

Contact Modelling in Multibody Systems by Means of a Boundary Element Co-simulation and a DIRICHLET-to-NEUMANN Algorithm

János Zierath and Christoph Woernle

Abstract The present contribution introduces the modelling of elastic contacts by coupled multibody and boundary element systems. Compared to contacts modelled by impact laws, physically more accurate results can be obtained. Due to the use of boundary element systems, the contact stresses are obtained within the contact calculation.

A new three-dimensional contact element for boundary element systems is developed. The mortar element uses the mixed formulation of boundary element formulations. The algorithm for the iteration of contact states is based on an DIRICHLET-to-NEUMANN algorithm. Herein, both contacting bodies are calculated serially. In the first calculation step one of the contacting bodies represents a rigid obstacle for the other elastic one. The resulting reaction forces on the elastic body are partially transferred on the other one, which is for the second calculation step no longer rigid. As a result, the obstacle is deformed and the next iteration starts. The algorithm converges if the numerical equilibrium in the contact interface is reached.

1 Introduction

Simulation of elastic multibody systems based on a finite element analysis has become usual in commercial multibody programs, such as MSC.Adams or SIMPACK. To model contacts between elastic bodies, the classical impact laws used in rigid-body dynamics, such as NEWTON's kinematical approach or POISSON's impact law [29] based on integrated forces, are not suitable. Instead contact models derived from the elastic body description have to be applied. Typically elastic body models are formulated by means of a modal approach like the Craig-Bampton method [12].

J. Zierath (✉)

W2E Wind to Energy GmbH, Strandstrasse 96, Rostock, Germany
e-mail: jzierath@wind-to-energy.de

C. Woernle

Chair of Technical Dynamics, University of Rostock, Justus-von-Liebig-Weg 6, Rostock, Germany
e-mail: woernle@uni-rostock.de

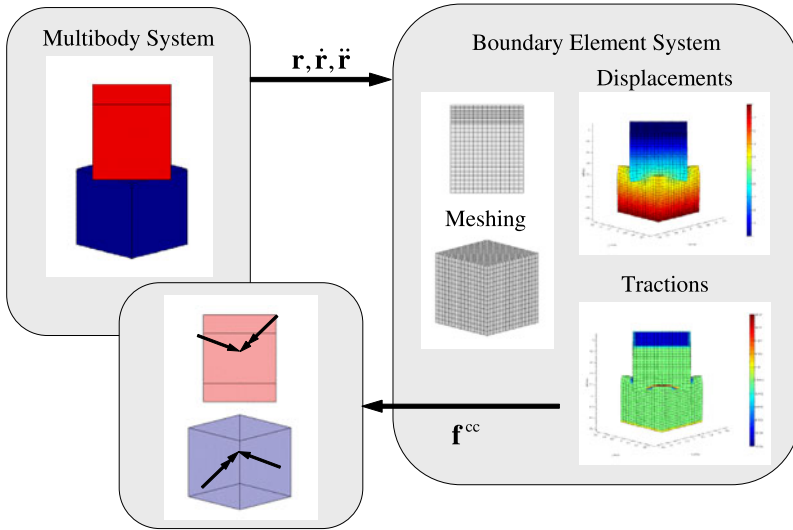


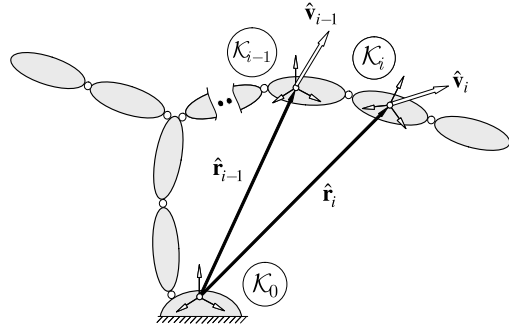
Fig. 1 Coupling of multibody system and boundary element models

Modal functions describe the elastic deformations by a superposition of global mode shapes with static mode shapes or frequency response modes. However, the global mode shapes are not able to describe the elastic deformations of the bodies in the contact area. Consequently, the number of modes used for the contacting elastic bodies has to be increased. Another method to describe elastic contacts is the coupling between a finite element (FE) model and a multibody system (MBS), see [15]. As a FE body has typically a higher degree of freedom (DOF) than a complete MBS, the coupling of FE models and MBS results in a large computational amount.

The aim of this chapter is to present a method to couple multibody systems with boundary element systems for contact simulation as shown in Fig. 1. This approach is seen as an alternative to model contact by coupled multibody/finite element simulations as presented by [2]. In this chapter three-dimensional problems are considered only. Due to the fact that three-dimensional contact algorithms for boundary element systems are sparsely described in literature, a new contact element based on mortar methods has been developed, see [37].

This contribution is organised as follows. In Sect. 2 the fundamentals of multibody systems are introduced. The description restricts on differential algebraic equations. Based on the boundary integral formulation for elastostatics introduced in Sect. 3, the DIRICHLET-to-NEUMANN algorithm for contact calculation in boundary element systems is described in Sect. 4. This algorithm is based on mortar methods which are presented in Sect. 5. In Sect. 6 the numerical implementation of the mortar methods is described. The contact stresses are obtained from the contact calculation in boundary element systems. The calculation of the resulting contact force from the stresses is described in Sect. 7. As an example the dynamic simulation of two impacting spheres is presented in Sect. 8.

Fig. 2 Kinematics of rigid multibody systems



2 Fundamentals of Multibody Systems

In this section the theoretical background of multibody systems is shortly described. Due to the fact that the contact acts as a force element inside the multibody environment an introduction to constrained systems of rigid bodies is given only. The equations of motion are formulated as differential-algebraic equations (DAE). This formulation is typically used in commercial multibody simulation packages. A detailed description of multibody system dynamics is given in [32] and [35].

The position of the i th body of a constrained multibody system is described by a spatial vector $\hat{\mathbf{r}}_i$ consisting of the position vector \mathbf{r}_i from an inertial system \mathcal{K}_0 to a body-fixed coordinate system \mathcal{K}_i and $m_i \geq 3$ coordinates γ_i describing the spatial orientation of \mathcal{K}_i relative to \mathcal{K}_0 (Fig. 2),

$$\hat{\mathbf{r}}_i = \begin{bmatrix} \mathbf{r}_i \\ \gamma_i \end{bmatrix}. \quad (1)$$

Examples for the definition of γ_i are EULER angles or EULER parameters (unit quaternions). Accordingly, the absolute velocity of the i th body is described by the velocity \mathbf{v}_i and the angular velocity ω_i of \mathcal{K}_i put together in the six-dimensional spatial vector

$$\hat{\mathbf{v}}_i = \begin{bmatrix} \mathbf{v}_i \\ \omega_i \end{bmatrix}. \quad (2)$$

To describe the position and velocity of the overall system with N bodies, the vectors

$$\hat{\mathbf{r}} = \begin{bmatrix} \hat{\mathbf{r}}_1 \\ \vdots \\ \hat{\mathbf{r}}_N \end{bmatrix}, \quad \hat{\mathbf{v}} = \begin{bmatrix} \hat{\mathbf{v}}_1 \\ \vdots \\ \hat{\mathbf{v}}_N \end{bmatrix} \quad (3)$$

are introduced. The relation between the time derivatives of $\hat{\mathbf{r}}_i$ and $\hat{\mathbf{v}}_i$ is given by kinematic differential equations of the form

$$\begin{aligned} \dot{\hat{\mathbf{r}}}_i &= \hat{\mathbf{H}}_i(\hat{\mathbf{r}}_i) \hat{\mathbf{v}}_i, \quad i = 1, \dots, N \\ \text{or} \\ \dot{\hat{\mathbf{r}}} &= \hat{\mathbf{H}}(\hat{\mathbf{r}}) \hat{\mathbf{v}}, \quad \hat{\mathbf{H}} = \text{diag}(\hat{\mathbf{H}}_1 \dots \hat{\mathbf{H}}_N). \end{aligned} \quad (4)$$

The constraints between the rigid bodies at the position, velocity, and acceleration levels are formulated in the implicit form,

$$\mathbf{g}(\hat{\mathbf{r}}, t) = \mathbf{0}, \quad (5)$$

$$\dot{\mathbf{g}} \equiv \mathbf{G}(\hat{\mathbf{r}}, t)\hat{\mathbf{v}} + \bar{\mathbf{g}}(\hat{\mathbf{r}}, t) = \mathbf{0}, \quad (6)$$

$$\ddot{\mathbf{g}} \equiv \mathbf{G}(\hat{\mathbf{r}}, t)\dot{\hat{\mathbf{v}}} + \bar{\bar{\mathbf{g}}}(\hat{\mathbf{r}}, \hat{\mathbf{v}}, t) = \mathbf{0}. \quad (7)$$

The kinetic differential equations are derived from the principles of linear momentum and angular momentum. Using spatial force vectors $\hat{\mathbf{f}}_i$ containing a pair of a force and a torque the kinetic equations of the i th body are

$$\widehat{\mathbf{M}}_i \dot{\hat{\mathbf{v}}}_i = \hat{\mathbf{f}}_i^c + \hat{\mathbf{f}}_i^a + \hat{\mathbf{f}}_i^r + \hat{\mathbf{f}}_i^{cc}, \quad i = 1, \dots, N, \quad (8)$$

with the symmetric, positive definite (6, 6) mass matrix $\widehat{\mathbf{M}}_i$, gyroscopic and CORIOLIS forces $\hat{\mathbf{f}}_i^c$, applied forces $\hat{\mathbf{f}}_i^a$, and reaction forces (constraint forces) $\hat{\mathbf{f}}_i^r$. Additionally, the vector $\hat{\mathbf{f}}_i^{cc}$ is introduced to represent the contact forces. Inside the multibody environment, the contact force is represented by an applied force. The overall kinetic equations can be written as

$$\widehat{\mathbf{M}}\dot{\hat{\mathbf{v}}} = \hat{\mathbf{f}}^c + \hat{\mathbf{f}}^a + \hat{\mathbf{f}}^r + \hat{\mathbf{f}}^{cc}$$

with $\widehat{\mathbf{M}} = \text{diag}(\widehat{\mathbf{M}}_1 \dots \widehat{\mathbf{M}}_N)$, $\hat{\mathbf{f}}^{c/a/r/cc} = \begin{bmatrix} \hat{\mathbf{f}}_1^{c/a/r/cc} \\ \vdots \\ \hat{\mathbf{f}}_N^{c/a/r/cc} \end{bmatrix}$. (9)

The reaction forces $\hat{\mathbf{f}}_i^r$ have components in the constrained spatial directions only, given by the row vectors of the constraint matrix \mathbf{G} from (6). Accordingly, they can be expressed by means of the explicit reaction force equations

$$\hat{\mathbf{f}}^r = \mathbf{G}^T \boldsymbol{\lambda} \quad (10)$$

with reaction force coordinates (LAGRANGIAN multipliers) $\boldsymbol{\lambda}$. If the positions $\hat{\mathbf{r}}$ and the velocities $\hat{\mathbf{v}}$, which have to be consistent with the constraints (5) and their first-order time derivatives (6), are given, (9) with (10) and (7) together represent a set of linear equations to determine uniquely the accelerations $\dot{\hat{\mathbf{v}}}$ and the reaction force coordinates $\boldsymbol{\lambda}$,

$$\begin{bmatrix} \widehat{\mathbf{M}} & \mathbf{G}^T \\ \mathbf{G} & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} \dot{\hat{\mathbf{v}}} \\ -\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{f}}^c + \hat{\mathbf{f}}^a + \hat{\mathbf{f}}^{cc} \\ -\bar{\bar{\mathbf{g}}} \end{bmatrix}. \quad (11)$$

Numerical integration of the velocity $\hat{\mathbf{v}}$ obtained from the kinematic differential equation (4) and of the acceleration $\dot{\hat{\mathbf{v}}}$ obtained from (11) yields the motion of the system, described by the position $\hat{\mathbf{r}}(t)$ and the velocity $\hat{\mathbf{v}}(t)$. The reaction force coordinates $\boldsymbol{\lambda}$ obtained from (11) yield the reaction forces $\hat{\mathbf{f}}^r$ by means of the explicit reaction force equations (10). During numerical integration, the constraints at the position and velocity levels may be violated necessitating a constraint stabilisation [4].

3 Boundary Integral Formulation in Elastostatics

The boundary integral formulation starts from the partial differential equation representing the quasistatic equilibrium inside the volume V ,

$$\sigma_{ij,j} + b_i = 0 \quad \text{in } V \quad \text{for } i = x, y, z. \quad (12)$$

Herein the variable σ_{ij} represents the stress tensor inside the volume V and b_i the body forces due to the acceleration of the body. In addition to the equilibrium inside the volume V , the equilibrium conditions on the surface S ,

$$t_i = \sigma_{ij}n_j \quad \text{on } S, \quad (13)$$

have to be fulfilled. Herein t_i is the stress vector, also called traction vector, representing the equilibrium on the surface S with the outward normal vector n_j .

The surface is divided into regions with different types of boundary conditions. There are regions on the surface S where the DIRICHLET boundary conditions

$$u_i = \bar{u}_i \quad \text{on } S_u \quad (14)$$

have to be fulfilled, and regions with NEUMANN boundary conditions,

$$t_i = \bar{t}_i \quad \text{on } S_t. \quad (15)$$

Due to the fact that each point on the surface S has three degrees of freedom, each direction $i = x, y, z$ has to be treated separately. Thus, the requirements $S_i = S_{ui} \cup S_{ti}$ and $S_{ui} \cap S_{ti} = 0$ have to be fulfilled for each coordinate direction i .

By using the weighted residual formulation

$$\int_V (\sigma_{ij,j} + b_i) w_i \, d\tilde{V} + \int_{S_u} (u_i - \bar{u}_i) \bar{w}_i \, d\tilde{S} + \int_{S_t} (t_i - \bar{t}_i) \bar{w}_i \, d\tilde{S} = 0 \quad (16)$$

and introducing the fundamental solutions u_{ij}^* and t_{ij}^* as weighting functions, the SOMIGLIANA identity

$$u_i(\mathbf{x}) = \int_S u_{ij}^*(\mathbf{x}, \mathbf{y}) t_j(\mathbf{y}) \, d\tilde{S} - \int_S t_{ij}^*(\mathbf{x}, \mathbf{y}) u_j(\mathbf{y}) \, d\tilde{S} + \int_V u_{ij}^*(\mathbf{x}, \mathbf{y}) b_j(\mathbf{y}) \, d\tilde{V} \quad (17)$$

is obtained which is valid for an arbitrary load point \mathbf{x} inside the volume V . The fundamental solutions for elastostatics have been developed by Lord KELVIN, see [33]. They represent the response of the system at any arbitrary field point \mathbf{y} due to a unit load at a chosen load point \mathbf{x} . In this chapter, three-dimensional boundary element formulations are treated only. Because each point has three degrees of freedom, the coordinate direction has also to be considered. A volume integral with the specific body forces b_j is given on the left hand side of (17).

According to [1], the displacement fundamental solution, also called DIRICHLET fundamental solution, at a field point \mathbf{y} in direction j due to a unit load at a load point \mathbf{x} in direction i is given by

$$u_{ij}^*(\mathbf{x}, \mathbf{y}) = \frac{1 + \nu}{8\pi E(1 - \nu)r} [(3 - 4\nu)\delta_{ij} + r_{,i} r_{,j}]. \quad (18)$$

Herein, the variable E represents YOUNG's modulus, ν POISSON's ratio, and δ_{ij} the KRONECKER delta. The distance r between a load point \mathbf{x} and a field point \mathbf{y} is given by the EUCLIDIAN norm

$$r = \|\mathbf{y} - \mathbf{x}\|. \quad (19)$$

The partial derivatives of the EUCLIDIAN distance r with respect to the coordinate direction i can easily be obtained by

$$r_{,i} = \frac{y_i - x_i}{r}. \quad (20)$$

The stress fundamental solution, also known as NEUMANN fundamental solution, at a field point \mathbf{y} in direction j due to a unit load at a load point \mathbf{x} in direction i is given by

$$t_{ij}^*(\mathbf{x}, \mathbf{y}) = -\frac{1}{8\pi(1-\nu)r^2} \left[\frac{\partial r}{\partial n} [(1-2\nu)\delta_{ij} + 3r_{,i}r_{,j}] - (1-2\nu)(n_i r_{,i} - n_j r_{,j}) \right], \quad (21)$$

where n_i represents the i th component of the outward normal vector \mathbf{n} . Herein, the derivative in normal direction is calculated by

$$\frac{\partial r}{\partial n} = r_{,i} n_i. \quad (22)$$

According to the third summand in (17), a remaining volume integral with the specific body forces b_j has to be taken into account. There are different procedures to calculate body forces. A method is the discretisation of the volume by using cells, see [1]. For a boundary element formulation this method is less appropriate because an additional volume mesh has to be introduced. A proposal for the transformation of the body forces from the volume V on the surface S was made in [13]. In the present chapter, quasistatic body forces are taken into account only. By this, elastic vibrations of the body are neglected.

Starting from SOMIGLIANA's identity (17) for a load point \mathbf{x} on the surface S , the integral equation under consideration of the boundary factor c_{ij} is given by

$$c_{ij}u_j(\mathbf{x}) = \int_S u_{ij}^*(\mathbf{x}, \mathbf{y})t_j(\mathbf{y})d\tilde{S} - \int_S t_{ij}^*(\mathbf{x}, \mathbf{y})u_j(\mathbf{y})d\tilde{S} + B_i(\mathbf{x}). \quad (23)$$

The boundary factor c_{ij} is a correction factor to consider load points on the surface S of the volume V . Due to the introduction of surface elements, the surface integrals in (23) are evaluated element-wise (summation over all elements e),

$$c_{ij}u_j(\mathbf{x}) = \sum_e \int_{S_e} u_{ij}^*(\mathbf{x}, \mathbf{y})t_j(\mathbf{y})d\tilde{S} - \sum_e \int_{S_e} t_{ij}^*(\mathbf{x}, \mathbf{y})u_j(\mathbf{y})d\tilde{S} + \sum_e B_{ie}(\mathbf{x}). \quad (24)$$

The continuous functions $u_j(\mathbf{y})$ for the displacements and $t_j(\mathbf{y})$ for the tractions can be replaced by discrete nodal values which leads to

$$u_j(\mathbf{y}) = N_k(\mathbf{y}(\xi, \eta))u_{kj} \quad \text{and} \quad t_j(\mathbf{y}) = N_k(\mathbf{y}(\xi, \eta))t_{kj}. \quad (25)$$

Herein, ξ and η are the local element coordinates. The index k represents the k th node of element e and j represents one of the coordinate directions x , y , or z . Caused by the fact that the values u_{kj} and t_{kj} can belong to more than one element, it is necessary to replace the summation over all elements e in (24) by an assembly,

$$\begin{aligned} c_{ij} N_k(\mathbf{x}(\xi, \eta)) u_{kj} &= \bigcup_e \int_{S_e} u_{ij}^*(\mathbf{x}, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} t_{kj} \\ &\quad - \bigcup_e \int_{S_e} t_{ij}^*(\mathbf{x}, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} u_{kj} + \sum_e B_{ie}(\mathbf{x}). \end{aligned} \quad (26)$$

Within this chapter, collocation methods are applied to obtain a discretised system of equations from (26). Assuming that the body is discretised by n nodes, the load point \mathbf{x} is set to the location of each node \mathbf{x}_l with $l = 1, \dots, n$. That means, instead of the arbitrary load point \mathbf{x} in (26), the nodal coordinates \mathbf{x}_l of the discretised geometry are used as collocation points leading to

$$\begin{aligned} c_{ij} u_{1j} &= \bigcup_e \int_{S_e} u_{ij}^*(\mathbf{x}_1, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} t_{kj} \\ &\quad - \bigcup_e \int_{S_e} t_{ij}^*(\mathbf{x}_1, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} u_{kj} + \sum_e B_{ie}(\mathbf{x}_1), \\ &\quad \vdots \\ c_{ij} u_{lj} &= \bigcup_e \int_{S_e} u_{ij}^*(\mathbf{x}_l, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} t_{kj} \\ &\quad - \bigcup_e \int_{S_e} t_{ij}^*(\mathbf{x}_l, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} u_{kj} + \sum_e B_{ie}(\mathbf{x}_l), \\ &\quad \vdots \\ c_{ij} u_{nj} &= \bigcup_e \int_{S_e} u_{ij}^*(\mathbf{x}_n, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} t_{kj} \\ &\quad - \bigcup_e \int_{S_e} t_{ij}^*(\mathbf{x}_n, \mathbf{y}(\xi, \eta)) N_k(\mathbf{y}(\xi, \eta)) d\tilde{S} u_{kj} + \sum_e B_{ie}(\mathbf{x}_n). \end{aligned} \quad (27)$$

The integrals from (27) can be evaluated with respect to the element surface S_e . By assembling (27) over all elements e , the system of equations

$$(\mathbf{C} + \mathbf{H}^*)\mathbf{u} = \mathbf{G}\mathbf{t} + \mathbf{B} \quad (28)$$

is obtained. Herein, the matrix \mathbf{C} contains the boundary factors c_{ij} . According to (27), this matrix is block diagonal. Due to the use of the fundamental solutions u_{ij}^* and t_{ij}^* , the matrices \mathbf{H}^* and \mathbf{G} are fully populated, unsymmetrical and not necessarily positive definite. The vector \mathbf{B} contains the projected body forces. The system of equations (28) can be written as

$$\mathbf{H}\mathbf{u} = \mathbf{G}\mathbf{t} + \mathbf{B} \quad (29)$$

which is the form known from literature [1, 5, 17].

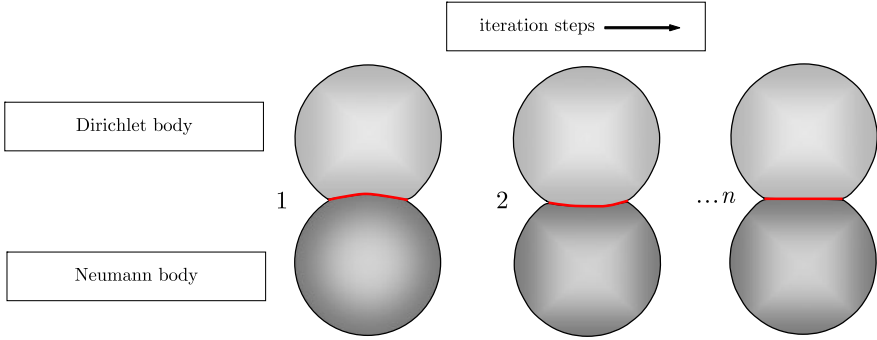


Fig. 3 Principle of DIRICHLET-to-NEUMANN algorithm

4 Contact Calculation by a DIRICHLET-to-NEUMANN Algorithm

This contribution presents an extension to the LAGRANGIAN multiplier approach introduced in [42]. The algorithm is introduced to reduce the amount of random access memory and to decrease the calculation time for large systems. The algorithm presented here is based on [25] and [38]. The principle of this DIRICHLET-to-NEUMANN algorithm which is called a nonlinear GAUSS-SEIDEL block in [24] is shown in Fig. 3.

The two contacting bodies are divided into a NEUMANN body and a DIRICHLET body. In this chapter the mortar and non-mortar are equivalent to the NEUMANN and DIRICHLET body, respectively. Within each iteration, a linear NEUMANN problem and a nonlinear DIRICHLET problem is solved. Starting from (29), the system of equations for the NEUMANN body can be expressed by

$$\mathbf{H}^N \Delta \mathbf{u}^{N(j)} = \mathbf{G}^N \Delta \mathbf{t}^{N(j)} + \mathbf{B}^{N(j)}, \quad (30)$$

where the index j represents the iteration step within the DIRICHLET-to-NEUMANN algorithm and the upper right index N denotes the NEUMANN body. The system of equations can be partitioned into degrees of freedom that are in contact and such that are not,

$$\begin{bmatrix} \mathbf{H}_{\text{nc}}^N & \mathbf{H}_{\text{cc}}^N \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u}_{\text{nc}}^{N(j)} \\ \Delta \mathbf{u}_{\text{cc}}^{N(j)} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{\text{nc}}^N & \mathbf{G}_{\text{cc}}^N \end{bmatrix} \begin{bmatrix} \Delta \mathbf{t}_{\text{nc}}^{N(j)} \\ \Delta \mathbf{t}_{\text{cc}}^{N(j)} \end{bmatrix} + \mathbf{B}^{N(j)}. \quad (31)$$

Herein, the degrees of freedom which are not in contact are denoted by the lower right indices nc and those which are in contact by cc, respectively. The body forces $\mathbf{B}^{N(j)}$ are typically applied within the first iteration, that means $\mathbf{B}^{N(j)} \stackrel{!}{=} \mathbf{0}$ for $j > 1$. As indicated by the name of the algorithm, the tractions caused by the contact are applied to the NEUMANN body, where for the first iteration step, thus $j = 1$, the contact tractions $\Delta \mathbf{t}_{\text{cc}}^{N(1)}$ are assumed to be zero. Considering also the other boundary

conditions in the non-contact area and sorting the system of equations (31) with respect to known and unknown values leads to

$$\begin{bmatrix} -\mathbf{G}_{\text{ncD}}^{\text{N}} & \mathbf{H}_{\text{ncN}}^{\text{N}} & \mathbf{H}_{\text{cc}}^{\text{N}} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{t}_{\text{ncD}}^{\text{N}(j)} \\ \Delta \mathbf{u}_{\text{ncN}}^{\text{N}(j)} \\ \Delta \mathbf{u}_{\text{cc}}^{\text{N}(j)} \end{bmatrix} = \begin{bmatrix} -\mathbf{H}_{\text{ncD}}^{\text{N}} & \mathbf{G}_{\text{ncN}}^{\text{N}} & \mathbf{G}_{\text{cc}}^{\text{N}} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u}_{\text{ncD}}^{\text{N}(j)} \\ \Delta \mathbf{t}_{\text{ncN}}^{\text{N}(j)} \\ \Delta \mathbf{t}_{\text{cc}}^{\text{N}(j)} \end{bmatrix} + \mathbf{B}^{\text{N}(j)}. \quad (32)$$

Equation (32) represents a fully populated quadrangular system of equation of the form $\mathbf{A}^{\text{N}} \mathbf{x}^{\text{N}} = \mathbf{b}^{\text{N}}$ which can be solved by GAUSS or LU decomposition. The deformed mesh of the NEUMANN body is stored after each iteration and acts as rigid obstacle for the DIRICHLET body. In order to obtain the overall deformations and tractions the incremental deformations $\Delta \mathbf{u}^{\text{N}(j)}$ and tractions $\Delta \mathbf{t}^{\text{N}(j)}$ are summarised over all iterations j . According to [25] a numerical damping parameter δ_{D} with $0 < \delta_{\text{D}} \leq 1$ is introduced, resulting in

$$\mathbf{u}^{\text{N}(j)} = \mathbf{u}^{\text{N}(j-1)} + \delta_{\text{D}} \Delta \mathbf{u}^{\text{N}(j)} \quad \text{and} \quad \mathbf{t}^{\text{N}(j)} = \mathbf{t}^{\text{N}(j-1)} + \delta_{\text{D}} \Delta \mathbf{t}^{\text{N}(j)}. \quad (33)$$

The actual nodal coordinates of the NEUMANN body are obtained by

$$\mathbf{x}^{\text{N}(j)} = \mathbf{x}^{\text{N}(j-1)} + \delta_{\text{D}} \Delta \mathbf{u}^{\text{N}(j)}. \quad (34)$$

The second step within each iteration is the solution of a nonlinear DIRICHLET problem. The formulation starts from the system of equations

$$\mathbf{H}^{\text{D}} \mathbf{u}^{\text{D}(j)} = \mathbf{G}^{\text{D}} \mathbf{t}^{\text{D}(j)} + \mathbf{B}^{\text{D}}, \quad (35)$$

where the upper right index D denotes the DIRICHLET body and j the number of iteration. Comparing (30) and (35) differences can be determined between the systems of equations of the NEUMANN and the DIRICHLET body. The calculation of the DIRICHLET body always starts from the reference configuration $\mathbf{x}^{\text{D}(1)}$ within each iteration step j , which means in general the undeformed configuration. Hence, no incremental update has to be done for the displacements $\mathbf{u}^{\text{D}(j)}$ and $\mathbf{t}^{\text{D}(j)}$. The body forces \mathbf{B}^{D} have to be applied within each iteration step or by superposition once in a preprocessing step.

The system of equations (35) can be partitioned and sorted leading to

$$\begin{bmatrix} -\mathbf{G}_{\text{ncD}}^{\text{D}} & \mathbf{H}_{\text{ncN}}^{\text{D}} & \mathbf{H}_{\text{cc}}^{\text{D}} & -\mathbf{G}_{\text{cc}}^{\text{D}} \end{bmatrix} \begin{bmatrix} \mathbf{t}_{\text{ncD}}^{\text{D}(j)} \\ \mathbf{u}_{\text{ncN}}^{\text{D}(j)} \\ \mathbf{u}_{\text{cc}}^{\text{D}(j)} \\ \mathbf{t}_{\text{cc}}^{\text{D}(j)} \end{bmatrix} = \begin{bmatrix} -\mathbf{H}_{\text{ncD}}^{\text{D}} & \mathbf{G}_{\text{ncN}}^{\text{D}} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{\text{ncD}}^{\text{D}(j)} \\ \mathbf{t}_{\text{ncN}}^{\text{D}(j)} \end{bmatrix} + \mathbf{B}^{\text{D}}, \quad (36)$$

which represents a system of linear equations of the form $\mathbf{A}^{\text{D}} \mathbf{x}^{\text{D}(j)} = \mathbf{b}^{\text{D}(j)}$. Due to the fact that within the potential contact area the displacements $\mathbf{u}_{\text{cc}}^{\text{D}(j)}$ and the tractions $\mathbf{t}_{\text{cc}}^{\text{D}(j)}$ are unknown, the system of equations (36) is rectangular, that means the number of equations is lower than the number of unknown nodal values. This results in an infinite number of possible solutions.

To solve (36), a quadratic optimisation problem with equality and inequality constraints is formulated,

$$\begin{aligned} \frac{1}{2} \mathbf{x}^{\text{D}(j)\text{T}} \mathbf{Q}^{\text{D}(j)} \mathbf{x}^{\text{D}(j)} + \mathbf{x}^{\text{D}(j)\text{T}} \mathbf{p}^{\text{D}(j)} &= \min_{\mathbf{x}^{\text{D}(j)}} \\ \text{subject to } \mathbf{A}^{\text{D}} \mathbf{x}^{\text{D}(j)} &= \mathbf{b}^{\text{D}(j)} \quad \text{and} \quad \mathbf{N}^{\text{D}(j)} \mathbf{x}^{\text{D}(j)} + \mathbf{d}_0^{\text{D}(j)} \geq \mathbf{0}. \end{aligned} \quad (37)$$

Herein $\mathbf{Q}^{\text{D}(j)}$ represents the objective function matrix containing the weighting factors of the nodal values $\mathbf{x}^{\text{D}(j)}$, and $\mathbf{p}^{\text{D}(j)}$ contains the coefficients of the linear part of the quadratic objective function. Typically, $\mathbf{Q}^{\text{D}(j)}$ is a unit matrix, and the elements of $\mathbf{p}^{\text{D}(j)}$ are zero. If the objective function matrix $\mathbf{Q}^{\text{D}(j)}$ is equal to the identity matrix, the result of such an optimisation is identical to the MOORE-PENROSE inverse of the left-hand side matrix \mathbf{A}^{D} given in (36). As equality condition the system of equations (36) is used. The inequality equations $\mathbf{N}^{\text{D}(j)} \mathbf{x}^{\text{D}(j)} + \mathbf{d}_0^{\text{D}(j)} \geq \mathbf{0}$ represent the one-sided constraints which are built up with respect to the deformed surface of the NEUMANN body representing the mortar side of the contacting bodies. The matrix $\mathbf{N}^{\text{D}(j)}$ containing the normal vectors and the vector $\mathbf{d}_0^{\text{D}(j)}$ containing the initial gaps have to be built up within each iteration step j .

The theoretical background and computational implementation of the quadratic programming is not explained here. A description and implementation on that topic is given by [26].

If the optimisation is successful, the reaction forces in form of the tractions $\mathbf{t}_{\text{cc}}^{\text{D}(j)}$ are obtained. The sum of the tractions inside the contact area has to vanish,

$$\mathbf{t}_{\text{cc}}^{\text{D}(j)} + \mathbf{t}_{\text{cc}}^{\text{N}(j)} = \mathbf{0}. \quad (38)$$

Because of the possibility of nonconforming meshes of the contacting bodies, dynamic constraints based on a mortar method not explained so far are introduced. This leads to a reformulation of (38),

$$\mathbf{M}_t^{\text{D}(j)} \mathbf{t}_{\text{cc}}^{\text{D}(j)} + \mathbf{M}_t^{\text{N}(j)} \mathbf{t}_{\text{cc}}^{\text{N}(j)} = \mathbf{0}, \quad \mathbf{0} \in \mathbb{R}^N, \quad (39)$$

where the mortar matrices $\mathbf{M}_t^{\text{D}(j)}$ and $\mathbf{M}_t^{\text{N}(j)}$ have to be built up within each iteration based on the actual geometry of the NEUMANN body as it will be shown in Sect. 5. Due to the fact that mortar matrices are formed with respect to the degrees of freedom in contact on the NEUMANN body representing the mortar side, the matrix $\mathbf{M}_t^{\text{N}(j)}$ is quadratic and has a full rank. Inversion of the matrix $\mathbf{M}_t^{\text{N}(j)}$ in (39) leads to

$$\mathbf{t}_{\text{cc}}^{\text{N}(j)} = -\mathbf{M}_t^{\text{N}(j)-1} \mathbf{M}_t^{\text{D}(j)} \mathbf{t}_{\text{cc}}^{\text{D}(j)}, \quad (40)$$

which represents a mapping of the tractions in the contact area from the DIRICHLET body to the NEUMANN body. Because the system of equations (30) is given in an incremental form, the incremental tractions $\Delta \mathbf{t}_{\text{cc}}^{\text{N}(j)}$ are calculated by

$$\Delta \mathbf{t}_{\text{cc}}^{\text{N}(j)} = \delta_N (\mathbf{t}_{\text{cc}}^{\text{N}(j)} - \mathbf{t}_{\text{cc}}^{\text{N}(j-1)}), \quad (41)$$

where δ_N is an additional numerical damping parameter with $0 < \delta_N \leq 1$. For further information on building up mortar matrices see Sect. 5. A summary of the global solution algorithm is given in Algorithm 1.

| |
|---|
| <p>input : A geometric set comprising <i>elements</i> and <i>nodes</i>, System matrices $\mathbf{H}^D, \mathbf{H}^N, \mathbf{G}^D, \mathbf{G}^N$, Boundary conditions $\Delta \mathbf{u}_{ncD}^{N(j)}, \Delta \mathbf{t}_{ncN}^{N(j)}, \mathbf{u}_{ncD}^{D(j)}, \mathbf{t}_{ncN}^{D(j)}$</p> <p>output: Displacements ($\mathbf{u}^D, \mathbf{u}^N$) and tractions ($\mathbf{t}^D, \mathbf{t}^N$) caused by contact</p> <pre> 1 for $j = 1, 2, \dots$, <i>until convergence</i> do 2 do collision detection 3 \rightarrow List of element pairs which are potentially in contact 4 Build-up of mortar matrices for tractions according to Sect. 5 5 $\rightarrow \mathbf{M}_t^{D(j)}, \mathbf{M}_t^{N(j)}$ 6 Transfer tractions: $\mathbf{t}_{cc}^{N(j)} = -\mathbf{M}_t^{N(j)-1} \mathbf{M}_t^{D(j)} \mathbf{t}_{cc}^{D(j-1)}$, (40) 7 <i>Remark</i>: $\mathbf{t}_{cc}^{D(j-1)} = \mathbf{0}$ for $j = 1$ 8 Solve linear NEUMANN problem: $\mathbf{H}^N \Delta \mathbf{u}^{N(j)} = \mathbf{G}^N \Delta \mathbf{t}^{N(j)} + \mathbf{B}^{N(j)}$, (30) 9 Update geometry of NEUMANN body: $\mathbf{x}^{N(j)} = \mathbf{x}^{N(j-1)} + \delta_D \Delta \mathbf{u}^{N(j)}$, (34) 10 Project nodes from DIRICHLET body on actual geometry of the NEUMANN body 11 \rightarrow normal vector matrix $\mathbf{N}^{D(j)}$, initial gap vector $\mathbf{d}^{D(j)}$ 12 Solve nonlinear DIRICHLET problem: $\frac{1}{2} \mathbf{x}^{D(j)T} \mathbf{Q}^{D(j)} \mathbf{x}^{D(j)} + \mathbf{x}^{D(j)T} \mathbf{p}^{D(j)} \rightarrow \text{MIN}$ 13 subject to: $\mathbf{A}^D \mathbf{x}^{D(j)} = \mathbf{b}^{D(j)}$ and $\mathbf{N}^{D(j)} \mathbf{x}^{D(j)} + \mathbf{d}^{D(j)} \geq \mathbf{0}$, (37) 14 \rightarrow contact tractions $\mathbf{t}_{cc}^{D(j)}$ 15 Check for convergence: $\ \mathbf{t}_{cc}^{N(j)} - \mathbf{t}_{cc}^{N(j-1)}\ \leq \text{TOL} \rightarrow \text{exit loop}$ 16 end </pre> |
|---|

Algorithm 1: DIRICHLET-to-NEUMANN algorithm

Note that if the damping parameters δ_D in (36) and δ_N in (41) are equal to 1, no convergence will be achieved. In [25] it is suggested to choose 0.7 for δ_D and δ_N . A value of 1.0 for the parameter δ_D and a value of 0.5 for the parameter δ_N for the algorithm described above is recommended. Due to the fact that the surface tractions are calculated, fast convergence is achieved if the half of the tractions are transferred from the DIRICHLET body to the NEUMANN body.

5 Build-up of Mortar Matrices

In this section the theoretical background of the mortar methods for contact and the numerical implementation to create the mortar matrices \mathbf{M}_t and \mathbf{M}_u from (39) are described. The aim of the mortar methods is to find a mapping matrix between the

degrees of freedom of the contacting bodies. The basic concept presented here was published by [30] for finite elements. It is here adapted for boundary elements. Similar to Fig. 3, the contacting bodies are divided into a mortar and non-mortar body, denoted by the indices m and nm , respectively, that are equivalent to the NEUMANN body and the DIRICHLET body, respectively.

As mentioned before, mortar contact belongs to the group of contacts which is based on LAGRANGIAN multipliers. The unilateral constraint for contact formulation can be defined using the KUHN-TUCKER-KARUSH conditions,

$$\begin{array}{llll} \text{1-dimensional} & d \geq 0 & \lambda \leq 0 & \lambda d = 0, \\ \text{2- or 3-dimensional} & \mathbf{d} \geq \mathbf{0} & \boldsymbol{\lambda} \leq \mathbf{0} & \boldsymbol{\lambda}^T \mathbf{d} = 0. \end{array} \quad (42)$$

The first line of (42) presents the constraints for one-dimensional problems. As the gap cannot be negative, $d \geq 0$, and only pressure forces can be transferred, $\lambda \leq 0$, the product of the gap d and the LAGRANGIAN multiplier λ always vanishes.

For two- and three-dimensional problems typically a distinction has to be made between normal and tangential contact. For normal contact, the same constraints as for one-dimensional contact can be used where vectors of the gap \mathbf{d} and of the LAGRANGIAN multiplier $\boldsymbol{\lambda}$ have to be projected in normal direction. Stiction and sliding may occur in tangential direction. In case of sliding, the tangential forces act as applied forces in opposite directions of the relative motion so that force laws such as COULOMB's law can be applied. In case of stiction, no motion in tangential direction occurs. In that case, the tangential contact can also be treated as a constraint and the resulting tangential tractions have to be treated as reaction stresses. Hence, the normal and tangential contact can be summarised according to the second line of (42).

The starting point for building up three-dimensional mortar matrices is the weak form of the KUHN-TUCKER-KARUSH condition,

$$\Pi_c^{LM} = \int_{S_{cc}} \boldsymbol{\lambda}^T \mathbf{d} \, dS. \quad (43)$$

Due to the weak form, the constraints are fulfilled in an integral meaning over the potential contact area S_{cc} only. The variation of the weak form from (43) leads to

$$\delta \Pi_c^{LM} \equiv C_c^{LM} = \int_{S_{cc}} \delta \boldsymbol{\lambda}^T \mathbf{d} \, dS + \int_{S_{cc}} \boldsymbol{\lambda}^T \delta \mathbf{d} \, dS = 0. \quad (44)$$

The first integral in (44) represents the fulfilment of the gap function \mathbf{d} choosing an arbitrary LAGRANGIAN multiplier vector $\boldsymbol{\lambda}$. The second integral represents the fulfilment of the reaction forces for an arbitrary chosen gap function \mathbf{d} . In case of contact the gap between the two contacting bodies is closed, and the gap function \mathbf{d} can be replaced by the displacements of the mortar body ${}^m \mathbf{u}^m$ and of the non-mortar body ${}^m \mathbf{u}^{nm}$, and the initial gap ${}^m \mathbf{d}_0^m$,

$$C_c^{LM} = \int_{S_{cc}} \delta \boldsymbol{\lambda}^T ({}^m \mathbf{u}^{nm} - {}^m \mathbf{u}^m - {}^m \mathbf{d}_0^m) \, dS + \int_{S_{cc}} ({}^m \boldsymbol{\nu}^{nm} + {}^m \boldsymbol{\nu}^m) \delta \mathbf{d}^T \, dS = 0. \quad (45)$$

The upper left index m denotes that the displacement vectors are expressed with respect to the body-fixed mortar reference frame \mathcal{K}_m . Additionally, the LAGRANGIAN

multiplier λ in the second integral can be replaced by the sum of the tractions of the mortar body ${}^m\mathbf{t}^m$ and of the non-mortar body ${}^m\mathbf{t}^{nm}$. Hence, the equilibrium in the contact area is fulfilled by an integral meaning only.

The LAGRANGIAN multipliers λ and their variations $\delta\lambda$ can be discretised in the same way as it was done for displacements and tractions. This leads to

$$\lambda(\xi, \eta) = \sum_i M_i^\lambda(\xi, \eta)\lambda_i, \quad \delta\lambda(\xi, \eta) = \sum_i M_i^\lambda(\xi, \eta)\delta\lambda_i, \quad (46)$$

where $M_i^\lambda(\xi, \eta)$ represents the general shape functions over one element e and λ_i and $\delta\lambda_i$ the corresponding nodal values. The same procedure can be applied to the gap function \mathbf{d} and the corresponding variation $\delta\mathbf{d}$,

$$\mathbf{d}(\xi, \eta) = \sum_i M_i^d(\xi, \eta)\mathbf{d}_i, \quad \delta\mathbf{d}(\xi, \eta) = \sum_i M_i^d(\xi, \eta)\delta\mathbf{d}_i. \quad (47)$$

Inserting (46) and (47) into (45) leads to the constraint equations for one pair of contact elements,

$$\begin{aligned} C_{cc}^{LM} = & \sum_i \delta\lambda_i^T \int_{S_{ecc}} \left(\sum_j M_i^\lambda(\xi_{nm}, \eta_{nm}) N_j(\xi_{nm}, \eta_{nm}) {}^m\mathbf{u}_j^{nm} \right. \\ & - \sum_j M_i^\lambda(\xi_m, \eta_m) N_j(\xi_m, \eta_m) {}^m\mathbf{u}_j^m \\ & \left. - \sum_j M_i^\lambda(\xi_m, \eta_m) N_j(\xi_m, \eta_m) {}^m\mathbf{d}_{0j}^m \right) dS \\ & + \sum_i \delta\mathbf{d}_i^T \int_{S_{ecc}} \left(\sum_j M_i^d(\xi_{nm}, \eta_{nm}) N_j(\xi_{nm}, \eta_{nm}) {}^m\mathbf{t}_j^{nm} \right. \\ & \left. + \sum_j M_i^d(\xi_m, \eta_m) N_j(\xi_m, \eta_m) {}^m\mathbf{t}_j^m \right) dS = 0. \end{aligned} \quad (48)$$

The nodal values ${}^m\mathbf{u}_j^{nm}$ and ${}^m\mathbf{t}_j^{nm}$ for the degrees of freedom on the non-mortar side are given with respect to the body-fixed reference frame \mathcal{K}_m , see (48). Typically the system matrices \mathbf{H}^{nm} and \mathbf{G}^{nm} are calculated with respect to the body-fixed reference frame \mathcal{K}_{nm} . The corresponding displacements and tractions are also given with respect to \mathcal{K}_{nm} . An additional transformation for the current orientation between \mathcal{K}_m and \mathcal{K}_{nm} is introduced so that

$${}^m\mathbf{u}_i^{nm} = {}^{nm}{}^m\mathbf{T}^{nm}\mathbf{u}_i^{nm} \quad \text{and} \quad {}^m\mathbf{t}_i^{nm} = {}^{nm}{}^m\mathbf{T}^{nm}\mathbf{t}_i^{nm}, \quad (49)$$

where ${}^{nm}{}^m\mathbf{T}$ is the transformation matrix between body-fixed reference frames on mortar body m and the non-mortar body nm . If (48) vanishes, the constraints for displacements and tractions are fulfilled. Considering (49) and using the fact that

(48) vanishes for arbitrary variations of the LAGRANGIAN multipliers $\delta\boldsymbol{\lambda}$ and gap functions $\delta\mathbf{d}$, the contact constraints for one pair of contact elements become

$$\begin{aligned} & \int_{S_{\text{ecc}}} \left(\sum_j M_i^\lambda(\xi_{nm}, \eta_{nm}) N_j(\xi_{nm}, \eta_{nm})^{mnm} \mathbf{T}^{nm} \mathbf{u}_j^{nm} \right. \\ & \quad - \sum_j M_i^\lambda(\xi_m, \eta_m) N_j(\xi_m, \eta_m)^m \mathbf{u}_j^m \\ & \quad \left. - \sum_j M_i^\lambda(\xi_m, \eta_m) N_j(\xi_m, \eta_m)^m \mathbf{d}_{0j}^m \right) dS = \mathbf{0}, \\ & \int_{S_{\text{ecc}}} \left(\sum_j M_i^d(\xi_{nm}, \eta_{nm}) N_j(\xi_{nm}, \eta_{nm})^{mnm} \mathbf{T}^{nm} \boldsymbol{\tau}_j^{nm} \right. \\ & \quad \left. + \sum_j M_i^d(\xi_m, \eta_m) N_j(\xi_m, \eta_m)^m \boldsymbol{\tau}_j^m \right) dS = \mathbf{0}. \end{aligned} \quad (50)$$

For a pair of bilinear elements, the first equation of (50) results for the displacements on the non-mortar side to the matrix

$$\mathbf{M}_u^{nm e} = \begin{bmatrix} \int_{S_{\text{ecc}}} M_1^\lambda N_1 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_1^\lambda N_2 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_1^\lambda N_3 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_1^\lambda N_4 dS^{mnm} \mathbf{T} \\ \int_{S_{\text{ecc}}} M_2^\lambda N_1 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_2^\lambda N_2 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_2^\lambda N_3 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_2^\lambda N_4 dS^{mnm} \mathbf{T} \\ \int_{S_{\text{ecc}}} M_3^\lambda N_1 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_3^\lambda N_2 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_3^\lambda N_3 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_3^\lambda N_4 dS^{mnm} \mathbf{T} \\ \int_{S_{\text{ecc}}} M_4^\lambda N_1 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_4^\lambda N_2 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_4^\lambda N_3 dS^{mnm} \mathbf{T} & \int_{S_{\text{ecc}}} M_4^\lambda N_4 dS^{mnm} \mathbf{T} \end{bmatrix} \quad (51)$$

and for the displacements on the mortar side to the matrix

$$\mathbf{M}_u^m = \begin{bmatrix} \int_{S_{\text{ecc}}} M_1^\lambda N_1 dSE & \int_{S_{\text{ecc}}} M_1^\lambda N_2 dSE & \int_{S_{\text{ecc}}} M_1^\lambda N_3 dSE & \int_{S_{\text{ecc}}} M_1^\lambda N_4 dSE \\ \int_{S_{\text{ecc}}} M_2^\lambda N_1 dSE & \int_{S_{\text{ecc}}} M_2^\lambda N_2 dSE & \int_{S_{\text{ecc}}} M_2^\lambda N_3 dSE & \int_{S_{\text{ecc}}} M_2^\lambda N_4 dSE \\ \int_{S_{\text{ecc}}} M_3^\lambda N_1 dSE & \int_{S_{\text{ecc}}} M_3^\lambda N_2 dSE & \int_{S_{\text{ecc}}} M_3^\lambda N_3 dSE & \int_{S_{\text{ecc}}} M_3^\lambda N_4 dSE \\ \int_{S_{\text{ecc}}} M_4^\lambda N_1 dSE & \int_{S_{\text{ecc}}} M_4^\lambda N_2 dSE & \int_{S_{\text{ecc}}} M_4^\lambda N_3 dSE & \int_{S_{\text{ecc}}} M_4^\lambda N_4 dSE \end{bmatrix} \quad (52)$$

with the 3×3 identity matrix \mathbf{E} . The compatibility matrices for the tractions $\mathbf{M}_t^{m e}$ and $\mathbf{M}_t^{nm e}$ are created with the same procedure. Assembling the matrices of all element pairs which are possibly in contact leads to

$$\begin{aligned} \mathbf{M}_u^m &= \bigcup_e \mathbf{M}_u^{m e}, & \mathbf{M}_u^{nm} &= \bigcup_e \mathbf{M}_u^{nm e}, \\ \mathbf{M}_t^m &= \bigcup_e \mathbf{M}_t^{m e}, & \mathbf{M}_t^{nm} &= \bigcup_e \mathbf{M}_t^{nm e}. \end{aligned} \quad (53)$$

Concluding some remarks on the shape functions $M_i(\xi, \eta)$ used for the interpolation of the gap \mathbf{d} and the LAGRANGIAN multipliers $\boldsymbol{\lambda}$ are given. According to [18, 37, 39], the shape functions M_i^λ for the interpolation of the LAGRANGIAN multipliers $\boldsymbol{\lambda}$ in (46) and M_i^d for the interpolation of gap \mathbf{d} according to (47) have to fulfil the BABUŠKA-BREZZI or *inf-sup* condition. A detailed description on that topic is provided by [9]. This condition ensures a unique solution and the maximum rank of the mortar matrices \mathbf{M}_u^{nm} and \mathbf{M}_t^m . To realise this condition, the function

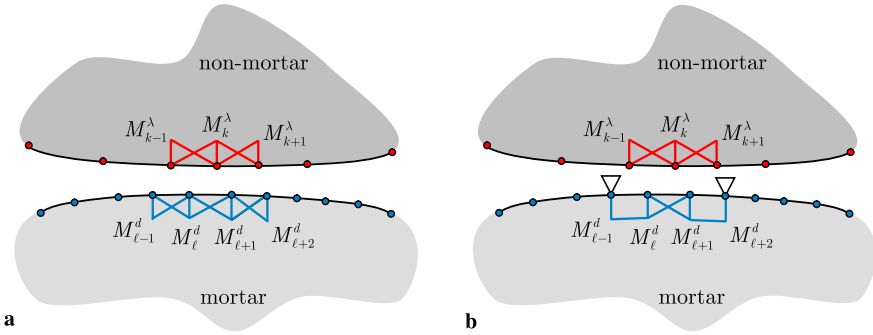


Fig. 4 Approximation of the LAGRANGIAN multiplier λ and the gap function \mathbf{d} within the contact area. **a** Without boundary conditions. **b** With DIRICHLET boundary conditions

space of the shape functions M_i^λ and M_i^d has to be sufficiently rich. Many research work has been done to obtain an optimal set of shape functions used for interpolation of LAGRANGIAN multipliers for domain decomposition within finite element calculations, see [36]. A dual LAGRANGIAN multiplier formulation was proposed by [36] which leads to diagonal mortar matrices. In the present work the original approach as described by [6] is implemented. The same interpolation functions are used for the approximation of the LAGRANGIAN multiplier λ and the gap function \mathbf{d} within the contact area as used for the interpolation of the displacements \mathbf{u} and the tractions \mathbf{t} . The use of that shape functions typically ensures the fulfilment of the BABUŠKA-BREZZI conditions. For two-dimensional contact problems, the interpolation functions are schematically shown in Fig. 4.

For the interpolation of the LAGRANGIAN multipliers λ the mesh on the non-mortar side is used, where for the interpolation of the gap function \mathbf{d} the mesh on the mortar side is used, as presented by M_i^λ and M_i^d in Fig. 4. Due to that formulation, a construction of an intermediate mesh as described by [34] is not necessary. Nevertheless to obtain mortar matrices with a maximum rank, the shape functions have to fulfil the BABUŠKA-BREZZI condition, which leads according to [34] to the simple requirement

$$\min(N^m, N^{nm}) \leq N^\lambda, N^d \leq \max(N^m, N^{nm}), \quad (54)$$

where N^m and N^{nm} represent the number of contact nodes of the mortar and non-mortar body, respectively. The corresponding numbers of nodes for the interpolation of the LAGRANGIAN multiplier and the gap function \mathbf{d} are given by N^λ and N^d . Due to the facts that the LAGRANGIAN multipliers are approximated using the potential contacting elements on the non-mortar side and the gap function is approximated using contacting elements on the mortar side, the requirement (54) is always fulfilled if no boundary conditions are applied at the boundary of the contact area. If DIRICHLET boundary conditions at the boundary of the contact area are applied, the shape functions M_i^λ and M_i^d at the boundary have to be modified, see Fig. 4b. According to [18, 34], the shape functions are kept constant to ensure the maximum rank of \mathbf{M}_u^{nm} and \mathbf{M}_t^m .

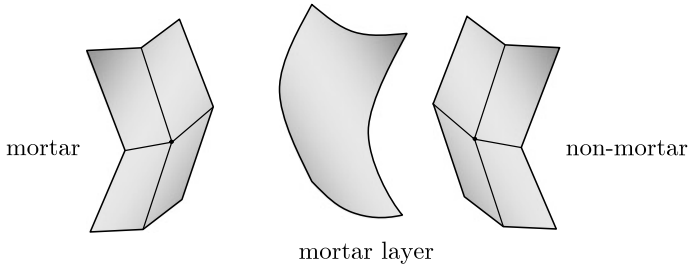


Fig. 5 Introduction of a mortar layer

6 Numerical Implementation of Mortar Matrices

First, a mortar layer is introduced between the mortar and non-mortar side, see Fig. 5.

The mortar layer can either be located as an intermediate surface between the mortar and the non-mortar surface as presented by [27, 28, 31], or one of the contacting surfaces can be chosen as shown in [25, 36]. According to [30], the numerical integration scheme is shown in Fig. 6 and in Algorithm 2.

First, the element facets are projected onto the common mortar layer, see Fig. 6a. Therefore, only the corner nodes are taken into account. For bilinear elements, this consideration is sufficient because two-dimensional bilinear elements consist of four nodes only, but in case of biquadratic elements where the edges could be curved errors may occur. For simplification and an easier handle of the projected elements, this error is neglected. This approach is valid if the differences between the geometry of the biquadratic element and the corresponding 4-node element facet are not too large.

In a second step, the common area due to the overlap of the projected elements is determined, see Fig. 6b. Therefore, a modified version of the SUTHERLAND-HODGMAN polygon clipping algorithm is used as presented in [16]. The algorithm described in [16] is valid for an axis parallel clipping only. The algorithm modifications presented here are valid for clipping with arbitrary 4-point polygons as presented in Fig. 7a.

The principle of the algorithm is to clip the parts *outside* of one polygon with respect to another one. This is done by dividing one of the elements into four clipping edges represented by the four projected edges of the element facet. According to Fig. 7a, the element 1234 is chosen to be the clipping element. The element 1'2'3'4' is clipped with respect to the element 1234. Therefore, a flow direction according to the arrow in Fig. 7a is chosen. In case of the chosen direction the area of the element 1'2'3'4' located on the right side of the clipping edges is cut off, because they are *outside* the element 1234. To create new corner points of the overlap polygon, the edges of the element 1'2'3'4' are cut off with respect to the clipping edges.

A calculation whether one point is lying *inside* or *outside* with respect to the clipping edge is necessary. According to Fig. 7b, this is done by a perpendicular projection of the point with respect to the clipping edge. The distance vector \mathbf{r}_{12}

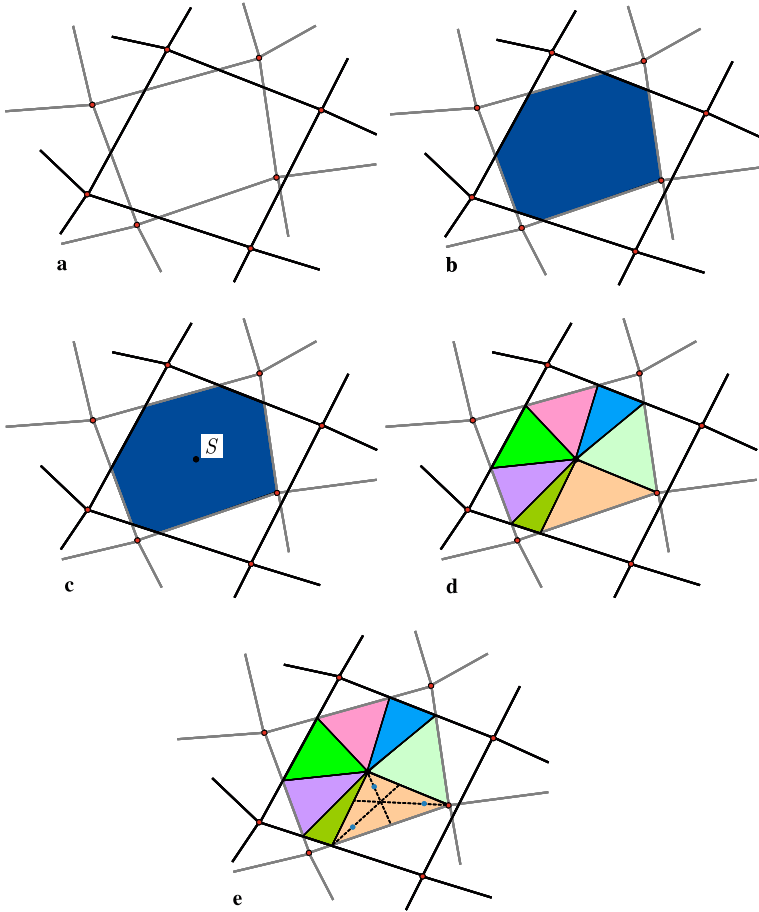


Fig. 6 Integration scheme for 3D mortar elements. **a** Projections of the element facets onto the mortar layer. **b** Common area of the projected facets. **c** Centre of area S . **d** Division of the polygon into triangles. **e** Locating GAUSS-RADAU integration points on the triangles

of the two points of the clipping edge and the corresponding normal vector \mathbf{n}_{12} are given by

$$\mathbf{r}_{12} = \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \end{bmatrix} \quad \text{and} \quad \mathbf{n}_{12} = \begin{bmatrix} y_2 - y_1 \\ x_1 - x_2 \end{bmatrix}. \quad (55)$$

The resulting clipping test is given by

$$\mathbf{n}_{12}^T \begin{bmatrix} x_P - x_1 \\ y_P - y_1 \end{bmatrix} \begin{cases} > 0 \rightarrow \textit{outside} \\ \leq 0 \rightarrow \textit{inside}. \end{cases} \quad (56)$$

The other parts of the algorithm remain unchanged, compare to [16]. Thereafter the centre of the area S of the resulting polygon, see Fig. 6c, is calculated by the

input : List of element pairs $Elp(i)$, which is possibly in contact
output: Mortar matrices \mathbf{M}_u^{nm} , \mathbf{M}_u^m , \mathbf{M}_t^{nm} , \mathbf{M}_t^m

1 **forall the element pairs $Elp(i)$ do**

1. projection of the element facets on a common mortar layer, see Fig. 6a
2. use clipping algorithm to form polygon of the overlap of the projected element facets, see Fig. 6b
3. locate geometric centre of the polygon, see Fig. 6c
4. divide polygon into n_t triangles, see Fig. 6d

forall the n_t triangles do

- locate GAUSS-RADAU integration points on the triangle, see Fig. 6e
- project GAUSS integration points on the mortar and non-mortar element to obtain ξ_m, η_m and ξ_{nm}, η_{nm}
- compute integrals over triangles, see (57)
- sum integral contributions, (58), to obtain $\mathbf{M}_u^{nme}, \mathbf{M}_u^{me}, \mathbf{M}_t^{nme}, \mathbf{M}_t^{me}$ according to (51)

end

2 **end**

3 assemble the element matrices according to (53)

Algorithm 2: Build-up of mortar matrices

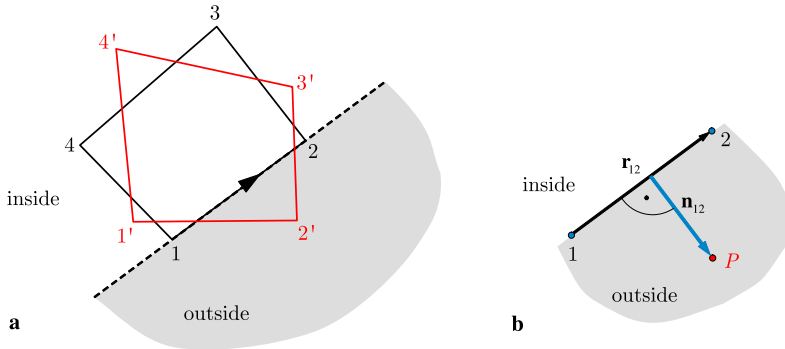
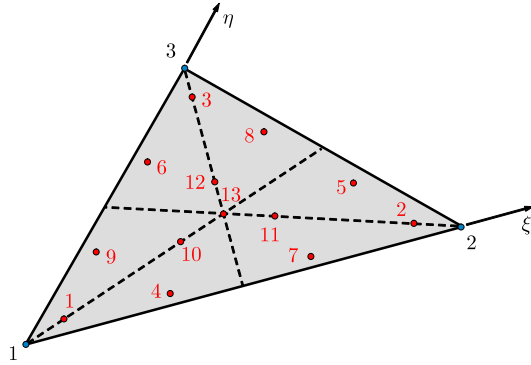


Fig. 7 Scheme of clipping algorithm. **a** Clipping with 4-point polygons. **b** Projection of an edge point P onto the normal of the clipping edge

arithmetic middle of the corner coordinates of the polygon. This results in a division of the polygon into triangles according to Fig. 6d. In the next step the GAUSS-RADAU integration points can be determined for each of the triangles according to

Fig. 8 GAUSS-RADAU integration points for triangles



[11, 14]. In [30] a thirteen point GAUSS-RADAU rule is recommended, see Fig. 8, to overcome the problem that warped meshes can not be integrated exactly.

In contrast to typical numerical integration, not the local but the absolute coordinates are necessary. These coordinates are projected back on the mortar and the non-mortar elements, respectively. As a result, the local coordinates ξ_m, η_m and ξ_{nm}, η_{nm} will be obtained. The integrals over one triangle from (51) can be expressed by

$$\int_{S_{et}} M_i(\xi_k, \eta_k) N_j(\xi_k, \eta_k) dS = \sum_{l=1}^{n_g} M_i(\xi_{kl}, \eta_{kl}) N_j(\xi_{kl}, \eta_{kl}) W_l, \quad (57)$$

where S_{et} is the surface of a triangle t of the element e while k represents the projection on the mortar m and non-mortar side nm , respectively. The variable n_g represents the number of integration points and W_l the corresponding weighting factors. The integral over the element surface S_{ecc} can be evaluated by

$$\int_{S_{ecc}} M_i(\xi_k, \eta_k) N_j(\xi_k, \eta_k) dS = \sum_{t=1}^{n_t} \int_{S_{et}} M_i(\xi_k, \eta_k) N_j(\xi_k, \eta_k) dS, \quad (58)$$

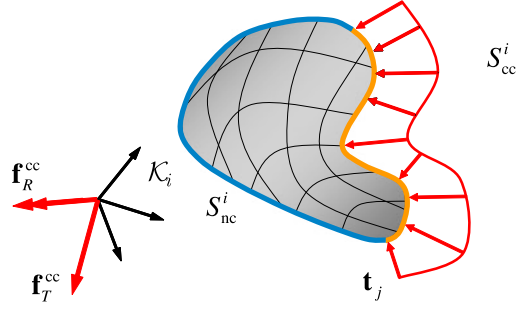
which represents the sum over all triangles. Finally, the integrals can be assembled according to (53).

7 From Contact Traction to Applied Forces

If the contact calculation inside the boundary element environment has converged, the tractions and displacements within the contact area are given. However, within the multibody environment only forces and torques can be treated. Therefore, the contact tractions \mathbf{t}_j have to be integrated with respect to the contact area to form the resulting wrench, consisting of the force \mathbf{f}_T^{cc} and torque \mathbf{f}_R^{cc} , see Fig. 9.

According to Fig. 9, contact tractions have to be integrated for each contacting body with respect to the body-fixed reference frame \mathcal{K}_i . The upper right index i denotes the mortar body m and non-mortar body nm , respectively. The body-fixed

Fig. 9 From contact tractions to resulting wrench



reference frame \mathcal{K}_i within the boundary element system is equal to the force application point within the multibody system. The resulting contact force \mathbf{f}_T^{cc} can be calculated by

$$\mathbf{f}_T^{\text{cc}} = \sum_e^{n_c} \left[\sum_k \int_{S_e} N_k \mathbf{t}_k d\tilde{S} \right], \quad (59)$$

where \mathbf{t}_i represents the surface traction vector of the i th node, N_i the corresponding shape functions, and S_e the element surface. The integral in (59) has to be evaluated over all n_c contacting elements. The resulting contact torque \mathbf{f}_R^{cc} built up by the tractions \mathbf{t}_i can be calculated by

$$\mathbf{f}_R^{\text{cc}} = \sum_e^{n_c} \left[\sum_k \sum_l \int_{S_e} \tilde{x}_{kl} N_l \mathbf{t}_l d\tilde{S} \right], \quad (60)$$

where \tilde{x}_{kl} are the components of the skew symmetric matrix representing the location of a point on the surface S_e with respect to the body-fixed reference frame \mathcal{K}_i . The integrals can be evaluated using standard GAUSS integration. The overall contact wrench $\hat{\mathbf{f}}^{\text{cc}}$ from (11) is obtained by assembling the calculated contact force \mathbf{f}_T^{cc} and torque \mathbf{f}_R^{cc} .

8 Dynamic Simulation of Two Contacting Spheres

To test the functionality of the algorithm, a simple multibody model is considered. A schematic representation of the dynamic model is given in Fig. 10a.

The one-dimensional model consists of two identical spheres, where the lower sphere is fixed at its lower side. The upper sphere is suspended at its upper side by a spring-damper element in vertical direction. The properties of the spring-damper element are defined by the stiffness coefficient k_{spring} and the damping coefficient b_{damper} . Because of the gravity g_{grav} , the upper sphere moves downwards until the initial gap d_0 is closed. Then elastic contact between both spheres based on the elastic behaviour of both curves occurs. The multibody model consists of two rigid spheres, see Fig. 10b. The multibody data of the spheres such as mass are calculated by the density ρ and the diameter d_{sph} . The calculation of the contact forces

Fig. 10 Model for dynamic simulation of two spheres.
a Schematic representation.
b Multibody model

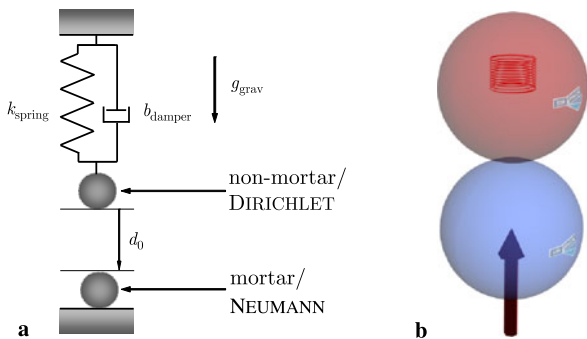


Table 1 Properties of the dynamic model

| Property | Value |
|---|--|
| YOUNG's modulus E | 210,000 MPa |
| POISSON's ratio ν | 0.3 |
| density ρ | 7,850.0 $\frac{\text{kg}}{\text{m}^3}$ |
| diameter d_{sph} | 1 m |
| initial gap d_0 | 0.2 m |
| gravity g_{grav} | 9.81 $\frac{\text{m}}{\text{s}^2}$ |
| stiffness coefficient k_{spring} | 40,000 $\frac{\text{N}}{\text{m}}$ |
| damping coefficient b_{damper} | 110 $\frac{\text{Ns}}{\text{m}}$ |

depends mainly on the motion of the reference frames of the spheres, which are located at the centres of the spheres. The data of the model are summarised in Table 1.

For comparison, a reference model with existing force elements of the multibody program SIMPACK™ was created. Here, the force element containing the developed BEM contact model is replaced by a HERTZIAN pressure element. Here, the bodies remain undeformed and body forces are also not considered in the reference model. The transition between no contact and contact is realised by an event function. The event function depends on the relative position and on the geometry of the contacting bodies. If the gap between the spheres is closed, the integrator is stopped and restarted with new initial conditions. The *Sodarst2* integrator from the multibody program SIMPACK™ is chosen. This integrator is based on an implicit formulation with automatic step size calculation.

The co-simulation is done by the use of the explicit EULER integrator. The constant step size is equal to 0.0002 s. This step size is chosen to overcome the impact problem of both spheres. The simulation time is 1 s. In contrast to the reference model, the body forces are considered. The contact gap of both simulations is shown in Fig. 11.

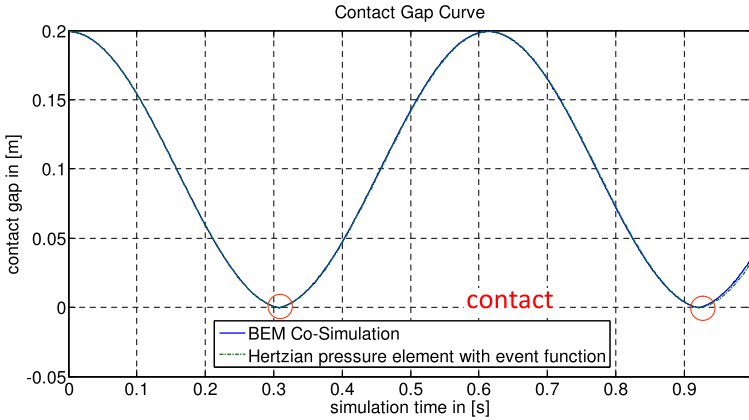


Fig. 11 Contact gap vs. time for BEM co-simulation and HERTZIAN pressure element

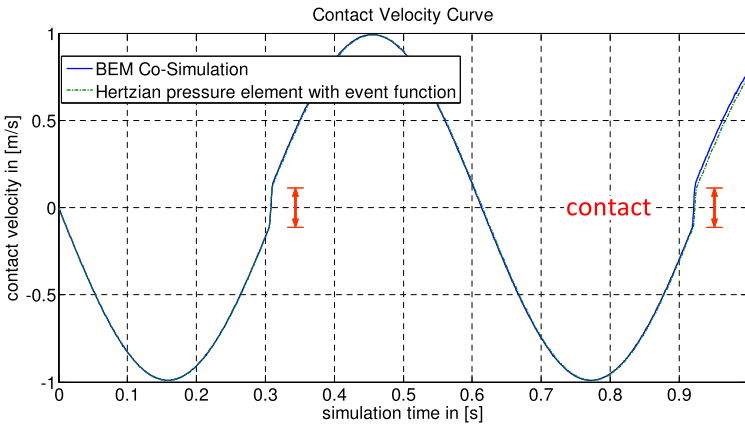


Fig. 12 Contact velocity vs. time for BEM co-simulation and HERTZIAN pressure element

The two curves agree well. Only small differences can be denoted at the end of the simulation time. The corresponding relative contact velocity of the upper contact point with respect to the lower contact point is shown in Fig. 12.

Herein also small differences between the two curves can be noticed at the end of the simulation time. The contact forces of both simulations are shown in Fig. 13.

The peaks of the two contact force curves differ from each other. The first peak of the HERTZIAN reference model reaches a value of -244.942 kN. The corresponding contact force value of the BEM co-simulation is obtained with -276.003 kN, which means a difference of -12.68 % compared to the reference model. Higher pressure force are to be expected due to the consideration of body forces. Additional differences occur due to the integrators chosen. These differences can be seen at the second peak of both curves. The contact force values are obtained as -223.215 kN

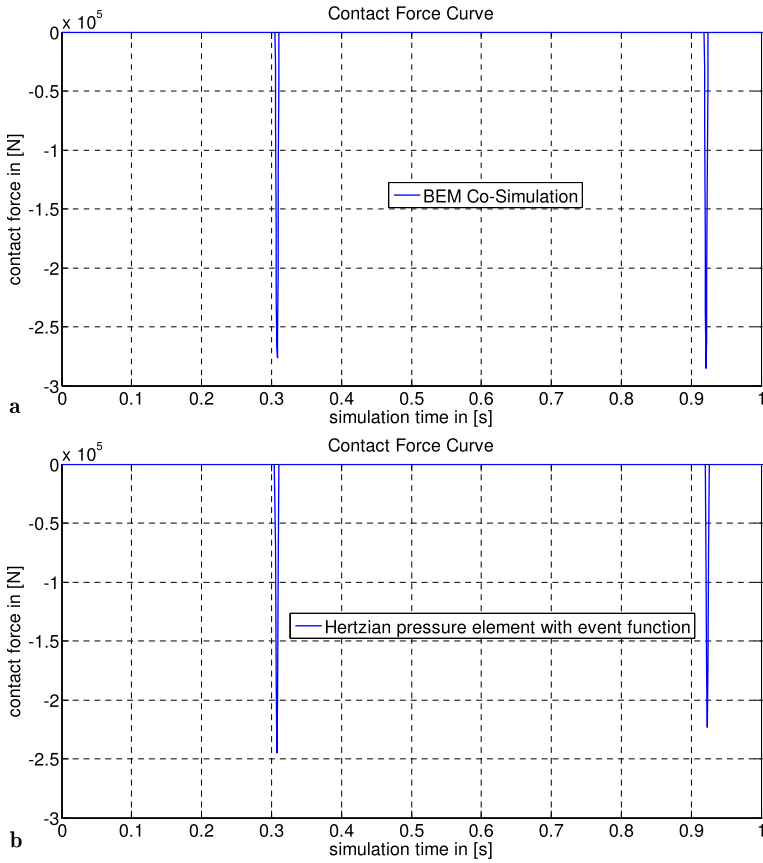


Fig. 13 Contact force vs. time. **a** BEM co-simulation. **b** HERTZIAN pressure element

and -285.308 kN for the reference model and the BEM co-simulation, respectively. The second peak of the reference model is lower than the first one because of the modelled damping d_{damper} . Additional damping occurs due to the implicit integrator. In contrast the second peak of the co-simulation is higher. This results from the explicit EULER integrator which leads to an excitation of the numerical solution.

9 Conclusions and Outlook

The present chapter introduces the modelling of elastic contacts by coupled multi-body and boundary element systems. Compared to contacts modelled by impact laws, physically more accurate results can be obtained. Due to the use of boundary element systems, the contact stresses are obtained within the contact calculation.

The contact formulation is based on mortar methods, which enables the contact calculation of non-conforming meshes. A new three-dimensional contact element for boundary element systems is developed. The mortar element uses the mixed formulation of boundary element formulations. Constraints in a weak form are defined for the displacements in the contact interface. This principle is also known from mortar formulations of finite elements. Additionally, a weak equilibrium is introduced for the tractions in the contact interface. The algorithm for the iteration of contact states is based on a DIRICHLET-to-NEUMANN algorithm. Herein, both contacting bodies are calculated serially. In the first calculation step, one of the contacting bodies represents a rigid obstacle for the other elastic one. The resulting reaction forces on the elastic body are partially transferred on the other one, which is for the second calculation step no longer rigid. As a result the obstacle is deformed and the next iteration starts. The algorithm converges if the numerical equilibrium in the contact interface is reached.

The incorporation of the multibody and boundary element program is realised by interprocess communication. Further details are described in [41]. The multibody program stops during the calculation of the contact forces by the boundary element program. Unix domain sockets are used for communication of both programs. This coupling scheme is applied because both programs work under the same operating system on the same computer. The applied communication protocol can easily be extended to network sockets. In contrast to Unix domain sockets, network sockets allow both programs working on different computers. This property is very important for the developed BEM co-simulation because the boundary element program needs more computational resources than the multibody program. The position data provided by the multibody program are used by the boundary element program to calculate the displacements and tractions on the contacting bodies. The resulting contact tractions are summarised to a contact wrench by numerical integration.

Altogether the MBS-BEM co-simulation is an appropriate way for contact calculation in multibody systems. The main advantage is that contact stresses are obtained within the dynamic calculation. These data can be used for strength, fatigue and durability analyses. Despite the fact that the calculation of complex models needs a large calculation time the coupled simulation of multibody and boundary element systems offers various applications.

As an application for the BEM contact algorithm, the femoral-patellar joint with sliding contact between a human patella and a femur bone is under investigation. First tests show a good convergence of the BEM contact algorithm, see Fig. 14.

A future task should be the speed-up of the co-simulation. The main advantage of the developed BEM contact algorithm is the possibility to integrate fast boundary element methods, because the two contacting bodies are treated separately. Fast boundary methods are developed to overcome large calculation times caused by fully populated system matrices. The panel clustering method described in [19] approximates the matrix-vector multiplication so that instead of a system matrix \mathbf{A} of the size $n \times n$ only two vectors of the size n are calculated. Thus, the memory requirements are strongly reduced. The build-up of a binary tree or

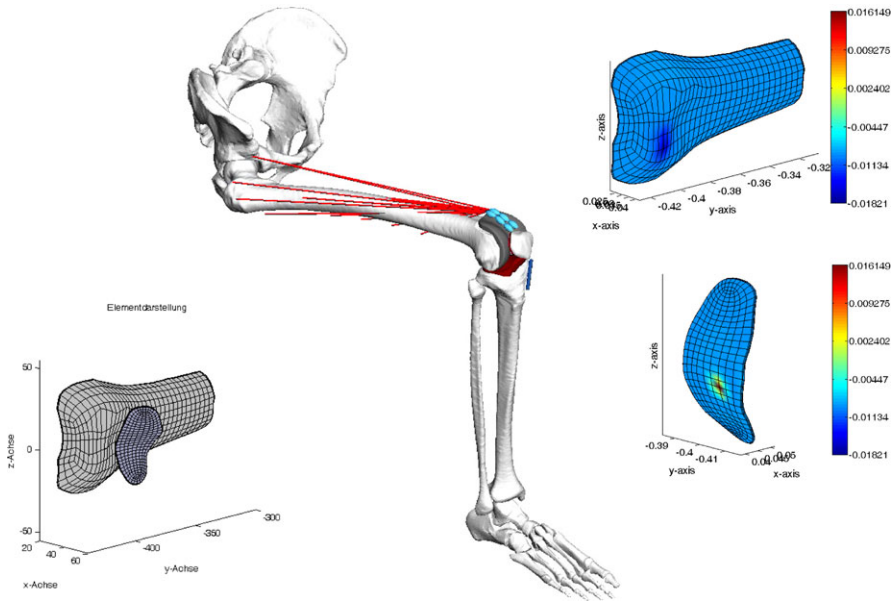


Fig. 14 Simulation of a patellar joint (in cooperation with Department of Orthopaedics, University Medicine, Rostock)

octree for the boundary elements is necessary, so that the collision detection algorithms described in [40, 41] can be used. The panel clustering method is extended to three-dimensional elastostatics in [20, 21] where the integral free term c_{ij} from (23) is assumed to be equal to $\frac{1}{2}\delta_{ij}$. From theoretical background this is valid for flat surfaces only, which is not typical for physical shapes in mechanical engineering. Therefore, these algorithms have to be checked carefully. Panel clustering uses hierarchical matrices which are explained in detail by [7]. First implementations of the panel clustering method for temperature distribution problems is shown in [22].

In the present contribution, quasistatic boundary element formulations are considered only. To take local vibrations and wave propagation of the elastic bodies into account dynamic co-simulation is recommended. A possible implementation is the dual reciprocity method described in [8]. The result of that method are dynamic system matrices, whereby also a system matrix in analogy to the mass matrix of finite element systems is obtained. The coupling of multibody and boundary element systems based on dual reciprocity methods to model flexible bodies is described by [3].

The integration of dynamic boundary element formulations leads to better results for elastic impact problems because the energy loss due to wave propagation is taken into account. The system matrices for that formulation are fully populated and not necessarily positive definite so that complex eigenfrequencies are obtained. To overcome these problem internal nodes are included to represent the mass distribution.

In [23] it is shown that lower eigenfrequencies become real if a sufficient number of internal nodes is used. For dynamic co-simulation a separate integrator has to be implemented within the boundary element formulation. In addition other coupling schemes have to be implemented because the two simulations have to be synchronised. Special coupling schemes for co-simulation of multibody and finite element systems are described in [10] which are also applicable for the co-simulation of multibody and boundary element systems.

Acknowledgements Thanks to Prof. Klaus Schittkowski from the University of Bayreuth for providing the source code of quadratic programming.

References

1. Aliabadi, F.: *The Boundary Element Method: Applications in Solids and Structures*. Wiley, New York (2002)
2. Ambrósio, J., Rauter, F., Pereira, M., Pombo, J.: A flexible multibody pantograph model for the analysis of the catenary-pantograph contact problem. In: Fracek, J., Wojtyra, M. (eds.) *ECCOMAS Thematic Conference: Multibody Dynamics*, Warsaw (2009)
3. Amirouche, F.: *Fundamentals of Multibody Dynamics. Theory and Applications*. Birkhäuser, Boston (2005)
4. Baumgarte, J.: Stabilization of constraints and integrals of motion in dynamical systems. *Comput. Methods Appl. Mech. Eng.* **1**, 1–16 (1972)
5. Beer, G., Smith, I., Duenser, C.: *The Boundary Element Method with Programming: For Engineers and Scientists*. Springer, Berlin (2008)
6. Bernardi, C., Maday, Y., Patera, A.: A new conforming approach to domain decomposition: the mortar element. In: Brezis, H., Lions, J.L. (eds.) *Nonlinear Partial Differential Equations and Their Applications*, pp. 13–51. Wiley, New York (1992)
7. Börm, S., Grasedyck, L., Hackbusch, W.: Hierarchical matrices. Lecture note No. 21, Max-Planck-Institut für Mathematik in den Naturwissenschaften, Leipzig (2003)
8. Brebbia, C.A., Wrobel, L.C., Partridge, P.W.: *Dual Reciprocity Boundary Element Method*. Springer, Berlin (1991)
9. Brezzi, F., Fortin, M.: *Mixed and Hybrid Finite Element Methods*. Springer, Berlin (1991)
10. Busch, M., Schweizer, B.: Numerical stability and accuracy of different co-simulation techniques: analytical investigations based on a 2-DOF test model. In: Mikkola, A. (ed.) *Proceedings of the 1st Joint International Conference on Multibody System Dynamics—IMSD*, Lappeenranta, pp. 71–83 (2010)
11. Cowper, G.R.: Gaussian quadrature formulas for triangles. *Int. J. Numer. Methods Eng.* **7**, 405–408 (1973)
12. Craig, R.R., Bampton, M.C.C.: Coupling of substructures for dynamic analyses. *J. Aircr.* **6**(7), 1313–1319 (1968)
13. Danson, D.J.: A boundary element formulation of problems in linear isotropic elasticity with body forces. In: *3rd International Conference on Boundary Element Methods*, pp. 105–122 (1981)
14. Dunavant, D.A.: High degree efficient symmetrical Gaussian quadrature rules for the triangle. *Int. J. Numer. Methods Eng.* **21**, 1129–1148 (1985)
15. Eberhard, P.: *Kontaktuntersuchungen durch hybride Mehrkörpersystem/Finite Elemente Simulationen*. Shaker, Aachen (2000)
16. Foley, J.D., Van Dam, A., Feiner, S.K., Hughes, J.F., Phillips, R.L.: *Introduction to Computer Graphics*. Addison-Wesley Longman, Amsterdam (1993)

17. Gaul, L., Fiedler, C.: *Methode der Randelemente in Statik und Dynamik*. Vieweg, Braunschweig (1998)
18. Gaul, L., Fischer, M.: Large-scale simulations of acoustic-structure interaction using the fast multipole BEM. *Z. Angew. Math. Mech.* **86**, 4–17 (2006)
19. Hackbusch, W., Nowak, Z.P.: On the fast matrix multiplication in the boundary element method by panel clustering. *Numer. Math.* **54**, 463–491 (1989)
20. Hayami, K., Sauter, S.: Application of the panel clustering method to the three-dimensional elastostatic problem. In: *Boundary Elements XIX*, pp. 625–635. Computational Mechanics Publications, Southampton (1997)
21. Hayami, K., Sauter, S.: Panel clustering for 3-D elastostatics using spherical harmonics. In: *Boundary Elements XX*, pp. 289–299. Computational Mechanics Publications, Southampton (1998)
22. Koch, A.: Implementierung eines direkten Randelementverfahrens und einer Panel-Clustering-Methode zur Berechnung von Temperaturproblemen. Master thesis, Universität Rostock (2011)
23. Kögl, M.: A boundary element method for dynamic analysis of anisotropic elastic, piezoelectric, and thermoelastic solids. Ph.D. thesis, Institut A für Mechanik (Bericht 6/2000), Universität Stuttgart (2000)
24. Krause, R.H.: Monotone multigrid methods for Signorini’s problem with friction. Ph.D. thesis, FU Berlin (2001)
25. Krause, R.H., Wohlmuth, B.I.: A Dirichlet-Neumann type algorithm for contact problems with friction. *Comput. Vis. Sci.* **5**, 139–148 (2002)
26. Lehmann, T., Schittkowski, K., Spickenreuther, T.: MIQL: a Fortran code for convex mixed-integer quadratic programming—user’s guide. Tech. Rep. 1, Department of Computer Science, University of Bayreuth (2009)
27. McDevitt, T.W., Laursen, T.A.: A mortar-finite element formulation for frictional contact problems. *Int. J. Numer. Methods Eng.* **48**, 1525–1547 (2000)
28. Park, K., Felippa, C., Rebel, G.: A simple algorithm for localized construction of non-matching structural interfaces. *Int. J. Numer. Methods Eng.* **53**, 2117–2142 (2002)
29. Pfeiffer, F., Glocker, C.: *Multibody Dynamics with Unilateral Contacts*. Wiley, New York (1996)
30. Puso, M.A.: A 3D mortar method for solid mechanics. *Int. J. Numer. Methods Eng.* **59**, 315–336 (2004)
31. Rebel, G., Park, K., Felippa, C.: A contact formulation based on localized Lagrange multipliers: formulation and application to two-dimensional problems. *Int. J. Numer. Methods Eng.* **54**, 263–298 (2002)
32. Schiehlen, W., Eberhard, P.: *Technische Dynamik*. Teubner, Leipzig (2004)
33. Thomson, W.: Note on the integration of the equations of equilibrium of an elastic solid. *Cambridge and Dublin Mathematical Journal* **3**, 87–89 (1848)
34. Vodička, R., Mantič, V., Paris, F.: SGBEM with Lagrange multipliers applied to elastic domain decomposition problems with curved interfaces using non-matching meshes. *Int. J. Numer. Methods Eng.* **83**, 91–128 (2010)
35. Woernle, C.: *Mehrkörpersysteme: Eine Einführung in die Kinematik und Dynamik von Systemen starrer Körper*. Springer, Berlin (2011)
36. Wohlmuth, B.I.: A mortar finite element method using dual spaces for the Lagrange multiplier. *SIAM J. Numer. Anal.* **38**, 989–1012 (2000)
37. Wohlmuth, B.I.: *Discretization Methods and Iterative Solvers Based on Domain Decomposition*. Springer, Berlin (2001)
38. Wohlmuth, B.I., Krause, R.H.: Monotone methods on non-matching grids for nonlinear contact problems. *SIAM J. Sci. Comput.* **25**, 324–347 (2003)
39. Wriggers, P.: *Computational Contact Mechanics*. Springer, Berlin (2006)
40. Zachmann, G.: Virtual reality in assembly simulation—collision detection, simulation algorithms, and interaction techniques. Ph.D. thesis, Technische Universität Darmstadt (2000)

41. Zierath, J.: Modelling of Three-Dimensional Contacts by Coupled Multibody- and Boundary-Element-Systems. Verlag Dr. Hut, München (2011)
42. Zierath, J., Woernle, C.: Elastic contact analyses with hybrid boundary element/multibody systems. In: Mikkola, A. (ed.) Proceedings of the 1st Joint International Conference on Multibody System Dynamics, Lappeenranta, pp. 327–336 (2010)