$$

which implies the symmetry of the Cauchy stress tensor. The second Piola Kirchhoff $\mathbf{S}$ stress tensor is symmetric too, which is a result of the definition in Eq. (2.25).

2.4.1. Boundary value problem

Now we transfer the spatial translational equilibrium (from Eq. (2.26)) in a suitable formulation to apply the variational principle. As we will deal with multiple bodies for solving the contact problem we already consider the boundary value problem for multiple bodies. At a given time t we can formulate our problem for the "mortar" (2) and "non-mortar" (1) body in the material configuration Ω (visualization is given in Figure 2.3) as

$$\begin{aligned} \mathrm{Div}(\mathbf{P}^{(i)}) + \mathbf{f}_{B,0}^{(i)} &= \mathbf{0} & \text{in} & \quad \Omega^{(i)} \\ \mathbf{P}^{(i)} \, \mathbf{N}^{(i)} &= \mathbf{t}_{\sigma,0}^{(i)} & \text{on} & \quad \Gamma_{\sigma}^{(i)} \cup \Gamma_{C}^{(i)} \\ \mathbf{u}^{(i)} &= \mathbf{u}_u^{(i)} & \text{on} & \quad \Gamma_u^{(i)} \end{aligned}$$

with

$\mathbf{P}^{(i)} := \mathbf{F}^{(i)} \, \mathbf{S}^{(i)}$ being the first Piola-Kirchhoff stress tensor for body (i)

$\mathbf{f}_{B,0}^{(i)} :=$ back transformation of the prescribed body force $\mathbf{f}_B$ in body (i)

$\mathbf{N}^{(i)} :=$ outward normal in material configuration for body (i)

and assume for the boundary sets

$$\partial\Omega^{(i)} = \Gamma_u^{(i)} \cup \Gamma_{\sigma}^{(i)} \cup \Gamma_C^{(i)}$$

$$\Gamma_u^{(i)} \cap \Gamma_{\sigma}^{(i)} = \Gamma_u^{(i)} \cap \Gamma_C^{(i)} = \Gamma_{\sigma}^{(i)} \cap \Gamma_C^{(i)} = \emptyset$$

where $\Gamma_C^{(i)}$ denotes the contact boundary. This back transformation of the real contact surface in the spatial frame on to the material configuration is of course artificial. So is the contact traction in the material configuration. This contact surface and the corresponding contact traction are therefore transformed forward onto the current configuration in Section 5.2. This allows a more natural interpretation of the contact quantities. At this point no contact conditions between the two bodies are taken into account. That means the contact traction is treated the same way as any given load at the Neumann boundary $\Gamma_{\sigma}^{(i)}$. The contact mechanics will be analyzed in Section 5.
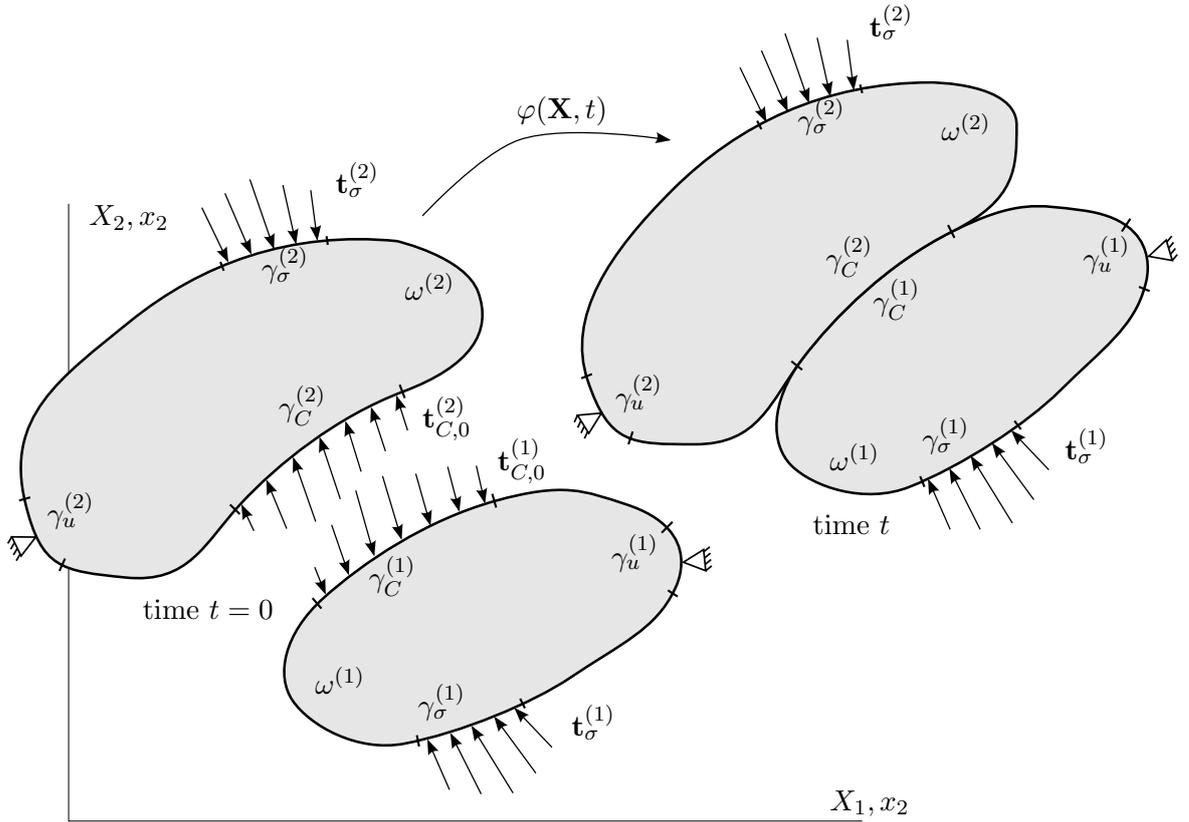
Figure 2.3.: Two bodies in the material configuration (time $t = 0$) and in the spatial config-
uration at time t are shown. Those bodies have Dirichlet boundary conditions
applied on $\Gamma_u^{(i)}$ ($\gamma_u^{(i)}$ in spatial frame) and Neumann conditions on $\Gamma_\sigma^{(i)}$ ($\gamma_\sigma^{(i)}$ in
spatial frame). At the initial frame there is no physical contact, nevertheless
one can do a back transformation of the contact traction $\mathbf{t}_C^{(i)}$ and contact sur-
face $\gamma_C^{(i)}$ onto the material configuration. The corresponding tractions $\mathbf{t}_{C,0}^{(i)}$ are
therefore artificial surface force densities.

2.4.2. Weak formulation

To use the finite element method we have to apply the variational principle to obtain the
weak formulation. This procedure is well described in various books, e.g. Zienkiewicz and
Taylor [46, 47], Bathe [2].

The stationary condition for the multi (two) body system reads as

$$\delta\Pi\left(\mathbf{u}, \delta\mathbf{u}\right) = \sum_{i=1}^{2} \delta\Pi^{(i)}\left(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}\right) = 0 \tag{2.29}$$

We can write the weak formulation for each body as

$$
\begin{aligned}
\delta\Pi^{(i)}\left(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}\right) &= \int_{\Omega^{(i)}} \left(\mathbf{S}^{(i)} : \delta\mathbf{E}^{(i)} - \mathbf{f}_{B,0}^{(i)} \cdot \delta\mathbf{u}^{(i)}\right) \mathrm{d}\Omega - \int_{\Gamma_\sigma^{(i)} \cup \Gamma_C^{(i)}} \mathbf{t}_{,0}^{(i)} \cdot \delta\mathbf{u}^{(i)} \mathrm{d}\Gamma \\
&= \underbrace{\int_{\Omega^{(i)}} \mathbf{F}^{(i)} \cdot \mathbf{S}^{(i)} : \delta\mathbf{F}^{(i)} \mathrm{d}\Omega}_{\delta\Pi_{int}^{(i)}} \underbrace{- \int_{\Omega^{(i)}} \mathbf{f}_{B,0}^{(i)} \cdot \delta\mathbf{u}^{(i)} \mathrm{d}\Omega - \int_{\Gamma_\sigma^{(i)}} \mathbf{t}_{\sigma,0}^{(i)} \cdot \delta\mathbf{u}^{(i)} \mathrm{d}\Gamma}_{\delta\Pi_{ext}^{(i)}} \\
&\quad \underbrace{- \int_{\Gamma_C^{(i)}} \mathbf{t}_{C,0}^{(i)} \cdot \delta\mathbf{u}^{(i)} \mathrm{d}\Gamma}_{\delta\Pi_C^{(i)}}
\end{aligned}
\tag{2.30}
$$

where we still have not made any assumptions on the contact conditions. This formulation is generally applicable for multi body systems. We have already split up the integral over the Neumann boundary in the contact boundary $\Gamma_C^{(i)}$ and the "classical" Neumann boundary $\Gamma_\sigma^{(i)}$. In Section 5.2 we will introduce the conditions on the contact tractions $\mathbf{t}_{C,0}^{(i)}$ as they are not independent for two bodies in contact.

2.5. Constitutive law

Detailed description of material theory can be found, e.g. in Truesdell and Noll [34], Marsden and Hughes [25]. Throughout this work we will focus on contact mechanics. Therefore we choose two rather simple elastic materials. One for finite deformations, the compressible neo-Hookean material and one for small deformations, the St. Venant-Kirchhoff material.

To describe the material parameters of those materials we use the Lamé parameters inside the energy strain function ψ (see e.g. Truesdell and Noll [34]). One can convert the lame parameters λ, μ into the more common material parameters E (Young's modulus) and ν (Poisson's ratio) with

$$
E = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu}
\tag{2.31}
$$

$$
\nu = \frac{\lambda}{2(\lambda + \mu)}
\tag{2.32}
$$

and $\mu = G$ being the shear modulus.

2.5.1. St. Venant-Kirchhoff material

The strain energy function is given as

$$
\psi(\mathbf{F}(\mathbf{X}), \mathbf{X}) = \frac{1}{2}\lambda \left(\mathrm{tr}(\mathbf{E})\right)^2 + \mu\mathbf{E} : \mathbf{E}
\tag{2.33}
$$

where λ and μ are the Lamé material coefficients. The second Piola-Kirchhoff stress then reads

$$\mathbf{S} = \frac{\partial \psi}{\partial \frac{1}{2}\mathbf{C}}$$

$$\mathbf{S} = \lambda \left(\mathrm{tr}(\mathbf{E}) \right) + 2\mu \mathbf{E} \tag{2.34}$$

We will solve the non-linear system by application of the Newton-Raphson procedure (see Section 6.1). Therefore we need the stress increment for a given state $\mathbf{S}, \mathbf{E}$

$$\Delta \mathbf{S} = \frac{\partial \mathbf{S}}{\partial \frac{1}{2}\mathbf{C}} : \Delta \mathbf{E} = \mathbb{C} : \Delta \mathbf{E} \tag{2.35}$$

with $\mathbb{C}$ being the elasticity tensor. It can be written in index notation as

$$\mathbb{C}_{ijkl} = \lambda \, \delta_{ij}\delta_{kl} + \mu \, 2\,\frac{1}{2}\left[\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk} \right] \tag{2.36}$$

which includes the fourth order unit tensor

$$\mathbb{I}_{ijkl} = \frac{1}{2}\left[\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk} \right] \tag{2.37}$$

where δ_{ij} is the Kronecker delta (see Klingbeil [21]).

2.5.2. Compressible Neo-Hookean material

To define the general strain energy function we need the three invariants of the right Cauchy Green deformation tensor $\mathbf{C}$

$$\mathrm{I}_C = \mathrm{tr}\left(\mathbf{C}\right) = \mathbf{C} : \mathbb{I} \tag{2.38}$$

$$\mathrm{II}_C = \mathrm{tr}\left(\mathbf{C}\right)\mathbf{C} = \mathbf{C} : \mathbf{C} \tag{2.39}$$

$$\mathrm{III}_C = \det\left(\mathbf{C}\right) = J^2 \tag{2.40}$$

The strain energy function is then defined as

$$\psi(\mathbf{F}(\mathbf{X}), \mathbf{X}) = \frac{\mu}{2}\left(\mathrm{I}_C - 3 \right) - \mu \ln\left(\mathrm{III}_C \right) + \frac{\lambda}{2}\left[\ln\left(\mathrm{III}_C \right) \right]^2 \tag{2.41}$$

where λ and μ are once again the Lamé material coefficients. Based on this elastic potential one can determine the stress tensor

$$\mathbf{S} = \frac{\partial \psi}{\partial \frac{1}{2}\mathbf{C}}$$

$$\mathbf{S} = \mu \left(\mathbf{I} - \mathbf{C}^{-1} \right) + \lambda \ln\left(\mathrm{I}_C \right) \mathbf{C}^{-1} \tag{2.42}$$

The fourth order elasticity tensor $\mathbb{C}_{ijkl}$ can be calculated with

$$\mathbb{C}_{ijkl} = \lambda \left(\mathbf{C}^{-1} \right)_{ij} \left(\mathbf{C}^{-1} \right)_{kl} + 2\left[\mu - \lambda \ln\left(\mathrm{I}_C \right) \right] \mathcal{I}_{ijkl} \tag{2.43}$$

which includes the fourth order tensor $\mathcal{I}_{ijkl}$

$$\mathcal{I}_{IJKL} = \frac{1}{2}\left\{ \left(\mathbf{C}^{-1} \right)_{ik} \left(\mathbf{C}^{-1} \right)_{jl} + \left(\mathbf{C}^{-1} \right)_{il} \left(\mathbf{C}^{-1} \right)_{jk} \right\} \tag{2.44}$$

3. Contact kinematics

Throughout this chapter we will cover the two dimensional contact kinematics for the two body contact problem. If there are more than two bodies in contact the following is valid for one chosen contact pairing.

We will start with a rather general description of the contact kinematics. That means, in a first step, we do not focus on the influence of discretization. This is similar to the approach of Willner [36] or the work of Konyukhov and Schweizerhof [22] on the covariant description of contact kinematics.

We will consider discretization in Chapter 4. Here we will try to keep as much information of the continuous contact surface as possible.

3.1. Nearest point projection

The procedure of finding a corresponding contact point $\mathbf{x}^{(2)}$ on the mortar side for a given point $\overline{\mathbf{x}}^{(1)}$ on the non mortar side is called nearest point projection (NPP). This means all objects marked by $\overline{\square}$ depend on a non-mortar point $\mathbf{x}^{(1)}$ by application of the NPP.
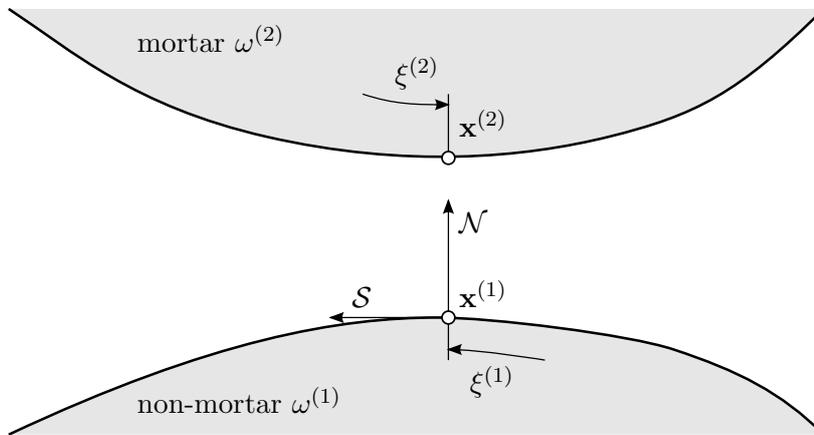
Figure 3.1.: We define the Frenet frame for an arbitrary point $\mathbf{x}^{(1)}$. At first we do not differ if $\mathcal{S}$ is seen as tangential to $\gamma_C^{(1)}$ or $\gamma_C^{(2)}$ as they coincide (only have different directions). The positive normal vector $\mathcal{N}$ is defined to point from the non-mortar $\omega^{(1)}$ to the mortar domain $\omega^{(2)}$.

We define and show the main objects of the contact kinematics, the Frenet frame (see Sokolnikoff [33] or Gray [12] for further explanations), in Figure 3.1. This is the parametrization

of the non-mortar surface $\gamma^{(1)}$ with the parameter $\xi^{(1)}$ and $\gamma^{(2)}$ with the mortar surface $\xi^{(2)}$.

The tangential vector $\mathcal{S}$, which is defined on the non-mortar surface, can be calculated via

$$\mathcal{S} = \frac{\partial \mathbf{x}^{(1)}}{\partial \xi^{(1)}} \left\| \frac{\partial \mathbf{x}^{(1)}}{\partial \xi^{(1)}} \right\|^{-1} \qquad \text{or} \qquad \mathcal{S} = -\frac{\partial \mathbf{x}^{(2)}}{\partial \xi^{(2)}} \left\| \frac{\partial \mathbf{x}^{(2)}}{\partial \xi^{(2)}} \right\|^{-1} \tag{3.1}$$

or in an even more general way if discretization is considered (see Section 4.2). We try to develop the contact kinematics fully independently (as synthetically as possible) from the choice of $\mathcal{S}$. This allows us later in the discretization to compare two different methods of defining the Frenet frame.

The projection itself is done through minimization of the distance d of a fixed point $\mathbf{x}^{(1)}$ on the non-mortar side and an arbitrary point $\mathbf{x}^{(2)}$ on the mortar side identified with the convective coordinate $\xi^{(2)}$. The situation is visualized in Figure 3.2 and reflects the physical "meaning" of contact.
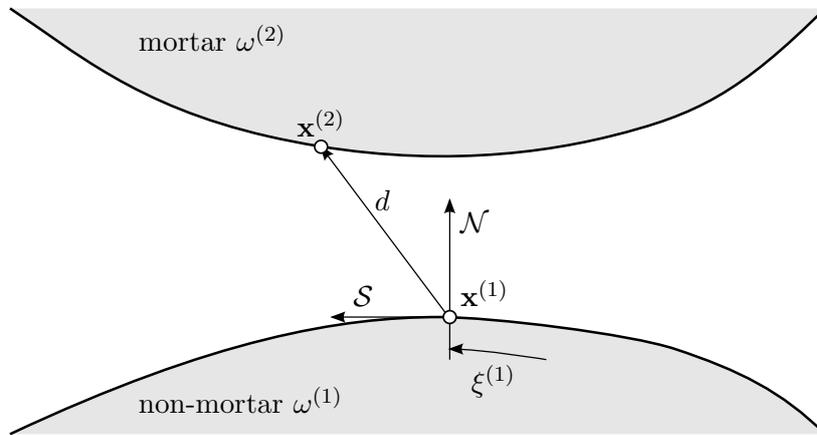
Figure 3.2.: Two arbitrary points $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ on the surface of the contact body are shown. The difference vector $\mathbf{d}$ has the length d.

With this setup we obtain the distance

$$d = \|\mathbf{x}^{(2)}(\xi^{(2)}) - \mathbf{x}^{(1)}\| \to \text{MIN}$$

and define the nearest point projection.

> **Definition:** The nearest point projection is defined as minimization of the distance d of two points $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ on the corresponding surfaces of the contact bodies. We keep the parameter $\xi^{(1)}$ fixed and minimize the distance with respect to the convective coordinate $\xi^{(2)}$.

As we will see later, this procedure is advantageous as the numerical algorithm is based on a surface quadrature on the non-mortar side, where the parameter of the integration point is fixed too. This is true for our concentrated integration scheme used and presented in Section 6.5.

In mathematical sense this means

$$\frac{\partial}{\partial \overline{\xi}^{(2)}} d(\overline{\xi}^{(2)}, t) = 0$$

$$\frac{\overline{\mathbf{x}}^{(2)}(\overline{\xi}^{(2)}) - \mathbf{x}^{(1)}}{\|\overline{\mathbf{x}}^{(2)}(\overline{\xi}^{(2)}) - \mathbf{x}^{(1)}\|} \cdot \frac{\partial \overline{\mathbf{x}}^{(2)}}{\partial \overline{\xi}^{(2)}} = 0$$

$$- \left(\overline{\mathbf{x}}^{(2)}(\overline{\xi}^{(2)}) - \mathbf{x}^{(1)} \right) \cdot \mathcal{S} = 0 \tag{3.2}$$

what is illustrated in Figure 3.3.
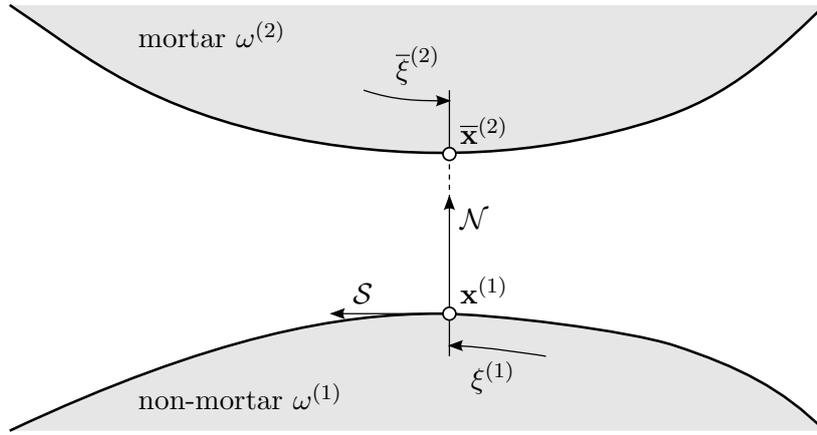
Figure 3.3.: Two bodies in contact with continuous surfaces without discretization. As a result of the NPP we see that the projection point $\overline{\mathbf{x}}^{(2)}$ lies on the section of the normal vector $\mathcal{N}$ with the mortar surface. To show the properties we have to separate the two bodies being in contact at point $\mathbf{x}^{(1)}$. For physical contact situation the points $\overline{\mathbf{x}}^{(2)}$ and $\mathbf{x}^{(1)}$ would coincide.

3.1.1. Solvability of the nearest point projection

The solution of the nearest point projection might not be unique. This is the case if the Frenet frame is defined on the mortar side and the mortar surface is non convex. Until know we did not discuss how the Frenet frame $[\mathcal{N}, \mathcal{S}, \mathbf{e}_3]$ is constructed. It is of course possible (and common) to use the mortar surface $\gamma_C^{(2)}$ for this construction. If we do not apply any averaging this choice is even advantageous. In Figure 3.4 the situation is shown.

This issue is a rather small problem for the discrete algorithm. On the other hand a few other problems due to discretization might occur and might lead to oscillations of the active contact set. Those issues are discussed in Section 4.3.
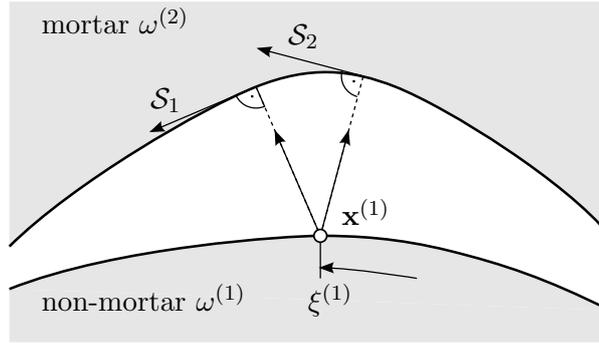
Figure 3.4.: For non convexity of the mortar surface a non unique solution of the nearest point projection is possible. This situation might occur if one defines the tangential vector $\mathcal{S}$ on the mortar side. Then the projection of a non-mortar point $\mathbf{x}^{(1)}$ onto the mortar side might not be unique. See Section 4.3 for further discussions and solutions.

3.2. Normal contact

Between the non-mortar surface point and a possible contact point on the mortar side we define the following relation

$$\overline{\mathbf{x}}^{(2)}(\overline{\xi}^{(2)}) = \mathbf{x}^{(1)}(\xi^{(1)}) + g_N \mathcal{N} \tag{3.3}$$

$$g_N = \left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \mathcal{N} \tag{3.4}$$

with g_N being the gap function (see Figure 3.5). Here we can already note a very important property of an active contact situation. For a "real" (physical) contact situation the gap function g_N has to be zero. We will reflect those characteristics of physical contact in Section 5.1.
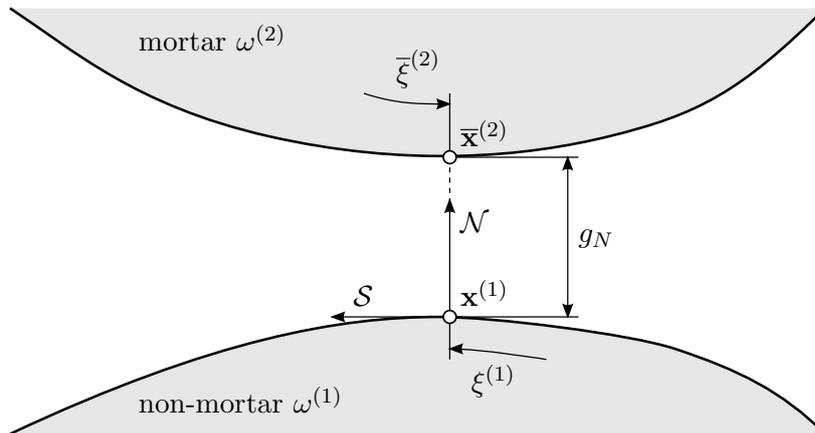
Figure 3.5.: We define the scalar gap function g_N as pointed direction from the non-mortar point $\mathbf{x}^{(1)}$ to the projected mortar point $\overline{\mathbf{x}}^{(2)}$.

Due to the discretization of the contact surface different tangential vectors for the mortar

and the non-mortar side may occur. Depending on the used algorithm we will therefore use different vectors $\mathcal{S}$ and normal vectors $\mathcal{N}$. This will be covered in Section 4. But whatever normal or tangential vector we choose the relation Eq. (3.3) remains valid.

3.2.1. Variation of gap function

As the solution algorithm is based on a variational calculus we need the variation of the gap function. Therefore we vary Eq. (3.3) and obtain

$$\delta\overline{\mathbf{u}}^{(2)} + \overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} = \delta\mathbf{u}^{(1)} + \delta g_N\,\mathcal{N} + g_N\,\delta\mathcal{N} \tag{3.5}$$

with $\mathbf{a}^{(i)}$ being defined throughout the thesis as non normalized tangent vector $\mathbf{a}^{(i)} = \frac{\partial\mathbf{x}^{(i)}}{\partial\xi^{(i)}}$.

> **Remark:** We distinguish between the tangential vector $\mathcal{S}$ from the defined Frenet frame and the derivative of $\overline{\mathbf{x}}^{(2)}$ with respect to the convective coordinate $\xi^{(2)}$. Although we know that for the continuous case $\mathcal{S}$ is parallel to $-\overline{\mathbf{a}}^{(2)}$.

At this point we should note that there is no inner variation of $\mathbf{u}^{(1)}$ because $\delta\xi^{(1)} = 0$. This is only true for our chosen integration scheme with fixed integration limits and integration points. This fact is further discussed in Section 6.5.

Multiplying Eq. (3.5) with $\mathcal{N}$ gives the variation of the gap function

$$\delta g_N = \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right)\cdot\mathcal{N} + \mathcal{N}\cdot\overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} \tag{3.6}$$

$\delta\xi^{(2)}$ is part of the derivation regarding tangential contact and is shown in Section 3.3.

3.3. Tangential contact

Multiplying Eq. (3.5) with the tangential vector $\mathcal{S}$ yields

$$\mathcal{S}\cdot\overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} = -\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right)\cdot\mathcal{S} + g_N\,\delta\mathcal{N}\cdot\mathcal{S} + \delta g_N\,\underbrace{\mathcal{N}\cdot\mathcal{S}}_{=\,0}$$

$$\mathcal{S}\cdot\overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} = -\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right)\cdot\mathcal{S} + g_N\,\delta\mathcal{N}\cdot\mathcal{S} \tag{3.7}$$

One should keep in mind that due to $\mathcal{N}\cdot\mathcal{S} = 0$, $\delta\mathcal{N}\cdot\mathcal{S} = -\mathcal{N}\cdot\delta\mathcal{S}$. We will show in Section 4.1 and Section 4.2 that this equation leads to the calculation of $\delta\xi^{(2)}$. The calculation of $\delta\xi^{(2)}$ has a rather strong dependency on the chosen discretization of the normal and tangential vector field.

The virtual contact work in tangential direction for the stick case depends on δg_T (see Eq. (5.22)). We use the nearest point projection condition from Eq. (3.2)

$$\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right)\cdot\mathcal{S} = 0 \tag{3.8}$$

and apply the δ-process

$$0 = \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S} + \mathcal{S} \cdot \overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} + \left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \delta\mathcal{S} \tag{3.9}$$

$$0 = \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S} + \mathcal{S} \cdot \overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} + g_N\mathcal{N} \cdot \delta\mathcal{S} \tag{3.10}$$

$$-\mathcal{S} \cdot \overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} = \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S} - g_N\delta\mathcal{N} \cdot \mathcal{S} \tag{3.11}$$

The same result can of course be obtained if one considers Eq. (3.7)

$$\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S} - g_N \, \delta\mathcal{N} \cdot \mathcal{S} = -\mathcal{S} \cdot \overline{\mathbf{a}}^{(2)} \, \delta\xi^{(2)} \tag{3.12}$$

We will need $\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S}$ for our variational formulation (see Eq. (5.22)) and we call this term δg_T. If we assume that $g_N = 0$ we get

$$\delta g_T = \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \mathcal{S} \tag{3.13}$$

$$\delta g_T = -\mathcal{S} \cdot \overline{\mathbf{a}}^{(2)} \, \delta\xi^{(2)} \tag{3.14}$$

This relation can also be found in Laursen [24], Wriggers [40] or Willner [36]. The inner product $-\mathcal{S} \cdot \overline{\mathbf{a}}^{(2)}$ is equal to $\|\overline{\mathbf{a}}^{(2)}\|$ for the continuous case. As already mentioned we try to keep as much information as possible. Keeping this product and not substituting with $\|\overline{\mathbf{a}}^{(2)}\|$ will lead to a somehow "artificial" asymmetry mentioned in the work of Konyukhov and Schweizerhof [22] and others. On the other hand keeping information should enhance the quality of results.
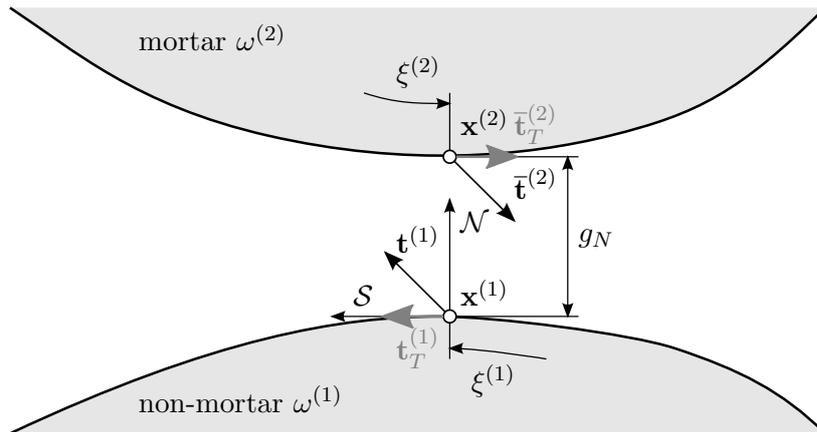
Figure 3.6.: If there is a non zero gap g_N between the non-mortar point $\mathbf{x}^{(1)}$ and the projection point $\overline{\mathbf{x}}^{(2)}$ the traction $\mathbf{t}^{(1)}$ and $\overline{\mathbf{t}}^{(2)}$ will not coincide. This pair of forces $\mathbf{t}_T^{(i)}$ with normal distance g_N would result in a moment which physically does not exist.

For the continuous situation $g_N = 0$ is true as it is our contact condition. This is not the only argument for setting $g_N = 0$. We neglect the g_N term because our equilibrium in tangential direction is only satisfied for $g_N = 0$. Else we would have a resulting moment which is in fact not the case. This has been also discussed in Yang et al. [44]. For a discrete arbitrarily curved surface $g_N \neq 0$, but the equilibrium has to be satisfied and therefore we are not removing information here but incorporate an important physical fact. The whole problem is presented in Figure 3.6.

3.3.1. Relative velocity $\mathbf{v}_T$

As stated in Wriggers [40] and Laursen [24] let us consider the distance vector between the mortar and non-mortar surface point $\mathbf{g} = \overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}$. Of course this has to be zero for perfect contact and so has $g_T = \mathbf{g} \cdot \mathcal{S} = 0$.

We will construct an objective velocity measure by applying the Lie derivative $\mathcal{L}_\Phi$ onto g_T as shown in Section 2.1.3.1. Of course the Lie derivative $\mathcal{L}_\Phi \{g_T\}$ has to be zero too.

$$
\begin{aligned}
0 &= \mathcal{L}_\Phi \left\{ \left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \mathcal{S} \right\} \\
&= \Phi_* \left[\frac{\mathrm{d}}{\mathrm{d}t} \left\{ \Phi_*^{-1} \left(\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \mathcal{S} \right] \right\} \right] \\
&= \Phi_* \left[\frac{\mathrm{d}}{\mathrm{d}t} \left\{ \left(\overline{\mathbf{X}}^{(2)} - \mathbf{X}^{(1)} \right) \cdot \Phi_*^{-1} (\mathcal{S}) \right\} \right] \\
&= \Phi_* \left[\left(\dot{\overline{\mathbf{X}}}^{(2)} - \dot{\mathbf{X}}^{(1)} + \overline{\mathbf{X}}^{(2)}_{,\xi^{(2)}} \dot{\xi}^{(2)} \right) \cdot \Phi_*^{-1} [\mathcal{S}] + \left(\overline{\mathbf{X}}^{(2)} - \mathbf{X}^{(1)} \right) \cdot \frac{\mathrm{d}}{\mathrm{d}t} \left\{ \Phi_*^{-1} [\mathcal{S}] \right\} \right] \\
&= \left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)} \right) \cdot \mathcal{S} + \overline{\mathbf{a}}^{(2)} \dot{\xi}^{(2)} \cdot \mathcal{S} + \underbrace{\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)} \right)}_{g_N \mathcal{N} = 0} \cdot \Phi_* \left[\frac{\mathrm{d}}{\mathrm{d}t} \left\{ \Phi_*^{-1} [\mathcal{S}] \right\} \right]
\end{aligned}
$$

$$
\left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)} \right) \cdot \mathcal{S} = -\overline{\mathbf{a}}^{(2)} \dot{\xi}^{(2)} \cdot \mathcal{S}
$$

This allows us to define the *convective* velocity

$$
v_T = \left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)} \right) \cdot \mathcal{S} \tag{3.15}
$$

$$
v_T = -\overline{\mathbf{a}}^{(2)} \dot{\xi}^{(2)} \cdot \mathcal{S} \tag{3.16}
$$

This formulation is suitable for the continuous case, where the gap function $g_N = 0$ for all points being in contact.

3.3.1.1. Objectivity of v_T

In the following we show that the velocity from Eq. (3.15) is not objective if we insert the geometric interpolation for the discrete case. This is a problem for finite deformations and large slidings if we use v_T as criteria for detection of stick or slip respectively. We define the transformation according to Section 2.1.3

$$
\mathbf{x}^* = \mathbf{c}(t) + \mathbf{Q}(t) \cdot \mathbf{x} \tag{3.17}
$$

An objective velocity measure v_T must fulfill

$$
\mathbf{v}_T^* = \mathbf{Q}(t) \, \mathbf{v}_T \tag{3.18}
$$

$$
v_T^* = v_T \tag{3.19}
$$

25

3. Contact kinematics

We insert the transformation from Eq. (2.10) into our definition of the velocity from Eq. (3.15).

$$v_T^* = -\left(\mathbf{QS}\right)^T \left(\mathbf{Q}\dot{\overline{\mathbf{x}}}^{(2)} + \dot{\mathbf{Q}}\overline{\mathbf{x}}^{(2)} - \mathbf{Q}\dot{\mathbf{x}}^{(1)} - \dot{\mathbf{Q}}\mathbf{x}^{(1)}\right) \tag{3.20}$$

$$v_T^* = \underbrace{-\mathcal{S}\cdot\left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)}\right)}_{v_T} - \mathcal{S}\cdot\left[\mathbf{Q}^T\dot{\mathbf{Q}}\underbrace{\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right)}_{g_N\mathcal{N}=0}\right] \tag{3.21}$$

The second term on the right hand side is zero only if the gap function g_N is zero. This is true for the continuous case but not for discretized and arbitrarily curved surfaces.

To construct an objective velocity measure we follow the approach presented in the paper of Yang et al. [44]. Let us reconsider the definition of the scalar velocity v_T. Next we extend the velocity with the zero vector $\dot{\mathbf{g}}$. For perfect contact the gap vector $\mathbf{g}$ has to be zero and so has $\dot{\mathbf{g}} = \mathbf{0}$.

$$v_T = -\left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)}\right)\cdot\mathcal{S}$$

$$v_T = -\left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)} - \underbrace{\dot{\mathbf{g}}}_{=\mathbf{0}}\right)\cdot\mathcal{S} \tag{3.22}$$

Now we to introduce the shape functions $\mathbf{N}_\alpha^{(i)}$ on the contact interface which interpolate the surface node coordinates $\hat{\mathbf{x}}_\alpha^{(i)}$

$$\overline{\mathbf{x}}^{(2)} = \overline{\mathbf{N}}_\alpha^{(2)}\hat{\mathbf{x}}_\alpha^{(2)} \qquad\qquad \mathbf{x}^{(1)} = \mathbf{N}_\alpha^{(1)}\hat{\mathbf{x}}_\alpha^{(1)} \tag{3.23}$$

$$\dot{\overline{\mathbf{x}}}^{(2)} = \overline{\mathbf{N}}_\alpha^{(2)}\dot{\hat{\mathbf{x}}}_\alpha^{(2)} \qquad\qquad \dot{\mathbf{x}}^{(1)} = \mathbf{N}_\alpha^{(1)}\dot{\hat{\mathbf{x}}}_\alpha^{(1)} \tag{3.24}$$

This allows us to calculate $\dot{\mathbf{g}} = \overline{\mathbf{N}}_\alpha^{(2)}\dot{\hat{\mathbf{x}}}_\alpha^{(2)} + \dot{\overline{\mathbf{N}}}_\alpha^{(2)}\hat{\mathbf{x}}_\alpha^{(2)} - \mathbf{N}_\alpha^{(1)}\dot{\hat{\mathbf{x}}}_\alpha^{(1)} - \dot{\mathbf{N}}_\alpha^{(1)}\hat{\mathbf{x}}_\alpha^{(1)}$ which can be substituted in Eq. (3.22) to obtain

$$v_T = -\mathcal{S}\cdot\left[\overline{\mathbf{N}}_\alpha^{(2)}\dot{\hat{\mathbf{x}}}_\alpha^{(2)} - \mathbf{N}_\alpha^{(1)}\dot{\hat{\mathbf{x}}}_\alpha^{(1)} - \left(\overline{\mathbf{N}}_\alpha^{(2)}\dot{\hat{\mathbf{x}}}_\alpha^{(2)} + \dot{\overline{\mathbf{N}}}_\alpha^{(2)}\hat{\mathbf{x}}_\alpha^{(2)} - \mathbf{N}_\alpha^{(1)}\dot{\hat{\mathbf{x}}}_\alpha^{(1)} - \dot{\mathbf{N}}_\alpha^{(1)}\hat{\mathbf{x}}_\alpha^{(1)}\right)\right] \tag{3.25}$$

$$v_T = \mathcal{S}\cdot\left(\dot{\overline{\mathbf{N}}}_\alpha^{(2)}\hat{\mathbf{x}}_\alpha^{(2)} - \dot{\mathbf{N}}_\alpha^{(1)}\hat{\mathbf{x}}_\alpha^{(1)}\right) \tag{3.26}$$

Eq. (3.26) represents an objective velocity measure even for the discrete case. For our concentrated integration scheme $\dot{\mathbf{N}}_\alpha^{(1)}$ vanishes, as $\dot{\xi}^{(1)} = 0$. It is notable that we do not need the time derivation of the nodal coordinates but the time derivation of the shape functions $\dot{\overline{\mathbf{N}}}_\alpha$. We further apply an implicit backward Euler scheme

$$\frac{\mathrm{d}(\cdot)}{\mathrm{d}t} \approx \frac{(\cdot)(t^{n+1}) - (\cdot)(t^n)}{\Delta t} = \frac{\Delta^t(\cdot)}{\Delta t} \tag{3.27}$$

where t^n is the time of the last converged state of equilibrium and we obtain

$$v_T = \mathcal{S}\cdot\frac{\Delta^t\overline{\mathbf{N}}_\alpha^{(2)}}{\Delta t}\hat{\mathbf{x}}_\alpha^{(2)} = \mathcal{S}\cdot\frac{\Delta^t\mathbf{g}_T}{\Delta t} \tag{3.28}$$
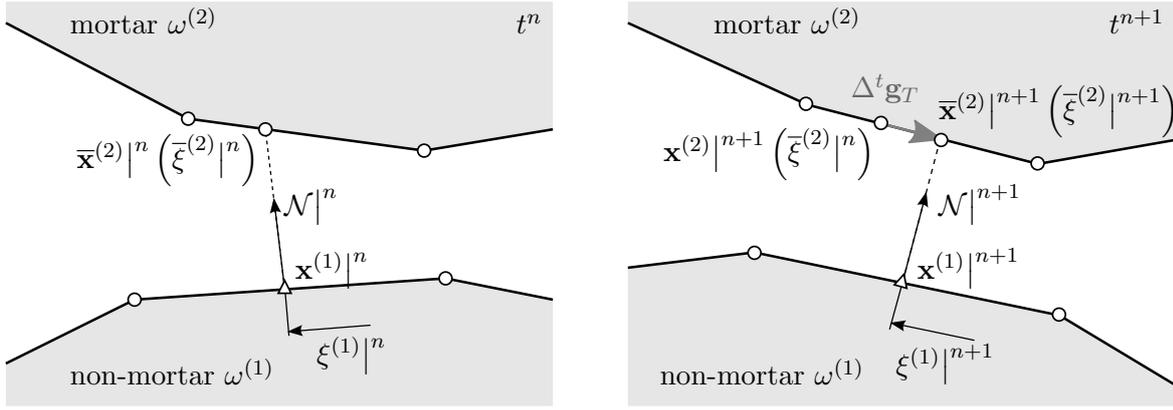
Figure 3.7.: The relative tangential slip can be seen as difference vector between the actual nearest point $\overline{\mathbf{x}}^{(2)}|^{n+1}\left(\overline{\xi}^{(2)}|^{n+1}\right) = \mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}|^{n+1}\right)\hat{\mathbf{x}}_\alpha^{(2)}|^{n+1}$ and the point $\mathbf{x}^{(2)}|^{n+1}\left(\overline{\xi}^{(2)}|^{n}\right) = \mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}|^{n}\right)\hat{\mathbf{x}}_\alpha^{(2)}|^{n+1}$. The latter point can be interpreted as being marked on the surface of the previous converged time step (green dot) and "transformed" into the actual time step.

$\Delta^t\overline{\mathbf{N}}_\alpha^{(2)}$ can be calculated with $\mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}(t^{n+1})\right) - \mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}(t^{n})\right)$. That means we have to store the convective coordinate $\overline{\xi}^{(2)}(t^n)$ of the last converged time step. For simplification we will write in the following $\square(t^n) = \square|^n$. This result is equal to the method presented in Tur et al. [35]. The presented derivation unifies the suggestion of Tur et al. [35] with the procedure presented by Yang et al. [44].

3.3.2. Relative tangential slip increment $\Delta^t g_T$

With the relative tangential velocity v_T we can now calculate a relative tangential slip increment $\Delta^t g_T = v_T \Delta t$

$$\begin{aligned} \Delta^t g_T &= \mathcal{S} \cdot \Delta^t\overline{\mathbf{N}}_\alpha^{(2)}\hat{\mathbf{x}}_\alpha^{(2)}|^{n+1} \\ &= \mathcal{S} \cdot \left\{\mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}|^{n+1}\right) - \mathbf{N}_\alpha^{(2)}\left(\overline{\xi}^{(2)}|^{n}\right)\right\}\hat{\mathbf{x}}_\alpha^{(2)}|^{n+1} \end{aligned} \tag{3.29}$$

We present a graphical interpretation of this objective velocity in Figure 3.7.

> **Remark:** It is import to distinguish the time increment $\Delta^t\square$ which is always based on the last equilibrium state and the increment $\Delta\square$ resulting from the linearization process.

3.4. Jacobian of contact surface

We have to evaluate the integral on the non-mortar side and need therefore the Jacobian determinant $J^{(1)}$.

$$J^{(1)} = \|\mathbf{x}_{,\xi^{(1)}}^{(1)}\| = \left(\mathbf{x}_{,\xi^{(1)}}^{(1)} \cdot \mathbf{x}_{,\xi^{(1)}}^{(1)}\right)^{\frac{1}{2}} = \left(\mathbf{a}^{(1)} \cdot \mathbf{a}^{(1)}\right)^{\frac{1}{2}} \tag{3.30}$$

With variational methods we obtain

$$\delta J^{(1)} = \frac{\mathbf{a}^{(1)}}{J^{(1)}} \cdot \delta \mathbf{u}^{(1)}_{,\xi^{(1)}} \tag{3.31}$$

4. Discrete contact kinematics

In Chapter 3 we discussed the continuous and rather synthetic contact kinematics. For a practical use we need to handle discrete contact surfaces with discontinuities.

In the following we will incorporate the difficulties arising through discretization. We will also provide different types of implementations for the discrete situation with respect to the main kinematic property - the normal field.

We will distinguish between the non-continuous normal field defined on the mortar side (see Section 4.1) and the averaged normal field defined on the non-mortar side (see Section 4.2). The idea of averaging was first presented in a work by Yang et al. [44] and got very popular by other authors implementing the mortar method later on.

4.1. Non-continuous mortar side normal

This idea is based on the work of Fischer and Wriggers [6]. Here the normal vector is defined on the mortar domain $\omega^{(2)}$. This is advantageous because the variation of the gap function g_N simplifies as we will see later on. The situation is sketched in Figure 4.1.
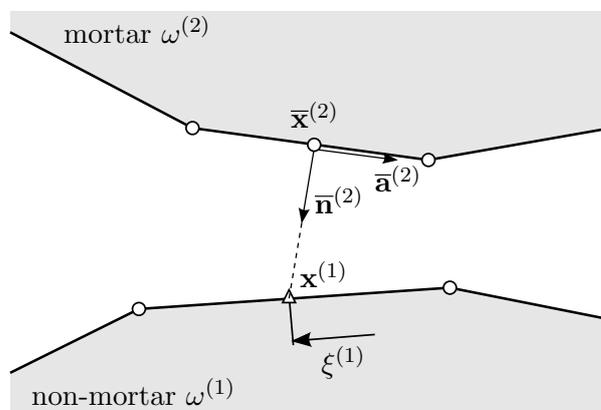
Figure 4.1.: The normal vector $\overline{\mathbf{n}}^{(2)}$ is defined on the mortar side. It is perpendicular to the corresponding edge. For linear shape functions (but also for higher order Lagrangian shape functions) there is a kink (C^1 discontinuity) on the edge nodes. Therefore the normal vector field has a singularity on these edge nodes.

We define a non unit tangent vector on the mortar side

$$\overline{\mathbf{a}}^{(2)} = \overline{\mathbf{x}}_{,\xi}^{(2)}$$

4. Discrete contact kinematics

Now we can construct the Frenet frame with

$$\overline{\mathbf{n}}^{(2)} = \frac{\overline{\mathbf{a}}^{(2)}}{\|\overline{\mathbf{a}}^{(2)}\|} \times \mathbf{e}_3 = \overline{\mathbf{s}}^{(2)} \times \mathbf{e}_3 \tag{4.1}$$

With this definitions we can set $\mathcal{N} = -\overline{\mathbf{n}}^{(2)}$ and $\mathcal{S} = -\overline{\mathbf{s}}^{(2)}$. We are able to use the kinematic relations from Chapter 3. The main relation for the gap function, Eq. (3.3), reads

$$\overline{\mathbf{x}}^{(2)} = \mathbf{x}^{(1)} - {}^{M}g_N \overline{\mathbf{n}}^{(2)} \tag{4.2}$$

In the following we will denote all values related to the mortar side non-continuous normal field with ${}^{M}\square$.

4.1.1. Nearest point projection

Before we proceed we define two often used quantities throughout the whole work.

$$\overline{\alpha}^{(2)} = \left\|\overline{\mathbf{a}}^{(2)}\right\| = \sqrt{\overline{\mathbf{a}}^{(2)} \cdot \overline{\mathbf{a}}^{(2)}} \tag{4.3}$$

$$\overline{\beta}^{(2)} = \overline{\mathbf{x}}^{(2)}_{,\xi\xi} \cdot \overline{\mathbf{n}}^{(2)} \tag{4.4}$$

> **Remark:** It should be noted that $\overline{\mathbf{x}}^{(2)}_{,\xi\xi} = \mathbf{0}$ and thus $\overline{\beta}^{(2)} = 0$ for linear shape functions.

Multiplying Eq. (4.2) with the tangent $\overline{\mathbf{a}}^{(2)}$ (it is not necessary to use the normalized tangent $\overline{\mathbf{s}}^{(2)}$) and keeping in mind that $\overline{\mathbf{n}}^{(2)} \cdot \overline{\mathbf{a}}^{(2)} \equiv 0$ we get

$$\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \overline{\mathbf{a}}^{(2)} = 0 \tag{4.5}$$

See Section 7.2.2.1 for details on implementation.

4.1.2. Normal gap

Multiplying Eq. (4.2) with the normal $\overline{\mathbf{n}}^{(2)}$ (see also Eq. (3.4)) yields

$${}^{M}g_N = -\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \overline{\mathbf{n}}^{(2)} \tag{4.6}$$

as gap function.

Let us reconsider Eq. (3.5). The last term on the right hand side vanishes for the mortar side normal field, because of the orthogonality between $\overline{\mathbf{n}}^{(2)}$ and $\overline{\mathbf{a}}^{(2)}$. This is the big advantage of this method and simplifies the variation and the linearization of all quantities significantly.

$${}^{M}\delta g_N = -\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \overline{\mathbf{n}}^{(2)} \tag{4.7}$$

4.1.3. Tangential slip

Now we specialize the Eq. (3.14) for the mortar side normal field and get

$$^M\delta g_T = \overline{\alpha}^{(2)} \ ^M\delta\xi^{(2)} \tag{4.8}$$

which means we need $^M\delta\xi^{(2)}$ if we consider tangential contact. We insert into Eq. (3.7) the specialization for the non-continuous mortar side normal field ($\mathcal{N} = -\overline{\mathbf{n}}^{(2)}$, $\mathcal{S} = -\overline{\mathbf{s}}^{(2)}$) and obtain

$$\overline{\mathbf{a}}^{(2)} \cdot \overline{\mathbf{a}}^{(2)}\delta\xi^{(2)} = -\left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right) \cdot \overline{\mathbf{a}}^{(2)} + g_N \cdot \delta\overline{\mathbf{a}}^{(2)} \tag{4.9}$$

$$\overline{\alpha}^{(2)2} \ ^M\delta\xi^{(2)} = -\left(\delta\overline{\mathbf{u}}^{(2)} - \mathbf{u}^{(1)}\right) \cdot \overline{\mathbf{a}}^{(2)} + g_N \, \overline{\mathbf{n}}^{(2)} \cdot \left(\delta\overline{\mathbf{u}}^{(2)}_{,\xi^{(2)}} + \overline{\mathbf{x}}^{(2)}_{,\xi\xi}{}^M\delta\xi^{(2)}\right)$$

$$^M\delta\xi^{(2)} = \frac{1}{\overline{\alpha}^{(2)2} - g_N \, \overline{\beta}^{(2)}} \left[-\left(\delta\overline{\mathbf{u}}^{(2)} - \mathbf{u}^{(1)}\right) \cdot \overline{\mathbf{a}}^{(2)} + g_N \, \overline{\mathbf{n}}^{(2)}\delta\overline{\mathbf{u}}^{(2)}_{,\xi^{(2)}}\right] \tag{4.10}$$

With $^M\delta\xi^{(2)}$ we are now able to evaluate Eq. (4.8).

4.2. Averaged non-mortar side normal

This method is based on the idea presented by Yang et al. [44] and then adopted and used by Puso [28] and Tur et al. [35]. There are different possibilities for carrying our the averaging of the normal field. The already published methods are collected and explained in Table 4.1.

In this first averaging attempt of Yang et al. [44] the normal vector is weighted by the length of the adjacent edge. $$\hat{\hat{\mathbf{n}}}_{k+1}^{(1)} = \frac{l_{k+1}\,\mathbf{n}_k^{(1)}\Big|_{\xi^{(1)}=+1} + l_k\,\mathbf{n}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}}{\left\|l_{k+1}\,\mathbf{n}_k^{(1)}\Big|_{\xi^{(1)}=+1} + l_k\,\mathbf{n}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}\right\|}$$	
One might simplify the first attempt, as done in the work of Popp et al. [27], by omitting the weighting. The disadvantage might be that short edges have a to big influence on the normal field. $$\hat{\hat{\mathbf{n}}}_{k+1}^{(1)} = \frac{\mathbf{n}_k^{(1)}\Big|_{\xi^{(1)}=+1} + \mathbf{n}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}}{\left\|\mathbf{n}_k^{(1)}\Big|_{\xi^{(1)}=+1} + \mathbf{n}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}\right\|}$$	
This averaging method is based on the tangent vector itself and was first presented by Tur et al. [35]. $$\hat{\hat{\mathbf{n}}}_{k+1}^{(1)} = \frac{\mathbf{a}_k^{(1)}\Big|_{\xi^{(1)}=+1} + \mathbf{a}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}}{\left\|\mathbf{a}_k^{(1)}\Big|_{\xi^{(1)}=+1} + \mathbf{a}_{k+1}^{(1)}\Big|_{\xi^{(1)}=-1}\right\|} \times \mathbf{e}_3$$	
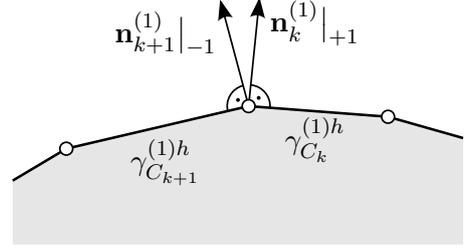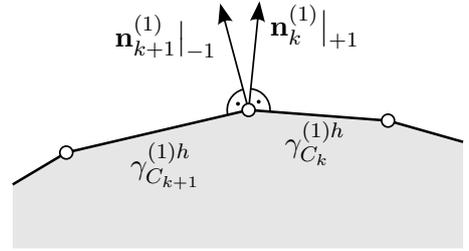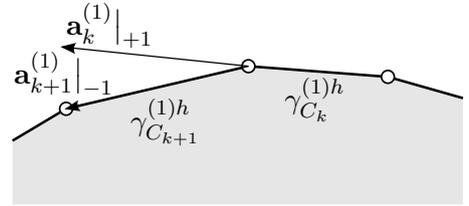
Table 4.1.: Possible averaging methods

We use the two tangent vectors at one corner node before normalization and calculate the arithmetic mean. This way the influence of the edge length is kept and the variation and linearization can be done with limited effort. For the following we assume $\xi \in [-1, 1]$.

$$\hat{\tilde{\mathbf{a}}}_k^{(1)} = \frac{1}{2}\left(\mathbf{x}_{,\xi_k}^{(1)}\Big|_{\xi=+1} + \mathbf{x}_{,\xi_{k+1}}^{(1)}\Big|_{\xi=-1}\right) = \frac{1}{2}\left(\mathbf{a}_k^{(1)}\Big|_{\xi=+1} + \mathbf{a}_{k+1}^{(1)}\Big|_{\xi=-1}\right) \qquad (4.11)$$

This is somehow a mixture between the second and third variant of Table 4.1.

We can use the averaged tangent vectors in the corner nodes to interpolate the average tangent on arbitrary points along the edge and calculate the normal

$$\tilde{\mathbf{a}}^{(1)} = \hat{\tilde{\mathbf{a}}}_k^{(1)} N_k^{(1)} + \hat{\tilde{\mathbf{a}}}_{k+1}^{(1)} N_{k+1}^{(1)} \qquad ; \qquad \tilde{\mathbf{n}}^{(1)} = \frac{\tilde{\mathbf{a}}^{(1)}}{\|\tilde{\mathbf{a}}^{(1)}\|} \times \mathbf{e}_3 \qquad (4.12)$$

It does not matter if one interpolates first and build the normal afterwards or vice versa like shown in following equations

$$\hat{\tilde{\mathbf{n}}}_k^{(1)} = \frac{\hat{\tilde{\mathbf{a}}}_k^{(1)}}{\|\hat{\tilde{\mathbf{a}}}_k^{(1)}\|} \times \mathbf{e}_3 \qquad ; \qquad \tilde{\mathbf{n}}^{(1)} = \hat{\tilde{\mathbf{n}}}_k^{(1)} N_k^{(1)} + \hat{\tilde{\mathbf{n}}}_{k+1}^{(1)} N_{k+1}^{(1)} \qquad (4.13)$$

With this definition of the normal field we can use the kinematic relations from Chapter 3
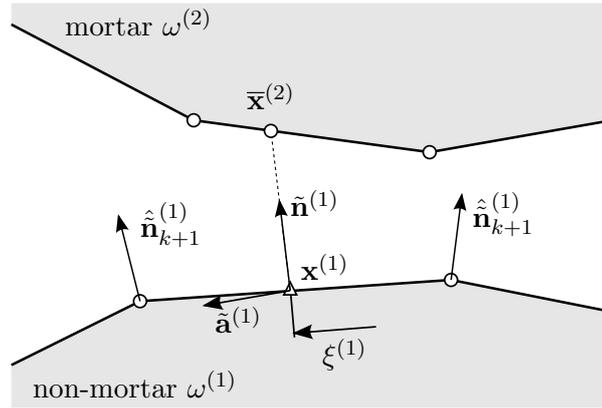
Figure 4.2.: One can see that the corner nodal normal vectors $\hat{\tilde{\mathbf{n}}}_k^{(1)}$ and $\hat{\tilde{\mathbf{n}}}_{k+1}^{(1)}$ are not normal to any of the adjacent edges. With the interpolation from Eq. (4.12) we obtain the normal vector $\tilde{\mathbf{n}}^{(1)}$ at the integration point $\mathbf{x}_{IP}^{(1)}$.

and set $\mathcal{N} = \tilde{\mathbf{n}}^{(1)}$

$$\overline{\mathbf{x}}^{(2)} = \mathbf{x}^{(1)} + {}^A g_N \tilde{\mathbf{n}}^{(1)} \qquad (4.14)$$

4.2.1. Nearest point projection

Multiplying Eq. (4.14) with the average tangent $\tilde{\mathbf{a}}^{(1)}$ yields the projection condition

$$\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \tilde{\mathbf{a}}^{(1)} = 0 \qquad (4.15)$$

See Section 7.2.2.2 for details on implementation.

4.2.2. Normal gap

Multiplying Eq. (4.14) with the averaged normal $\tilde{\mathbf{n}}^{(1)}$ (see also Eq. (3.4)) yields the gap function

$$^{A}g_N = \left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{n}}^{(1)} \tag{4.16}$$

Variation of Eq. (4.14) leads to

$$\delta \overline{\mathbf{u}}^{(2)} + \overline{\mathbf{a}}^{(2)} \delta \xi^{(2)} = \delta \mathbf{u}^{(1)} + {}^{A}\delta g_N\, \tilde{\mathbf{n}}^{(1)} + {}^{A}g_N\, \delta \tilde{\mathbf{n}}^{(1)} \tag{4.17}$$

We multiply Eq. (4.17) with $\tilde{\mathbf{n}}^{(1)}$ and obtain (see Eq. (3.5))

$$^{A}\delta g_N = \tilde{\mathbf{n}}^{(1)} \cdot \left(\delta \overline{\mathbf{u}}^{(2)} - \delta \mathbf{u}^{(1)} + \overline{\mathbf{a}}^{(2)}\, {}^{A}\delta \xi^{(2)} \right) \tag{4.18}$$

> **Remark:** In comparison to Equation (4.7) we have to calculate the variation $^{A}\delta \xi^{(2)}$ - because $\tilde{\mathbf{n}}^{(1)}$ is not orthogonal to $\overline{\mathbf{a}}^{(2)}$.

The variation $^{A}\delta \xi^{(2)}$ is shown in the following subsection as it is strongly related to tangential contact conditions.

4.2.3. Tangential slip

To calculate $^{A}\delta \xi^{(2)}$ we multiply Eq. (4.17) with $\tilde{\mathbf{a}}^{(1)}$ (see also Eq. (3.7)) and obtain

$$^{A}\delta \xi^{(2)} = \frac{\tilde{\mathbf{a}}^{(1)}}{\left(\tilde{\mathbf{a}}^{(1)} \cdot \overline{\mathbf{a}}^{(2)} \right)} \cdot \left[- \left(\delta \overline{\mathbf{u}}^{(2)} - \delta \mathbf{u}^{(1)} \right) + {}^{A}g_N\, \delta \tilde{\mathbf{n}}^{(1)} \right] \tag{4.19}$$

> **Remark:** As $\tilde{\mathbf{n}}^{(1)}$ is fixed in the integration point, it does not depend on $^{A}\xi^{(2)}$. Therefore the resulting variation $^{A}\delta \xi^{(2)}$ does not directly depend on the curvature of the mortar or non-mortar surface. Nevertheless the curvature influences the averaging and is included in $\tilde{\mathbf{a}}^{(1)}$ and its variation.

We end up with the necessity to calculate the variation of the averaged normal $\delta \tilde{\mathbf{n}}^{(1)}$

$$\delta \tilde{\mathbf{n}}^{(1)} = \delta \tilde{\mathbf{s}}^{(1)} \times \mathbf{e}_3$$

$$\delta \tilde{\mathbf{n}}^{(1)} = \left[\frac{\delta \tilde{\mathbf{a}}^{(1)}}{\|\tilde{\mathbf{a}}^{(1)}\|} - \tilde{\mathbf{a}}^{(1)} \frac{\delta \tilde{\mathbf{a}}^{(1)} \cdot \tilde{\mathbf{a}}^{(1)}}{\|\tilde{\mathbf{a}}^{(1)}\|^3} \right] \times \mathbf{e}_3 \tag{4.20}$$

We specialize Eq. (3.14) for the averaged normal field to get

$$^{A}\delta g_T = -\tilde{\mathbf{s}}^{(1)} \cdot \overline{\mathbf{a}}^{(2)A}\delta \xi^{(2)} \tag{4.21}$$

4.3. Uniqueness and solvability of the nearest point projection

The singularity for the nearest point projection as discussed in Section 3.1.1 is a rather small problem for the discrete case. If we have a non-convex mortar surface we are able

to find two possible solutions for the nearest point projection. We can simply select one of them by choosing the solution with the smaller gap value g_N.

However, for the discrete case there are more special cases one should be aware of. A very good and complete study of the different cases can be found in Zavarise and de Lorenzis [45] for the penalty method. It is possible to transfer their results onto the mortar method by simply replacing the slave node with the non-mortar integration point.

The following situations might occur

- The **"in-of-both" case** (see Figure 4.3) can occur for the mortar side normal vector field but not for the averaged normal field. Zavarise and de Lorenzis [45] suggest three possible solutions for this case. We used in our case a very simple one. We choose the non-mortar edge with the smaller gap value g_N. This solution is not unproblematic but really simple and therefore chosen. By choosing the shortest g_N, oscillation effects could be observed which require special treatment (see Section 7.3).
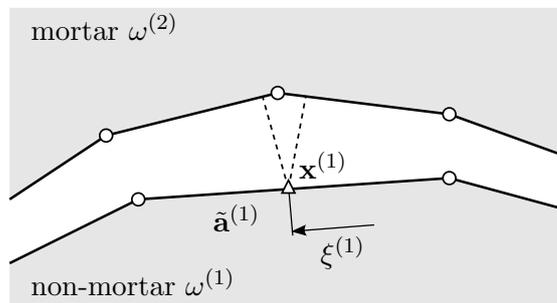
Figure 4.3.: The **"in-of-both" case** where the projection is defined for two possible mortar edges.

- The **"out-of-both" case** can again only occur for the mortar side normal vector field. We overcome this situation once again by choosing the segment with the smaller gap value g_N. In Figure 4.4 one can see that both projection points reside outside the contact surface. They are constructed through extension of the corresponding edges. This means also that these points do physically not exist and cannot be in contact.
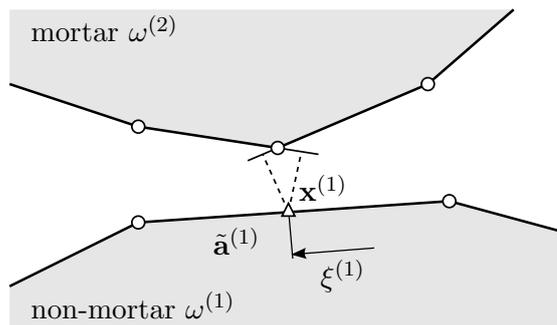
Figure 4.4.: For the **"out-of-both" case** the nearest point projection is not defined at all. Nevertheless we have to chose an appropriate mortar edge to calculate the kinematic properties.

- The **"out-of-first" case** and **"out-of-last" case** are possible, independently from the chosen normal field. As we are using a concentrated integration scheme, (Section 6.5.2) we neglect the terms corresponding to the actual non-mortar integration point $\mathbf{x}_{IP}^{(1)}$.
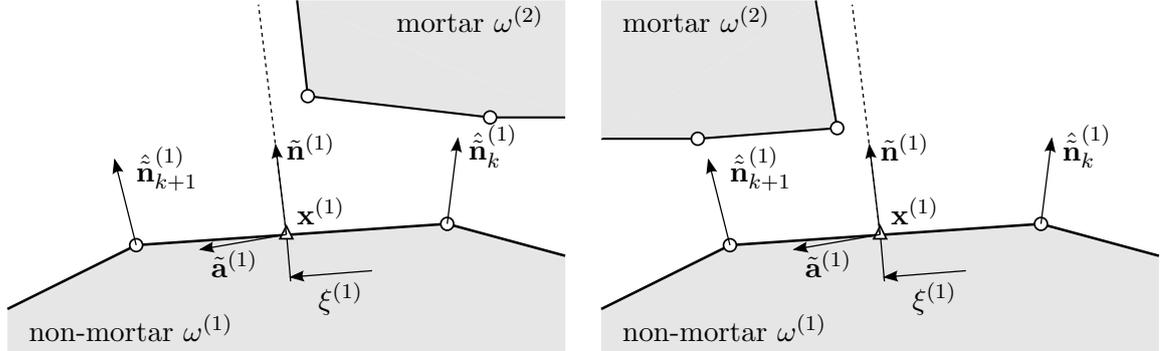
(a) For the "out-of-first" case the projection fails because the first mortar edge of a possible contact surface is out of range.

(b) For the "out-of-last" case the projection fails because the first mortar edge of a possible contact surface is out of range.

Figure 4.5.: For the **"out-of-first" case** and the **"out-of-last" case** no contact partner edge can be determined.

5. Contact mechanics

In the previous chapters we have prepared the kinematic properties needed to analyze the contact situation. In this chapter the kinetics and the contact constraints are discussed.

We start with the mathematical formulation of the contact constraints - the Karush-Kuhn-Tucker conditions for contact. These conditions are the basis for the following incorporation into the weak formulation of our boundary value problem. This virtual work of the contact traction is needed for the discretization with the finite element method.

The mortar method is characterized through the weak enforcement, not only of the contact traction, but also of the contact constraints. It is therefore a mixed method with the contact traction being additional unknowns, the Lagrange parameters. These additional unknowns require additional equations, the weak contact enforcement.

For comparison we have also implemented a penalty regularized method, where the contact tractions depend on the gap function. Therefore we do not have additional unknowns and we do not have to integrate weak constraint conditions into the equation system. A similar mortar condition is used to decide the active set.

5.1. Karush-Kuhn-Tucker conditions

The contact constraints can be summarized with the so called Karush-Kuhn-Tucker conditions for the normal and the tangential contact.

5.1.1. Normal contact conditions

The Karush-Kuhn-Tucker conditions are for the normal direction

$$g_N(\mathbf{x}, t) \geq 0 \tag{5.1}$$

$$t_N \leq 0 \tag{5.2}$$

$$t_N\, g_N = 0 \tag{5.3}$$

The first condition describes the kinematic constraint of non-penetration which means that the gap function g_N has to be zero. The second prevents adhesive traction t_N (t_N represents the contact traction in normal direction on the contact surface Γ_C, see Figure 5.3) The third condition is called the complementary condition. This condition forces the gap to be zero if the pressure is not zero and the pressure to be zero if the gap function is greater then zero. In other words this condition selects only one of the two conditions to be "active".

These conditions can be seen as constitutional law for the normal contact as they connect the kinematics with the kinetics. This law is singular at $g_N = 0$. This can be physically interpreted as the contact traction t_N is a reaction force and can not be calculated by the constitutional law but is a result of the equilibrium.

Different regularization methods can be applied (see for penalty regularization Fischer and Wriggers [6] and for augmented Lagrangian Puso and Laursen [30]). The situation is visualized in Figure 5.1.
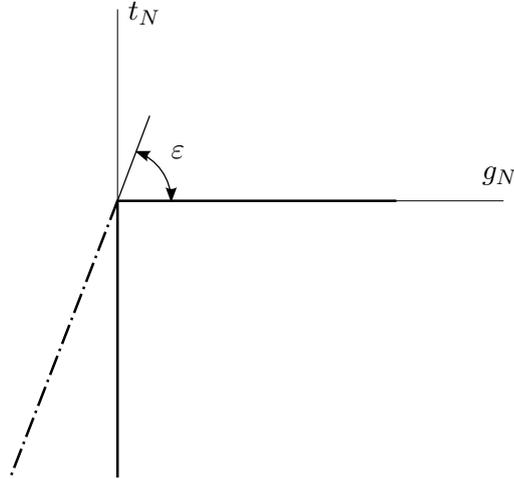
Figure 5.1.: Non-regularized (continuous line) and regularized (dashed line) version of the normal contact condition. A penalty regularization can be achieved with $t_N = \varepsilon g_N$. This is used for the penalty method implementation in Section 6.2.2.

5.1.2. Tangential contact conditions

For the tangential part one can write (see Laursen [24], Wriggers [40] or Willner [36])

$$\mathbf{v}_T - \dot{\gamma}\frac{\mathbf{t}_T}{\|\mathbf{t}_T\|} = 0 \tag{5.4}$$

$$\Psi := \|\mathbf{t}_T\| - \mu\|\mathbf{t}_N\| \le 0 \tag{5.5}$$

$$\dot{\gamma} \ge 0 \tag{5.6}$$

$$\Psi\dot{\gamma} = 0 \tag{5.7}$$

as non-regularized Coulomb friction law. The presented friction law is not differentiable at $\mathbf{v}_T = \mathbf{0}$ (see Figure 5.2), because during active stick the traction in tangential direction is a reaction force. That means the traction is only limited by the constitutional law but can not be calculated with it.

Different regularized versions of the Coulomb friction law can be found in literature to overcome this issue. We are using the non-regularized Coulomb friction law to show the possibility to use it with the mortar method and also to show the performance of this method.
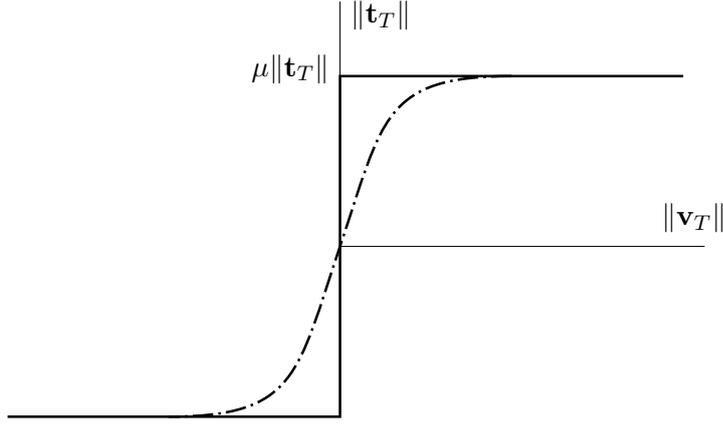
Figure 5.2.: One can see the comparison between the non-regularized (continuous line) and a regularized (dotdashed line) Coulomb friction law. For the non regularized version the singularity (jump) at $g_T = 0$ is shown. There are different methods for doing regularizations. One can find a suitable compilation in Wriggers [40].

5.2. Contact virtual work

Based on the virtual work Eq. (2.30) from Section 2.4.2 we derive the virtual work of contact forces according to Laursen and Simo [23], for the mortar method according to Yang et al. [44], Puso [28].

We use the weak form of the boundary value problem in the initial configuration defined in Eq. (2.30).

$$\delta\Pi^{(i)}\left(\mathbf{u}^{(i)}, \delta\mathbf{u}^{(i)}\right) = \int_{\Omega^{(i)}}\left(\mathbf{S}^{(i)} : \delta\mathbf{E}^{(i)} - \mathbf{f}^{(i)} \cdot \delta\mathbf{u}^{(i)}\right)\mathrm{d}\Omega - \int_{\Gamma_\sigma^{(i)}}\mathbf{t}_{\sigma,0}^{(i)}\cdot\delta\mathbf{u}^{(i)}\mathrm{d}\Gamma - \underbrace{\int_{\Gamma_C^{(i)}}\mathbf{t}_{C,0}^{(i)} \cdot \delta\mathbf{u}^{(i)}\mathrm{d}\Gamma}_{\delta\Pi_C^{(i)}}$$

where $\mathbf{t}_{C,0}^{(i)}$ denotes the traction in the contact surface in the initial configuration. This is done for each of the two bodies $^{(i)}$, the mortar and the non-mortar body.

Now we separate the contact virtual work and insert $i = 1, 2$ which yields

$$\delta\Pi_C\left(\mathbf{u}, \delta\mathbf{u}\right) = -\int_{\Gamma_C{}^{(1)}}\mathbf{t}_{C,0}^{(1)} \cdot \delta\mathbf{u}^{(1)}\mathrm{d}\Gamma^{(1)} - \int_{\Gamma_C{}^{(2)}}\mathbf{t}_{C,0}^{(2)} \cdot \delta\mathbf{u}^{(2)}\mathrm{d}\Gamma^{(2)}$$

$\mathbf{t}_{C,0}^{(i)}$ is an artificial quantity, as in the initial configuration normally no contact occurs. So we carry out a transformation of the contact traction $\mathbf{t}_{C,0}^{(i)}$ from the material to the spatial configuration and get $\mathbf{t}_C^{(i)}$ (which is of course not known a priori)

$$\delta\Pi_C\left(\mathbf{u}, \delta\mathbf{u}\right) = -\int_{\gamma_C{}^{(1)}}\mathbf{t}_C^{(1)} \cdot \delta\mathbf{u}^{(1)}\mathrm{d}\gamma^{(1)} - \int_{\gamma_C{}^{(2)}}\mathbf{t}_C^{(2)} \cdot \delta\mathbf{u}^{(2)}\mathrm{d}\gamma^{(2)} \qquad (5.8)$$

The virtual work of the contact traction Eq. (5.8) on $\gamma_C^{(i)}$ is the contact part of the weak formulation of our boundary value problem. The boundary value problem is presented in
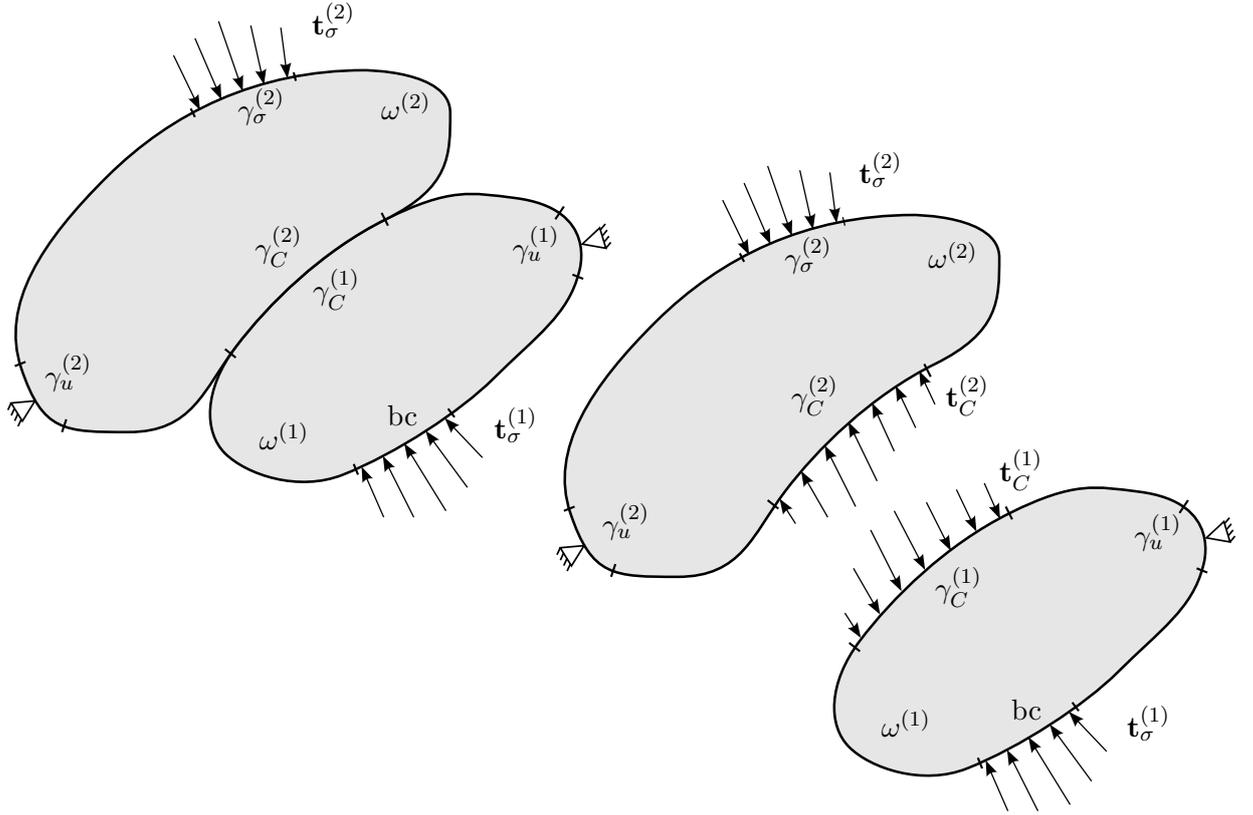
Figure 5.3.: We extend Figure 2.3 for the initial boundary value problem by separating the two bodies in contact at the spatial configuration. The common surface is the contact surface $\gamma_C^{(1)} = \gamma_C^{(2)}$. This contact surface $\gamma_C^{(i)}$ is disjunctive with the Neumann $\gamma_\sigma^{(i)}$ and Dirichlet $\gamma_u^{(i)}$ boundary. If we separate the two bodies we have to introduce a contact traction $\mathbf{t}_C^{(i)}$ which reflects the kinetic effect of one body onto the other.

Figure 5.3. Until now we did not assume or insert any constraint between the two bodies getting into contact.

5.2.1. Equilibrium - momentum conservation

It is quite obvious that the equilibrium between the mortar and non-mortar traction holds.

$$\mathbf{t}_C^{(1)}(\xi^{(1)})\mathrm{d}\gamma^{(1)} = -\mathbf{t}_C^{(2)}(\xi^{(2)})\mathrm{d}\gamma^{(2)}$$

There is a connection through projection between $\xi^{(1)}$ and $\xi^{(2)}$ based on the nearest point projection from Section 3.1. This projection leads to $\bar{\xi}^{(2)}(\xi^{(1)})$.

$$\mathbf{t}_C^{(1)}(\xi^{(1)})\mathrm{d}\gamma^{(1)} = -\mathbf{t}_C^{(2)}\left(\bar{\xi}^{(2)}(\xi^{(1)})\right)\mathrm{d}\gamma^{(2)} \tag{5.9}$$

$$\mathbf{t}_C^{(2)}\left(\bar{\xi}^{(2)}(\xi^{(1)})\right) = -\mathbf{t}_C^{(1)}(\xi^{(1)})\frac{\mathrm{d}\gamma^{(1)}}{\mathrm{d}\gamma^{(2)}} \tag{5.10}$$

Insertion of Eq. (5.10) into Eq. (5.8) yields

$$\delta\Pi_C = -\int_{\gamma_C} \mathbf{t}_C{}^{(1)} \cdot \delta\mathbf{u}^{(1)} \mathrm{d}\gamma^{(1)} + \int_{\gamma_C} \mathbf{t}_C{}^{(1)} \cdot \delta\mathbf{u}^{(2)} \mathrm{d}\gamma^{(1)}$$

$$\delta\Pi_C = \int_{\gamma_C} \mathbf{t}_C{}^{(1)} \cdot \left\{ \delta\mathbf{u}^{(2)} - \delta\mathbf{u}^{(1)} \right\} \mathrm{d}\gamma^{(1)} \tag{5.11}$$

The next step is to split the traction $\mathbf{t}_C{}^{(1)}$ into a normal t_N and a tangential t_T part. We omit the marker $^{(1)}$ because in the following all tractions are defined on the non-mortar side.

> **Remark:** The contact tractions are defined on the non-mortar surface.

$$\mathbf{t}_C{}^{(1)} = \mathbf{t}_N + \mathbf{t}_T = t_N \,\mathcal{N} + \mathbf{t}_T \tag{5.12}$$

We also omit $^{(1)}$ for the differential contact surface $\mathrm{d}\gamma$.

> **Remark:** All the integration is done on the non-mortar side.

Using the virtual gap function Eq. (3.5) leads to

$$\delta\Pi_C = \int_{\gamma_C} t_N \,\mathcal{N} \cdot \left\{ \delta\left[\mathbf{u}^{(2)}(\bar{\xi}^{(2)}) \right] - \delta\mathbf{u}^{(1)} \right\} \mathrm{d}\gamma + \int_{\gamma_C} \mathbf{t}_T \cdot \left\{ \delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)} \right\} \mathrm{d}\gamma$$

$$\delta\Pi_C = \underbrace{\int_{\gamma_C} t_N \,\delta g_N \,\mathrm{d}\gamma}_{\delta\Pi_{CN}} + \underbrace{\int_{\gamma_C} \mathbf{t}_T \,\delta\mathbf{g} \,\mathrm{d}\gamma}_{\delta\Pi_{CT}} \tag{5.13}$$

As a result we get two expressions, one for the normal contact part $\delta\Pi_{CN}$, and one for the tangential part $\delta\Pi_{CT}$. This kind of formulation is often referred to as "split formulation". If one wants to calculate a domain decomposition (a contact situation where only perfect stick occurs) the splitting is not needed and a full formulation is straight forward. We are working towards a formulation suitable for frictional contact so we need to distinguish between the tangential and the normal part. This is because we have different constitutional laws for the normal and the tangential contact.

5.2.2. Normal contact

At first we will have a look onto the normal contact part of the virtual work. As already mentioned in the introduction of Chapter 5 it is characteristic for the mortar method to incorporate the non penetration condition in weak form.

5.2.2.1. Weak non penetration condition

To obtain the weak from of the non penetration condition Eq. (5.1) we test the gap function g_N with the virtual contact traction δt_N (KKT $\equiv$ Karush-Kuhn-Tucker).

$$\delta\Pi_{KKT_N} = \int_{\gamma_C} \delta t_N \, g_N \,\mathrm{d}\gamma = 0 \tag{5.14}$$

We add the virtual contact work Eq. (5.13) and the virtual non penetration condition Eq. (5.14) to our virtual work as both must be zero independent from each other

$$\delta\Pi_N = \delta\Pi_{CN} + \delta\Pi_{KKT_N} = \int_{\gamma_C} t_N\,\delta g_N\,\mathrm{d}\gamma + \int_{\gamma_C} \delta t_N\,g_N\,\mathrm{d}\gamma = 0 \qquad (5.15)$$

We will see in the following (compare Eq. (5.17)) that it is advantageous to add the zero term $\int_{\gamma_c} t_N\,g_N\,\delta(\mathrm{d}\gamma)$. This integral is zero because of the complementary condition Eq. (5.3) which is fulfilled for perfect contact but not for the discrete situation. By adding this term full symmetry is recovered even in the discrete case.

5.2.3. Contact potential for normal contact

Definition of a contact potential is a very common method, as already presented by Laursen and Simo [23] and applied by Fischer and Wriggers [6] or Tur et al. [35].

$$\Pi_N = \int_{\gamma_c} t_N\,g_N\,\mathrm{d}\gamma \qquad (5.16)$$

Variation of this contact potential for the normal contact part leads to

$$\delta\Pi_N = \underbrace{\int_{\gamma_c} \delta t_N g_N\,\mathrm{d}\gamma}_{\delta\Pi_{KKT_N}} + \underbrace{\int_{\gamma_c} t_N\,\delta g_N\,\mathrm{d}\gamma}_{\delta\Pi_{CN}} + \int_{\gamma_c} \underbrace{t_N\,g_N}_{0}\,\delta(\mathrm{d}\gamma) = 0 \qquad (5.17)$$

The first term is nothing else then the weak form of the non penetration condition Eq. (5.1), $\delta\Pi_{KKT_N}$. The second term is part of the weak formulation as derived in Eq. (5.13), $\delta\Pi_{CN}$. The third term has no contribution to the contact virtual work because of the complementary condition Eq. (5.3).

> **Remark:** We are adding the variation of the contact surface (third term) to the variation of the contact potential of the normal part to obtain a symmetric linearization for the discrete case.

We can see that it does not matter if we derive the weak contact formulation from the common weak formulation for the structural problem Eq. (2.30) or from a contact potential Eq. (5.16).

$$\delta\Pi_N = \delta\Pi_{CN} + \delta\Pi_{KKT_N} = 0 \qquad (5.18)$$

5.2.4. Tangential contact

If we want to use the non regularized Coulomb friction law as given in Eqs. (5.4)-(5.7) we have to differ between the stick and slip case as the transition has a singularity.

This is similar to the distinction between active and not active contact in the normal direction. A big difference to the normal condition is, that in the case of not active normal contact the contribution to the contact virtual work of the normal part $(\delta\Pi_N)$ is zero. This is neither true for slip nor stick case where we have contributions to the virtual work of the tangential part $(\delta\Pi_T)$ for both cases.

The contribution of the tangential tractions to the virtual contact work is stated in Eq. (5.13)

$$\delta\Pi_{CT} = \int_{\gamma_c} \mathbf{t}_T \cdot \delta\mathbf{g}\,\mathrm{d}\gamma \tag{5.19}$$

For the two dimensional case we can insert $\mathbf{t}_T = t_T\,\mathcal{S}$ and get

$$\delta\Pi_{CT} = \int_{\gamma_c} t_T\,\mathcal{S} \cdot \delta\mathbf{g}\,\mathrm{d}\gamma \tag{5.20}$$

$$= \int_{\gamma_c} t_T\,\mathcal{S} \cdot \left(\delta\overline{\mathbf{u}}^{(2)} - \delta\mathbf{u}^{(1)}\right)\mathrm{d}\gamma \tag{5.21}$$

$$= \int_{\gamma_c} t_T\,\delta g_T\,\mathrm{d}\gamma \tag{5.22}$$

The difference between the stick and slip case is the way $\mathbf{t}_T$ is calculated. That means we have to switch between two different constitutional laws.

Remark:
- sticking $\Rightarrow$ $\mathbf{t}_T$ is a reaction force and depends on the load. The kinematic condition, that the relative velocity $\mathbf{v}_T$ has to be zero, must be fulfilled (see Eq. (5.4) for $\dot{\gamma} = 0$)
- sliding $\Rightarrow$ $\mathbf{t}_T$ is an impressed surface force. For Coulombs friction law the tangential traction depends on the normal traction and a friction coefficient μ (see Eq. (5.5)).

5.2.4.1. Stick case

For the stick case the situation in tangential direction is very similar to the normal direction. We have to fulfill the tangential stick condition Eq. (5.4) (for $\dot{\gamma}$ being 0) which means the relative velocity $\mathbf{v}_T$ has to be zero. We use the defined incremental slip $\Delta^t g_T = v_T\,\Delta t$ from Eq. (3.29).

We transfer the Karush-Kuhn-Tucker condition Eq. (5.4) into a pure geometric condition without time dependency. This condition is now fulfilled in a weak sense.

$$\delta\Pi_{KKT_{ST}} = \int_{\gamma_c} \delta t_T\,\Delta^t g_T\,\mathrm{d}\gamma \tag{5.23}$$

The contribution to the virtual work is given as

$$\delta\Pi_{T_{ST}} = \delta\Pi_{KKT_{ST}} + \delta\Pi_{CT} + 0 = \int_{\gamma_c} \delta t_T\,\Delta^t g_T\,\mathrm{d}\gamma + \int_{\gamma_c} t_T\,\delta g_T\,\mathrm{d}\gamma + \int_{\gamma_c} \underbrace{t_T\,\Delta^t g_T}_{0}\,\delta(\mathrm{d}\gamma) \tag{5.24}$$

Remark: We are adding the variation of the contact surface here to obtain a symmetric linearization later on. This procedure is similar to the one shown in Eq. (5.17). The contribution is zero due to the stick condition and the relative velocity being zero.

5.2.4.2. Slip case

As already stated, $\mathbf{t}_T$ is no longer a reaction for the slip case but an impressed surface force. For the Coulomb friction law it depends on the normal traction and a friction coefficient μ. We have defined the scalar tangential traction t_T to point into the direction of $\mathcal{S}$. The same is true for a positive direction of $\mathbf{v}_T$. We know from Coulombs friction law that the tangential traction has to point into the opposite direction of the relative velocity $\mathbf{v}_T$. Therefore we get

$$t_T = -\mu\, t_N\, \frac{\mathbf{v}_T \cdot \mathcal{S}}{\|\mathbf{v}_T \cdot \mathcal{S}\|} = -\mu\, t_N \operatorname{sign}(\mathbf{v}_T, \mathcal{S}) \tag{5.25}$$

and we can see that Eq. (5.5) is just the absolute value of this equation.

As we do not introduce any additional unknown we also do not need any additional equation or inequation. The virtual contact work for the sliding case can be written therefore as

$$\begin{aligned}
\delta\Pi_{T_{SL}} &= \int_{\gamma_c} -\mu\, t_N\, \frac{\mathbf{v}_T \cdot \mathcal{S}}{\|\mathbf{v}_T \cdot \mathcal{S}\|}\, \delta g_T\, \mathrm{d}\gamma \\
&= \int_{\gamma_c} -\mu\, t_N \operatorname{sign}(\Delta^t g_T)\, \delta g_T\, \mathrm{d}\gamma
\end{aligned} \tag{5.26}$$

It is not possible to construct a symmetric operator for the slip case.

6. The finite element formulation

The aim of this chapter is to introduce the finite element method as solution method for the contact boundary value problem. We will omit the discussion of implementing and realizing the classical finite element procedure for the weak formulation of the boundary value problem from Eq. (2.30). One can find excellent literature on this topic as in Zienkiewicz and Taylor [46, 47], Bathe [2] or Bonet and Wood [5].

First the Newton-Raphson procedure is discussed. This method is used to solve the non-linear system of equations.

Second we discuss the possible methods of contact enforcement, namely the Lagrange method as classical mortar method and the penalty method as regularizing method.

Third the mortar element with all the needed shape functions and degrees of freedom is introduced. We will see that this contact element is not a finite element in the "classic" sense but somehow artificial to fit into classical finite element implementations.

Forth we present the active set strategy to decide on the active contact properties.

Last the integration scheme applied in this work is explained and a short overview of available other methods is given.

6.1. Newton-Raphson procedure

We are calculating the equilibrium at the time step $n + 1$ which means the weak form of the equilibrium has to be fulfilled (see Eq. (2.30)).

$$\delta\Pi_{int}\Big|_{n+1} + \delta\Pi_{ext}\Big|_{n+1} = 0 \tag{6.1}$$

The external loads at the time step $n+1$, $\mathbf{t}_\sigma\Big|_{n+1}$ are known as they are part of the problem definition. We can therefore calculate the potential of external loads $\delta\Pi_{ext}\Big|_{n+1}$ at the current time step. But we are not able to calculate $\delta\Pi_{int}\Big|_{n+1}$. So we apply a Taylor series expansion

$$\delta\Pi_{int}\Big|_{n+1} = \underbrace{\delta\Pi_{int}\Big|_{n} + \Delta\left(\delta\Pi_{int}\right)\Big|_{n}}_{\delta\widetilde{\Pi}} + \dots \tag{6.2}$$

We stop the expansion after the linear term and insert $\delta\widetilde{\Pi}$ it into our equilibrium Eq. (6.1) which yields

$$\delta\Pi_{int}\Big|_{n} + \Delta\left(\delta\Pi_{int}\right)\Big|_{n} + \delta\Pi_{ext}\Big|_{n+1} \equiv 0 \tag{6.3}$$

6. The finite element formulation

We end up with a fixed point equation which can be solved iteratively. $\Delta\left(\delta\Pi_{int}\right)\Big|_{n}$ depends on increments of the degrees of freedom. In each iteration step $\Big|^{k}$ we solve

$$\Delta\left(\delta\Pi_{int}\right)\Big|_{n}^{k} = -\delta\Pi_{int}\Big|_{n}^{k} - \delta\Pi_{ext}\Big|_{n+1} \tag{6.4}$$

Here we can extract the well known linearized stiffness matrix $\Delta\mathbf{K}\Big|_{n}^{k}$

$$\Delta\left(\delta\Pi_{int}\right)\Big|_{n}^{k} = \int_{\Omega}\left(\Delta\mathbf{F}\Big|_{n}^{k}\cdot\mathbf{S}\Big|_{n}^{k}:\delta\mathbf{F} + \mathbf{F}\Big|_{n}^{k}\cdot\Delta\mathbf{S}\Big|_{n}^{k}:\delta\mathbf{F}\right)\mathrm{d}\Omega$$

$$= \delta\hat{\mathbf{u}}^{T}\cdot\Delta\mathbf{K}\Big|_{n}^{k}\cdot\Delta\hat{\mathbf{u}}\Big|_{n}^{k}$$

and the residual vector $\mathbf{R}\Big|_{n}^{k}$ is defined as

$$\delta\hat{\mathbf{u}}^{T}\cdot\mathbf{R}\Big|_{n}^{k} = -\delta\Pi_{int}\Big|_{n}^{k} - \delta\Pi_{ext}\Big|_{n+1}$$

$$= -\int_{\Omega}\left(\mathbf{F}\Big|_{n}^{k}\cdot\mathbf{S}\Big|_{n}^{k}:\delta\mathbf{F}\right)\mathrm{d}\Omega + \int_{\omega}\mathbf{f}_{B}\Big|_{n+1}\cdot\delta\mathbf{u}\,\mathrm{d}\omega + \int_{\gamma_{\sigma}}\mathbf{t}_{\sigma}\Big|_{n+1}\cdot\delta\mathbf{u}\,\mathrm{d}\gamma$$

which leads to the finite element equation system

$$\Delta\mathbf{K}\Big|_{n}^{k}\cdot\Delta\hat{\mathbf{u}}\Big|_{n}^{k} = \mathbf{R}\Big|_{n}^{k} \tag{6.5}$$

This system is set up and solved in a given time step n for all iterations k. The updated displacement vector is calculated with $\hat{\mathbf{u}}\Big|_{n}^{k+1} = \hat{\mathbf{u}}\Big|_{n}^{k} + \Delta\hat{\mathbf{u}}\Big|_{n}^{k}$. This iteration is done until the abort criterion is fulfilled

$$\mathbf{R}\Big|_{n}^{k}\cdot\Delta\hat{\mathbf{u}}\Big|_{n}^{k} < \varepsilon \tag{6.6}$$

Throughout the following $\Big|_{n}^{k}$ is omitted as all the values have to be calculated for a given time step n in an iteration step k.

6.2. Contact enforcement

In Section 5.2 the necessity of an additional relation between kinetic and kinematic in the weak formulation has been already discussed. We can either use the Lagrange method, which is the classical mortar method, or a penalty regularization.

6.2.1. Lagrange method

For the Lagrange method we introduce additional unknowns λ_N for the normal contact and λ_T for the stick case in tangential direction.

6.2.1.1. Normal contact

We substitute the unknown normal traction t_N with the negative Lagrange multiplier. This means we add new unknowns to the equation system.

$$t_N = -\lambda_N$$

To solve this additional unknowns we need the weak form of the Karush-Kuhn-Tucker conditions $\delta\Pi_{KKTN}$ (see Eq. (5.14)) in addition to the virtual work from the equilibrium $\delta\Pi_{CN}$ (see Eq. (5.13)).

$$\delta\Pi_N = -\int_{\gamma_C} \delta\lambda_N g_N \mathrm{d}\gamma - \int_{\gamma_C} \lambda_N \delta g_N \mathrm{d}\gamma - \int_{\gamma_C} \lambda_N g_N \delta(\mathrm{d}\gamma) \tag{6.7}$$

As already mentioned for Eq. (5.17) we add a third term, which is zero, to get a symmetric linearization as a consequence.

Linearization Each line of the increment is symmetric

$$\begin{aligned}
\Delta\left(\delta\Pi_N\right) = {} & -\int_{\gamma_C} \delta\lambda_N \Delta g_N \mathrm{d}\gamma - \int_{\gamma_C} \Delta\lambda_N \delta g_N \mathrm{d}\gamma \\
& -\int_{\gamma_C} \lambda_N \Delta\left(\delta g_N\right) \mathrm{d}\gamma \\
& -\int_{\gamma_C} \lambda_N \delta g_N \Delta(\mathrm{d}\gamma) - \int_{\gamma_C} \lambda_N \Delta g_N \delta(\mathrm{d}\gamma) \\
& -\int_{\gamma_C} \delta\lambda_N g_N \Delta(\mathrm{d}\gamma) - \int_{\gamma_C} \Delta\lambda_N g_N \delta(\mathrm{d}\gamma) \\
& -\int_{\gamma_C} \lambda_N g_N \Delta\left[\delta(\mathrm{d}\gamma)\right]
\end{aligned} \tag{6.8}$$

Here we can see that the third in the virtual contact work is needed for symmetry in the third and fourth line of the virtual contact work increment.

6.2.1.2. Tangential contact for stick case

For the stick case the tangential traction is a reaction force and therefore unknown. Similar to the normal direction (Section 6.2.1.1) we introduce the Lagrange parameter.

$$t_T = \lambda_T$$

This yields

$$\delta\Pi_{CT_{ST}} = \int_{\gamma_C} \delta\lambda_T \Delta^t g_T \mathrm{d}\gamma + \int_{\gamma_C} \lambda_T \delta g_T \mathrm{d}\gamma + \int_{\gamma_C} \lambda_T \Delta^t g_T \delta\left(\mathrm{d}\gamma\right) \tag{6.9}$$

once again with the additional zero term at the end.

6. The finite element formulation

Linearization It is important to note that the increment $\Delta\left(\Delta^t g_T\right)$ is equivalent to Δg_T which can be derived from the variation of the tangential slip δg_T from Eq. (3.14) with $\delta \to \Delta$.

$$
\begin{aligned}
\Delta\left(\Delta^t g_T\right) &= \Delta\left(v_T\,\Delta t\right) \\
&= \Delta\left[\left(\dot{\overline{\mathbf{x}}}^{(2)} - \dot{\mathbf{x}}^{(1)}\right)\cdot\mathcal{S}\,\Delta t\right] \\
&= \Delta\left[\left(\frac{\overline{\mathbf{x}}^{(2)}|^{n+1} - \overline{\mathbf{x}}^{(2)}|^{n}}{\Delta t} - \frac{\mathbf{x}^{(1)}|^{n+1} - \mathbf{x}^{(1)}|^{n}}{\Delta t}\right)\cdot\mathcal{S}\,\Delta t\right] \\
&= \left(\Delta\overline{\mathbf{u}}^{(2)} - \Delta\mathbf{u}^{(1)}\right)\cdot\mathcal{S} \\
\Delta\left(\Delta^t g_T\right) &= -\mathcal{S}\cdot\overline{\mathbf{a}}^{(2)}\Delta\xi^{(2)} = \Delta g_T
\end{aligned}
\tag{6.10}
$$

Once again each line of the increment is symmetric

$$
\begin{aligned}
\Delta\left(\delta\Pi_{T_{ST}}\right) = &+ \int_{\gamma_C} \delta\lambda_T\Delta g_T\mathrm{d}\gamma + \int_{\gamma_C} \Delta\lambda_T\delta g_T\mathrm{d}\gamma \\
&+ \int_{\gamma_C} \lambda_T\Delta\left(\delta g_T\right)\mathrm{d}\gamma \\
&+ \int_{\gamma_C} \lambda_T\delta g_T\Delta(\mathrm{d}\gamma) + \int_{\gamma_C} \lambda_T\Delta g_T\delta(\mathrm{d}\gamma) \\
&+ \int_{\gamma_C} \delta\lambda_T\Delta^t g_T\Delta(\mathrm{d}\gamma) + \int_{\gamma_C} \Delta\lambda_T\Delta^t g_T\delta(\mathrm{d}\gamma) \\
&+ \int_{\gamma_C} \lambda_T\Delta^t g_T\Delta\left[\delta(\mathrm{d}\gamma)\right]
\end{aligned}
\tag{6.11}
$$

6.2.1.3. Tangential contact for slip case

For the slip case the contact traction in tangential direction is given by the constitutive law, namely Coulomb's friction law from Eq. (5.4) - Eq. (5.7). We can directly substitute the tangential traction t_T with the relation given in Eq. (5.25).

$$
\delta\Pi_{CT_{SL}} = \int_{\gamma_c} \mu\,\lambda_N\,\mathrm{sign}(\mathbf{v}_T,\mathcal{S})\,\delta g_T\,\mathrm{d}\gamma
\tag{6.12}
$$

> **Remark:** The virtual work of the tangential part for the slip case does not depend on any new unknown Lagrange parameters in tangential direction but depends on the Lagrange parameters in normal direction.

Linearization The linearization procedure is straight forward. As already mentioned, the resulting operator is non symmetric and the symmetry can not be recovered for this formulation. But the numerical experiments show, that the influence on the solvability (with an

appropriate solver) is negligible.

$$\Delta\left(\delta\Pi_{CT_{SL}}\right) = + \int_{\gamma_c} \mu\,\Delta\lambda_N \operatorname{sign}(\mathbf{v}_T,\mathcal{S})\,\delta g_T\,\mathrm{d}\gamma$$
$$+ \int_{\gamma_c} \mu\,\lambda_N \operatorname{sign}(\mathbf{v}_T,\mathcal{S})\,\Delta\left(\delta g_T\right)\,\mathrm{d}\gamma$$
$$+ \int_{\gamma_c} \mu\,\lambda_N \operatorname{sign}(\mathbf{v}_T,\mathcal{S})\,\delta g_T\,\Delta\left(\mathrm{d}\gamma\right) \tag{6.13}$$

6.2.2. Penalty method

Although the Lagrange method represents the classical mortar method, we implemented as a starting point the penalty method as a regularized method. This has been done for the normal part only. If one is interested in doing penalty for tangential direction, see Fischer and Wriggers [7] and for an augmented Lagrangian method Yang et al. [44] or Puso [28].

Nevertheless our penalty implementation is still a segment-to-segment method as the contact condition is checked in the integration points. For a node-to-segment method the penalty regularization of the non penetration would be done at the nodes.

6.2.2.1. Normal contact

We introduce a regularized constitutive law which connects the traction and the gap function $t_N = \varepsilon\,g_N$ inside the contact surface with the help of a parameter ε. This means that our contact potential is given as

$$^P\Pi_N = \frac{1}{2}\int_{\gamma_c} \varepsilon g_n^2 \mathrm{d}\gamma$$

which leads to the virtual contact work

$$\delta^P\Pi_N = \int_{\gamma_C} \delta g_N\,\varepsilon\,g_N \mathrm{d}\gamma + \frac{1}{2}\int_{\gamma_C} \varepsilon\,g_N^2\delta(\mathrm{d}\gamma) \tag{6.14}$$

Linearization For the penalty method we obtain a symmetric increment for the normal part as we do for the Lagrange method.

$$\Delta\left(\delta^P\Pi_N\right) = + \int_{\gamma_C} \Delta(\delta g_N)\,\varepsilon\,g_N \mathrm{d}\gamma + \int_{\gamma_C} \delta g_N\,\varepsilon\,\Delta g_N \mathrm{d}\gamma$$
$$+ \int_{\gamma_C} \delta g_N\,\varepsilon\,g_N\Delta(\mathrm{d}\gamma) + \int_{\gamma_C} \Delta g_N\,\varepsilon\,g_N\delta(\mathrm{d}\gamma)$$
$$+ \frac{1}{2}\int_{\gamma_C} \varepsilon\,g_N^2\Delta\left[\delta(\mathrm{d}\gamma)\right]$$

6. The finite element formulation

6.3. The mortar element

One can not find a defined patch (an element) in classical sense with nodes related to each other by a boundary of edges or surfaces. It also depends on the kind of integration scheme (see Section 6.5) used which mortar element definition seems to be more "natural". We will present a solution which is well applicable to the concentrated integration scheme and to the application of rather abstract assembling algorithms.
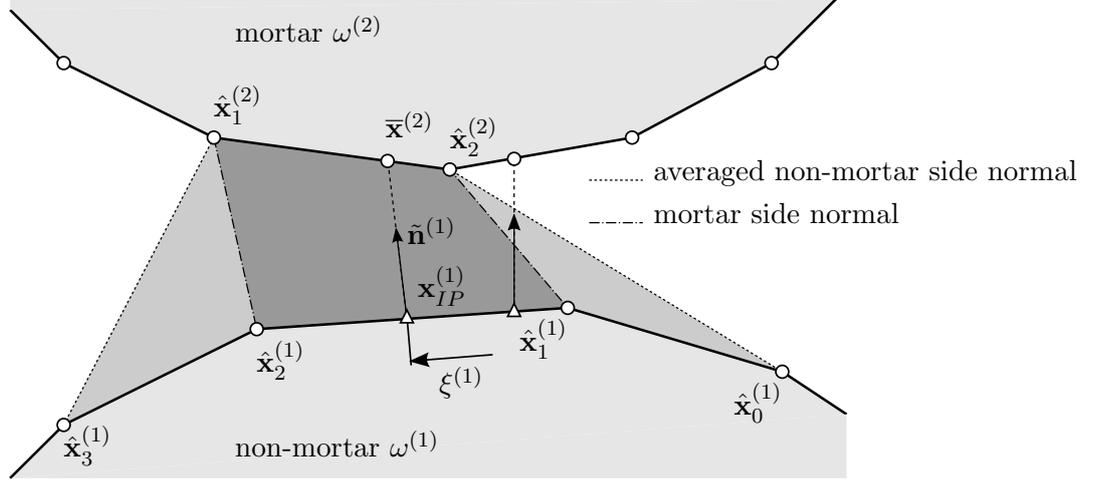
Figure 6.1.: The mortar element patch for mortar side normal (dark gray) and the mortar element patch for averaged non-mortar side normal are shown. The elements are defined for an integration point at $\mathbf{x}_{IP}^{(1)}$. This integration point depends on the degrees of freedoms (DOFs) of its corresponding non mortar edge but also on the DOFs of the corresponding mortar edge. This edge is determined with the nearest point projection $\overline{\mathbf{x}}^{(2)}$. For the averaged normal field we also need the DOFs of the preceding and the following edge (see Section 6.3.3). One can see that depending on the nearest projection edge two integration points on the same non-mortar edge can have different mortar edges. (right integration point)

In this work a mortar element is defined as a patch with a minimum amount of degrees of freedom needed to calculate a contribution to the global stiffness. A principal sketch of possible mortar elements is shown in Figure 6.1. We have to define such a mortar element for each integration point, because two different integration points on the same non-mortar edge might have different projection edges. For the averaged normal we also need the preceding and the following edge to a non-mortar edge. In Figure 6.2 a visualization is given.

6.3.1. Lagrange degrees of freedom

At first we define the degrees of freedom for the Lagrange values. Each **active node** (see Section 6.4) on the non-mortar edge has a Lagrangian degree of freedom. That is the Lagrange factor in normal direction λ_N.

Figure 6.2.: The mortar element patch for the averaged non-mortar side normal field is shown in gray. The nodal normal vectors $\hat{\tilde{\mathbf{n}}}_k^{(1)}$ and $\hat{\tilde{\mathbf{n}}}_{k+1}^{(1)}$ define the integration basis for the averaged normal vector $\tilde{\mathbf{n}}^{(1)}$ itself. To calculate this nodal normal vectors we also need the nodes of the preceding and the following non-mortar edge.

At least one node has to be active - if none of the non-mortar nodes is active we do not have to calculate any contribution to the global stiffness.

6.3.2. Mortar side normal

We define a nodal value object for each integration point as mentioned in Figure 6.1. We give the displacement matrix here but a nodal coordinate matrix $^M\hat{\mathbf{x}}_{IP}$ can be defined similarly.

$$^M\hat{\mathbf{u}}_{IP} = \left[\begin{array}{cc|cc} \hat{u}_{x1}^{(1)} & \hat{u}_{x2}^{(1)} & \hat{u}_{x1}^{(2)} & \hat{u}_{x2}^{(2)} \\ \hat{u}_{y1}^{(1)} & \hat{u}_{y2}^{(1)} & \hat{u}_{y1}^{(2)} & \hat{u}_{y2}^{(2)} \end{array} \right] \rightarrow {}^M\hat{u}_{IP_{ij}}$$

6.3.3. Averaged non-mortar side normal

We define a nodal value tensor for each integration point in a similar way. But for averaging we need the end points of the proceeding and the following edge on the non-mortar side (see Figure 6.2). Once again we give the displacement matrix here but a nodal coordinate matrix $^A\hat{\mathbf{x}}_{IP}$ can be defined similarly.

$$^A\hat{\mathbf{u}}_{IP} = \left[\begin{array}{cccc|cc} \hat{u}_{x0}^{(1)} & \hat{u}_{x1}^{(1)} & \hat{u}_{x2}^{(1)} & \hat{u}_{x3}^{(1)} & \hat{u}_{g_{x1}}^{(2)} & \hat{u}_{g_{x2}}^{(2)} \\ \hat{u}_{y0}^{(1)} & \hat{u}_{y1}^{(1)} & \hat{u}_{y2}^{(1)} & \hat{u}_{y3}^{(1)} & \hat{u}_{g_{y1}}^{(2)} & \hat{u}_{g_{y2}}^{(2)} \end{array} \right] \rightarrow {}^M\hat{u}_{IP_{ij}}$$

6.3.4. Shape functions

$N_j^{(i)}(\xi^{(i)})$ denotes the classical Lagrange shape function for $\xi^{(i)} \in [-1, +1]$ on the surface $\gamma_C^{(i)}$ for the node j.

6.3.4.1. Dual shape functions

These are used to interpolate the Lagrange multipliers along the non-mortar edge.

$$\mathbf{\Phi}^{me}(\xi^{(1)}) = \left[\begin{array}{cc} \Phi_1(\xi^{(1)}) & \Phi_2(\xi^{(1)}) \end{array} \right] \rightarrow \Phi_i^{me}(\xi^{(1)})$$

The dual shape functions ϕ_i for the linear case are

$$\Phi_1(\xi^{(1)}) = \frac{1}{2}(1 - 3\xi^{(1)}) \qquad ; \qquad \Phi_2(\xi^{(1)}) = \frac{1}{2}(1 + 3\xi^{(1)}) \tag{6.15}$$

It is not necessary to use dual shape functions. We could also use standard Lagrange shape functions but with the help of dual shape functions we can implement a condensation algorithm to simplify the saddle point problem and remove the Lagrange multiplier from the global set of unknowns. Those dual mortar methods have advantages for solving the linear set of equations. See especially the work of Hüeber and Wohlmuth [17], Hüeber and Wohlmuth [18] or Flemisch et al. [8] but also the application on non linear problems by Popp et al. [27] or Gitterle et al. [10]. We did not implement the reduction for this work so either the dual or the classical shape functions might be used.

6.3.4.2. Mortar side normal

Corresponding to the defined nodal values we need the shape functions for interpolating the gap function along the non-mortar edge. As we assume that $\mathcal{N}$ directs from non-mortar to mortar side (see Eq. (3.4)) we need to calculate $\mathbf{g} = (\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)})$. With the definition of the following interpolation matrix

$$^M\mathbf{N}^{me} = \left[\begin{array}{cc|cc} -N_1^{(1)}(\xi^{(1)}) & -N_2^{(1)}(\xi^{(1)}) & N_1^{(2)}(\xi^{(2)}) & N_2^{(2)}(\xi^{(2)}) \end{array} \right] \rightarrow {}^M N_i^{me}$$

we can write the components of the gap vector at a given integration point in a compressed index notation

$$g_i = {}^M N_j^{me} \, {}^M \hat{x}_{ij}$$

6.3.4.3. Averaged non-mortar side normal

In the case of averaged normal field we have more degrees of freedom per non-mortar edge. Therefore we have to extend the shape functions too.

$$^A\mathbf{N}^{me} = \left[\begin{array}{ccccc|cc} 0 & -N_1^{(1)}(\xi^{(1)}) & -N_2^{(1)}(\xi^{(1)}) & 0 & N_{g_1}^{(2)}(\xi^{(2)}) & N_{g_2}^{(2)}(\xi^{(2)}) \end{array} \right] \rightarrow {}^A N_i^{me}$$

This allows us once again to calculate the components of the gap vector at a given integration point

$$g_i = {}^A N_j^{me} \, {}^A \hat{x}_{ij}$$

6.4. Active set strategy

This is a very crucial part of the mortar method. We have transferred the strong (means local pointwise) defined contact conditions (see Section 5.1) into weak global form. This means the contact conditions are fulfilled in mean globally and not locally in each point. But for the algorithm we need a discrete active set strategy. What we do in this section is to derive a criterion for active or non active nodes. This does not mean that we do a node to segment method in the classical sense. The criterion is not based on a pointwise but edgewise fulfillment of the contact conditions. This procedure of defining the active set in a global sense and also the comparison to a local decision is discussed in depth by Hild [16].

6.4.1. Lagrange method

The whole active set strategy is mainly developed for the enforcement of the contact conditions with Lagrange multipliers. However, we will also show (based on the work of Fischer and Wriggers [6]) how an active set strategy can be realized for the penalty method (see Section 6.4.2).

6.4.1.1. Normal condition

We have to find two conditions. One for nodes which are not in contact. They might get into contact (which means the non-penetration condition Eq. (5.1) gets violated) and get activated therefore. The other condition must detect if active nodes are still in contact. They have to be deactivated when the surface traction becoms positive Eq. (5.2).

Actually no contact

If we do not have active contact we have to check the non-penetration condition (Eq. (5.1)) in weak manner. The strong condition is now transformed into a weak non-penetration condition for a discrete non-mortar edge. It can be formulated with $\delta\lambda_N$ being the test function for the kinematic contact constraint defined in Eq. (5.1).

$$\int_{\gamma_C^h} \delta\lambda_N \ g_N \ \mathrm{d}\gamma \geq 0 \tag{6.16}$$

$$\sum_P \delta\hat{\lambda}_{N_P} \underbrace{\int_{\gamma_C^h} \Phi_P(\xi^{(1)}) \ g_N \ \mathrm{d}\gamma}_{\tilde{g}_{N_P}} \geq 0 \tag{6.17}$$

For the quadrature of $\tilde{g}_{N_P}$ we need the value of the actual gap function g_N in each integration point. With this definition it is possible to decide per node with index P whether node P is still not active.

$$\tilde{g}_{N_P} \geq 0 \tag{6.18}$$

As this integral has to be evaluated over the whole contact surface γ_C^h a global loop over all non-mortar edges has to be done. The contributions of Φ_P on one node P have to be assembled and taken into account.

6. The finite element formulation

Already in contact

For all the nodes already in contact we have to check if the kinetic contact condition (Eq. (5.2)) is still fulfilled.

$$t_N \leq 0 \qquad (6.19)$$

Once again this strong condition is transformed into a weak one. Now we are using a virtual gap function $\delta g_N = N_P^{(1)}(\xi^{(1)})\, \delta \hat{g}_{N_P}$ as test function. (Attention: $\lambda_N = -t_N$)

$$\int_{\gamma_C^h} \delta g_N \lambda_N \mathrm{d}\gamma \geq 0$$

$$\sum_P \delta \hat{g}_{N_P} \underbrace{\int_{\gamma_C^h} N_P^{(1)}(\xi^{(1)})\lambda_N \mathrm{d}\gamma}_{-\tilde{t}_{N_P}} \geq 0 \qquad (6.20)$$

Now we are able to decide the active contact on nodal basis. The node remains active as long as

$$\tilde{t}_{N_P} \leq 0 \qquad (6.21)$$

6.4.1.2. Tangential condition

It is obvious that the tangential condition has to be checked only if the node is active. Else no tangential traction occurs. If we have active contact we have to differ between the stick and the slip case. Nodes which are getting into new contact are assumed to stick. This is just an assumption and has to be checked within the inner active set loop.

Stick Case

For the stick case the tangential traction λ_T is a reactive traction. The parameter $\dot{\gamma}$ in Eq. (5.4) is zero and so is the relative velocity v_T. One has to check that the tangential traction is less then the maximum traction defined by Coulombs law in Eq. (5.5). (see also Figure 5.2)

$$\Psi \leq 0$$
$$\|\lambda_T\| \leq \mu \|t_N\|$$

This strong condition is transferred into a weak one by using $\delta g_T = N_P^{(1)} \delta \hat{g}_{T_P}$

$$\int_{\gamma_c} \delta g_T\, \Psi\, \mathrm{d}\gamma \leq 0$$

$$\int_{\gamma_c} \delta g_T\, (\|\lambda_T\| - \mu\, \|\lambda_N\|)\, \mathrm{d}\gamma \leq 0$$

$$\delta \hat{g}_{T_P} \underbrace{\int_{\gamma_c} N_P^{(1)}(\xi^{(1)})\, \|\lambda_T\|\, \mathrm{d}\gamma}_{\tilde{t}_{T_P}} \leq \delta \hat{g}_{T_P}\, \mu \|\tilde{t}_{N_P}\| \qquad (6.22)$$

We now have a nodal based weak condition whether node P is sticking

$$\tilde{t}_{T_P} \leq \mu \|\tilde{t}_{N_P}\| \tag{6.23}$$

Slip Case

For the slip case the parameter $\dot{\gamma}$ in Eq. (5.7) is non-zero and thus Ψ from Eq. (5.5) has to be zero. With Eq. (5.4) and Eq. (5.5) we are able to define value and direction of λ_T. The slip case is active as long as Eq. (5.6) holds. This means that $\dot{\gamma}$ and λ_T must have opposite directions. As soon as $\Delta^t g_T$ gets zero (which should be a rather rare case for numerical reasons) or changes its sign we fall back to the stick condition. Thus we use the virtual Lagrange parameter as test function for the tangential slip $\Delta^t g_T$.

$$\int_{\gamma_c} \delta\lambda_T \, \Delta^t g_T \, \mathrm{d}\gamma = \sum_P \delta\hat{\lambda}_{T_P} \underbrace{\int_{\gamma_c} \Phi_P(\xi^{(1)}) \, \Delta^t g_T \, \mathrm{d}\gamma}_{\Delta^t \tilde{g}_{T_P}} \tag{6.24}$$

6.4.2. Penalty method

For the penalty method we do not transfer the strong pointwise conditions into a weak form. That is not needed as there are no additional unknowns. We decide the active set per integration point here. (see Fischer and Wriggers [6] for further explanations)

No contact on integration point

For the calculation of contact stiffness we need the gap value g_N at each integration point, namely g_N^{IP}. This value is a result of the nearest point projection algorithm.

We do not add any contribution of the integration point at $\mathbf{x}_{IP}^{(1)}$ to the virtual contact work if

$$g_N^{IP} \geq 0 \tag{6.25}$$

Active contact on integration point

We add the contribution of the integration point at $\mathbf{x}_{IP}^{(1)}$ to the virtual contact work as long as

$$t_N^{IP} < 0$$
$$\varepsilon \, g_N^{IP} < 0 \tag{6.26}$$

6.5. Integration scheme

It is part of the mortar contact algorithm to numerically evaluate integrals like

$$\delta\Pi_C = \int_{\gamma_C} t_\square \, \delta g_\square \, \mathrm{d}\gamma \tag{6.27}$$

6. The finite element formulation

This integral is split by applying the finite element discretization procedure ($\sum_h$ should represent the assembling operator).

$$\delta\Pi_C^h = \sum_h \int_{\gamma_C^h} t_\square \, \delta g_\square \, \mathrm{d}\gamma \tag{6.28}$$

There are different possibilities to realize the quadrature over the discrete contact area segment γ_C^h. Two of them are presented in the following as they are the commonly used ones.

6.5.1. Segmented integration scheme

This integration method is well described in the work by McDevitt and Laursen [26] for 2D and was extended by Puso [28], Puso and Laursen [29, 30] for 3D contact. This method is very popular (see Yang et al. [44], Yang and Laursen [42, 43], Popp et al. [27]). The discrete non-mortar contact surface γ_C^h is split up into segments. This is carried out by projection of the in between mortar side nodes onto the non-mortar edge. This means we split up the integral over γ_C^h into multiple sub integrals over the segments $\gamma_C^{h,\rho}$ with $\gamma_C^h = \gamma_C^{h,1} \cup \gamma_C^{h,2} \cup \cdots \cup \gamma_C^{h,\rho}$. The situation is pictured in Figure 6.3.

$$\delta\Pi_C^h = \sum_h \left[\sum_\rho \int_{\gamma_C^{h,\rho}} t_\square \, \delta g_\square \, \mathrm{d}\gamma \right] \tag{6.29}$$
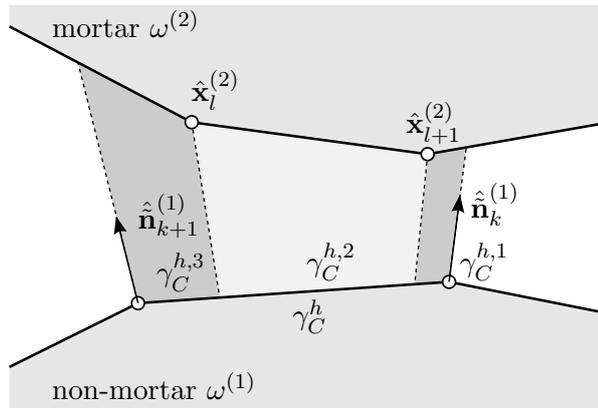
Figure 6.3.: The mortar nodes $\hat{\mathbf{x}}_l^{(2)}$ and $\hat{\mathbf{x}}_{l+1}^{(2)}$ are projected onto the non-mortar edge γ_C^h with the corresponding normal vector. With this procedure three sub segments $\gamma_C^{h,1}, \gamma_C^{h,2}, \gamma_C^{h,3}$ are created.

- In Figure 6.3 the averaged normal field is used. To do a unique projection of a mortar node onto the non-mortar edge the projection has to be unique. Therefore we need a unique normal vector defined in the mortar node. For the mortar side normal field (see Section 4.1) this is not the case. This problem is an important reason for doing the averaging. The argument does not hold for the concentrated integration of Section 6.5.2.

56

- It is important to note, that the in interior limits of the segments $\gamma_C^{h,\rho}$ are deformation dependent (which is not the case for the concentrated integration of Section 6.5.2). Therefore the linearization effort rises. See Yang et al. [44] for detailed explanation of the linearization procedure for this integration scheme.

- On the other hand this integration method is exact. For the linear discretization the sub segments are bounded with straight edge segments. Therefore the numerical quadrature can be done exactly.

As we would like to use the mortar side normal field we have not chosen to use this scheme. Nevertheless one could also implement a non-continuous normal field with the segmented integration scheme. The normal field should be defined on the non-mortar side to do so.

6.5.2. Concentrated integration scheme

The numerical quadrature of $\delta\Pi_C^h$ might be realized with

$$\delta\Pi_C^h = \sum_h \int_{\gamma_C^h} t_\square \, \delta g_\square \, \mathrm{d}\gamma = \sum_h \left[\sum_{IP} t_\square \Big|_{IP} \delta g_\square \Big|_{IP} J^{(1)} \Big|_{IP} \omega_{IP} \right] \qquad (6.30)$$

and ω_{IP} being the weight of the integration point at $\mathbf{x}_{IP}^{(1)}$. All the values inside the integral (like $t_\square\big|_{IP}, \delta g_\square\big|_{IP}, J^{(1)}\big|_{IP}$ for this example) are evaluated at the integration point. The situation is visualized in Figure 6.4.
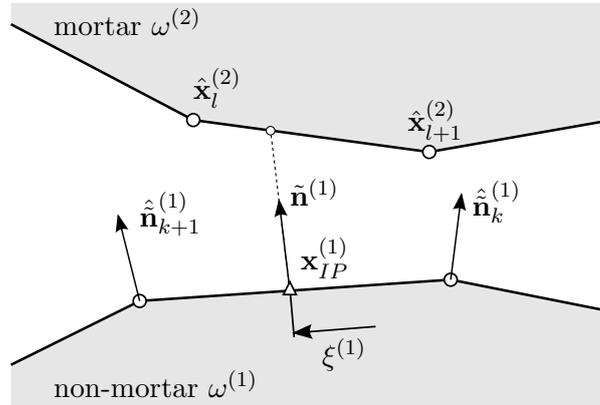
Figure 6.4.: For the concentrated integration scheme all properties inside the integral are evaluated at the integration point $\mathbf{x}_{IP}^{(1)}$. This integration scheme is not able to "detect" the kinks of the mortar boundary at $\hat{\mathbf{x}}_l^{(2)}$ or $\hat{\mathbf{x}}_{l+1}^{(2)}$.

- For this method the boundaries of Γ_C^h (the initial configuration of γ_C^h) are fixed. This simplifies the linearization procedure significantly in comparison to the segmented integration scheme of Section 6.5.1.

- The amount of integration points must be increased. The integration volume is no longer bounded with linear edges (in comparison to the segmented integration of

Section 6.5.1). On the mortar side there may occur kinks (C_1 singularities) which prevent an exact numerical quadrature. Numerical experiments (see also Fischer and Wriggers [6] or Tur et al. [35]) have shown that this influence is negligible and high convergence rate can be achieved.

For the implementation we have chosen to use this concentrated integration scheme. Therefore all linearization and implementation details of Chapter A are given for the concentrated integration scheme.

7. Implementation details

Throughout this chapter essential details of implementing the mortar method are given. We start by giving a brief summary of the global solution algorithm. Then we discuss the realization of the nearest point projection for the two different kinematic implementations; namely the non-continuous mortar and the averaged non-mortar side normal field. This is followed by the presentation of contact search. It is vital to find the possible contact zones to calculate the kinematic properties. Last some further insights into the implementation of the active set strategy are given.

7.1. Solution algorithm

In Algorithm 1 the global solution algorithm of a fixed point Newton-Raphson mortar analysis is given.

> **for** *time_step* **in** *Model.getTimeBar()* **do**
> > integrateBoundaryConditions(time_step);
> > **repeat**
> > > contact_set_changed = false;
> > > active_nodes_changed = false;
> > > **repeat**
> > > > unbalanced_energy = solveFESystem(time_stamp);
> > > > mortarCalcKinematics(time_stamp);
> > > **until** *unbalanced_energy <= convergence_criteria*;
> > > **if** *mortarContactSearch( )* **then**
> > > > contact_set_changed = true;
> > > > mortarCalcKinematics(time_stamp);
> > > **end**
> > > **if** *mortarUpdateActiveSet()* **then**
> > > > active_nodes_changed = true;
> > > **end**
> > **until not** *contact_set_changed* **and not** *active_nodes_changed*;
> **end**

Algorithm 1: Global solution algorithm for mortar based fixed point contact analysis.

The function `integrateBoundaryConditions()` is responsible for updating the Dirichlet and Neumann Boundaries in the Model. This is not different to a classical finite element code.

Inside this outer time step loop there is the inner loop for the active contact set. This means we search for the equilibrium (solve the finite element equation system) as long as

the contact set changes or the active nodes change. The algorithm is quite different from a semi-smooth Newton method like that presented in Hüeber and Wohlmuth [17] or Popp et al. [27] for non linear problems.

Inside this contact loop we need the classical iteration procedure of the Newton-Raphson algorithm. The calculation of the stiffness matrices and the assembling are all done inside the `solveFESystem()` function. This innermost loop, corresponding to the Newton-Raphson procedure, is the same compared with a classical non linear finite element method. Therefore (exact numerical integration assumed) quadratic convergence for the unbalanced energy should be reached. As our numerical experiments have shown, (see Chapter 8) the influence of the approximative concentrated integration method on the convergence rate can be neglected.

> **Remark:** For the averaged non-mortar side normal method one either calculates the average nodal tangent $\tilde{\mathbf{a}}^{(1)}$ before the `solveFESystem()` function is called, or each time the tangent is needed. The latter method is of course less efficient.

Some modifications inside this innermost loop in comparison to a calculation without contact have to be done. The kinematic properties like the gap function g_N, the projected convective coordinate $\overline{\xi}^{(2)}$ have to be recalculated for edge non-mortar integration point after equilibrium is reached. These properties are mostly results of the nearest point projection described in Section 7.2. The two implementations for the two normal fields are shown in Section 7.2.2.1 and Section 7.2.2.2 respectively. The method `mortarCalcKinematics()` is responsible for those calculations.

After the equilibrium is reached we have to search potential new contact partner nodes first. The reason for not doing this before stiffness and equilibrium calculation is, that we start with a given configuration (normally the Lagrangian state) where no contact is assumed. This means a search of contact before the first displacement was calculated makes no sense. The method we call is `mortarContactSearch()` which returns true if the possible set of contact partner nodes has changed. The implementation of this problem is described in Section 7.3. If we find new contact partners we have to recalculate the kinematic properties.

After we have detected the new set of contact partners we have to actualize the active set itself. This means we have to decide whether an integration point (for the penalty method) or a node (for the Lagrange method) is active for the contact. The algorithm is explained in Section 7.4.

7.2. The nearest point projection

In this section we describe how the procedure of the nearest point projection and the calculation of the corresponding kinematic data is implemented.

7.2.1. Selection of best projection edge

We do this for both edges regardless of any mortar projection point being outside of the edge. Part of this algorithm is to determine the gap value $g_{N,k}$ and $g_{N,k+1}$ too. In Figure 7.1

the situation for the mortar side normal field is shown. The issue of selecting the best projection edge only occurs for the non-continuous mortar side normal field.
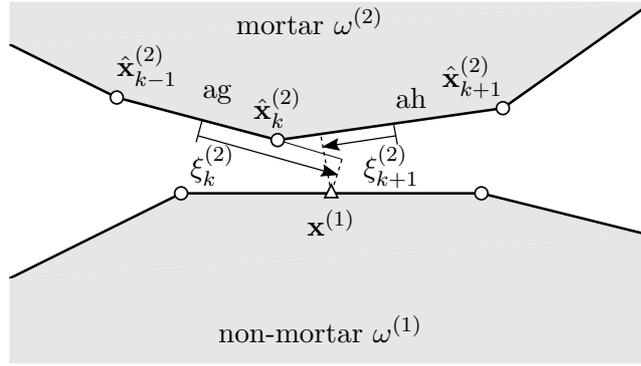
Figure 7.1.: Based on the set $\left\{\hat{\mathbf{x}}_{k-1}^{(2)}, \hat{\mathbf{x}}_{k}^{(2)}, \hat{\mathbf{x}}_{k+1}^{(2)}\right\}$ found by the contact search (see Section 7.3) we execute the nearest point projection. This is done for both edges $\gamma_{C,k}^{(2)}$ and $\gamma_{C,k+1}^{(2)}$. As we can see in this Figure only $\xi_{k+1}^{(2)}$ is in the edge boundary $[-1, +1]$. This means we have to select the right edge. This is done in the last step of the procedure.

1. Step: As a result of the contact search we have two possible projection edges (see Section 7.3). We do the nearest point projection, described in Section 7.2.2.1 and Section 7.2.2.2 respectively, for both edges.

2. Step: Select one of the two edges to be the main projection mortar edge for the given integration point $\mathbf{x}_{IP}^{(1)}$. If $\xi_{k}^{(2)} \in [-1,1] \cup \xi_{k+1}^{(2)} \notin [-1,1]$ the corresponding edge is $\gamma_{C,k}^{(2)}$ or if $\xi_{k}^{(2)} \notin [-1,1] \cup \xi_{k+1}^{(2)} \in [-1,1]$ the corresponding edge is $\gamma_{C,k+1}^{(2)}$.

As already discussed in Section 4.3 there are special cases to take care of. Those have been treated in depth already - so we only add the solutions here.

- The **"in-of-both" case**: If $\xi_{k}^{(2)} \in [-1,1] \cup \xi_{k+1}^{(2)} \in [-1,1]$ the corresponding edge is the one where the projected distance g_N is smaller.

- The **"out-of-both" case**: If $\xi_{k}^{(2)} \notin [-1,1] \cup \xi_{k+1}^{(2)} \notin [-1,1]$ the corresponding edge is the one where the projected distance g_N is smaller. If both possible projection mortar points are out of the mortar edge γ_C^h the projected point is somehow "artificial".

  To prevent this, one might define the corner node $\hat{\mathbf{x}}_{k}^{(2)}$ itself as projection point. This would introduce further special cases into the algorithm and therefore is not carried out. We simply use the artificial point with the smaller gap value g_N. Numerical experiments have shown that the choice has small influence on the algorithm and the solution quality.

- The **"out-of-first" case** and **"out-of-last" case**: Only one $\xi^{(2)}$ exists as only the following ("out-of-first") or the preceding ("out-of-last") non-mortar edge are part of the contact zone γ_C. As already stated in Zavarise and de Lorenzis [45] there is no general solution to overcome these situations. For our code we decided

that if this happens the stiffness term related to this non-mortar integration point $\mathbf{x}_{ip}^{(1)}$ is neglected. This prevents unwanted lifting forces. Although the implementation effort rises as one has to differ between "active" and "not-active" mortar integration points.

7.2.2. Inner Newton-Raphson procedure

For arbitrary edge shapes (quadratic or higher order ones) the projection is a non-linear fix point equation to solve. This nearest point projection is strongly dependent on the chosen normal field and therefore the two methods are explained each on its own.

7.2.2.1. Non-continuous mortar side normal

Here the implementation of the procedure described in Section 3.1 is shown. The Eq. (3.2) is the basis where we substitute $\mathcal{S}$ with $-\overline{\mathbf{a}}^{(2)}$

$$\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \overline{\mathbf{a}}^{(2)} = 0 \tag{7.1}$$

At this point $\overline{\xi}^{(2)}$ is unknown and therefore $\overline{\mathbf{x}}^{(2)}$ is unknown. We solve this fix point equation with Newton's method. Therefore we need the first derivative with respect to $\xi^{(2)}$

$$\left[\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \mathbf{a}^{(2)}\right]_{,\xi^{(2)}} = \mathbf{a}^{(2)} \cdot \mathbf{a}^{(2)} + \underbrace{\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\right)}_{-g_N\,\mathbf{n}^{(2)}} \cdot \mathbf{a}^{(2)}_{,\xi^{(2)}} \tag{7.2}$$

$$= \alpha^{(2)^2} - g_N\beta^{(2)} \tag{7.3}$$

The definition of $\alpha^{(2)}$ and $\beta^{(2)}$ can be found in Section 4.1.1.

> **Remark:** It should be noted that $\beta^{(2)} = 0$ for linear shape functions.

The increment $\Delta\xi^{(2)}\big|^{i+1}$ can be calculated with

$$\Delta\xi^{(2)}\big|^{i+1}\left[\alpha^{(2)^2} - g_N\beta^{(2)}\right]^i = \left[-\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \mathbf{a}^{(2)}\right]^i \tag{7.4}$$

$$\Delta\xi^{(2)}\big|^{i+1} = \left[-\frac{\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)}\right) \cdot \mathbf{a}^{(2)}}{\alpha^{(2)^2} - g_N\beta^{(2)}}\right]^i \tag{7.5}$$

where all the values are calculated with the values from the last Newton step i and the updated location $\xi^{(2)}\big|^{i+1}$ can be calculated with

$$\xi^{(2)}\big|^{i+1} = \xi^{(2)}\big|^i + \Delta\xi^{(2)}\big|^{i+1} \tag{7.6}$$

We are stopping the iteration after the following condition is fulfilled

$$\left\|\left(\mathbf{x}^{(2)}\big|^i - \mathbf{x}^{(1)}\right) \cdot \mathbf{a}^{(2)}\big|^i\right\| \leq \varepsilon \tag{7.7}$$

7.2.2.2. Averaged non-mortar side normal

Here the implementation of the procedure described in Section 3.1 is shown. Eq. (3.2) is the basis where we substitute $\mathcal{S}$ with $\tilde{\mathbf{a}}^{(1)}$

$$\left(\overline{\mathbf{x}}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{a}}^{(1)} = 0 \tag{7.8}$$

Once again we need the first derivative with respect to $\xi^{(2)}$

$$\left[\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{a}}^{(1)} \right]_{,\xi^{(2)}} = \mathbf{a}^{(2)} \cdot \tilde{\mathbf{a}}^{(1)} + \left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \underbrace{\left(\tilde{\mathbf{a}}^{(1)} \right)_{,\xi^{(2)}}}_{=0} \tag{7.9}$$

The increment $\Delta\xi^{(2)}\big|^{i+1}$ can be calculated with

$$\Delta\xi^{(2)}\big|^{i+1} \left[\mathbf{a}^{(2)} \cdot \tilde{\mathbf{a}}^{(1)} \right]^i = \left[- \left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{a}}^{(1)} \right]^i \tag{7.10}$$

$$\Delta\xi^{(2)}\big|^{i+1} = \left[- \frac{\left(\mathbf{x}^{(2)} - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{a}}^{(1)}}{\mathbf{a}^{(2)} \cdot \tilde{\mathbf{a}}^{(1)}} \right]^i \tag{7.11}$$

where all the values are calculated with the values from the last Newton step i.

Remark: One has to check that $\mathbf{a}^{(2)} \cdot \tilde{\mathbf{a}}^{(1)} \neq 0$. This might happen if the averaged tangential vector $\tilde{\mathbf{a}}^{(1)}$ is orthogonal to the mortar edge. (see Figure 7.2) This case is rather theoretical and occurs rather rarely. The physical contact of two orthogonal surfaces is useless. If this happens the contact segment is neglected.

---

mortar $\omega^{(2)}$

$\mathbf{a}^{(2)}$

$\tilde{\mathbf{a}}^{(1)}$ $\tilde{\mathbf{n}}^{(1)}$

non-mortar $\omega^{(1)}$
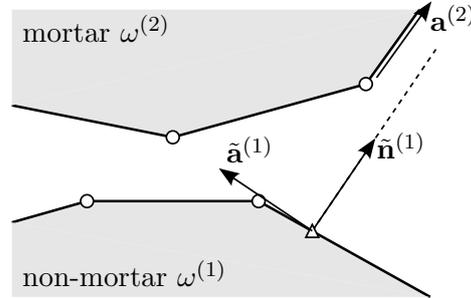
Figure 7.2.: If the averaged tangent vector $\tilde{\mathbf{a}}^{(1)}$ is orthogonal to the corresponding mortar edge the projection is not defined.

The updated location $\xi^{(2)}\big|^{i+1}$ can be calculated with

$$\xi^{(2)}\big|^{i+1} = \xi^{(2)}\big|^i + \Delta\xi^{(2)}\big|^{i+1} \tag{7.12}$$

We are stopping the iteration after the following condition is fulfilled

$$\left\| \left(\mathbf{x}^{(2)}\big|^i - \mathbf{x}^{(1)} \right) \cdot \tilde{\mathbf{a}}^{(1)}\big|^i \right\| \leq \varepsilon \tag{7.13}$$

7.2.3. **Algorithm**

for *mortar_element* **in** *model* **do**
    for *IP* **in** *mortar_element.getNonMortarEdge()* **do**
        IP.archiveLastResults();
        if *research_best_edge* **then**
            for *mortar_edge* **in** *IP.getMortarEdges()* **do**
                projection_results[mortar_edge] = nearestPointProjection(
                mortar_edge);
            end
            $[g_N, \overline{\xi}^{(2)}$, best_mortar_edge] = selectBestEdge(projection_results);
            if *only_one_mortar_edge* **and** $\|\overline{\xi}^{(2)}\| > 1.$ **then**
                IP.deactivate();
            else
                IP.setBestEdge(best_mortar_edge);
            end
        else
            $[g_N, \overline{\xi}^{(2)}]$ = nearestPointProjection(IP.getBestMortarEdge());
        end
        $\Delta^t g_T$ = slip(IP);
        IP.setGapData(g_N, $\Delta^t g_T$, $\overline{\xi}^{(2)}$);
    end
end

Algorithm 2: Algorithm of the nearest point projection and the corresponding selection of the best projection edge.

For each mortar element (see Section 6.3) we have to iterate over the integration points on the corresponding non-mortar edge. Inside this loop we first archive the previous results of the integration point as we need them for our tangential contact algorithm (see especially Section 3.3.1.1). Inside the function `archiveLastResults()` we store the best projection edge (the edge which fulfills the criteria explained in Section 7.2.1), $\overline{\xi}^{(2)}$ and $\Delta^t g_T$. It is not necessary to store $\overline{\xi}^{(2)}$ and $\Delta^t g_T$ but it saves some calculation time to store both. As the normal contact is path independent we do not need g_N.

Now we distinguish whether we have to search the best projection edge or only update the projection point. The first has to be done at last once for each time step at the beginning. We hold the best projection edge fixed afterwards, else it can happen that the best projection edge oscillates between the two possible solutions. This situation is visualized in Figure 7.3. A further argument against searching the best projection edge in each iteration is the loss of quadratic convergence.

If we want to research the best projection edge we first have to do the nearest point projection for both possible edges: `nearestPointProjection( mortar_edge )`. This is the implementation of the inner Newton-Raphson iteration described in Section 7.2.2.1 and Section 7.2.2.2 respectively. The result g_N and $\overline{\xi}^{(2)}$ is then stored in the `projection_results` map. The function `selectBestEdge( projection_results )` is the implementation of
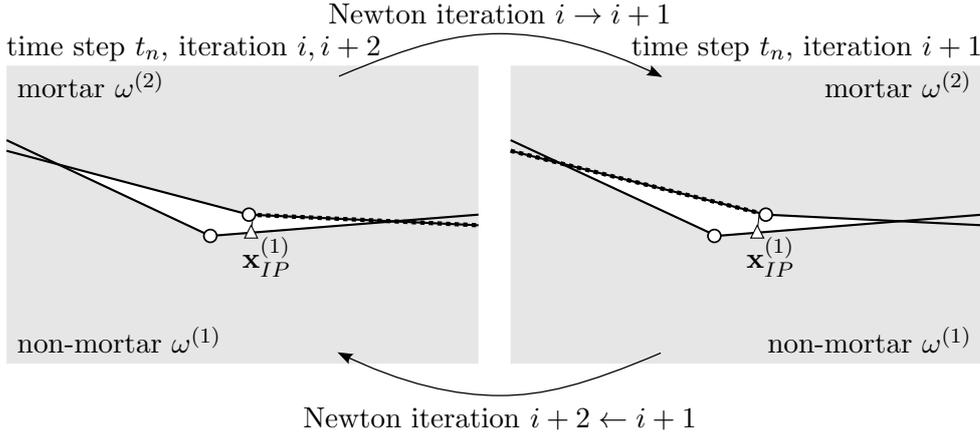
Figure 7.3.: It might happen during the Newton-Raphson iteration process that the best projection edge iterates between two states

Section 7.2.1.

If we only have one mortar side edge (this is the case for **"out-of-first" case** and **"out-of-last" case** - see Section 7.2.1) we check if the convective coordinate $\overline{\xi}^{(2)}$ is in the interval $\overline{\xi}^{(2)} \in [-1, +1]$. If not, we deactivate the integration point and it will not contribute to the stiffness matrix in the following. If we have two possible edges the best edge is stored inside the mortar integration point. Which also means it might happen (in rather rare cases), that $\overline{\xi}^{(2)} \notin [-1, +1]$ which has already been discussed in Section 7.2.1.

If we do not search the best projection edge we only have to do the nearest point projection.

After we have found g_N and $\overline{\xi}^{(2)}$ we can calculate the tangential slip $\Delta^t g_T$ based on Eq. (3.29) with the function `slip()`. Finally the results are stored for each integration point.

7.3. Contact search

To define a mortar element like in Figure 6.1 one has to know the corresponding mortar edge for a given integration point. To detect this we apply an algorithm which can be split up in multiple steps.

1. Step: Find for a given non-mortar integration point $\mathbf{x}_{IP}^{(1)}$ the nearest mortar node $\hat{\mathbf{x}}_k^{(2)}$. This is a global search for each integration point. We can restrict the global search to a local one after the first time step. One has to assume that the relative movement of the non-mortar point $\mathbf{x}_{IP}^{(1)}$ within one time step, is smaller than the radius of the choosen local neighbourhood. In our case we analyse for possible contact partner nodes on the preceding $\gamma_{C,k}^{(2)}$ and following edge $\gamma_{C,k+1}^{(2)}$ for the mortar node $\hat{\mathbf{x}}_k^{(2)}$ of the previous time step. This means the relative movement has to be smaller then the edge length which is equivalent to the discretization size (for detailed discussions see Benson and Hallquist [4]).
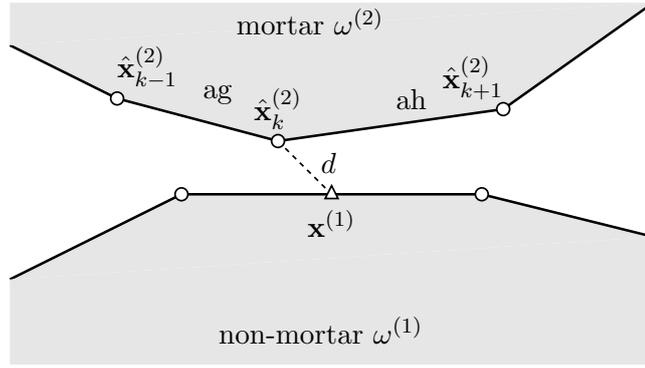
Figure 7.4.: For a given non-mortar integration point $\mathbf{x}_{IP}^{(1)}$ we search for the mortar node $\hat{\mathbf{x}}_k^{(2)}$ with the shortest distance d. Now we are able to build a set $\left\{\hat{\mathbf{x}}_{k-1}^{(2)}, \hat{\mathbf{x}}_k^{(2)}, \hat{\mathbf{x}}_{k+1}^{(2)}\right\}$ containing all nodes of the preceding $\gamma_{C,k}^{(2)}$ and following mortar edge $\gamma_{C,k+1}^{(2)}$.

There are more advanced techniques of finding the appropriate contact partner. In particular, methods based on bounding volumes can be found in literature, e.g. in Yang and Laursen [42, 43].

2. Step: For the preceding $\gamma_{C,k}^{(2)}$ and the following edge $\gamma_{C,k+1}^{(2)}$ of the mortar node $\hat{\mathbf{x}}_k^{(2)}$ do the nearest point projection algorithm to determine $\xi_k^{(2)}$ and $\xi_{k+1}^{(2)}$. See Section 7.2.1 for the selection of one of the two edges to be the projection edge for the contact algorithm.

7.3.1. Algorithm

In Algorithm 3 the implementation of the presented contact search is briefly sketched. If `IP.hasNearesetMortarNode()` evaluates to true we do a local search. Else no nearest mortar node has been stored in this integration point and we have to search globally. In particular for meshes with a lot of nodes this makes a big difference in run-time, although this procedure can be executed parallelly.

7.4. Active set strategy

As already mentioned in Section 6.4 there is a big difference in how we formulate the active set for the penalty and the Lagrangian contact enforcement. Therefore we will describe the corresponding algorithms separately.

7.4.1. Penalty contact enforcement

For the penalty contact enforcement we implement an active set criterion per integration point. The implementation of Eq. (6.25), Eq. (6.26) is shown in Algorithm 4. We only

```
distance = DBL_MAX;
for IP in mortar_element do
    if IP.hasNearestMortarNode() then
        neighborhood = model.getBoundaryNeighborhood(
        IP.getNearestMortarNode() );
        for node in neighborhood do
            new_distance = calcDistance( IP , node );
            if new_distance < distance then
                distance = new_distance;
                nearest_mortar_node = node;
            end
        end
    else
        for mortar_edge in model do
            new_distance = calcDistance( IP , mortar_edge.firstNode() );
            if new_distance < distance then
                distance = new_distance;
                nearest_mortar_node = mortar_edge.firstNode();
            end
            new_distance = calcDistance( IP , mortar_edge.lastNode() );
            if new_distance < distance then
                distance = new_distance;
                nearest_mortar_node = mortar_edge.lastNode();
            end
        end
    end
    IP.setNearestMortarNode( nearest_mortar_node );
    IP.setMortarEdges( model.getAdjacentEdges(nearest_mortar_node) );
end
```

Algorithm 3: Find the nearest mortar node for a non mortar integration point and set the two adjacent mortar edges for this mortar node. Attention: Only edges which are located on mortar boundaries should be added!

implemented the normal contact without constraints on the tangential contact (frictionless case).

7.4.2. Lagrangian contact enforcement

For the Lagrangian contact enforcement we formulate nodal based active set criteria. Further we have to distinguish in tangential direction between the stick and slip case.

First we have to initialize or reset the conditional properties before a new active set is searched. We need 5 conditional values which can be found in Algorithm 5. One can accumulate $(\hat{g}_{N_P}), (\hat{t}_{N_P})$ and $(\Delta^t \hat{g}_{T_P}), (\hat{t}_{T_P})$ into one property respectively as only one of the corresponding pair is needed depending on the nodal state. To clarify the algorithm we

```
for mortar_element in model do
 │  for IP in mortar_element do
 │   │  if IP.gN < 0. and ‖IP.ξ̄⁽²⁾‖ < 1. then
 │   │   │  IP.activate();
 │   │  else
 │   │   │  IP.deactivate();
 │   │  end
 │  end
end
```
Algorithm 4: Find the active integration points inside the non mortar edges.

```
for non_mortar_node in model do
 │  non_mortar_node.(ĝNP) = 0.;
 │  non_mortar_node.(t̂NP) = 0.;
 │  non_mortar_node.(ΔᵗĝTP) = 0.;
 │  non_mortar_node.(t̂TP) = 0.;
 │  non_mortar_node.sliding_direction = 0.;
end
```
Algorithm 5: Reset the nodal condition to 0.. This is needed as we are implementing the numerical integrations of Eq.(6.17), Eq. (6.20), Eq. (6.22), Eq. (6.24) in a recursive way.

refrained from doing so here in the documentation.

Next we have to implement the numerical quadrature of the integrals given in Eq. (6.17), Eq. (6.20), Eq. (6.22) and Eq. (6.24). This integration is done in the same way as we integrate the mortar stiffness matrix of Eq. (A.23)-(A.25), Eq. (A.26) and Eq. (A.56)-(A.58). The procedure is shown in Algorithm 6. The integration weight is written as ω_{IP}. As a result each non-mortar node $\hat{\mathbf{x}}_P^{(1)}$ holds the condition value needed to decide the actual state.

The algorithm to decide about this state is given in Algorithm 7. It is once again possible to reduce the decision tree by merging $(\hat{g}_{N_P})$ with $(\hat{t}_{N_P})$ and $(\Delta^t \hat{g}_{T_P})$ with $(\hat{t}_{T_P})$. This would compress the implementation but is less clear to read.

for *mortar_element* **in** *model* **do**
  for *IP* **in** *mortar_element* **do**
    if not *IP.active()* **then**
      | **continue**;
    end
    for *non_mortar_node* **in** *mortar_element* **do**
      if *non_mortar_node.active()* **then**
        non_mortar_node.$(\hat{t}_{N_P}) += N_P^{(1)}(\xi_{IP}^{(1)})\,(-\lambda_{N_{IP}})\,\omega_{IP}$;
        if *non_mortar_node.sliding()* **then**
          | non_mortar_node.$(\hat{t}_{T_P}) += N_P^{(1)}(\xi_{IP}^{(1)})\,\lambda_{T_{IP}}\,\omega_{IP}$;
        end
        if *non_mortar_node.sticking()* **then**
          | non_mortar_node.$(\Delta^t\hat{g}_{T_P}) += \Phi_P^{(1)}(\xi_{IP}^{(1)})\,\Delta^t g_{T_{IP}}\,\omega_{IP}$;
        end
      else
        | non_mortar_node.$(\hat{g}_{N_P}) += \Phi_P^{(1)}(\xi_{IP}^{(1)})\,g_{N_{IP}}\,\omega_{IP}$;
      end
    end
  end
end

Algorithm 6: Calculate conditional values needed for the active set criterion. (ω_{IP} being the integration weight for the corresponding integration point)

for *non_mortar_node* **in** *model* **do**

  if *non_mortar_node.active()* **then**

    if *non_mortar_node.$(\hat{t}_{N_P})$* $> 0.$ **then**

      non_mortar_node.deactivate();

    else

      if *non_mortar_node.sticking()* **then**

        if $\parallel$ *non_mortar_node.$(\hat{t}_{T_P})\parallel$* $>$ $\parallel$ *non_mortar_node.$(\hat{t}_{N_P}) \cdot \mu\parallel$* **then**

          non_mortar_node.isSliding();

          non_mortar_node.sliding_direction = sign(non_mortar_node.$(\hat{t}_{T_P})$);

        end

      end

      if *non_mortar_node.sliding()* **then**

        if *non_mortar_node.$(\hat{g}_{T_P}) \cdot$* *non_mortar_node.sliding_direction* $< 0.$ **then**

          non_mortar_node.isSticking();

        end

      end

    end

  end

  if *non_mortar_node.not_active()* **and** *non_mortar_node.$(\hat{g}_{N_P})$* $< 0.$ **then**

    non_mortar_node.activate();

    non_mortar_node.isSticking();

  end

end

Algorithm 7: Algorithm to detect the actual state for each node.

8. Numerical examples

In this chapter a selection of numerical examples is presented. The aim is to show the performance of the presented algorithm and to compare the two different normal fields implemented (namely the non continuous mortar side and the averaged non-mortar side normal field). The chosen problems are inspired by examples shown by many other people publishing in the field of contact mechanics.

The finite element software was written in a self-made framework - the SOOFEA (**S**oftware for **O**bject **O**riented **F**inite **E**lement) framework (see Hammer [14]). The chosen programming language is C++ and Python. The design is based on a paper from Archer et al. [1]. To visualize the deformed meshes and the various results matplotlib (Hunter [19]) was used. The meshes were created either by self written structural meshers or with GMSH (Geuzaine and Remacle [9]).

8.1. Simple benchmark

This example is based on a three dimensional problem presented in Puso et al. [31]. The dimensions of the problem are shown in Figure 8.1. The parameteres (especially the material properties) are given in Table 8.1.

Young's modulus of block	E_{block}	1000	MPa
Poisson's ratio of block	ν_{block}	0.3	
Young's modulus of slab	E_{slab}	1000	MPa
Poisson's ratio of slab	ν_{slab}	0.3	
Thickness of block and slab	h	1	mm
Vertical displacement	Δv	10	mm
Horizontal displacement	Δu	50	mm

Table 8.1.: Parameters for the simple benchmark example

8.1.1. Progress

In the beginning (see Figure 8.2(a) and Figure 8.3(a)) there is an initial gap of 1mm between the block and the slab. This gap was chosen to show that the implementation is able to capture a fully new contact surface.

At first the block is moved in vertical direction downwards. The end displacement $\Delta v = $ 10mm is reached after 1sec and 10 time steps. The state of deformation is shown in Fig-
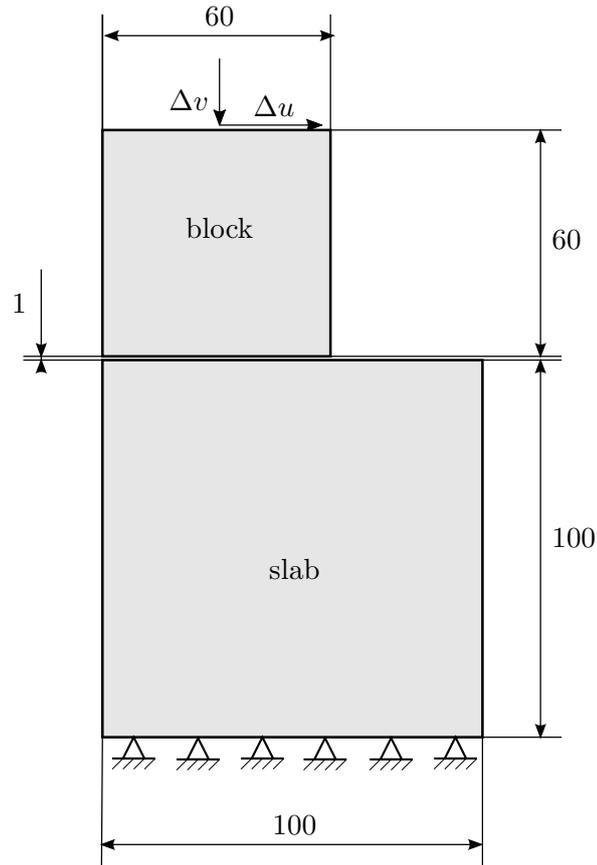
Figure 8.1.: A simple benchmark problem. A block is moved towards the slab in the direction of Δv. The boundary condition is applied on the top boundary of the block. After a certain impact is reached the block is moved over the slab in the direction of $\pm \Delta u$.

ure 8.2(b) and Figure 8.3(b). One can see for the frictionless calculation an overhang of the slab. For the friction case due to stick the left side of block and slab are still planar.

Now the block is moved in 30 time steps right until a horizontal displacement $\Delta u = 50$mm is reached at time $t = 4$sec. The algorithm converges for less time steps too, but we wanted to show intermediate results to discuss the different phases of deformation. At time step 17 after $t = 1.7$sec the slab starts to slide and we change from stick to slip. In Figure 8.3(c) one can see the shearing of slab and block due to the friction force. In Figure 8.2(c) no shear occurs as we assume perfect sliding there. The impact of the friction force on the deformation can be also seen in the comparison of Figure 8.2(d) and Figure 8.3(d).

After $t = 40$sec the moving direction of the block changes. Therefore the block starts to stick again on the block. Until time step 57 this sticking force is lower then the limit and the slab changes its shear direction. In Figure 8.2(e) one can see the rather straight and shearless deformation of the block in comparison to Figure 8.3(e). The movement continues until time $t = 70$sec is reached (Figure 8.2(f) and Figure 8.3(f)).

Now the block is lifted till the start position at time $t = 8$sec and $\Delta u = 0$ and $\Delta v = 0$

is reached. During this lift the deformation of the frictionless sliding is congruent to the deformation during the downwards movement. This can be already seen as the deformation in Figure 8.2(f) is identically to the one in Figure 8.2(b). This also means that the deformation at time step 75 is the same as for time step 5.

This is of course not true for the frictional contact calculation. One can see that for time step 75, Figure 8.3(f), the situation is different to that in Figure 8.2(b). This means for frictional calculation we have a path dependency. This is related to the stick effect and the dependency of the friction force direction on the moving direction. In mathematical and algorithmic sense this can be seen in the fact that we have to store the last convective coordinate $\overline{\xi}^{(2)}$ as internal variable. (See Section 3.3.1.1 and Eq. (3.28))

The shearing state is consequently reduced during lifting. The intermediate step in Figure 8.3(g) visualizes this procedure.

Finally time step 80 is reached and the initial state is recovered. As we have no contact in this last step the frictional case is identical to the frictionless case.

To show the (nearly) quadratic convergence of the Newton-Raphson procedure some examples of convergence rates of the unbalanced energy are given in Table 8.2 and Table 8.3. It is also interesting to see that the concentrated integration scheme we used has no significant influence on the convergence rate although it is an approximation of the exact integral only.

		Time step 10	Time step 17	Time step 40	Time step 57
Frictional case	iteration 1	+2.4179e+03	+2.1598e+03	+2.1481e+03	+2.1591e+03
	iteration 2	+7.3381e-02	+8.4075e-01	+2.8016e-01	+2.7159e-01
	iteration 3	+1.7256e-08	+3.4533e-07	+1.4972e-08	+2.7641e-08
	iteration 4	+1.6147e-22	+7.5659e-20	+1.8974e-22	+1.8560e-22
	iteration 1		+5.7085e+02	+3.2615e-24	+2.2148e+02
	iteration 2		+9.9229e-01		+9.2576e-02
	iteration 3		+3.6019e-06		+2.4516e-08
	iteration 4		+2.6846e-18		+1.3568e-22
Frictionless case	iteration 1	+2.4245e+03	+1.9669e+03	+2.2115e+03	+1.9720e+03
	iteration 2	+9.5056e-03	+7.5421e-02	+4.5648e-01	+8.0077e-02
	iteration 3	+8.2737e-12	+5.8048e-10	+1.7844e-07	+5.8707e-10
	iteration 4	+6.0677e-23	+3.0205e-21	+2.1941e-19	+2.1347e-24
	iteration 1		+3.0003e+01		+2.0580e+01
	iteration 2		+2.1412e-04		+3.6496e-03
	iteration 3		+5.4181e-14		+3.3724e-12
	iteration 4		+3.5354e-24		+3.3132e-24

Table 8.2.: Evolution of the unbalanced energy over Newton iterations on the non-continuous mortar side normal field.

		Time step 10	Time step 17	Time step 40	Time step 57
Frictionless case	iteration 1	+2.4179e+03	+2.1598e+03	+2.1481e+03	+2.1591e+03
	iteration 2	+7.3388e-02	+8.4075e-01	+2.8010e-01	+2.7153e-01
	iteration 3	+1.7184e-08	+3.4441e-07	+1.6015e-08	+2.7746e-08
	iteration 4	+1.6805e-22	+7.4172e-20	+1.3933e-22	+2.1243e-22
	iteration 1		+5.7088e+02	+2.9497e-24	+2.2154e+02
	iteration 2		+9.8701e-01		+9.1662e-02
	iteration 3		+3.4191e-06		+2.3042e-08
	iteration 4		+2.4365e-18		+1.2758e-22
Frictional case	iteration 1	+2.4245e+03	+1.9669e+03	+2.2111e+03	+1.9720e+03
	iteration 2	+9.4703e-03	+7.5540e-02	+4.5360e-01	+8.0220e-02
	iteration 3	+8.2598e-12	+5.7925e-10	+1.6284e-07	+5.8451e-10
	iteration 4	+4.7937e-23	+1.2982e-20	+1.3769e-19	+1.1331e-21
	iteration 1		+2.9856e+01		+1.8326e+01
	iteration 2		+1.9592e-04		+2.9215e-03
	iteration 3		+4.0828e-14		+1.6454e-12
	iteration 4		+4.5703e-24		+2.9256e-24

Table 8.3.: Evolution of the unbalanced energy over Newton iterations on the averaged non-mortar side normal field.

(a) time $t = 0$sec (b) time $t = 1$sec (time step 10) (c) time $t = 1.7\dot{3}$sec (time step 17)

(d) time $t = 4$sec (time step 40) (e) time $t = 5.7$sec (time step 57) (f) time $t = 7$sec (time step 70)

(g) time $t = 7.5$sec (time step 75) (h) time $t = 8$sec (time step 80)
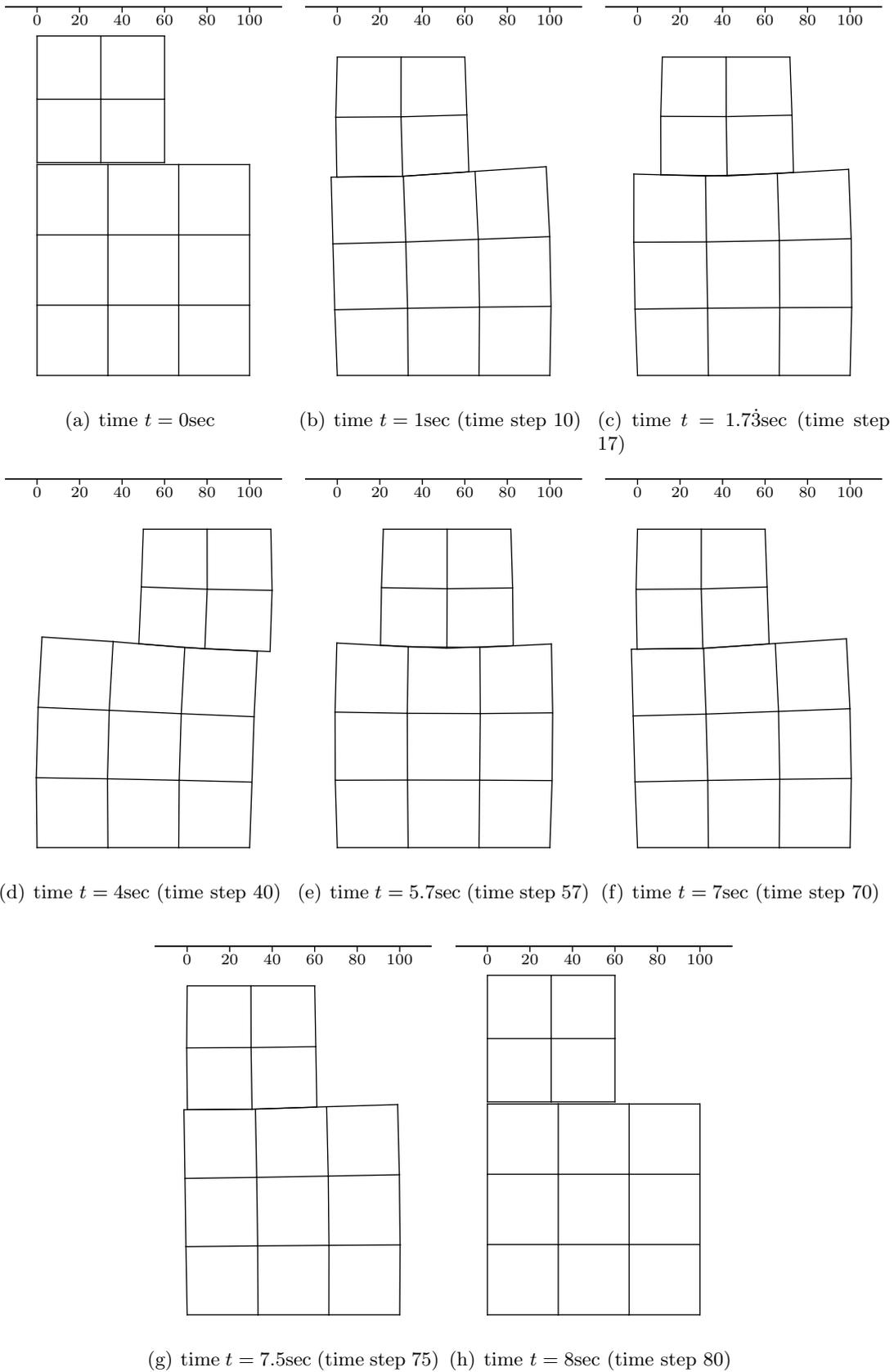
Figure 8.2.: The progress of the simple benchmark example without friction.

8. Numerical examples

(a) time $t = 0$sec   (b) time $t = 1$sec (time step 10)   (c) time $t = 1.7$sec (time step 17)

(d) time $t = 4$sec (time step 40)   (e) time $t = 5.7$sec (time step 57)   (f) time $t = 7$sec (time step 70)

(g) time $t = 7.5$sec (time step 75)   (h) time $t = 8$sec (time step 80)
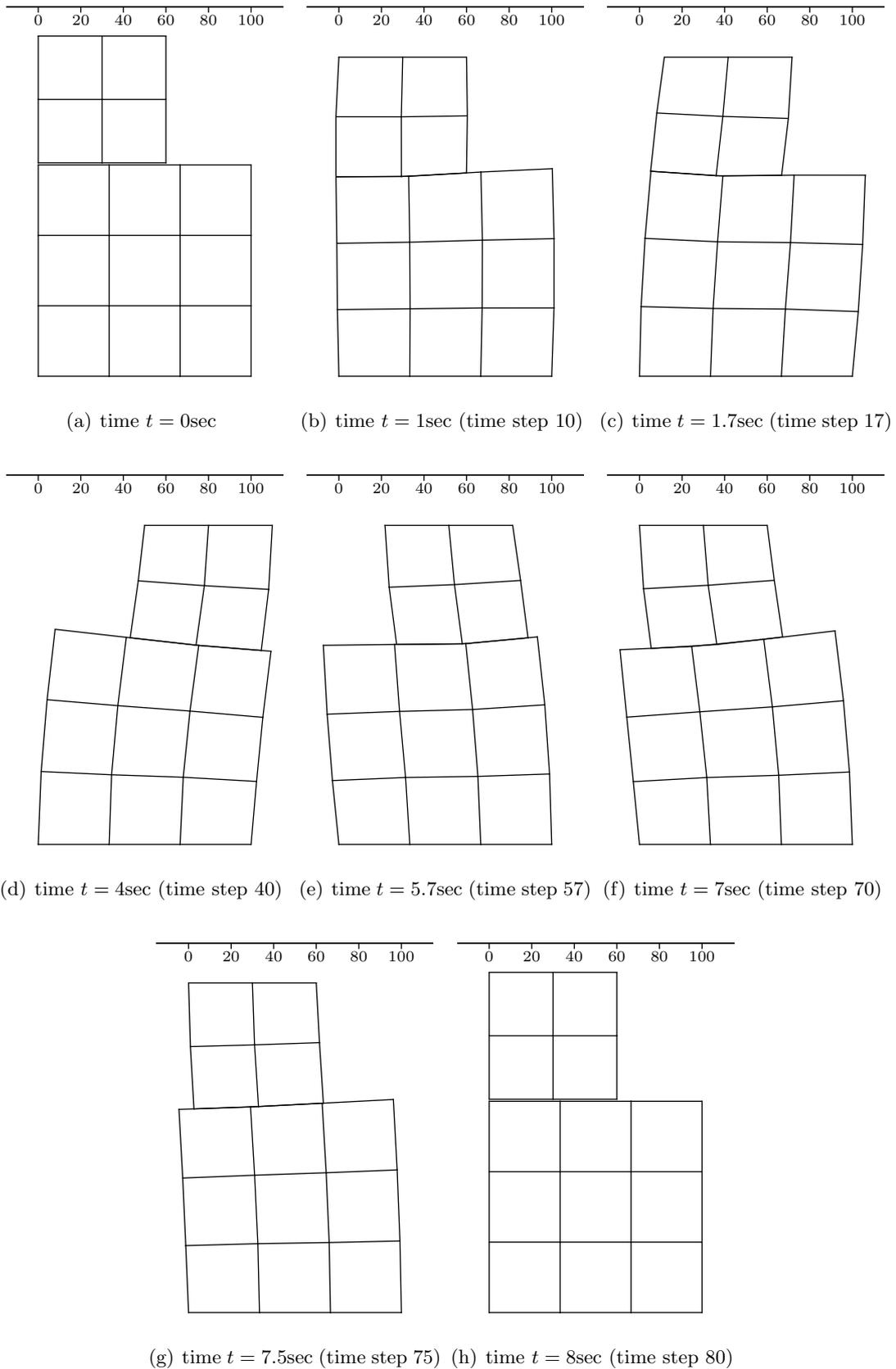
Figure 8.3.: The progress of the simple benchmark example for a Coulomb friction coefficient $\mu = 0.3$.

8.1.2. Reaction forces

In this section we provide plots of the reaction forces on top of the block. This reaction force is a sum of all internal forces of the nodes on the top boundary of the block where the Dirichlet boundary conditions are applied. To distinguish the different effects of friction we split the reaction force into a horizontal and a vertical part. As we will later see the differences in the result between the two normal vector fields are very small.

Frictionless case Firstly the frictionless case is analyzed. For the frictionless case the tangential traction $\mathbf{t}_T$ is zero. In the Lagrangian state the normal vector of the contact surface is vertical. This is not true for a later time step and for finite deformation. Due to the tilted position of the normal traction $\mathbf{t}_N$ a horizontal reaction force arises. This can be seen in Figure 8.4. The frictionless contact calculation is path independent. This means it does not matter how we achieved the current deformation state. The deformation states in time step 10 and time step 70 are identically (see Figure 8.2(b) and Figure 8.2(f)).
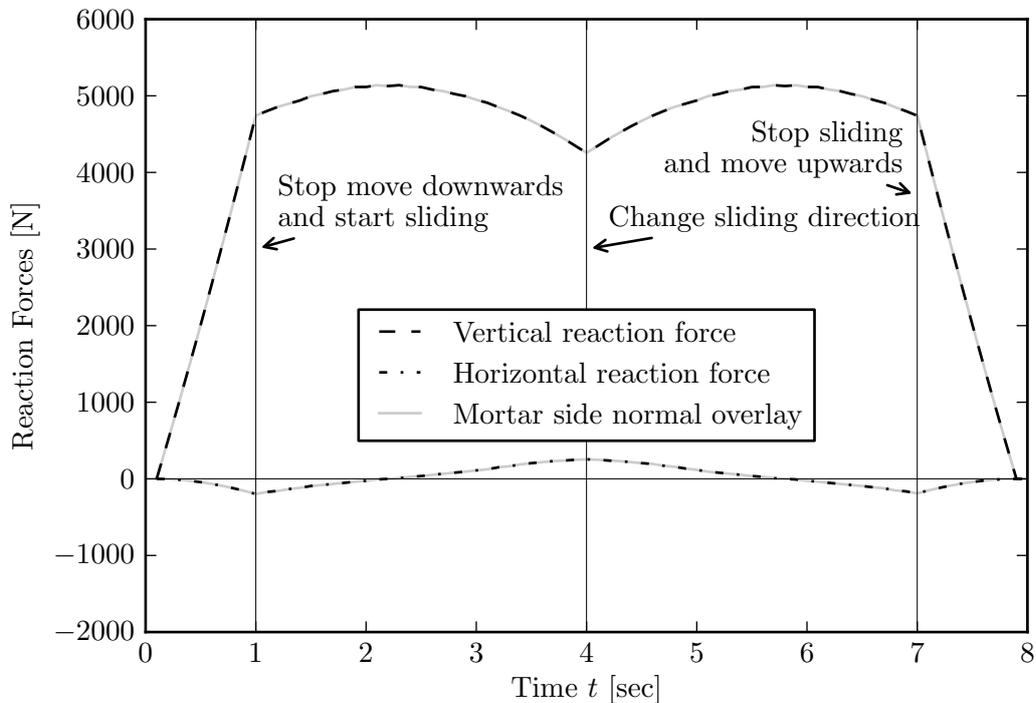
Figure 8.4.: Reaction forces on top side of block in horizontal and vertical direction for the frictionless case are shown. The reaction force in horizontal direction is not zero. This is due to the non symmetric finite deformation of the slab. This can be seen in Figure 8.2(b). Further one can observe the symmetry of the horizontal reaction forces due to the path independency of the frictionless calculation. The difference between the averaged non-mortar and the non-continuous mortar normal field are too small to be distinguished in the plots.

8. Numerical examples

Frictional Case Secondly we add friction to the simulation. Here we expect a horizontal reaction force due to the tangential traction $\mathbf{t}_T$ as a result of friction. The tangential traction $\mathbf{t}_T$ is correlated to the magnitude of the normal traction $\mathbf{t}_N$ by the Coulomb friction law (see Eq. (5.5)). This connection can bee seen in Figure 8.5. It is also notable that the reaction force is no longer path independent (this has already been discussed in Section 8.1.1). This means it differs for the motion towards the right side in comparison to the leftwards sliding. The maximum magnitude of the horizontal force for the motion towards right moving is -1440.56N and for the left moving 1389.06N. The absolute values differ because the changes from stick to slip are not exactly caught. One would have to do a refinement of the time steps near this time area to capture the switch from stick to slip more exactly.

Figure 8.5.: Reaction forces on top side of block in horizontal and vertical direction for the frictional case ($\mu = 0.3$) are shown. Once again the difference between the averaged non-mortar and the non-continuous mortar normal field are to small be distinguished in the plots. It can be seen that the horizontal reaction force is different for the leftwards and the rightwards motion. The correlation between the magnitude of the normal traction $\mathbf{t}_N$ and the tangential traction $\mathbf{t}_T$ is also well observable. It is also important to note that due to stick the deformation state and therefore also the horizontal reaction force is no longer path independent.

8.2. Deep ironing

This example is a very popular one in the field of contact mechanics. It can bee seen as an evolution of the simple benchmark example from Section 8.1. One can find a lot of results for comparison, e.g. in Yang and Laursen [43], Popp et al. [27], Gitterle et al. [10] and others. The idea behind this example is to show the performance of the algorithm for problems with finite deformations, large slidings and strong inconsistencies in discretization.

The dimensions of our problem are shown in Figure 8.6. The parameters (especially the material properties) are given in Table 8.4.

Young's modulus of block	E_{block}	1000	MPa
Poisson's ratio of block	ν_{block}	0.3	
Young's modulus of slab	E_{slab}	100	MPa
Poisson's ratio of slab	ν_{slab}	0.3	
Thickness of block and slab	h	1	mm
Vertical displacement	Δv	30	mm
Horizontal displacement 1	Δu_1	220	mm
Horizontal displacement 2	Δu_2	-110	mm

Table 8.4.: Parameters for the deep ironing example
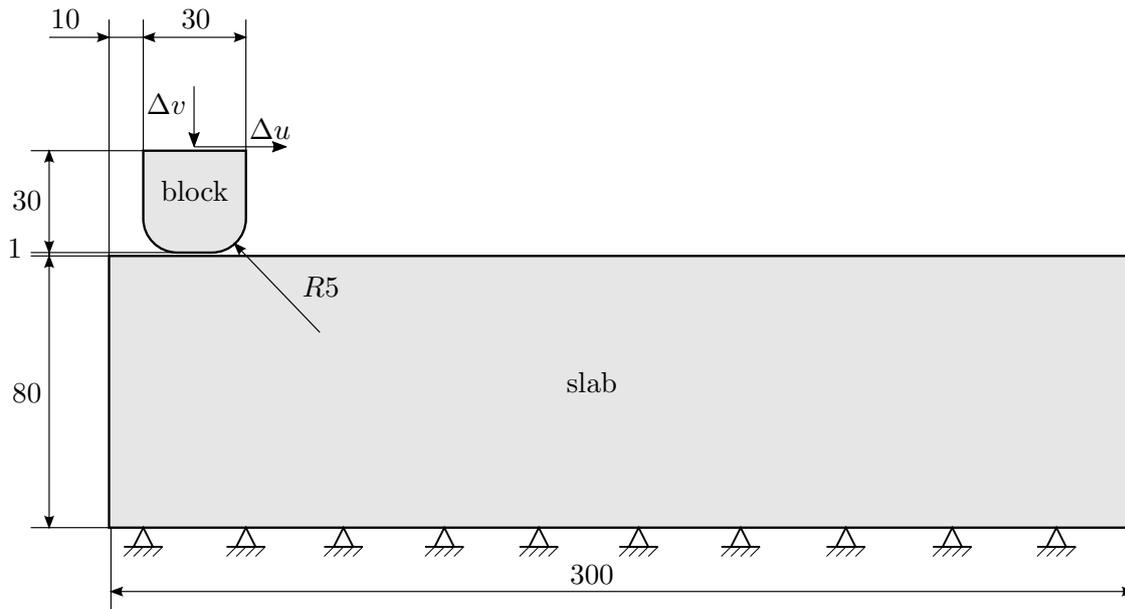
Figure 8.6.: The deep ironing problem. First the block is moved downwards in direction of Δv. The boundary condition is applied on the top boundary of the block. The block is pressed downwards until the top support of the block has reached the undeformed height of the slab (30mm for this example). Then the block is "ironed" over the slab. For this example we first moved the block in direction of Δu by Δu_1 and then moved backwards by Δv_2.

8.2.1. Progress

The progress is rather similar to the one of Section 8.1.1 and there are similar effects. We will only discuss the differences to the simple benchmark example here.

Frictionless case The deformation results shown in Figure 8.7 were calculated with non-continuous mortar side normals. As we will see in the following sections the differences between mortar and averaged non-mortar normals are really small. They can not be shown by overlaying, therefore we show these results only. We do a more elaborate comparison for the frictional case.

(a) time $t = 0$sec

(b) time $t = 1$sec

(c) time $t = 7.\dot{1}$sec

(d) time $t = 1.\dot{4}$sec

(e) time $t = 12.045$sec

(f) time $t = 16.0$sec
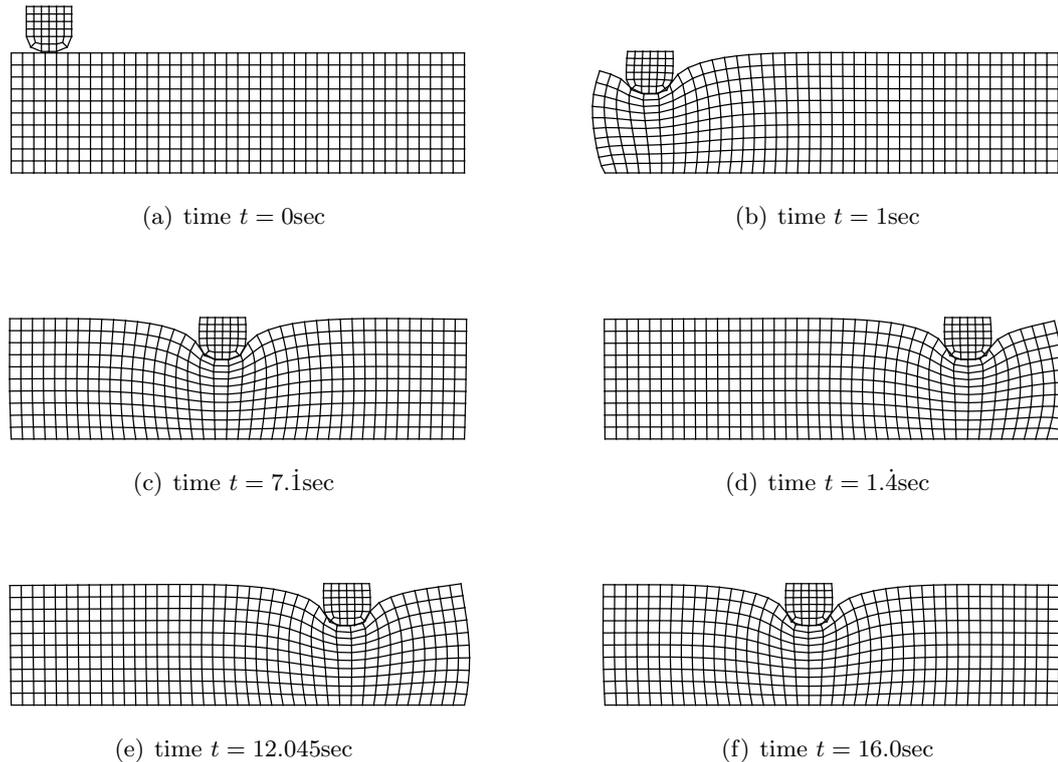
Figure 8.7.: The progress of the deep ironing example without friction. Non-continuous mortar side normals were used for the calculation of this example.

Frictional case In Figure 8.8 we compare the deformation of the non-continuous mortar side normal method with the averaged non-mortar side normal method. We do not lay the two deformations on top of each other, because the differences are that small that one could not distinguish them. We just wanted to clarify this fact by adding the comparison. The deformations are quite large and we have big changes of the active set throughout the calculation. In Yang and Laursen [43] the used block was not rounded. We have chosen to use the rounded version because we expected a higher influence of the averaging for this example. As we can see there is no influence on the quality of results (see also Section 8.2.2 for some further comparisons).

(a) time $t = 0$sec

(b) time $t = 0$sec

(c) time $t = 1$sec

(d) time $t = 1$sec

(e) time $t = 7.\dot{1}$sec

(f) time $t = 7.\dot{1}$sec

(g) time $t = 1.\dot{4}$sec

(h) time $t = 1.\dot{4}$sec

(i) time $t = 12.045$sec

(j) time $t = 12.045$sec

(k) time $t = 16.0$sec

(l) time $t = 16.0$sec
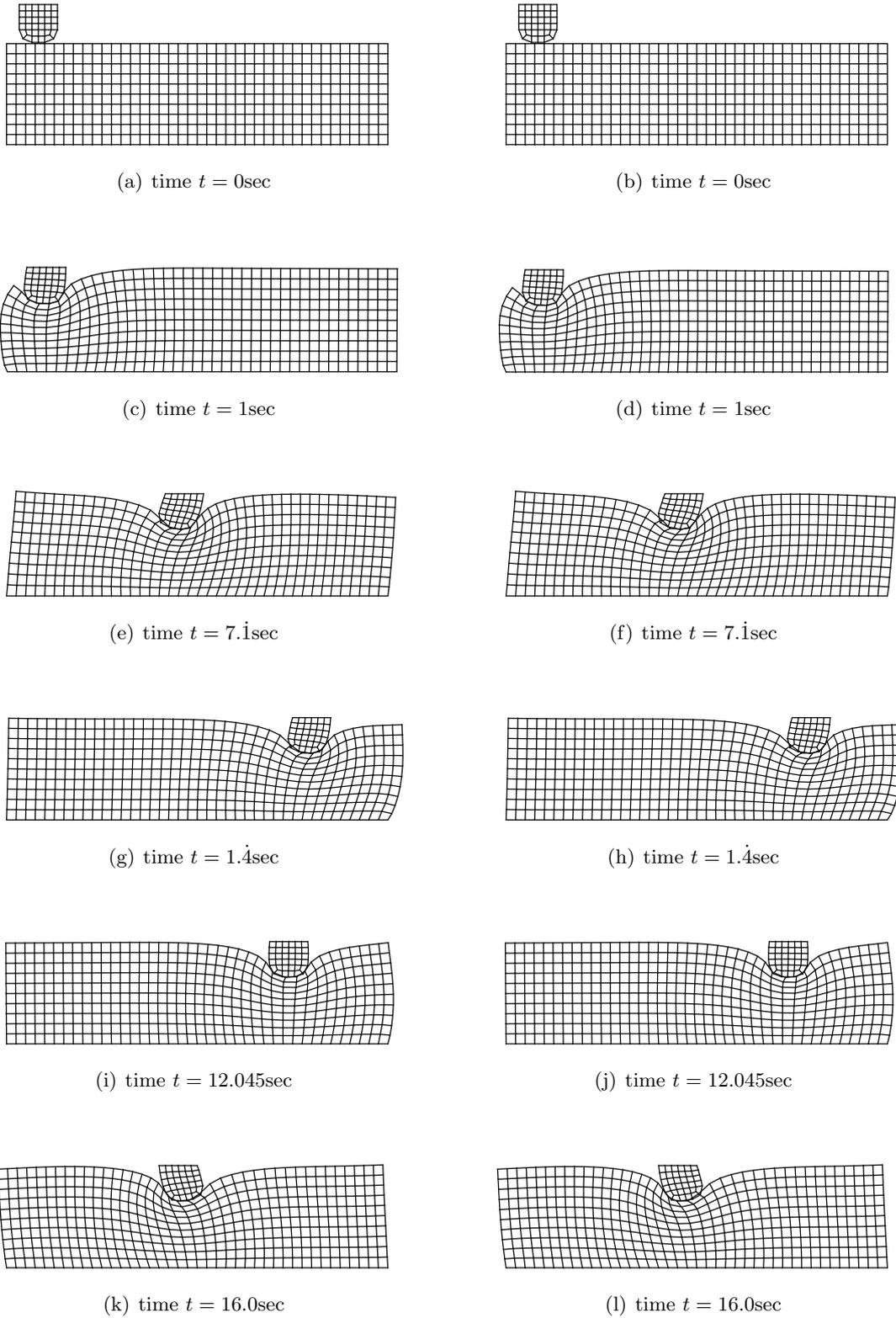
Figure 8.8.: The progress of the simple benchmark example with friction ($\mu = 0.3$) is shown. Each line shows the non-continuous mortar side normal method on the left and the averaged non-mortar side normal method on hte right.

8. Numerical examples

8.2.2. Reaction forces

Similar to the simple benchmark example in Section 8.1.2 we would like to analyze and compare the reaction forces on the top support of the block. Once again we split the overall reaction force for the top nodes into a horizontal and a vertical component. Of course those components should not be mixed up with the tangential and normal traction in the contact surface locally. This is a global view of the situation.

Frictionless case   In Figure 8.9 we compare the averaged non-mortar side normal method with the non-continuous mortar side normal method. The vertical reaction force for the averaged normal is rather smooth (shown in green) and there is no big difference to the mortar side normal (shown in the background with red color).

More interesting is the horizontal reaction force during the sliding of the block. One can see oscillating reaction forces. They occur only because of the non-continuity of the discrete surface. It is also important to note that the averaging of the normal field could not improve the results. This happens, because the averaging of the normal does not flatten the surface and the gap function g_N itself. It also cannot influence the stiffness changes due to the element jumps.

One has to note here, that those oscillations are getting rapidly smaller through refinement of the meshes. We used this rather coarse mesh by intention to show the effects.
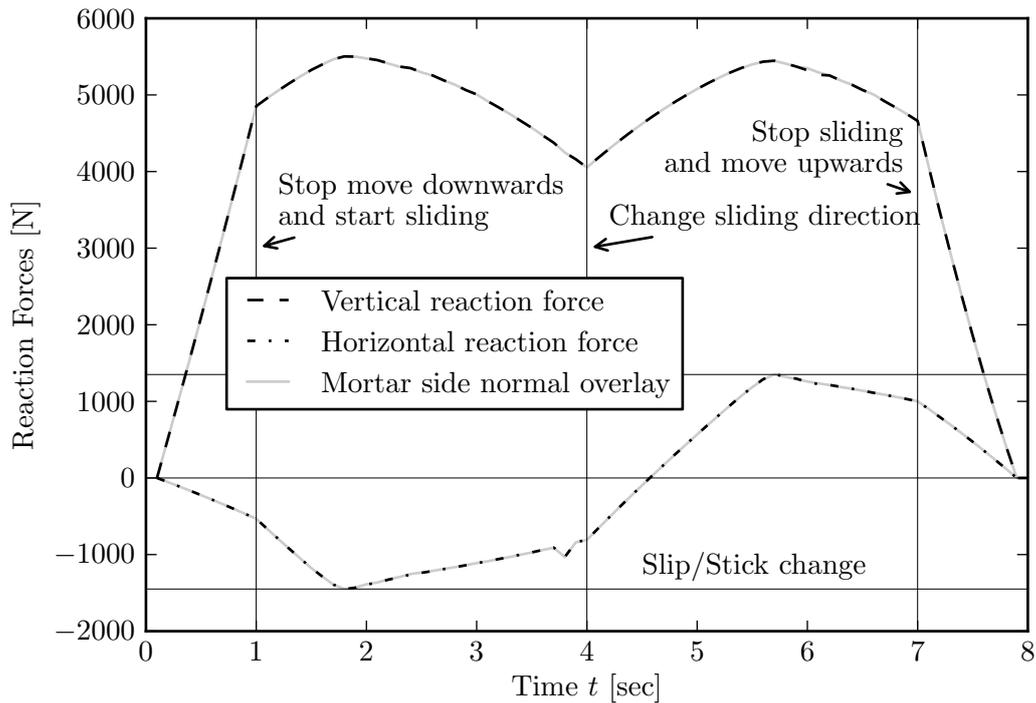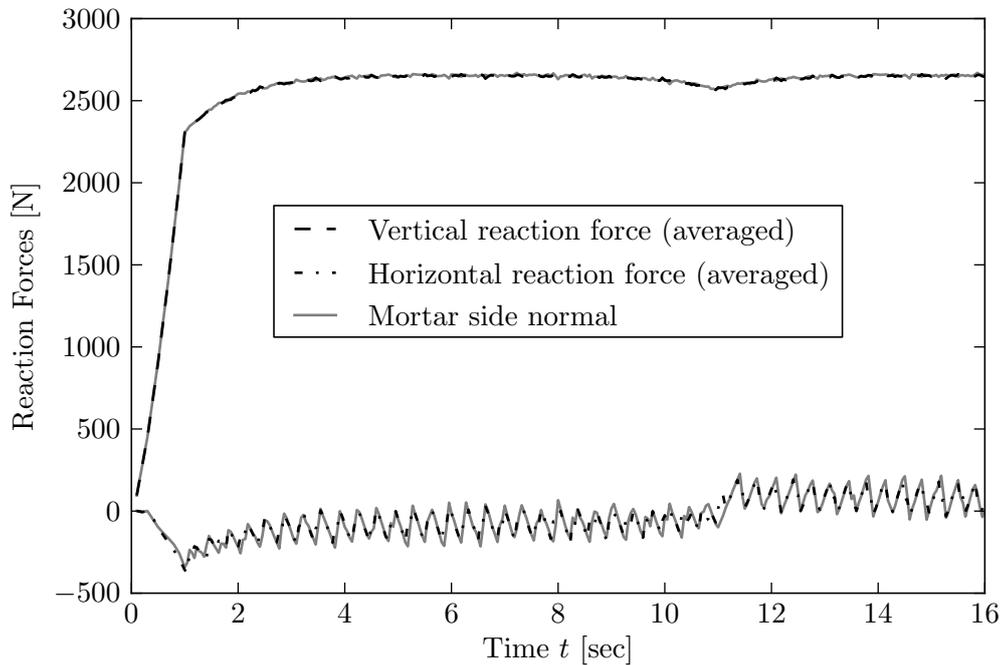

Figure 8.9.: Reaction forces on top side of block in horizontal and vertical direction for the frictionless case are shown.

Frictional case   We expect for the frictional case a rather similar behavior compared to frictionless case despite the fact that the horizontal reaction force is much higher. This can be seen in Figure 8.10.

Once again the difference between the two "normal" methods are negligible. There are only a few time steps where the red mortar side normal method is visible behind the averaged normal method laying on top.

The oscillation behavior for the frictional case is similar to the one of the frictionless case. This also means that the oscillations are not due to a slip/stick effect. They are only because of the non continuity of the surface. The reason for having larger magnitudes then for the frictionless case is the higher mean value of the horizontal force. Once again we can influence the height of the oscillations by refinement of the mesh.
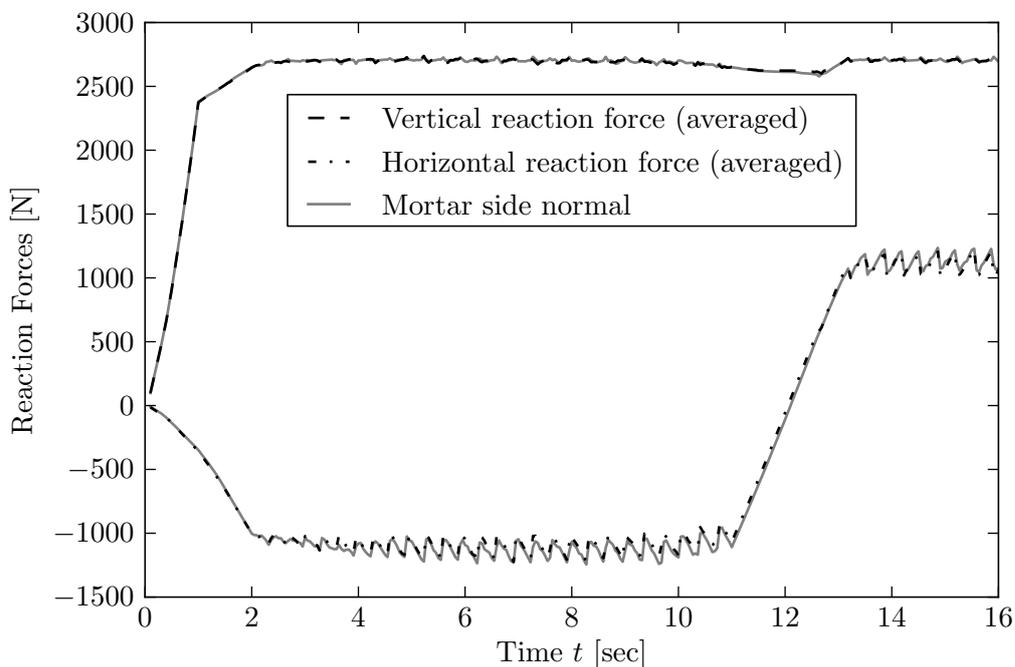
Figure 8.10.: Reaction forces on top side of block in horizontal and vertical direction for the frictional case ($\mu = 0.3$) are shown.

The oscillations (no matter if frictionless or frictional) are purely artificial due to the used numerical method and the needed finite element discretization. They can not be seen in real world experiments. It has to be a goal of future attempts in contact mechanics to get rid of those oscillating forces.

To show the effect of mesh refinement on the oscillations in tangential direction we solved the same problem with a finer mesh (see Figure 8.11 for some snapshots of the deformed mesh).

In Figure 8.12 we have once again plotted the reaction forces on the top support of the

(a) time $t = 0$sec

(b) time $t = 6$sec

(c) time $t = 11$sec

(d) time $t = 16$sec
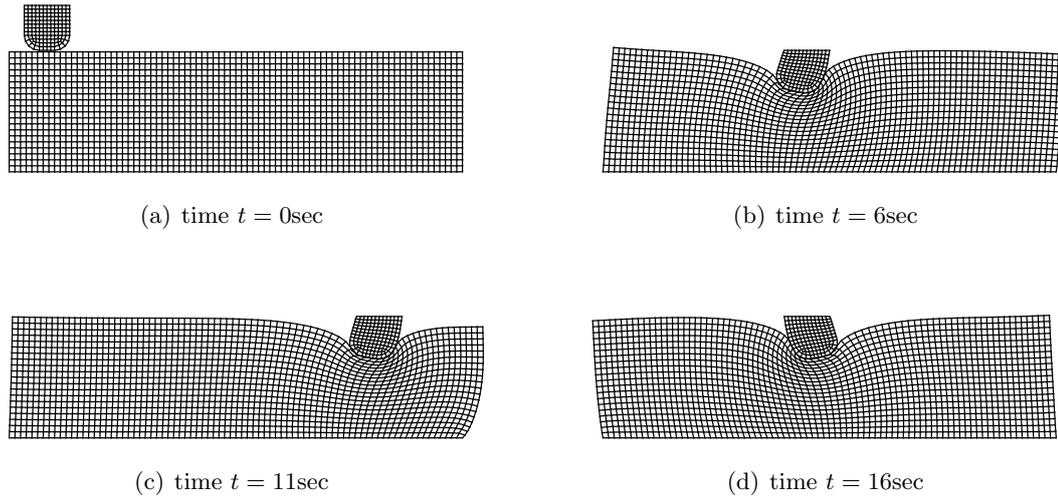
Figure 8.11.: The progress of the deep ironing example with friction ($\mu = 0.3$) is shown for a refined mesh.

block. It can be seen that the oscillation significantly decreases with mesh refinement if one compares the plot with Figure 8.10.
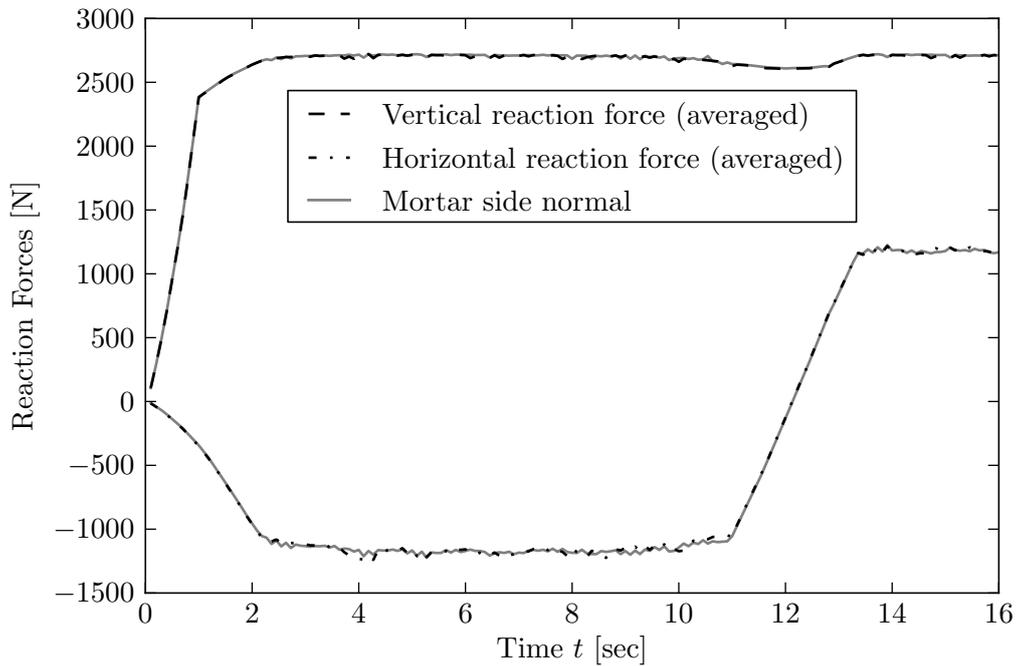
Figure 8.12.: Reaction forces on top side of block in horizontal and vertical direction for the frictional case ($\mu = 0.3$) on the refined mesh are shown.

8.3. Hertz problem

This is a very classical example in contact mechanics. The Hertz problem can be solved in an analytical manner for a general 3D contact of two paraboloids (see [11] for a detailed derivation of the analytical solution). We reduced the problem to a contact of an cylinder (Part 1) with a plane body (Part 2). The dimensions of our problem are shown in Figure 8.13.

As material we choose aluminum with the average parameters given in Table 8.5. For this example we used the St. Venant-Kirchhoff material (see Section 2.5.1) and not the compressible Neo-Hookean material. This was done because we expected this example to have small deformations and wanted to be as close as possible to the linear analytical solution.

Young's modulus	E_1, E_2	96	GPa
Poisson's ratio	ν_1, ν_2	0.35	
Vertical displacement	Δv	-0.015	mm
Penalty factor	ε	8.e5	

Table 8.5.: Parameters for the Hertz problem

Once again the loading is deformation controlled. For this example it would be possible to do force loading, as there are no global stick/slip changes. Therefore no negative slope of the force displacement curve can occur. Nevertheless the displacement based loading is a very robust method and is therefore used. The displacement of the top support in vertical direction can be easily connected with a corresponding loading by summing up the internal forces of the nodes on the top boundary. For our displacement $\Delta v = -0.015$ the resulting reaction force is $F = 206.44549\text{N}$

We are especially interested in the pressure distribution on the contact surface. For the special case of two cylinders with parallel axes coming into contact we can find in Goldsmith [11]

$$p(x) = \frac{2\,F}{\pi b^2}\sqrt{b^2 - x^2} \tag{8.1}$$

where x is the coordinate in tangential direction (the origin is in the symmetry line), F is the load applied on the two cylinders. b is the half-width of the contact zone and can be calculated with

$$b = 2\sqrt{\frac{F(\delta_1 + \delta_2)R_1 R_2}{R_1 + R_2}} \tag{8.2}$$

where R_1 and R_2 are the two radii of the cylinders being in contact. For the special case of a plain surface instead of cylinder 2 we can do $\lim_{R_2 \to \infty}(b)$ which yields

$$b = 2\sqrt{F(\delta_1 + \delta_2)R} \tag{8.3}$$

The stiffness values δ_i are given as

$$\delta_i = \frac{1 - \nu_i^2}{E_i \pi} \quad , \qquad i = 1, 2 \tag{8.4}$$
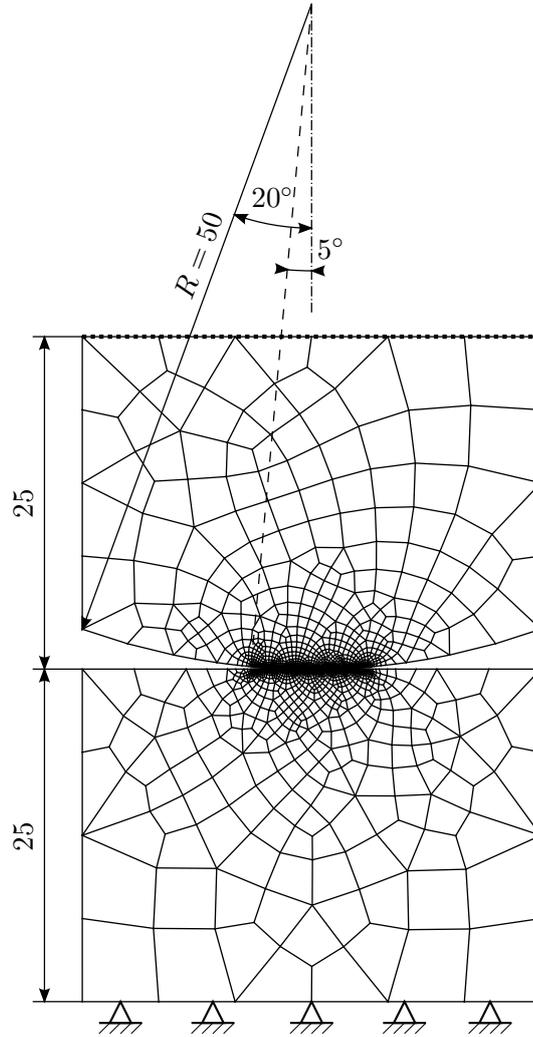
Figure 8.13.: The Hertz problem. The geometry is symmetric for the center plane, but not the mesh. In the area around the contact zone the mesh was refined. This refined surface area is defined for an angle of $[-5°, +5°]$. For this example we used an unstructured mesh as a comparison to the structured meshes of the previous examples. The top boundary (marked with a dashed line) is moved downwards during loading.

with Poisson's ratio ν_i and Young's modulus E_i.

The calculations were done frictionless. During vertical loading it is nearly impossible to create a stick condition with physically realistic properties (see also Kikuchi and Oden [20] for similar calculations). One would have to raise the friction coefficient to high values (e.g. $\nu = 0.7$ like in the example in Tur et al. [35]) and allow stresses in the body higher than a realistic elastic limit.

For this example we also compare the Lagrange contact enforcement with the penalty contact enforcement. We have done the calculations with the averaged non-mortar side

normal field and the non-continuous mortar side normal field. The differences for this small deformation example are that small that one cannot distinguish the results in the plot given in Figure 8.14. Therefore only a "Lagrange" solution (which is equivalent to the averaged **and** the mortar side normal field) is compared with the penalty solution.
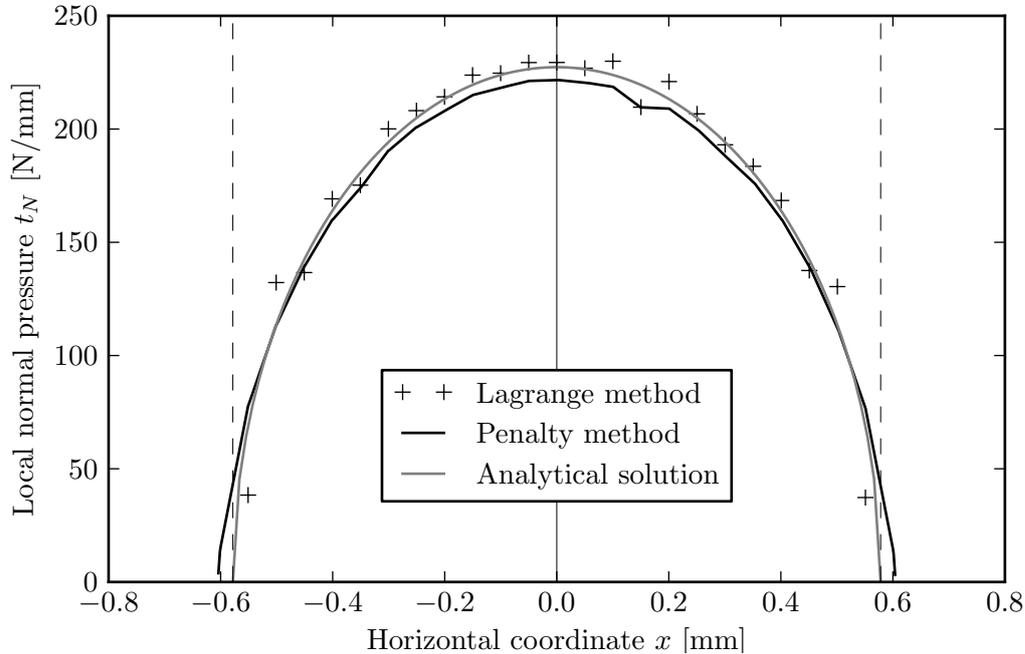
Figure 8.14.: The pressure distribution t_N. The origin is in the symmetry (vertical solid gray line) of the problem. The analytical half-width of the contact zone is $b = 0.5781716$ and drawn with a dashed gray line. The solution of the Lagrange contact enforcement is given point wise (in the nodes) with blue stars. The solution of the penalty contact enforcement is given as a solid green line. The penalty method provides us with a pressure in each integration point on the edge $t_N \approx \varepsilon g_N$. The integration points are much denser distributed and therefore a point wise plot would make no sense. The red solid line represents the analytical solution. Despite some discretization influences the numerical solutions fit the analytical one quite well.

8.4. Elastic ring

This example is based on that presented in Tur et al. [35]. For this problem big changes in the active set and finite deformations can be expected. The parameters can be found in Table 8.6.

Young's modulus of beam	E_{beam}	10000	MPa
Poisson's ratio of beam	ν_{beam}	0.3	
Young's modulus of slab	E_{slab}	300	MPa
Poisson's ratio of slab	ν_{slab}	0.3	
Thickness of beam and slab	h	1	mm
Vertical displacement	Δv	70	mm

Table 8.6.: Parameters for the elastic ring example

The geometrical dimension are shown in Figure 8.15.
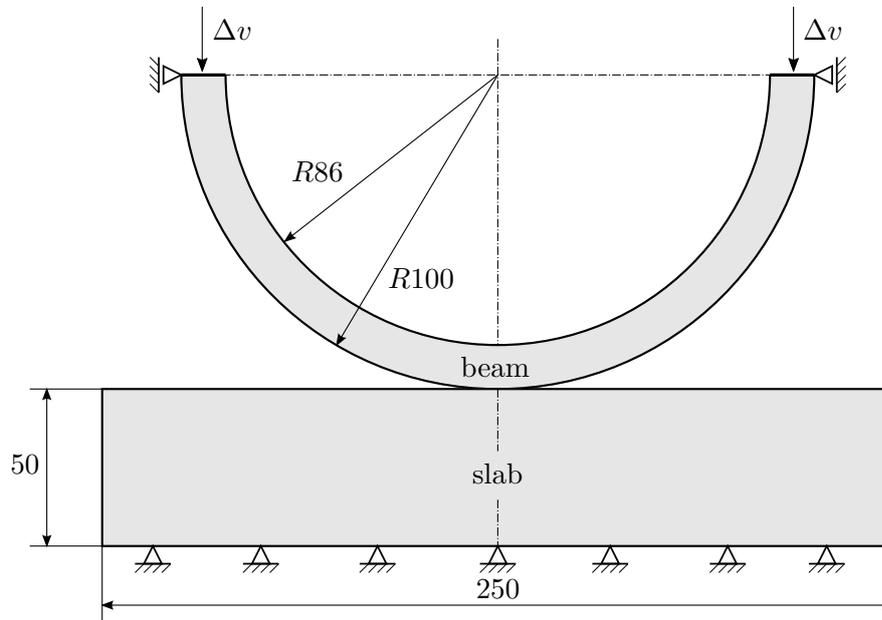
Figure 8.15.: The elastic ring problem. The ring beam is moved downwards the slab in direction Δv. The top endings of the ring beam are perfectly supported in horizontal direction. The slab is fixed on the lower boundary.

8.4.1. Progress and reaction forces

We calculated this example without friction and with a friction coefficient $\mu = 0.5$. The motion was calculated in 30 time steps and lasts 1sec. In this section we compare the two calculations.

Frictionless case For the frictionless case there is an interesting time stamp 23 at time 0.76̇sec (see Figure 8.16(c)) where the stiffness of the beam rapidly decreases. This can be seen in the plot (see Figure 8.17) of the reaction forces too. At this time the reaction forces start to decrease rapidly, we observe sone kind of a snap through point. We could go through this with force controlled loading only by application of a path-following algorithm or arc-length method like Riks method (e.g. Wriggers [41]).

(a) time $t = 0$sec

(b) time $t = 0.5$sec (time step 15)

(c) time $t = 0.76̇$sec (time step 23)

(d) time $t = 1.0$sec (time step 30)
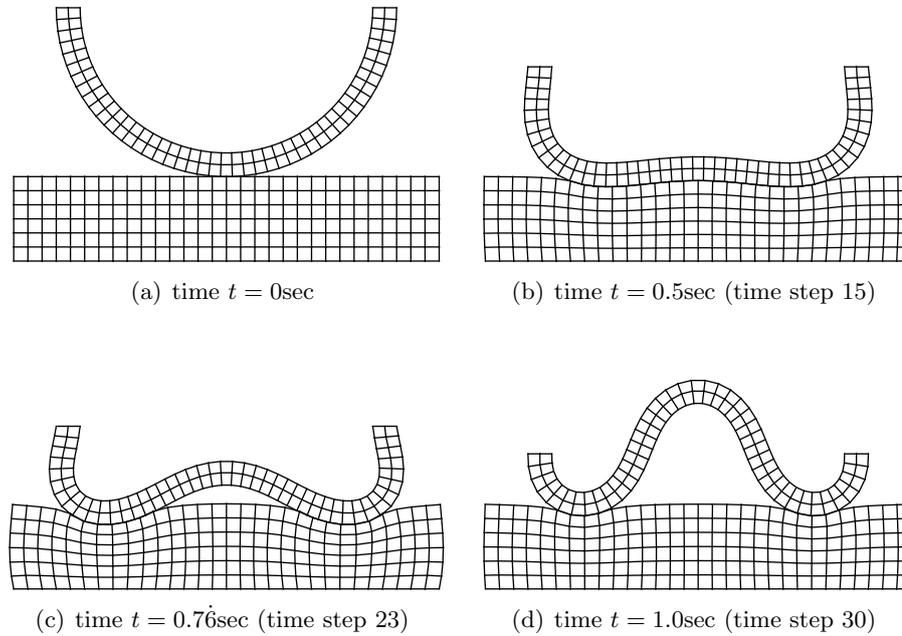
Figure 8.16.: The progress of the deformed meshes of the elastic ring example without friction.
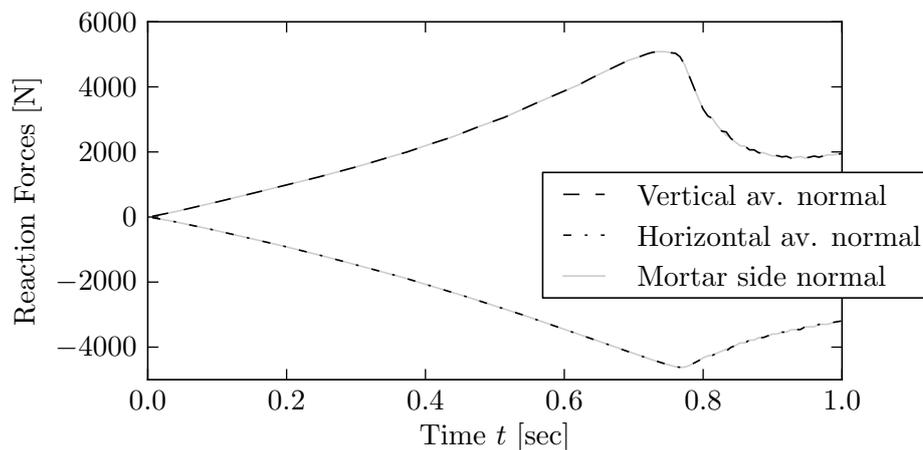
Figure 8.17.: Reaction forces on left support of the elastic ring in horizontal and vertical direction for the frictionless case.

Frictional case The deformed bodies can be seen in the progress Figure 8.18. The greatest difference to Figure 8.16 can be seen in the last plot Figure 8.18(d). The friction leads to a strong horizontal deformation of the slab. This sticking of the elastic ring on the slab leads to a stabilization of the snap through behavior of the frictionless case. This means in Figure 8.19 we can see no decrease in stiffness anymore.

(a) time $t = 0$sec

(b) time $t = 0.5$sec (time step 15)

(c) time $t = 0.7\dot{6}$sec (time step 23)

(d) time $t = 1.0$sec (time step 30)
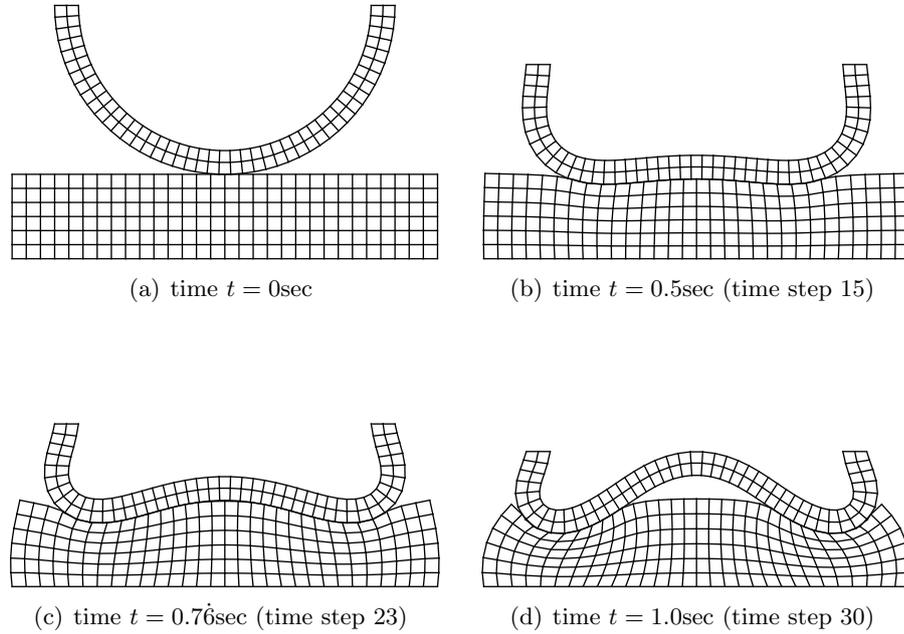
Figure 8.18.: The progress of the deformed meshes of the elastic ring example with friction ($\mu = 0.5$).
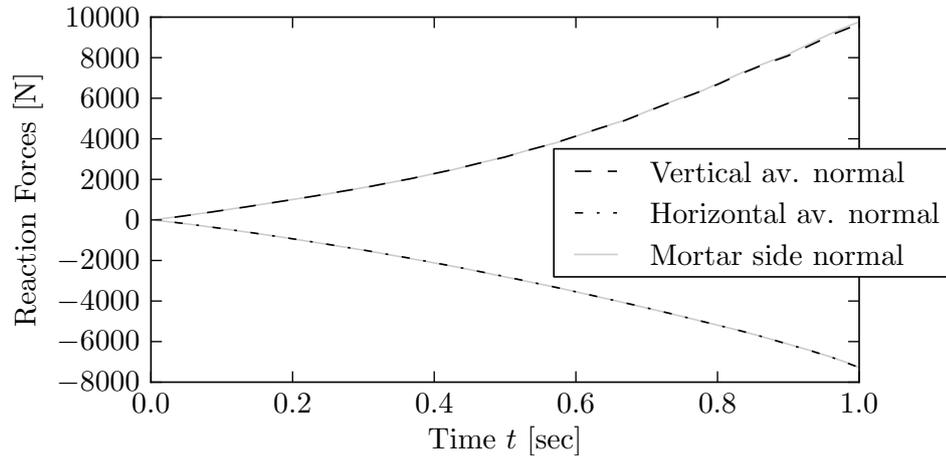
Figure 8.19.: Reaction forces on top side of block in horizontal and vertical direction for the frictional case ($\mu = 0.5$).

9. Conclusion

In this thesis we implemented the 2D mortar method and applied it to contact problems in the field of structural mechanics, with finite deformations and large slidings. The contact enforcement was realized first as penalty regularized condition. As this is not standard for the mortar method we then implemented the Lagrangian contact enforcement. For the Lagrangian contact enforcement the tangential contact was also treated. We implemented Coulomb's friction law which implied a distinction between the stick and the slip state.

The contact kinematics was described in a rather synthetic way. This enabled us to switch between an averaged continuous non-mortar side normal field and a non-continuous mortar side normal field. As we showed there is no remarkable difference in the results between the two methods. This also means one may omit the averaging if one uses the concentrated integration scheme as we did. The benefit is faster calculation and a more concise mathematical formulation.

We presented the algorithms and special cases in quite detail. We also tried to close the gap between different approaches in the engineering community to the mortar method in the field of contact mechanics. Further a great attention was given to closed derivations of the needed equations. Therefore no secondary literature should be needed to comprehend them.

9.1. Outlook and future work

The presented algorithm should be extended to three dimensional problems. As the concentrated integration scheme is easier to handle then the segmentation scheme, this should be worth a try.

As the examples in this work have shown, the influence of the normal averaging on the quality of solution is negligible. The problem is, that the influence of the non-continuity in the surface, has a much greater influence then the non-continuity of the normal field. Therefore some kind of surface smoothing should enhance the quality of the solution, especially the oscillation behavior. This has been done by various people with different methods and can be integrated in our algorithm.

It would be advantageous to condensate the Lagrange multipliers out of the set of linear equations. This has already been done for the segmented integration scheme and should be transferred to the concentrated scheme.

The whole analysis was done as a quasi-static analysis. There have not been considered any inertia effects. It would be a great enhancement to include the mass matrix and implement

an appropriate time integrator. As various works on this topic have shown, it is challenging to do this time integration in an energy preserving way.

For real technical applications the thermo-mechanical problem is of interest (e.g. any kind of metal forming process). For this problem, the heat transfer problem inside the domain, but also over the contact surface, has to be solved and a thermo-mechanical coupling is necessary. The dissipated energy by friction would lead to a heating process of the bodies.

Finally it is still a challenge to select which body is the master (mortar) and which one is the slave (non-mortar). This choice might seam easy if one decides the non-mortar surface to be the one with the finer discretization, although this is hard to detect in general. The bigger issue is that the solution of the problem is not invariant to the choice, so that makes no sense for the physical problem. In reality bodies in contact do not distinguish between a master and a slave. They are always equal contact partners. Therefore if we want to represent the physical problem with our mathematical model it is indispensable to eliminate the dependency on the master/slave choice.

A. Linearization

A.1. Linearization of contact surface Jacobian

As already presented in Section 3.4 we need the Jacobian of the contact surface in the current configuration. The Eq. (3.31) can be written in index notation as follows

$$\Delta J^{(1)} = \left\{ \frac{a_k^{(1)}}{J^{(1)}} \left(-N_l^{me,\xi^{(1)}} \right) \right\} \Delta \hat{u}_{kl}$$
$$= \{ \mathcal{J} \}_{kl} \, \Delta \hat{u}_{kl} \tag{A.1}$$

It is important to note the -1 in front of $N_l^{me,\xi^{(1)}}$ because we do not calculate the gap function here but the first derivative but we also want to reuse the already defined $\mathbf{N}^{me}$. One can of course use $\{ \mathcal{J} \}_{ij}$ to calculate the variation $\delta J^{(1)}$ and the increment $\Delta J^{(1)}$. The increment of the virtual Jacobian leads to

$$\Delta \left(\delta J^{(1)} \right) = \delta \mathbf{u}_{,\xi^{(1)}}^{(1)} \cdot \frac{1}{J^{(1)}} \left(\underline{\mathbf{1}} - \frac{1}{J^{(1)2}} \mathbf{x}_{,\xi^{(1)}}^{(1)} \otimes \mathbf{x}_{,\xi^{(1)}}^{(1)} \right) \cdot \Delta \mathbf{u}_{,\xi^{(1)}}^{(1)} \tag{A.2}$$

which can be written in index notation as

$$\Delta \left(\delta J^{(1)} \right) = \delta \hat{u}_{ij} \left\{ \left(-N_j^{me,\xi^{(1)}} \right) \frac{1}{J^{(1)}} \left(\delta_{ik} - \frac{a_i^{(1)} a_k^{(1)}}{J^{(1)2}} \right) \left(-N_l^{me,\xi^{(1)}} \right) \right\} \Delta \hat{u}_{kl}$$
$$= \delta \hat{u}_{ij} \left\{ \Delta \left(\delta \mathcal{J} \right) \right\}_{ijkl} \Delta \hat{u}_{kl} \tag{A.3}$$

The Jacobian variation, increment and variational increment is used for the mortar side normal field as well as the averaged normal field.

A.2. Non continuous mortar side normal

We have to implement the linearized contact virtual work Eq. (6.8). As we are talking about mortar side normal vectors throughout this Section A.2 we omit the ^{M} marker. All the presented values have to be calculated per integration point. As we are using the concentrated integration scheme we also omit any index or marker to reflect this, like $\square_{IP_\psi}$ in Section 6.5.2.

A.2.1. Normal gap function

The variation of the normal gap function and the normal gap function increment is given in index notation based on Eq. (4.7) as

$$\delta g_N = -\,n_i^{(2)}\,N_j^{me}\,\delta\hat{u}_{ij} \qquad\qquad \Delta g_N = -\,n_k^{(2)}\,N_l^{me}\,\Delta\hat{u}_{kl}$$
$$\delta g_N = \{\mathcal{G}_N\}_{ij}\,\delta\hat{u}_{ij} \qquad\qquad \Delta g_N = \{\mathcal{G}_N\}_{kl}\,\Delta\hat{u}_{kl} \tag{A.4}$$

A.2.1.1. Linearization of variation of normal gap function

At first we have to linearize the normal vector of Eq. (4.1) and obtain for the increment

$$\Delta\overline{\mathbf{n}}^{(2)} = \frac{\Delta\overline{\mathbf{a}}^{(2)}}{\|\overline{\mathbf{a}}^{(2)}\|}\left[\mathbf{1} - \frac{\overline{\mathbf{a}}^{(2)}\otimes\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)\,2}}\right]\times\mathbf{e}_3$$
$$= \frac{\Delta\overline{\mathbf{a}}^{(2)}}{\|\overline{\mathbf{a}}^{(2)}\|}\left[\frac{\overline{\mathbf{a}}^{(2)}\otimes\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)\,2}} + \overline{\mathbf{n}}^{(2)}\otimes\overline{\mathbf{n}}^{(2)} - \frac{\overline{\mathbf{a}}^{(2)}\otimes\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)\,2}}\right]\times\mathbf{e}_3$$
$$= -\frac{\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)\,2}}\left(\Delta\overline{\mathbf{a}}^{(2)}\cdot\overline{\mathbf{n}}^{(2)}\right) \tag{A.5}$$

which depends on the increment $\overline{\mathbf{a}}^{(2)}$

$$\Delta\overline{\mathbf{a}}^{(2)} = \Delta\left(\overline{\mathbf{x}}_{,\xi}^{(2)}\right) = \overline{\mathbf{x}}_{,\xi\xi}^{(2)}\,\Delta\xi^{(2)} + \Delta\overline{\mathbf{u}}_{,\xi}^{(2)} \tag{A.6}$$

We can write this in index notation (with $\overline{\mathbf{x}}_{,\xi\xi}^{(2)} = \overline{\mathbf{b}}^{(2)}$)

$$\Delta\overline{a}_i^{(2)} = \overline{b}_i^{(2)}\Delta\xi^{(2)} + N_j^{me,\xi\,(2)}\Delta\hat{u}_{ij}^{(2)} \tag{A.7}$$

The incremental tangent vector can be inserted to get

$$\Delta\overline{\mathbf{n}}^{(2)} = -\frac{\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)\,2}}\left[\left(\overline{\mathbf{x}}_{,\xi\xi}^{(2)}\,\Delta\xi^{(2)} + \Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\right)\cdot\overline{\mathbf{n}}^{(2)}\right] \tag{A.8}$$

Second we have to calculate the increment $\Delta\xi^{(2)}$. Based on Eq. (3.5) we replace δ with Δ and get

$$\Delta\overline{\mathbf{u}}^{(2)} + \overline{\mathbf{a}}^{(2)}\Delta\xi^{(2)} = \Delta\mathbf{u}^{(1)} - \Delta g_N\,\overline{\mathbf{n}}^{(2)} - g_N\,\Delta\overline{\mathbf{n}}^{(2)} \tag{A.9}$$

which is multiplied with $\overline{\mathbf{a}}^{(2)}$, inserting Eq. (A.8) yields

$$\overline{\alpha}^{(2)\,2}\,\Delta\xi^{(2)} = -\left(\Delta\overline{\mathbf{u}}^{(2)} - \Delta\mathbf{u}^{(1)}\right)\cdot\overline{\mathbf{a}}^{(2)} + g_N\left(\overline{\mathbf{x}}_{,\xi\xi}^{(2)}\,\Delta\xi^{(2)} + \Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\right)\cdot\overline{\mathbf{n}}^{(2)}$$
$$\left(\overline{\alpha}^{(2)\,2} - g_N\,\overline{\beta}^{(2)}\right)\Delta\xi^{(2)} = -\left(\Delta\overline{\mathbf{u}}^{(2)} - \Delta\mathbf{u}^{(1)}\right)\cdot\overline{\mathbf{a}}^{(2)} + g_N\,\Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\cdot\overline{\mathbf{n}}^{(2)}$$

$$\Delta\xi^{(2)} = \frac{1}{\overline{\alpha}^{(2)\,2} - g_N\,\overline{\beta}^{(2)}}\left[-\left(\Delta\overline{\mathbf{u}}^{(2)} - \Delta\mathbf{u}^{(1)}\right)\cdot\overline{\mathbf{a}}^{(2)} + g_N\,\Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\cdot\overline{\mathbf{n}}^{(2)}\right] \tag{A.10}$$

This is the same result as we got in Eq. (4.10) with $\delta \to \Delta$. Transferring the expression into index notation yields

$$\Delta \xi^{(2)} = \left\{ \frac{1}{\overline{\alpha}^{(2)^2} - g_N \, \overline{\beta}^{(2)}} \left[-N_l^{me} \, \overline{a}_k^{(2)} + g_N \, N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right] \right\} \Delta \hat{u}_{kl}$$

$$\Delta \xi^{(2)} = \{\mathcal{X}\}_{kl} \, \Delta \hat{u}_{kl} \tag{A.11}$$

with $N_j^{me,\xi^{(2)}}$ representing the derivation of the shape tensor with respect to $\xi^{(2)}$

$$\mathbf{N}^{me,\xi^{(2)}} = \left[\begin{array}{cc|cc} 0 & 0 & \frac{\partial N_1^{(2)}}{\partial \xi^{(2)}} & \frac{\partial N_2^{(2)}}{\partial \xi^{(2)}} \end{array} \right] \to N_i^{me,\xi^{(2)}}$$

$N_j^{me,\xi^{(2)}}$ is zero for the non-mortar side shape functions $j = 1, 2$. One can replace $\Delta \to \delta$ and obtain

$$\delta \xi^{(2)} = \{\mathcal{X}\}_{ij} \, \delta \hat{u}_{ij} \tag{A.12}$$

With Eq. (A.8) and Eq. (A.10) we can now calculate the virtual gap function increment by linearization of Eq. (4.7)

$$\Delta \left(\delta g_N \right) = -\Delta \overline{\mathbf{n}}^{(2)} \cdot \left(\delta \overline{\mathbf{u}}^{(2)} - \delta \mathbf{u}^{(1)} \right) - \overline{\mathbf{n}}^{(2)} \cdot \delta \overline{\mathbf{u}}_{,\xi}^{(2)} \, \Delta \xi^{(2)} \tag{A.13}$$

We write Eq. (A.8) in index notation

$$\Delta \overline{n}_i^{(2)} = -\frac{\overline{a}_i^{(2)}}{\overline{\alpha}^{(2)^2}} \left(\overline{x}_{,\xi \xi_k}^{(2)} \, \Delta \xi^{(2)} + N_l^{me,\xi^{(2)}} \, \Delta \hat{u}_{kl} \right) \overline{n}_k^{(2)}$$

$$= -\left\{ \frac{\overline{a}_i^{(2)}}{\overline{\alpha}^{(2)^2}} \left(\frac{\overline{\beta}^{(2)}}{\overline{\alpha}^{(2)^2} - g_N \, \overline{\beta}^{(2)}} \left(-N_l^{me} \, \overline{a}_k^{(2)} + g_N \, N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right) + N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right) \right\} \Delta \hat{u}_{kl}$$

$$= \{\mathcal{N}\}_{kli} \, \Delta \hat{u}_{kl}$$

Finally for the variational normal gap function increment (based on Eq. (A.13)) in index notation we make use of Eq. (A.8) and Eq. (A.10) and get

$$\Delta \left(\delta g_N \right) = -\Delta \overline{n}_i^{(2)} \, N_j^{me} \, \delta \hat{u}_{ij} - \overline{n}_i^{(2)} \, N_j^{me,\xi^{(2)}} \, \delta \hat{u}_{ij} \, \Delta \xi^{(2)}$$

$$= -\delta \hat{u}_{ij} \left\{ -N_j^{me} \frac{\overline{a}_i^{(2)}}{\overline{\alpha}^{(2)^2}} \left[\frac{\overline{\beta}^{(2)}}{\overline{\alpha}^{(2)} - g_N \, \overline{\beta}^{(2)}} \left(-N_l^{me} \, \overline{a}_k^{(2)} + g_N \, N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right) + N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right] \right\} \Delta \hat{u}_{kl} -$$

$$- \delta \hat{u}_{ij} \left\{ \overline{n}_i^{(2)} \, N_j^{me,\xi^{(2)}} \frac{1}{\overline{\alpha}^{(2)^2} - g_N \, \overline{\beta}^{(2)}} \left[-N_l^{me} \, \overline{a}_k^{(2)} + g_N \, N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right] \right\} \Delta \hat{u}_{kl}$$

Factorize $\frac{1}{\overline{\alpha}^{(2)^2} - g_N \, \overline{\beta}^{(2)}}$, reorder terms and collect to visualize symmetry

$$\Delta \left(\delta g_N \right) = \delta \hat{u}_{ij} \left\{ \left(-\overline{a}_i^{(2)} \, N_j^{me} \quad \frac{\overline{\beta}^{(2)}}{\overline{\alpha}^{(2)}} \, N_l^{me} \, \overline{a}_k^{(2)} \right. \right.$$

$$+ \overline{a}_i^{(2)} \, N_j^{me} \quad N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)}$$

$$+ \overline{n}_i^{(2)} \, N_j^{me,\xi^{(2)}} \quad N_l^{me} \, \overline{a}_k^{(2)}$$

$$\left. \left. - \overline{n}_i^{(2)} \, N_j^{me,\xi^{(2)}} \quad g_N \, N_l^{me,\xi^{(2)}} \, \overline{n}_k^{(2)} \right) \frac{1}{\overline{\alpha}^{(2)^2} - g_N \, \overline{\beta}^{(2)}} \right\} \Delta \hat{u}_{kl}$$

The term in the curly brackets is separately named as $\{\Delta\left(\delta\mathcal{G}_N\right)\}_{ijkl}$. This is an index object of fourth order. It is not a classical fourth order tensor as the indices j and l denote the nodal index. Only i and k are "tensor" indices as they denote the coordinate direction.

$$\Delta\left(\delta g_N\right) = \delta\hat{u}_{ij}\left\{\Delta\left(\delta\mathcal{G}_N\right)\right\}_{ijkl}\Delta\hat{u}_{kl} \tag{A.14}$$

This is a very central property for defining the stiffness matrix.

A.2.2. Tangential slip

We need δg_T and Δg_T respectively. It has been shown in Eq. (6.10) that these quantities are equivalent and they can be written in index notation (based on Eq. (4.8)) as

$$\delta g_T = \overline{\alpha}^{(2)}\delta\xi^{(2)} = \overline{\alpha}^{(2)}\left\{\mathcal{X}\right\}_{ij}\delta\hat{u}_{ij} \qquad\qquad \Delta g_T = \overline{\alpha}^{(2)}\left\{\mathcal{X}\right\}_{kl}\Delta\hat{u}_{kl}$$

$$\delta g_T = \left\{\mathcal{G}_T\right\}_{ij}\delta\hat{u}_{ij} \qquad\qquad\qquad\qquad \Delta g_T = \left\{\mathcal{G}_T\right\}_{kl}\Delta\hat{u}_{kl} \tag{A.15}$$

A.2.2.1. Linearization of variation of tangential slip

We obtain the variational tangential slip increment based on Eq. (A.15)

$$\Delta\left(\delta g_T\right) = \frac{\Delta\overline{\mathbf{a}}^{(2)}\cdot\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)}}\delta\xi^{(2)} + \overline{\alpha}^{(2)}\Delta\left(\delta\xi^{(2)}\right)$$

$$= \frac{\left(\overline{\mathbf{x}}_{,\xi\xi}^{(2)}\,\Delta\xi^{(2)} + \Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\right)\cdot\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)}}\delta\xi^{(2)} + \overline{\alpha}^{(2)}\Delta\left(\delta\xi^{(2)}\right)$$

$$= \delta\xi^{(2)}\frac{\Delta\overline{\mathbf{u}}_{,\xi}^{(2)}\cdot\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)}} + \delta\xi^{(2)}\frac{\overline{\mathbf{x}}_{,\xi\xi}^{(2)}\cdot\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)}}\Delta\xi^{(2)} + \overline{\alpha}^{(2)}\Delta\left(\delta\xi^{(2)}\right) \tag{A.16}$$

where the first term introduces an (already mentioned - see Section 3.3 and explanations to Eq. (3.14)) "artificial" asymmetry. As numerical experiments have shown, the influence of this first term is rather small and can be neglected. For completeness we added the term. The whole frictional mortar procedure already requires a solver for the set of linear equations which is able to deal with asymmetry (especially the slip case is non symmetric any way).

To obtain $\Delta\left(\delta\xi^{(2)}\right)$ we use the variational contact kinematic from Eq. (3.5) and do the linearization to get

$$\mathbf{N}^{me,\xi^{(2)}}\delta\hat{\mathbf{u}}\,\Delta\xi^{(2)} + \mathbf{N}^{me,\xi^{(2)}}\Delta\hat{\mathbf{u}}\,\delta\xi^{(2)} + \overline{\mathbf{a}}^{(2)}\Delta\left(\delta\xi^{(2)}\right) + \overline{\mathbf{b}}^{(2)}\Delta\xi^{(2)}\delta\xi^{(2)} =$$

$$-\Delta\left(\delta g_N\right)\overline{\mathbf{n}}^{(2)} - \delta g_N\Delta\overline{\mathbf{n}}^{(2)} - \Delta g_N\delta\overline{\mathbf{n}}^{(2)} - g_N\Delta\left(\delta\overline{\mathbf{n}}^{(2)}\right) \tag{A.17}$$

$\overline{\mathbf{b}}^{(2)}$ is the second derivative of $\mathbf{x}^{(2)}$ with respect to $\xi^{(2)}$.

$$\overline{\mathbf{b}}^{(2)} = \mathbf{x}_{,\xi^{(2)}\xi^{(2)}}^{(2)} = \frac{\partial\mathbf{x}^{(2)^2}}{\partial^2\xi^{(2)}} \tag{A.18}$$

> **Remark:** $\overline{\mathbf{b}}^{(2)}$ equals zero for linear shape functions!

We multiply this equation with $\overline{\mathbf{a}}^{(2)}$ and obtain after some reordering

$$\Delta\left(\delta\xi^{(2)}\right) = \frac{\overline{\mathbf{a}}^{(2)}}{\overline{\alpha}^{(2)}}\cdot\left[-\delta g_N\Delta\overline{\mathbf{n}}^{(2)} - \Delta g_N\delta\overline{\mathbf{n}}^{(2)} - g_N\Delta\left(\delta\overline{\mathbf{n}}^{(2)}\right) - \Delta\xi^{(2)}\overline{\mathbf{b}}^{(2)}\delta\xi^{(2)} - \delta\overline{\mathbf{u}}^{(2)}_{,\xi}\Delta\xi^{(2)} - \Delta\overline{\mathbf{u}}^{(2)}_{,\xi}\delta\xi^{(2)}\right]$$

(A.19)

which is fully symmetric. Once again we transform Eq. (A.19) into index notation

$$\Delta\left(\delta\xi^{(2)}\right) = \delta\hat{u}_{ij}\left\{\frac{\overline{a}^{(2)}_m}{\overline{\alpha}^{(2)}}\left[-\{\mathcal{G}_N\}_{ij}\{\mathcal{N}\}_{klm} - \{\mathcal{G}_N\}_{kl}\{\mathcal{N}\}_{ijm} - g_N\Delta\{\delta\left(\mathcal{N}\right)\}_{ijklm}\right.\right.$$
$$\left.\left. - \delta_{im}N^{me,\xi^{(2)}}_j\{\mathcal{X}\}_{kl} - \delta_{km}N^{me,\xi^{(2)}}_l\{\mathcal{X}\}_{ij}\right]\right\}\Delta\hat{u}_{kl} \quad \text{(A.20)}$$

$$\Delta\left(\delta\xi^{(2)}\right) = \delta\hat{u}_{ij}\{\Delta\left(\delta\mathcal{X}\right)\}_{ijkl}\Delta\hat{u}_{kl}$$

(A.21)

With these results we finally are able to formulate the variational tangential slip increment

$$\Delta\left(\delta g_T\right) = \delta\hat{u}_{ij}\left\{\{\mathcal{X}\}_{ij}N^{me,\xi^{(2)}}_l\overline{a}^{(2)}_k + \{\mathcal{X}\}_{ij}\overline{b}^{(2)}_m\overline{a}^{(2)}_m\{\mathcal{X}\}_{kl} + \sqrt{\overline{\alpha}^{(2)}}\{\Delta\left(\delta\mathcal{X}\right)\}_{ijkl}\right\}\Delta\hat{u}_{kl}$$
$$= \delta\hat{u}_{ij}\{\Delta\left(\mathcal{G}_T\right)\}_{ijkl}\Delta\hat{u}_{kl}$$

(A.22)

A.2.3. Linearized virtual contact work and weak contact conditions

We have now all properties needed to define the linearized virtual contact work and the linearized weak contact conditions. These terms are based on Eqs. (6.8), (6.11) and (6.13).

A.2.3.1. Normal direction

For the chosen concentrated integration scheme we have to evaluate all the values $\{\Delta\mathcal{G}_N\}_{ij}$, $\{\Delta\left(\delta\mathcal{G}_N\right)\}_{ijkl}$, $\{\delta\mathcal{J}\}_{ij}$ and $\{\Delta\left(\delta\mathcal{J}\right)\}_{ijkl}$ at the integration points. The integration itself is realized with a classical numerical Gaussian quadrature over the natural interval $\xi^{(1)} = [-1,+1]$.

$$\Delta\left(\delta^M\Pi_N\right) = \int_\square \begin{array}{ll} -\delta\hat{\lambda}_{N_j}\left\{ \phi_j \quad \left[J^{(1)}\{\mathcal{G}_N\}_{kl} + g_N\{\mathcal{J}_N\}_{kl}\right] \right\}\Delta\hat{u}_{kl} \\ -\delta\hat{u}_{ij}\left\{ \phi_l \quad \left[J^{(1)}\{\mathcal{G}_N\}_{ij} + g_N\{\mathcal{J}_N\}_{ij}\right] \right\}\Delta\hat{\lambda}_{N_l} \\ -\delta\hat{u}_{ij}\left\{ \lambda_N \left[\{\Delta\left(\delta\mathcal{G}_N\right)\}_{ijkl}J^{(1)} + \{\Delta\left(\delta\mathcal{J}\right)\}_{ijkl}g_N \right.\right. \\ \qquad\qquad\qquad \left.\left. + \{\mathcal{G}_N\}_{ij}\{\mathcal{J}\}_{kl} + \{\mathcal{J}\}_{ij}\{\mathcal{G}_N\}_{kl}\right] \right\}\Delta\hat{u}_{kl}\,\mathrm{d}\gamma_r \end{array}$$

(A.23)

One can see the symmetry of all operators. It is part of the assembly operator to add the three and four dimensional objects into the stiffness matrix.

A.2.3.2. Tangential direction

In tangential direction we have to differ between the stick case (with an additional unknown λ_T) and the slip case where the tangential traction t_T is a result of tangential constitutive equation given in Eq. (5.5).

Stick case

$$
\begin{aligned}
\Delta\left(\delta^M \Pi_{CT_{ST}}\right) = \int_\square \quad &-\delta\hat{\lambda}_{T_j} \left\{ \quad \phi_j \qquad \left[J^{(1)} \{\mathcal{G}_T\}_{kl} + \Delta^t g_T \{\mathcal{J}\}_{kl} \right] \qquad \right\} \Delta\hat{u}_{kl} \\
&-\delta\hat{u}_{ij} \left\{ \quad \phi_l \qquad \left[J^{(1)} \{\mathcal{G}_T\}_{ij} + \Delta^t g_T \{\mathcal{J}\}_{ij} \right] \qquad \right\} \Delta\hat{\lambda}_{T_l} \\
&-\delta\hat{u}_{ij} \left\{ \quad \lambda_T \quad \left[\{\Delta\left(\delta\mathcal{G}_T\right)\}_{ijkl} J^{(1)} + \{\Delta\left(\delta\mathcal{J}\right)\}_{ijkl} \Delta^t g_T \right.\right. \\
&\qquad\qquad\qquad \left.\left. + \{\mathcal{G}_T\}_{ij} \{\mathcal{J}\}_{kl} + \{\mathcal{J}\}_{ij} \{\mathcal{G}_T\}_{kl} \right] \qquad \right\} \Delta\hat{u}_{kl} \, \mathrm{d}\gamma_r
\end{aligned}
\tag{A.24}
$$

Slip case

$$
\begin{aligned}
\Delta\left(\delta^M \Pi_{CT_{SL}}\right) = \int_\square \quad &\delta\hat{u}_{ij} \left\{ \quad \mu\,\lambda_N \, \text{sign}(\Delta^t g_T) \, \{\mathcal{G}_T\}_{ij} \, J\,\phi_l \qquad \right\} \Delta\hat{\lambda}_{N_l} \\
&+\delta\hat{u}_{ij} \left\{ \quad \mu\,\lambda_N \, \text{sign}(\Delta^t g_T) \, \{\Delta\left(\delta\mathcal{G}_T\right)\}_{ijkl} \, J+ \right. \\
&\qquad\qquad \left. \mu\,\lambda_N \, \text{sign}(\Delta^t g_T) \, \{\mathcal{G}_T\}_{ij} \, \{\mathcal{J}\}_{kl} \qquad \right\} \Delta\hat{u}_{kl} \, \mathrm{d}\gamma_r
\end{aligned}
\tag{A.25}
$$

A.2.4. Linearized virtual contact work for the penalty method

For the penalty method the virtual contact work contains no additional unknowns. We can reuse the linearizations and variations from Eq. (A.4), Eq. (A.14), Eq. (A.1) and Eq. (A.3) and get the virtual contact work increment

$$
\begin{aligned}
\Delta\left(\delta^P \Pi_N\right) = \int_\square \delta\hat{u}_{ij} \left\{ \right. &+ \{\Delta\left(\delta\mathcal{G}_N\right)\}_{ijkl} \, \varepsilon\, g_N J^{(1)} \\
&+ \{\mathcal{G}_N\}_{ij} \, \varepsilon \, \{\mathcal{G}_N\}_{kl} \, J^{(1)} \\
&+ \{\mathcal{G}_N\}_{ij} \, \varepsilon\, g_N \{\mathcal{J}\}_{kl} + \{\mathcal{J}\}_{ij} \, \varepsilon\, g_N \{\mathcal{G}_N\}_{kl} \\
&+ \frac{1}{2}\varepsilon\, g_N^2 \, \{\Delta\left(\delta\mathcal{J}\right)\}_{ijkl} \qquad \left. \right\} \Delta\hat{u}_{kl} \, \mathrm{d}\gamma_r
\end{aligned}
\tag{A.26}
$$

A.3. Averaged non-mortar side normal

As we are talking about averaged mortar side normal vectors throughout this section we are omitting the ^{A} marker. We are using the concentrated integration scheme so we omit any index or marker to reflect this, like $\square_{IP_\psi}$ in Section 6.5.2.

A.3.1. Normal gap function

The variation of the normal gap function is based on g_N from Eq. (4.16) and given in Eq. (4.18). We write this in index notation

$$\delta g_N = \tilde{n}_i^{(1)} \left(N_j^{me} \delta \hat{u}_{ij} + \bar{a}_i^{(2)} \delta \xi^{(2)} \right) \tag{A.27}$$

This means we need the variation of $\xi^{(2)}$ based on Eq. (4.19) which reads in index notation as

$$\delta \xi^{(2)} = \frac{\tilde{a}_i^{(1)}}{\tilde{a}_k^{(1)} \bar{a}_k^{(2)}} \left(-N_j^{me} \delta \hat{u}_{ij} + g_N \, \delta \tilde{n}_i^{(1)} \right) \tag{A.28}$$

This expression contains the variation of the averaged normal vector $\delta \tilde{n}_i^{(1)}$. To calculate this we define

$$\tilde{\alpha}^{(1)} = \sqrt{\tilde{a}_i^{(1)} \, \tilde{a}_i^{(1)}} \qquad \text{length of averaged tangent} \tag{A.29}$$

in analogy to Eq. (4.3). Based on the avaraging rule from Eq. (4.11) we define an averaging matrix $\mathcal{A}_j$.

$$\tilde{a}_i^{(1)} = \mathcal{A}_j \, \hat{x}_{ij} \tag{A.30}$$

Remark: The variation of the averaged tangent vector in the linear case depends on 4 nodes including the nodes of the preceding and following edge. See Figure 6.2 for a visualization.

$$\mathcal{A} = \begin{bmatrix} N_1^{(1)} & N_2^{(1)} \end{bmatrix} \cdot \mathbf{A} \tag{A.31}$$

This averaging matrix $\mathcal{A}$ consists of the interpolation part of the normal vector depending on the shape functions $N_i^{(1)}$ and and averaging rule which will be collected in the matrix $\mathbf{A}$

$$\mathbf{A} = \frac{1}{2} \begin{bmatrix} N_{,\xi_1}^{(1)}(+1) & N_{,\xi_1}^{(1)}(-1) + N_{,\xi_2}^{(1)}(+1) & N_{,\xi_2}^{(1)}(-1) & 0 & 0 & 0 \\ 0 & N_{,\xi_1}^{(1)}(+1) & N_{,\xi_1}^{(1)}(-1) + N_{,\xi_2}^{(1)}(+1) & N_{,\xi_2}^{(1)}(-1) & 0 & 0 \end{bmatrix} \tag{A.32}$$

The averaging matrix $\mathbf{A}$ from Eq. (A.32) is valid for a non-mortar edge with a preceding and a following edge. On the end of a boundary there are edges with missing preceding or following edges. For this averaging tensor we have to distinguish the cases shown in Figure A.1. Therefore we define one for the case of missing preceding edge

$$\mathbf{A}^P = \frac{1}{2} \begin{bmatrix} 2N_{,\xi_1}^{(1)}(-1) & 2N_{,\xi_2}^{(1)}(-1) & 0 & 0 & 0 \\ N_{,\xi_1}^{(1)}(+1) & N_{,\xi_1}^{(1)}(-1) + N_{,\xi_2}^{(1)}(+1) & N_{,\xi_2}^{(1)}(-1) & 0 & 0 \end{bmatrix} \tag{A.33}$$

for the rare case of only one non-mortar edge

$$\mathbf{A}^0 = \begin{bmatrix} N_{,\xi_1}^{(1)}(-1) & N_{,\xi_2}^{(1)}(-1) & 0 & 0 \\ N_{,\xi_1}^{(1)}(+1) & N_{,\xi_2}^{(1)}(+1) & 0 & 0 \end{bmatrix} \tag{A.34}$$

and for the case of missing following edge

$$\mathbf{A}^F = \frac{1}{2} \begin{bmatrix} N_{,\xi_1}^{(1)}(+1) & N_{,\xi_1}^{(1)}(-1) + N_{,\xi_2}^{(1)}(+1) & N_{,\xi_2}^{(1)}(-1) & 0 & 0 \\ 0 & 2N_{,\xi_1}^{(1)}(+1) & 2N_{,\xi_2}^{(1)}(+1) & 0 & 0 \end{bmatrix} \tag{A.35}$$

A. Linearization

(a) Missing preceding edge, leads to definition of $\mathbf{A}^P$

(b) single edge without preceding nor following edge, leads to definition of $\mathbf{A}^0$

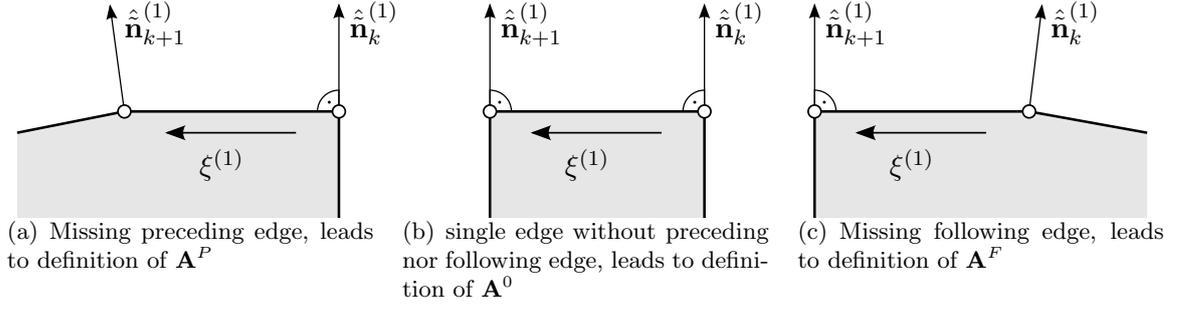(c) Missing following edge, leads to definition of $\mathbf{A}^F$

Figure A.1.: The special cases for the averaging tensor Eq. (A.31) are shown. The boundary without contact conditions is marked in green.

The variation of the averaged tangent vector as function of the nodal variational displacements yields

$$\delta \tilde{a}_i^{(1)} = \underbrace{\delta \mathcal{A}_j \, \hat{x}_{ij}}_{0} + \mathcal{A}_j \, \delta \hat{u}_{ij} = \mathcal{A}_j \, \delta \hat{u}_{ij} \tag{A.36}$$

As $\mathcal{A}_j$ is only dependent on non-mortar side shape functions it is constant with respect to the variation (the integration point coordinates are fixed). Therefore $\delta \mathcal{A}_j = 0$.

Based on Eq. (4.20) we can now write the variation of the averaged normal vector $\delta \tilde{\mathbf{n}}^{(1)}$ in index notation. It is important to note, that for the execution of the cross product we need 3 dimensional vectors. Therefore the averaged tangent $\tilde{a}_i^{(1)}$ is extended to the third dimension. After calculation we have to slice the indices i and m to $1, 2$.

$$\delta \tilde{n}_m^{(1)} = \left\{ \varepsilon_{mlk} \left[\frac{\delta_{li}}{\tilde{\alpha}^{(1)}} - \frac{\tilde{a}_l^{(1)} \, \tilde{a}_i^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3} \right] \mathcal{A}_j \, e_{3k} \right\} \Bigg|_{i,m=1,2} \delta \hat{u}_{ij}$$

$$= \{\mathcal{N}\}_{ijm} \, \delta \hat{u}_{ij} \tag{A.37}$$

and insert into Eq. (A.28)

$$\delta \xi^{(2)} = \frac{\tilde{a}_m^{(1)}}{\tilde{a}_k^{(1)} \overline{a}_k^{(2)}} \left(-N_j^{me} \delta_{im} + g_N \, \{\mathcal{N}\}_{ijm} \right) \delta \hat{u}_{ij}$$

$$= \{\mathcal{X}\}_{ij} \, \delta \hat{u}_{ij} \tag{A.38}$$

to finally get the variation of the normal gap function

$$\delta g_N = \left\{ \tilde{n}_k^{(1)} \left(N_j^{me} \delta_{ki} + \overline{a}_k^{(2)} \, \{\mathcal{X}\}_{ij} \right) \right\} \delta \hat{u}_{ij}$$

$$\delta g_N = \{\mathcal{G}_N\}_{ij} \, \delta \hat{u}_{ij} \tag{A.39}$$

To derive Δg_n we have to replace δ with Δ

$$\Delta g_N = \{\mathcal{G}_N\}_{kl} \, \Delta \hat{u}_{kl} \tag{A.40}$$

100

A.3.1.1. Linearization of variation of normal gap function

We need the increment $\Delta\left(\delta\tilde{\mathbf{n}}^{(1)}\right)$ of the virtual normal vector which is calculated based on Eq. (4.20)

$$
\Delta\left(\delta\tilde{\mathbf{n}}^{(1)}\right) = \left[-\delta\tilde{\mathbf{a}}^{(1)}\frac{\Delta\tilde{\mathbf{a}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3} - \Delta\tilde{\mathbf{a}}^{(1)}\frac{\delta\tilde{\mathbf{a}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3} - \right.
$$
$$
\left. -\tilde{\mathbf{a}}^{(1)}\left(\frac{\delta\tilde{\mathbf{a}}^{(1)}\cdot\Delta\tilde{\mathbf{a}}^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3} - \delta\tilde{\mathbf{a}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}\frac{3}{\left(\tilde{\alpha}^{(1)}\right)^5}\Delta\tilde{\mathbf{a}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}\right)\right] \times \mathbf{e}_3 \quad (A.41)
$$

which can be written in index notation. We keep in mind that we have to slice the indices i, k and o to $1, 2$ like above.

$$
\Delta\left(\delta\tilde{n}^{(1)}\right)_o = \delta\hat{u}_{ij}\left\{\varepsilon_{onm}\left[-\mathcal{A}_j\frac{\delta_{in}\tilde{a}_k^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3}\mathcal{A}_l - \mathcal{A}_j\frac{\tilde{a}_i^{(1)}\delta_{kn}}{\left(\tilde{\alpha}^{(1)}\right)^3}\mathcal{A}_l -\right.\right.
$$
$$
\left.\left.-\tilde{a}_n^{(1)}\left(\mathcal{A}_j\frac{\delta_{ik}}{\left(\tilde{\alpha}^{(1)}\right)^3}\mathcal{A}_l - \mathcal{A}_j\tilde{a}_i^{(1)}\frac{3}{\left(\tilde{\alpha}^{(1)}\right)^5}\tilde{a}_k^{(1)}\mathcal{A}_l\right)\right]e_{3m}\right\}\bigg|_{i,k,o=1,2}\Delta\hat{u}_{kl} \quad (A.42)
$$

$$
\Delta\left(\delta\tilde{n}^{(1)}\right)_o = \delta\hat{u}_{ij}\ \{\Delta\left(\delta\mathcal{N}\right)\}_{ijklo}\,\Delta\hat{u}_{kl} \quad (A.43)
$$

Now we linearize the variation of the contact kinematic Eq. (4.17) and obtain

$$
\mathbf{N}^{me,\xi^{(2)}}\delta\hat{\mathbf{u}}\,\Delta\xi^{(2)} + \mathbf{N}^{me,\xi^{(2)}}\Delta\hat{\mathbf{u}}\,\delta\xi^{(2)} + \overline{\mathbf{a}}^{(2)}\Delta\left(\delta\xi^{(2)}\right) + \Delta\xi^{(2)}\overline{\mathbf{b}}^{(2)}\delta\xi^{(2)} =
$$
$$
= \Delta\left(\delta g_N\right)\tilde{\mathbf{n}}^{(1)} + \delta g_n\Delta\tilde{\mathbf{n}}^{(1)} + \Delta g_n\delta\tilde{\mathbf{n}}^{(1)} + g_N\Delta\left(\delta\tilde{\mathbf{n}}^{(1)}\right) \quad (A.44)
$$

From Eq. (A.44) we can separate $\Delta\left(\delta\xi^{(2)}\right)$ by multiplying with $\cdot\tilde{\mathbf{a}}^{(1)}$ because then the term $\Delta\left(\delta g_N\right)\tilde{\mathbf{n}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}$ vanishes due to $\tilde{\mathbf{n}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)}$ being zero. Thus

$$
\Delta\left(\delta\xi^{(2)}\right) = \frac{\tilde{\mathbf{a}}^{(1)}}{\overline{\mathbf{a}}^{(2)}\cdot\tilde{\mathbf{a}}^{(1)}}\cdot\left[+\delta g_n\Delta\tilde{\mathbf{n}}^{(1)} \qquad +\Delta g_N\delta\tilde{\mathbf{n}}^{(1)}+\right.
$$
$$
+ g_n\Delta\left(\delta\tilde{\mathbf{n}}^{(1)}\right) \qquad -\Delta\xi^{(2)}\overline{\mathbf{b}}^{(2)}\delta\xi^{(2)}-
$$
$$
\left.-\mathbf{N}^{me,\xi^{(2)}}\delta\hat{\mathbf{u}}\,\Delta\xi^{(2)} - \mathbf{N}^{me,\xi^{(2)}}\Delta\hat{\mathbf{u}}\,\delta\xi^{(2)}\right] \quad (A.45)
$$

which can be written in index notation

$$
\Delta\left(\delta\xi^{(2)}\right) = \delta\hat{u}_{ij}\left\{\frac{\tilde{a}_o^{(1)}}{\overline{a}_m^{(2)}\tilde{a}_m^{(1)}}\left[+\{\mathcal{G}_N\}_{ij}\{\mathcal{N}\}_{klo} \quad +\{\mathcal{N}\}_{ijo}\{\mathcal{G}_N\}_{kl}+\right.\right.
$$
$$
+ g_n\{\Delta\left(\delta\mathcal{N}\right)\}_{ijklo} - \{\mathcal{X}\}_{ij}\overline{b}_o^{(2)}\{\mathcal{X}\}_{kl}-
$$
$$
\left.\left.-N_j^{me,\xi^{(2)}}\delta_{io}\{\mathcal{X}\}_{kl} - \{\mathcal{X}\}_{ij}\delta_{ko}N_l^{me,\xi^{(2)}}\right]\right\}\Delta\hat{u}_{kl} \quad (A.46)
$$

A. Linearization

$$\Delta\left(\delta\xi^{(2)}\right) = \delta\hat{u}_{ij}\ \{\Delta\left(\delta\mathcal{X}\right)\}_{ijkl}\,\Delta\hat{u}_{kl} \tag{A.47}$$

Now we separate from Eq. (A.44) the virtual normal gap function increment $\Delta\left(\delta g_N\right)$ by multiplying with the averaged tangent $\cdot\tilde{\mathbf{a}}^{(1)}$. We should be aware that $\Delta\tilde{\mathbf{n}}^{(1)}\cdot\tilde{\mathbf{n}}^{(1)}$ and $\delta\tilde{\mathbf{n}}^{(1)}\cdot\tilde{\mathbf{n}}^{(1)}$ equals to zero because of $\tilde{\mathbf{n}}^{(1)}$ being a unit vector.

$$\Delta\left(\delta g_N\right) = \tilde{\mathbf{n}}^{(1)}\cdot\left[-g_N\Delta\left(\delta\tilde{\mathbf{n}}^{(1)}\right)\qquad +\overline{\mathbf{a}}^{(2)}\Delta\left(\delta\xi^{(2)}\right)\right.$$
$$+\Delta\xi^{(2)}\overline{\mathbf{b}}^{(2)}\delta\xi^{(2)}$$
$$\left.+\mathbf{N}^{me,\xi^{(2)}}\delta\hat{\mathbf{u}}\,\Delta\xi^{(2)} + \mathbf{N}^{me,\xi^{(2)}}\Delta\hat{\mathbf{u}}\,\delta\xi^{(2)}\right] \tag{A.48}$$

in index notation

$$\Delta\left(\delta g_N\right) = \delta\hat{u}_{ij}\left\{\tilde{n}_o^{(1)}\left[-g_N\{\Delta\left(\delta\mathcal{N}\right)\}_{ijklo} + \overline{a}_o^{(2)}\{\Delta\left(\delta\mathcal{X}\right)\}_{ijkl}\right.\right.$$
$$+\{\mathcal{X}\}_{ij}\,\overline{b}_o^{(2)}\,\{\mathcal{X}\}_{kl}$$
$$\left.\left.+N_j^{me,\xi^{(2)}}\delta_{io}\{\mathcal{X}\}_{kl} + \{\mathcal{X}\}_{ij}\,\delta_{ko}N_l^{me,\xi^{(2)}}\right]\right\}\Delta\hat{u}_{kl} \tag{A.49}$$

$$\Delta\left(\delta g_N\right) = \delta\hat{u}_{ij}\ \{\Delta\left(\delta\mathcal{G}_N\right)\}_{ijkl}\,\Delta\hat{u}_{kl} \tag{A.50}$$

A.3.2. Tangential slip

Based on Eq. (4.21) we can calculate δg_T. As shown in Eq. (6.10) δg_T and Δg_T are equivalent. The values can be calculated in index notation as

$$\delta g_T = -\tilde{s}_k^{(1)}\overline{a}_k^{(2)}\delta\xi^{(2)} = -\tilde{s}_k^{(1)}\overline{a}_k^{(2)}\{\mathcal{X}\}_{ij}\,\delta\hat{u}_{ij}\qquad \Delta g_T = -\tilde{s}_k^{(1)}\overline{a}_k^{(2)}\{\mathcal{X}\}_{kl}\,\Delta\hat{u}_{kl}$$
$$\delta g_T = \{\mathcal{G}_T\}_{ij}\,\delta\hat{u}_{ij}\qquad\qquad\qquad\qquad \Delta g_T = \{\mathcal{G}_T\}_{kl}\,\Delta\hat{u}_{kl} \tag{A.51}$$

A.3.2.1. Linearization of variation of tangential slip

The increment can be calculated based on Eq. (A.51) and we obtain

$$\Delta\left(\delta g_T\right) = -\Delta\tilde{\mathbf{s}}^{(1)}\cdot\overline{\mathbf{a}}^{(2)}\,\delta\xi^{(2)} - \tilde{\mathbf{s}}^{(1)}\cdot\Delta\overline{\mathbf{a}}^{(2)}\,\delta\xi^{(2)} - \tilde{\mathbf{s}}^{(1)}\cdot\overline{\mathbf{a}}^{(2)}\,\Delta\left(\delta\xi^{(2)}\right) \tag{A.52}$$

where we need the following expressions

$$\Delta\tilde{\mathbf{s}}^{(1)} = \Delta\left(\frac{\tilde{\mathbf{a}}^{(1)}}{\tilde{\alpha}^{(1)}}\right) = \frac{\Delta\tilde{\mathbf{a}}^{(1)}}{\tilde{\alpha}^{(1)}} - \frac{\tilde{\mathbf{a}}^{(1)}}{\left(\tilde{\alpha}^{(1)}\right)^3}\,\Delta\tilde{\mathbf{a}}^{(1)}\cdot\tilde{\mathbf{a}}^{(1)} \tag{A.53}$$

$$\Delta\overline{\mathbf{a}}^{(2)} = \overline{\mathbf{b}}^{(2)}\Delta\xi^{(2)} + \Delta\mathbf{u}_{,\xi}^{(2)} \tag{A.54}$$

Once again we can write the variation of tangential slip increment in index notation as

$$
\begin{aligned}
\Delta\left(\delta g_{T}\right) = \delta\hat{u}_{ij}\Bigg[&-\frac{\mathcal{A}_{l}\,\bar{a}_{k}^{(2)}}{\tilde{\alpha}^{(1)}}\left\{\mathcal{X}\right\}_{ij} + \frac{\tilde{a}_{m}^{(1)}\,\bar{a}_{m}^{(2)}}{\left(\tilde{\alpha}^{(1)}\right)^{3}}\mathcal{A}_{l}\,\tilde{a}_{k}^{(1)} \\
&-\left\{\mathcal{X}\right\}_{ij}\bar{b}_{m}^{(2)}\tilde{s}_{m}^{(2)}\left\{\mathcal{X}\right\}_{kl} - \left\{\mathcal{X}\right\}_{ij}\tilde{s}_{k}^{(2)}N_{l}^{me,\xi(2)} \\
&-\left\{\Delta\left(\delta\mathcal{X}\right)\right\}_{ijkl}\Bigg]\Delta\hat{u}_{kl} \\
= \delta\hat{u}_{ij}&\left\{\Delta\left(\mathcal{G}_{T}\right)\right\}_{ijkl}\Delta\hat{u}_{kl}
\end{aligned}
\tag{A.55}
$$

$\left\{\Delta\left(\mathcal{G}_{T}\right)\right\}_{ijkl}$ is not symmetric for this formulation. The asymmetry vanishes as soon as we assume that the mortar side tangent $\bar{\mathbf{a}}^{(2)}$ is parallel to the non-mortar side tangent $\tilde{\mathbf{a}}^{(1)}$. As already mentioned in Section 3.3 this is an artificial asymmetry. But this formulation keeps as much information from the discretized surface as possible.

A.3.3. Linearized virtual contact work

All the geometric properties are contained in $\left\{\Delta\left(\delta\mathcal{G}_{N}\right)\right\}_{ijkl}$, $\left\{\Delta\left(\delta\mathcal{J}\right)\right\}_{ijkl}$ and the variational values. We obtain the same result for the incremental virtual contact work as for the mortar side normal field. Here the advantage of the chosen formulation can be seen. It was a goal of this work to abstract the contact kinematics to insert nearly arbitrary normal field and surface formulations.

A.3.3.1. Normal direction

$$
\begin{aligned}
\Delta\left(\delta^{A}\Pi_{N}\right) = \int_{\square} \ &-\delta\hat{\lambda}_{N_{j}}\left\{\quad\phi_{j}\qquad\left[J^{(1)}\left\{\mathcal{G}_{N}\right\}_{kl} + g_{N}\left\{\mathcal{J}\right\}_{kl}\right]\qquad\right\}\Delta\hat{u}_{kl} \\
&-\delta\hat{u}_{ij}\left\{\quad\phi_{l}\qquad\left[J^{(1)}\left\{\mathcal{G}_{N}\right\}_{ij} + g_{N}\left\{\mathcal{J}\right\}_{ij}\right]\qquad\right\}\Delta\hat{\lambda}_{N_{l}} \\
&-\delta\hat{u}_{ij}\left\{\quad\lambda_{N}\quad\Big[\left\{\Delta\left(\delta\mathcal{G}_{N}\right)\right\}_{ijkl}J^{(1)} + \left\{\Delta\left(\delta\mathcal{J}\right)\right\}_{ijkl}g_{N}\right.\right. \\
&\qquad\qquad\qquad\left.\left.+\left\{\mathcal{G}_{N}\right\}_{ij}\left\{\mathcal{J}\right\}_{kl} + \left\{\mathcal{J}\right\}_{ij}\left\{\mathcal{G}_{N}\right\}_{kl}\Big]\quad\right\}\Delta\hat{u}_{kl}\ \mathrm{d}\gamma_{r}
\end{aligned}
\tag{A.56}
$$

A.3.3.2. Tangential direction

Stick case

$$
\begin{aligned}
\Delta\left(\delta^{A}\Pi_{CT_{ST}}\right) = \int_{\square} \ &-\delta\hat{\lambda}_{T_{j}}\left\{\quad\phi_{j}\qquad\left[J^{(1)}\left\{\mathcal{G}_{T}\right\}_{kl} + \Delta^{t}g_{T}\left\{\mathcal{J}\right\}_{kl}\right]\qquad\right\}\Delta\hat{u}_{kl} \\
&-\delta\hat{u}_{ij}\left\{\quad\phi_{l}\qquad\left[J^{(1)}\left\{\mathcal{G}_{T}\right\}_{ij} + \Delta^{t}g_{T}\left\{\mathcal{J}\right\}_{ij}\right]\qquad\right\}\Delta\hat{\lambda}_{T_{l}} \\
&-\delta\hat{u}_{ij}\left\{\quad\lambda_{N}\quad\Big[\left\{\Delta\left(\delta\mathcal{G}_{T}\right)\right\}_{ijkl}J^{(1)} + \left\{\Delta\left(\delta\mathcal{J}\right)\right\}_{ijkl}\Delta^{t}g_{T}\right.\right. \\
&\qquad\qquad\qquad\left.\left.+\left\{\mathcal{G}_{T}\right\}_{ij}\left\{\mathcal{J}\right\}_{kl} + \left\{\mathcal{J}\right\}_{ij}\left\{\mathcal{G}_{T}\right\}_{kl}\Big]\quad\right\}\Delta\hat{u}_{kl}\ \mathrm{d}\gamma_{r}
\end{aligned}
\tag{A.57}
$$

A. Linearization

Slip case

$$\Delta\left(\delta^A\Pi_{CT_{SL}}\right) = \int_\square \quad \delta\hat{u}_{ij}\left\{\quad \mu\,\lambda_N\,\text{sign}(\Delta^t g_T)\,\{\mathcal{G}_T\}_{ij}\,J\,\phi_l \qquad\right\}\Delta\hat{\lambda}_{N_l}$$
$$+\delta\hat{u}_{ij}\left\{\quad \mu\,\lambda_N\,\text{sign}(\Delta^t g_T)\,\{\Delta\left(\delta\mathcal{G}_T\right)\}_{ijkl}\,J+\right.$$
$$\left.\mu\,\lambda_N\,\text{sign}(\Delta^t g_T)\,\{\mathcal{G}_T\}_{ij}\,\{\mathcal{J}\}_{kl}\quad\right\}\Delta\hat{u}_{kl}\;\mathrm{d}\gamma_r \qquad\text{(A.58)}$$

B. Bibliography

[1] G.C. Archer, G. Fenves, and C. Thewalt. A new object-oriented finite element analysis program architecture. *Computers and Structures*, 70:63–75, 1999.

[2] K.J. Bathe. *Finite Element Procedures*. Prentice Hall, 2007.

[3] F.B. Belgacem, P. Hild, and P. Laborde. The mortar finite element method for contact problems. *Mathematical and Computer Modelling*, 28(4–8):263–271, 1998.

[4] D.J. Benson and J.O. Hallquist. A single surface contact algorithm for postbucklung. *Computer Methods in Applied Mechanics and Engineering*, 78:141–163, 1990.

[5] Javier Bonet and Richard D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 2nd edition, 1997.

[6] K. A. Fischer and P. Wriggers. Frictionless 2D Contact formulations for finite deformations based on the mortar method. *Computational Mechanics*, 36:226–244, 2005.

[7] Kathrin A. Fischer and Peter Wriggers. Mortar based frictional contact formulation for higher order interpolations using the moving friction cone. *Computer Methods in Applied Mechanics and Engineering*, 195:5020–5036, 2006.

[8] B. Flemisch, M. A. Puso, and B. I. Wohlmuth. A new dual mortar method for curved interfaces: 2D elasticity. *International Journal for Numerical Methods in Engineering*, 63:813–832, 2005.

[9] C. Geuzaine and J.F. Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[10] Markus Gitterle, Alexander Popp, Michael W. Gee, and Wolfgang A. Wall. Finite deformation frictional mortar contact using a semi-smooth Newton method with consistent linearization. *International Journal for Numerical Methods in Engineering*, 84 (5):543–571, 2010.

[11] W. Goldsmith. *Impact: the theory and physical behaviour of colliding solids*. Dover books on engineering. Dover Publications, 2001.

[12] Alfred Gray. *Modern differential geometry of curves and surfaces with Mathematica*. CRC Press, 2nd edition, 1998.

[13] JO Hallquist, GL Goudreau, and Benson DJ. Sliding interfaces with contact-impact in large-scale Lagrangian computations. *Computer Methods in Applied Mechanics and Engineering*, 51:107–137, 1985.

B. Bibliography

[14] Michael E. Hammer. SOOFEA - Software for Object Oriented Finite Element Analysis, 2006. URL `http://soofea.fest.tugraz.at`.

[15] S. Hartmann and E. Ramm. A mortar based contact formulation for non-linear dynamics using dual Lagrange multipliers. *Finite Elements in Analysis and Design*, 44: 245–258, 2008.

[16] Patrick Hild. Numerical implementation of two nonconforming finite element methods for unilateral contact. *Computer Methods in Applied Mechanics and Engineering*, 184: 99–123, 2000.

[17] S. Hüeber and B.I. Wohlmuth. A primal–dual active set strategy for non-linear multi-body contact problems. *Computer Methods in Applied Mechanics and Engineering*, 194:3147–3166, 2005.

[18] S. Hüeber and B.I. Wohlmuth. Thermo-mechanical contact problems on non-matching meshes. *Computer Methods in Applied Mechanics and Engineering*, 198:1338–1350, 2009.

[19] John D. Hunter. Matplotlib, 2011. URL `http://matplotlib.sourceforge.net/index.html`.

[20] N. Kikuchi and J.T. Oden. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*. SIAM studies in applied mathematics. SIAM, 1988.

[21] Eberhard Klingbeil. *Tensorrechnung für Ingenieure*. Bibliographisches Institut, 1985.

[22] A. Konyukhov and K. Schweizerhof. A special focus on 2D formulations for contact problems using a covariant description. *International Journal for Numerical Methods in Engineering*, 66:1432–1465, 2006.

[23] T. A. Laursen and J. C. Simo. A continuum-based finite element formulation for the implicit solution of multibody, large deformation-frictional contact problems. *International Journal for Numerical Methods in Engineering*, 36(20):3451–3485, 1993.

[24] T.A. Laursen. *Computational Contact and Impact Mechanics*. Springer, 2002.

[25] Jerold E. Marsden and Thomas J. R. Hughes. *Mathematical Foundations of Elasticity*. Dover Publications, Inc., 1983.

[26] T. W. McDevitt and T. A. Laursen. A mortar-finite element formulation for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 48: 1525–1547, 2000.

[27] Alexander Popp, Michael W. Gee, and Wolfgang A. Wall. A finite deformation mortar contact formulation using a primal–dual active set strategy. *International Journal for Numerical Methods in Engineering*, 79(11):1354–1391, 2009.

[28] Michael A. Puso. A 3D mortar method for solid mechanics. *International Journal for Numerical Methods in Engineering*, 59:315–336, 2004.

[29] Michael A. Puso and Tod A. Laursen. A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193(6-8):601 – 629, 2004.

[30] Michael A. Puso and Tod A. Laursen. A mortar segment-to-segment frictional contact method for large deformations. *Computer Methods in Applied Mechanics and Engineering*, 193(45-47):4891 – 4913, 2004.

[31] Michael A. Puso, T.A. Laursen, and Jerome Solberg. A segment-to-segment mortar contact method for quadratic elements and large deformations. *Computer Methods in Applied Mechanics and Engineering*, 197(6-8):555 – 566, 2008.

[32] K. Schweizerhof and A. Konyukhov. Covariant description for frictional contact problems. *Computational Mechanics*, 35:190–213, 2005.

[33] I. S. Sokolnikoff. *Tensor Analysis Theory and Applicatoins to Geometry and Mechanics of Continua*. John Wiley & Sons, Inc., 1964.

[34] C. Truesdell and W. Noll. *The Non-Linear Field Theories of Mechanics*. Springer-Verlag, 1965.

[35] M. Tur, F.J. Fuenmayor, and P. Wriggers. A mortar-based frictional contact formulation for large deformations using Lagrange multipliers. *Computer Methods in Applied Mechanics and Engineering*, 198(37-40):2860 – 2873, 2009.

[36] Kai Willner. *Kontinuums- und Kontaktmechanik*. Springer, 2003.

[37] B. I. Wohlmuth. A mortar finite element method using dual spaces for the Lagrange multiplier. *SIAM Journal on Numerical Analysis*, 38:989–1012, 2000.

[38] B. I. Wolmuth. A mortar finite element method using dual spaces for the Lagrange multiplier. *SIAM J. Numer. Anal.*, 38:989–1012, 2000.

[39] P. Wriggers and G. Zavarise. A formulation for frictionless contact problems using a weak form introduced by Nitsche. *Computational Mechanics*, 41:407–420, 2008.

[40] Peter Wriggers. *Computational contact mechanics*. Springer, 2007.

[41] Peter Wriggers. *Nonlinear Finite Element Methods*. Springer, 2009.

[42] Bin Yang and Tod A. Laursen. A contact searching algorithm including bounding volume trees applied to finite sliding mortar formulations. *Computational Mechanics*, 41:189–205, 2008.

[43] Bin Yang and Tod A. Laursen. A large deformation mortar formulation of self contact with finite sliding. *Computer Methods in Applied Mechanics and Engineering*, 197:756–772, 2008.

[44] Bin Yang, Tod A. Laursen, and Xiaonong Meng. Two dimensional mortar contact methods for large deformation frictional sliding. *International Journal for Numerical Methods in Engineering*, 62(9):1183–1225, 2005.

[45] Giorgio Zavarise and Laura de Lorenzis. The node-to-segment algorithm for 2D frictionless contact: Classical formulation and special cases. *Computer Methods in Applied Mechanics and Engineering*, 198:3428–3451, 2009.

[46] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method. Its Basis and Fundamentals.*, volume 1. Butterworth-Heinemann, 6th edition, 2005.

[47] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method for Solid and Structural Mechanics*, volume 2. Butterworth-Heinemann, 6th edition, 2005.