A mortar finite element approach for point, line, and surface contact

P. Farah | W. A. Wall | A. Popp

Institute for Computational Mechanics, Technical University of Munich, Garching b. München, Germany

Correspondence
A. Popp, Institute for Computational Mechanics, Technical University of Munich, 85748 Garching b. München, Germany.
Email: popp@lnm.mw.tum.de

Funding information
German Federal Ministry of Economic Affairs and Energy, Grant/Award Number: 20T1306; Rolls Royce Deutschland Ltd & Co KG, Grant/Award Number: T004.017.000

Summary
An approach for investigating finite deformation contact problems with frictional effects with a special emphasis on nonsmooth geometries such as sharp corners and edges is proposed in this contribution. The contact conditions are separately enforced for point contact, line contact, and surface contact by employing 3 different sets of Lagrange multipliers and, as far as possible, a variationally consistent discretization approach based on mortar finite element methods. The discrete unknowns due to the Lagrange multiplier approach are eliminated from the system of equations by employing so-called dual or biorthogonal shape functions. For the combined algorithm, no transition parameters are required, and the decision between point contact, line contact, and surface contact is implicitly made by the variationally consistent framework. The algorithm is supported by a penalty regularization for the special scenario of nonparallel edge-to-edge contact. The robustness and applicability of the proposed algorithms are demonstrated with several numerical examples.

KEYWORDS
contact, edges and corners, finite deformations, friction, mortar finite element methods, nonsmooth geometries

1 | INTRODUCTION

Computational contact mechanics of nonlinear solids and structures is both highly relevant and challenging in many classical engineering tasks and in applied sciences. Finite element analysis is undoubtedly the dominating numerical approximation technique for the solution of partial differential equations in solid and structural mechanics and is therefore also frequently employed for investigating the occurring contact stresses in a finite deformation regime. In the past years, the main focus of research was based on the assumption of smooth physical surfaces, and thus, pure surface-to-surface contact settings have been almost exclusively considered in the literature. In the context of this contribution, the naming “smooth” is employed for surfaces that can be considered as being at least C1-continuous in the continuum description of the problem formulation. Thus, the aim for surface-to-surface contact discretizations was to calculate the corresponding smooth contact stress distributions and to exactly represent the geometries (ie, the contact surfaces) of the involved bodies. For this purpose, non–uniform-rational-B-splines (NURBS) have become of high interest because with these types of shape functions a computer–aided design-conforming geometry representation can be reached, see, for example, other works. Nevertheless, classical finite elements based on the first- and second-order Lagrangian polynomials are still the most commonly employed discretization type and are also considered in this contribution.
As fundamental technique for discretizing the contact constraints, mortar methods are well-established nowadays because they allow for a variationally consistent treatment of contact conditions despite the presence of nonmatching surface meshes, see other works. These methods have already been successfully extended to resolve complex interface phenomena such as wear, lubrication, and thermal effects. Despite the superior robustness of mortar methods, their applicability is strongly restricted by the requirement of smooth geometries (i.e., surface-to-surface contact). From an illustrative, slightly unmathematical perspective, this is due to their weak enforcement of the contact constraints, which results in a surface-based weighting of the gap function. Consequently, large penetrations at vertices and sharp edges would occur, and therefore, contact of nonsmooth geometries such as vertex-to-vertex, vertex-to-edge, vertex-to-surface, edge-to-edge, and edge-to-surface contact cannot be acceptably resolved with classical mortar methods.

Finite element–based contact discretization schemes that naturally lead to an adequate satisfaction of nonpenetration conditions at nonsmooth geometric entities are, for example, the well-known node-to-segment methods, see other works. In combination with a Lagrange multiplier approach for constraint enforcement, they lead to an exact constraint fulfillment at each slave node, and thus, no penetration at the nodes occurs. However, node-to-segment methods often lack important accuracy requirements as can be demonstrated with classical patch tests, see the work of Papadopoulos and Taylor.

In addition to the problem of constraint discretization, the numerical evaluation of a suitable contact normal direction is a highly complex task for arbitrary geometries, see the work of Konyukhov and Schweizerhof. Eventually, a discrete nodal normal field is required for defining the nonpenetration constraints based on the gap measurement between two bodies. This constitutes a classical, well-established approach in computational contact mechanics. An alternative nonpenetration condition can be formulated by employing the intersection volume instead of the gap function, see the works of Kane et al and Pandolfi et al. These methods were originally restricted to geometrically linear tetrahedral elements but have later been extended to more general element types, see the works of Cirak and West and Haikal and Hjelmstad. Similar to these procedures, the contact domain method has been developed in other works. The contact domains can be interpreted as regions connecting the potential contact surfaces of the involved bodies and are utilized to formulate the constraints. These methods are also able to pass classical contact patch tests.

When considering contact of vertices, edges, and surfaces, the so-called discontinuous deformation analysis should also be mentioned, which was originally introduced in the work of Shi. This method shares similarities with the finite element method since it is based on solving for displacements and stresses of discrete elements, but these elements are not connected in a finite element sense. Instead, they are disconnected discrete elements that interact with each other via contact constraints. Thus, it also contains characteristics of discrete element methods, see the work of Cundall. This method is well established in simulating jointed rock mass behavior, where edge-to-edge contact scenarios occur permanently.

Besides the finite element approaches, nonsmooth contact scenarios are also often appearing within rigid multibody simulations, but these approaches, while being highly efficient in calculating physically correct contact kinematics, are not able to determine accurate contact stresses in finite deformation settings. Therefore, this type of simulation is not considered in this contribution.

It can be seen from the short review given above, that despite the abundant literature on numerical approaches for the simulation of surface-to-surface contact, methods for the contact treatment of nonsmooth geometries (i.e., vertices, edges, and surfaces) can only rarely be found in the existing literature. Therefore, the aim of this contribution is to develop a unified mortar–based finite element framework for an accurate and robust calculation of the displacements and contact stresses for contacting vertices, edges, and surfaces in a finite deformation regime. To the best of the authors’ knowledge, this is not possible with any other mortar-based approach from the literature. For this purpose, a variationally consistent framework for computational contact of nonsmooth geometries based on dual mortar methods is developed. Therein, point contact, line contact, and surface contact are treated with 3 different sets of Lagrange multipliers. A suitable definition of the appropriate discrete Lagrange multiplier spaces prevents the system from being overconstrained. Furthermore, the discrete Lagrange multiplier unknowns are eliminated from the arising system of equations by employing dual shape functions for the line and surface Lagrange multipliers. For the combined framework, no heuristic transition parameters are required to switch between point contact, line contact, and surface contact. Only two special contact scenarios, namely, edge-to-edge contact and certain very complex situations, where different entities of the same bodies are interacting, require special treatment in the form of a penalty regularization to prevent the overall problem from being overconstrained also in such rare cases.

The outline of this paper is as follows. In Section 2, the basics on nonlinear continuum mechanics are briefly reviewed, and the strong formulations of the different sets of contact constraints (vertex, edge, surface) are given. Then, the weak
formulation of the contact problem, including frictional effects, is described in Section 3. In addition, the finite element discretization of the geometry and the displacement field are explained in Section 4. Therein, also suitable definitions for the nodal normal fields are introduced. In Section 5, the numerical treatment of point contact, line contact, and surface contact is explained. The combined framework for simulating contact of non-smooth geometries and the global solution scheme are outlined in Section 6. The introduced algorithm is validated with various examples in Section 7. Finally, a short conclusion and an outlook are given in Section 8.

2 | PROBLEM DEFINITION: STRONG FORMULATION

Our starting point for deriving the problem formulation is the consideration of two bodies \( B^{(1)} \) and \( B^{(2)} \) with one shared contact interface. The presence of only two bodies is no restriction to the following problem statement but rather a deliberately chosen simplification for making the following definitions more comprehensive. An extension to more than two bodies is absolutely straightforward. Throughout the entire contribution, \( B^{(1)} \) is denoted as slave body, and \( B^{(2)} \) is termed master body. The open sets \( \Omega_0^{(i)} \subset \mathbb{R}^3 \) and \( \Omega_1^{(i)} \subset \mathbb{R}^3 \) with \( i = 1, 2 \) represent these bodies in reference and spatial configuration, respectively. The reference configurations \( \tilde{\Omega}_0^{(i)} \) are occupied by all reference points \( \tilde{X}^{(i)} \), whereas the spatial configurations define the changed positions \( x^{(i)} \) due to finite deformations. Consequently, the displacement vectors read \( u^{(i)} = \tilde{x}^{(i)} - \tilde{x}^{(i)} \). The boundaries of the two involved bodies are divided into three disjoint boundary sets as

\[
\begin{align}
\partial \tilde{\Omega}_0^{(i)} &= \tilde{\Gamma}_u^{(i)} \cup \tilde{\Gamma}_{\sigma}^{(i)} \cup \tilde{\Gamma}_c^{(i)}, \\
\Gamma_u^{(i)} \cap \Gamma_{\sigma}^{(i)} &= \Gamma_u^{(i)} \cap \Gamma_c^{(i)} = \Gamma_{\sigma}^{(i)} \cap \Gamma_c^{(i)} = \emptyset,
\end{align}
\]

where \( \Gamma_u^{(i)} \) and \( \Gamma_{\sigma}^{(i)} \) are the well-known Dirichlet and Neumann boundaries, which are subjected to given displacements \( \tilde{u}^{(i)} \) and prescribed external boundary tractions \( \tilde{t}^{(i)} \), respectively. Additionally, \( \Gamma_c^{(i)} \) denotes the potential contact boundaries. The counterparts in the current configuration are denoted as \( \gamma_u^{(i)}, \gamma_{\sigma}^{(i)} \), and \( \gamma_c^{(i)} \). One of the characteristics of the contact problems is that the so-called active part of the contact boundary \( \Gamma_u^{(i)} \subseteq \Gamma_c^{(i)} \) is a priori unknown and possibly changing over time. Consequently, the currently inactive part of the contact boundary is defined by \( \Gamma_{i}^{(i)} = \Gamma_c^{(i)} \setminus \Gamma_u^{(i)} \). Without loss of generality, a hyperelastic (e.g., neo-Hookean) material model is assumed. For its description, the deformation gradients \( F^{(i)} \) and the Green-Lagrange strain tensors \( E^{(i)} \) are defined as

\[
F^{(i)} = \frac{\partial \tilde{x}^{(i)}}{\partial x^{(i)}}, \quad E^{(i)} = \frac{1}{2} \left( F^{(i)\top} F^{(i)} - I \right).
\]

Based on the hyperelastic strain energy functions \( \psi^{(i)} \), the second Piola-Kirchhoff stress tensors \( S^{(i)} \) and the fourth-order constitutive tensors \( C^{(i)} \) can be written as

\[
S^{(i)} = \frac{\partial \psi^{(i)}}{\partial E^{(i)}}, \quad C^{(i)} = \frac{\partial^2 \psi^{(i)}}{\partial E^{(i)} \partial E^{(i)}}.
\]

On each subdomain \( \Omega_0^{(i)} \), the initial boundary value problem of finite deformation elastodynamics needs to be satisfied, viz,

\[
\begin{align}
\text{Div}(F^{(i)} S^{(i)}) + b_0^{(i)} &= \rho_0^{(i)} u^{(i)} \quad \text{in } \Omega_0^{(i)} \times [0, T], \\
\tilde{u}^{(i)} &= \tilde{u}^{(i)} \quad \text{on } \Gamma_u^{(i)} \times [0, T], \\
(F^{(i)} S^{(i)}) \cdot N^{(i)} &= \tilde{t}^{(i)} \quad \text{on } \Gamma_{\sigma}^{(i)} \times [0, T], \\
\tilde{u}^{(i)}(X^{(i)}, 0) &= \tilde{u}_0^{(i)}(X^{(i)}) \quad \text{in } \Omega_0^{(i)}, \\
\dot{\tilde{u}}^{(i)}(X^{(i)}, 0) &= \dot{\tilde{u}}_0^{(i)}(X^{(i)}) \quad \text{in } \Omega_0^{(i)}.
\end{align}
\]

Here, \( b_0^{(i)} \) denote the external volume forces, \( \rho_0^{(i)} \) are the initial densities of the involved bodies, and \( N^{(i)} \) are the unit outward normals in the reference configuration.

The special focus of this contribution is to investigate contact situations involving non-smooth geometries. Therefore, the potential contact boundaries \( \Gamma_c^{(i)} \) are divided into three disjoint subsets, viz,

\[
\Gamma_c^{(i)} = \Gamma_{\sigma}^{(i)} \cup \Gamma_i^{(i)} \cup \Gamma_{\sigma}^{(i)}.
\]
\[ \Gamma_{\circ} \cap \Gamma_{\circ}^{(i)} = \Gamma_{\circ}^{(i)} \cap \Gamma_{\circ} = \Gamma_{\circ}^{(i)} \cap \Gamma_{\star} = \emptyset, \]

where \( \Gamma_{\circ}^{(i)} \) are the potential contact boundaries of surfaces, \( \Gamma_{\circ}^{(i)} \) represent edges, and \( \Gamma_{\star}^{(i)} \) are the sets of all vertices within the contact boundaries. Similar to the other boundaries, their spatial counterparts are denoted as \( \gamma_{\circ}^{(i)} \), \( \gamma_{\star}^{(i)} \), and \( \gamma_{\star}^{(i)} \) (see Figure 1). In a general finite deformation setting, the geometrical contact entities, namely, surfaces, edges, and vertices could deform significantly, meaning that an initial vertex could be flattened to become part of a new surface, or a surface could be deformed in a way to create a new edge. However, in this contribution, it is assumed that the spatial points are assigned to the set definition of its reference boundary and consequently extreme deformations, such as a complete flattening of an edge, are not allowed. In order to define suitable contact conditions, the possibly arising contact scenarios have to be specified. For this purpose, it is assumed that the contact entity with the lower geometrical dimension acts as slave part, and the corresponding contact entity of equal or higher geometrical order is defined to be the master boundary. Concretely, the first class of possible contact scenarios is characterized by the active contacting area reducing to a point (see Figure 2). Namely, these scenarios are vertex-to-vertex, vertex-to-edge, vertex-to-surface, and nonparallel edge-to-edge settings. These contact situations are denoted as point contact in the following. The next class is defined by the contacting area being a one-dimensional line, which could arise due to edge-to-surface and parallel edge-to-edge contact (see Figure 3). The last setting is the classical surface contact scenario, which is well-investigated in the context of computational contact mechanics and is schematically visualized in Figure 4. This classification represents a hierarchy of contact situations, where the contact boundary on which the contact constraints are going to be formulated is the involved slave entity with the lowest dimension. The only exception to this scheme is the nonparallel edge-to-edge contact. Here, the geometrical slave entity is an edge, but the contact scenario reduces to a point contact. Therefore, these special contact points are denoted as \( x_{\star} \) and the set of all slave contact points resulting from crossing edges is \( \gamma_{\star}^{(1)} \). For the
nonparallel edge-to-edge setting, the discrete enforcement of the contact constraints will be treated in a special way later on. However, in order to keep a convenient notation, we postulate that the set of edge-to-edge crossing points and the set of all vertices are united to the set of potential point contacts ψ(1) ∪ ψ(1) × ψ(1).

Furthermore, it is assumed that the potential line contact setting is defined on γ(1), and the potential surface contact scenario is defined on γ(1). The corresponding contact force quantity, which acts on the slave contact boundary are now different for the point, line, and surface contact (see Figure 5). The point contact formulation is subjected to the force vector f(1)c, which is a concentrated load on the contact point. For the line contact setting, a line load vector l(1)c is introduced, and consequently, a surface traction vector t(1)c is employed for the surface contact. Due to the balance of linear momentum, the corresponding master load vectors are identical except for the opposite sign, i.e.,

\[ f(1)c = -f(2)c, \quad l(1)c = -l(2)c, \quad t(1)c = -t(2)c. \]  

For describing the relative position between a slave point x(1) and the corresponding point on the master side x(2) in the spatial configuration, the gap vector g is introduced

\[ g(X, t) = x(1)(X(1), t) - x(2)(X(2)(X(1), t), t). \]  

For FIGURE 5, the contact force quantities for different contact scenarios: point contact (left), line contact (middle), and surface contact (right) are illustrated.
Here, the master point $\mathbf{x}^{(2)}$ is associated with the slave point $\mathbf{x}^{(1)}$ via a geometrical projection. In order to define the contact constraints, the scalar-valued gap function $g_n(\mathbf{X}, t)$ is defined, which acts as measurement for proximity and potential contact of the two involved bodies, ie,

$$g_n(\mathbf{X}, t) = -n \mathbf{g}(\mathbf{X}, t).$$

Herein, $n$ is the outward unit normal field in the spatial configuration, which is typically determined by a so-called closest-point-projection (CPP) procedure, see, eg, the aforementioned work. While the gap function characterizes the contact interaction in normal direction, the primary kinematic quantity for tangential contact interaction is the relative tangential velocity, ie,

$$\mathbf{v}_{\tau, \text{rel}} = (I - n \otimes n) \cdot [\dot{x}^{(1)}(\mathbf{X}^{(1)}, t) - \dot{x}^{(2)}(\mathbf{X}^{(1)}, t), t].$$

It represents a vector-valued quantity that is completely defined in the local tangential plane. Similar to the gap function and the relative tangential velocity, the contact load vectors $\mathbf{f}_c^{(1)}$, $\mathbf{t}_c^{(1)}$, and $\mathbf{t}_c^{(1)}$ on the slave surface can be split into their normal and tangential components, yielding

$$\mathbf{f}_c^{(1)} = f_n \mathbf{n} + \mathbf{f}_\tau, \quad \mathbf{t}_c^{(1)} = l_n \mathbf{n} + \mathbf{l}_\tau, \quad \mathbf{t}_c^{(1)} = t_n \mathbf{n} + \mathbf{t}_\tau.$$

Thus, the contact constraints in normal direction are given in the form of the well-known Hertz-Signorini-Moreau conditions for the point, line, and surface contact scenarios, ie,

$$g_n \geq 0 \quad \text{on } \gamma_c^{(1)} \times [0, T].$$

$$f_n \leq 0, \quad f_n g_n = 0 \quad \text{on } \gamma_c^{(1)} \times [0, T].$$

$$l_n \leq 0, \quad l_n g_n = 0 \quad \text{on } \gamma_l^{(1)} \times [0, T].$$

$$t_n \leq 0, \quad t_n g_n = 0 \quad \text{on } \gamma_s^{(1)} \times [0, T].$$

In addition, frictional sliding is formulated under the assumption that the purely phenomenological Coulomb law is also valid for point contact and line contact, see the aforementioned work. The frictional sliding constraints for point contact read

$$\Phi_\tau := ||\mathbf{f}_\tau|| - \gamma_s |f_n| \leq 0, \quad \mathbf{v}_{\tau, \text{rel}} + \beta_\tau \mathbf{f}_\tau = \mathbf{0}, \quad \beta_\tau \geq 0, \quad \Phi_\tau \beta_\tau = 0 \quad \text{on } \gamma_c^{(1)} \times [0, T].$$

The corresponding constraints for line contact are given as

$$\Phi_\tau := ||\mathbf{t}_\tau|| - \gamma_t |l_n| \leq 0, \quad \mathbf{v}_{\tau, \text{rel}} + \beta_\tau \mathbf{t}_\tau = \mathbf{0}, \quad \beta_\tau \geq 0, \quad \Phi_\tau \beta_\tau = 0 \quad \text{on } \gamma_l^{(1)} \times [0, T].$$

Finally, the tangential part of surface contact is defined by

$$\Phi_\tau := ||\mathbf{t}_\tau|| - \gamma_s |t_n| \leq 0, \quad \mathbf{v}_{\tau, \text{rel}} + \beta_\tau \mathbf{t}_\tau = \mathbf{0}, \quad \beta_\tau \geq 0, \quad \Phi_\tau \beta_\tau = 0 \quad \text{on } \gamma_s^{(1)} \times [0, T].$$

In (22)-(24), the friction coefficient $\gamma_s \geq 0$ is assumed to be equal for all contact scenarios for the sake of simplicity. Moreover, $|| \cdot ||$ denotes the $L^2$-norm in $\mathbb{R}^3$, and the parameters $\beta_i$ are complementarity parameters that are necessary to describe the separation of the stick and slip branch. These parameters will later vanish by reformulating the sets of tangential constraints within so-called nonlinear complementarity (NCP) functions (see Section 6.2).

3 | WEAK FORMULATION

For the derivation of a weak variational formulation, the solution spaces $\mathbf{U}^{(i)}$ and weighting spaces $\mathbf{V}^{(i)}$ for the displacement field are defined as

$$\mathbf{U}^{(i)} = \left\{ \mathbf{u}^{(i)} \in [H^1(\Omega)]^3 \mid \mathbf{u}^{(i)} = \mathbf{u}^{(i)} \text{ on } \Gamma_u^{(i)} \right\},$$

$$\mathbf{V}^{(i)} = \left\{ \mathbf{v}^{(i)} \in [H^1(\Omega)]^3 \mid \mathbf{v}^{(i)} = \mathbf{v}^{(i)} \text{ on } \Gamma_v^{(i)} \right\}.$$
\[ \mathcal{V}^{(i)} = \left\{ \delta \mathbf{u}^{(i)} \in [H^1(\Omega)]^3 \, \bigg| \, \delta \mathbf{u}^{(i)} = 0 \text{ on } \Gamma^i \right\} . \]  

Here, \( H^1(\Omega) \) denotes the usual Sobolev space of functions with square integrable values and first derivatives, respectively. In order to enforce the normal and tangential contact constraints, three vector-valued Lagrange multipliers are introduced. The first one is the surface Lagrange multiplier \( \lambda_s = -f_n \), which represents the negative slave side contact traction and is chosen from the convex cone \( \mathcal{M}_{s}(\lambda_s) \subset \mathcal{M}_{s} \) given by

\[ \mathcal{M}_{s}(\lambda_s) = \left\{ \mathbf{u} \in \mathcal{M}_{s} \bigg| \langle \mathbf{u}, \mathbf{v} \rangle_{\gamma^i} \leq \langle \mathcal{G}_{\lambda_s}, \| \mathbf{v} \| \rangle_{\gamma^i}, \mathbf{v} \in \mathcal{W}_{s} \text{ with } \mathbf{v}_n \leq 0 \right\} . \]  

Herein, \( \langle \cdot, \cdot \rangle_{\gamma^i} \) stands for the scalar or vector-valued duality pairing between \( H^{\alpha} \) and \( H^{\alpha+1} \) on the surface contact boundary \( \gamma^{(i)} \). Moreover, \( \mathcal{M}_{s} \) is the dual space of the trace space \( \mathcal{W}_{s} \) of \( \mathcal{V}^{(i)} \) restricted to the surface contact boundary \( \gamma_{s}(i) \). The second vector-valued Lagrange multiplier represents the negative slave side line traction \( \lambda_l = -f_l \) and is utilized to enforce the line contact constraints. It is chosen from the convex cone \( \mathcal{M}_{l}(\lambda_l) \subset \mathcal{M}_{l} \) defined by

\[ \mathcal{M}_{l}(\lambda_l) = \left\{ \mathbf{u} \in \mathcal{M}_{l} \bigg| \langle \mathbf{u}, \mathbf{v} \rangle_{\gamma^i} \leq \langle \mathcal{G}_{\lambda_l}, \| \mathbf{v} \| \rangle_{\gamma^i}, \mathbf{v} \in \mathcal{W}_{l}, \text{ with } \mathbf{v}_n \leq 0 \right\} . \]  

In complete analogy to (27), \( \mathcal{M}_{s} \) is the dual space of the trace space \( \mathcal{W}_{s} \) of \( \mathcal{V}^{(i)} \) restricted to the edge contact boundary \( \gamma^{(i)} \). Finally, the constraint enforcement for the point contact has to be defined. Since the Lagrange multipliers for the surface contact and line contact are already defined on all points of the potential slave contact boundary except the set of all vertices, only these vertices are free of constraints. Thus, the point contact for the vertices can be enforced by the vector-valued Lagrange multiplier \( \lambda_p \), which is chosen from the convex cone \( \mathcal{M}_{p}(\lambda_p) \subset \mathcal{M}_{p} \) defined by

\[ \mathcal{M}_{p}(\lambda_p) = \left\{ \mathbf{u} \in \mathcal{M}_{p} \bigg| \langle \mathbf{u}, \mathbf{v} \rangle_{\gamma^i} \leq \langle \mathcal{G}_{\lambda_p}, \| \mathbf{v} \| \rangle_{\gamma^i}, \mathbf{v} \in \mathcal{W}_{p}, \text{ with } \mathbf{v}_n \leq 0 \right\} . \]  

Again, \( \mathcal{M}_{p} \) is the dual space of the trace space \( \mathcal{W}_{p} \) of \( \mathcal{V}^{(i)} \) restricted to the vertex contact boundary \( \gamma_{p}(i) \). With \( \lambda_{p} \), only the point contact scenarios acting on vertices are defined and the edge-to-edge setting is not affected by this Lagrange multiplier. Unfortunately, it is mathematically impossible to define the point Lagrange multiplier also on the points of crossing edges since the edges are already subjected to the line Lagrange multiplier. From an engineering point of view, the scenario of two Lagrange multipliers acting on the same point can be considered as being overconstraint. Thus, the enforcement of the edge-to-edge contact constraints is relaxed via a penalty regularization. Therein, the contact constraints are explicitly removed by a penalization of any occurring constraint violation. Consequently, the normal part of the force vector reads

\[ f_n = \begin{cases} \epsilon_n (-g_n) & \text{if } g_n \leq 0 \\ 0 & \text{if } g_n > 0 \end{cases} \quad \text{on } \gamma_{p}(i) \times [0, T] . \]  

Parameter \( \epsilon_n \) penalizes the penetration of the two bodies. The tangential part of the force vector is defined via

\[ \mathcal{L} \mathbf{f}_t = \epsilon_t [\mathbf{v}_{t, \text{rel}} - \beta \mathbf{f}_t] \]  

\[ \Phi_{\times} := \| \mathbf{f}_t \| - \mathcal{G}_{\times} \leq 0, \quad \beta_{\times} \geq 0, \quad \Phi_{\times} \beta_{\times} = 0 \quad \text{on } \gamma_{p}(i) \times [0, T] . \]

Herein, \( \mathcal{L} \) is the Lie derivative of the tangential force, and \( \epsilon_t \) is the frictional penalty parameter. The scalar parameter \( \beta_{\times} \) has the same interpretation as in (22)-(24).

Finally, the weak saddle-point–type formulation can be summarized as follows. Find \( \mathbf{u}^{(i)} \in \mathcal{U}^{(i)}, \lambda_{p} \in \mathcal{M}_{p}(\lambda_{p}), \lambda_{s} \in \mathcal{M}_{s}(\lambda_{s}), \) \( \lambda_{l} \in \mathcal{M}_{l}(\lambda_{l}) \), and \( \lambda_{o} \in \mathcal{M}_{o}(\lambda_{o}) \) such that

\[ -\delta \mathcal{W}_{\text{kin,int,ext}} (\mathbf{u}^{(i)}, \delta \mathbf{u}^{(i)}) - \delta \mathcal{W}_{\text{co}} (\lambda_{p}, \lambda_{s}, \lambda_{o}, \delta \mathbf{u}^{(i)}) - \delta \mathcal{W}_{\text{pen}} (\mathbf{u}^{(i)}, \delta \mathbf{u}^{(i)}) = 0 \quad \forall \delta \mathbf{u}^{(i)} \in \mathcal{V}^{(i)} , \]

\[ \delta \mathcal{W}_{\lambda_{p}} (\mathbf{u}^{(i)}, \delta \lambda_{p}) \geq 0 \quad \forall \delta \lambda_{p} \in \mathcal{M}_{p}(\lambda_{p}) , \]

\[ \delta \mathcal{W}_{\lambda_{s}} (\mathbf{u}^{(i)}, \delta \lambda_{s}) \geq 0 \quad \forall \delta \lambda_{s} \in \mathcal{M}_{s}(\lambda_{s}) , \]

\[ \delta \mathcal{W}_{\lambda_{o}} (\mathbf{u}^{(i)}, \delta \lambda_{o}) \geq 0 \quad \forall \delta \lambda_{o} \in \mathcal{M}_{o}(\lambda_{o}) . \]
Here, the kinetic contribution $\delta W_{\text{kin}}$ and the internal and external virtual works $\delta W_{\text{int,ext}}$ are independent from the contact terms, and their derivation is well-known in nonlinear continuum mechanics and thus omitted here. The contact virtual work $\delta W_{\text{co}}$ due to the Lagrange multipliers is given as

$$-\delta W_{\text{co}} = \sum_{x \in \Gamma^{(c)}_{1,2}} \lambda_x \left( \delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)} \circ \chi \right) + \int_{\gamma^{(1)}} \lambda \left( \delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)} \circ \chi \right) dL + \int_{\gamma^{(2)}} \lambda_\sigma \left( \delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)} \circ \chi \right) dA,$$

with a suitable contact interface mapping $\chi : \gamma^{(1)} \rightarrow \gamma^{(2)}$, which is required due to the generally nonidentical contact boundaries $\gamma^{(1)}$ and $\gamma^{(2)}$. The weak form of the normal and tangential contact constraints reads for the point Lagrange multiplier, ie,

$$y \delta W_{\lambda_x} = \sum_{x \in \gamma^{(c)}_{1,2}} \left( \delta \lambda_{x,n} - \lambda_{x,n} \right) \mathbf{g}_n - \sum_{x \in \gamma^{(c)}_{1,2}} \left( \delta \lambda_{x,\tau} - \lambda_{x,\tau} \right) \mathbf{v}_{t,rel}.$$

The weak constraints for the line Lagrange multiplier are given as

$$\delta W_{\lambda} = \int_{\gamma^{(1)}} (\delta \lambda_{x,n} - \lambda_{x,n}) \mathbf{g}_n dL - \int_{\gamma^{(2)}} (\delta \lambda_{x,\tau} - \lambda_{x,\tau}) \mathbf{v}_{t,rel} dL.$$

Finally, the weak constraints due to the surface Lagrange multiplier read

$$\delta W_{\lambda_x} = \int_{\Gamma^{(c)}_{1,2}} (\delta \lambda_{\sigma,n} - \lambda_{\sigma,n}) \mathbf{g}_n dA - \int_{\Gamma^{(c)}_{1,2}} (\delta \lambda_{\sigma,\tau} - \lambda_{\sigma,\tau}) \mathbf{v}_{t,rel} dA.$$

The penalty contribution $\delta W_{\text{pen}}$ that becomes inevitable to avoid overconstraining due to the point contact contribution of crossing edges is schematically given by

$$\delta W_{\text{pen}} = \sum_{x \in \gamma^{(c)}_{1,2}} \mathbf{f}^{(1)}_c \left( \delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)} \circ \chi \right),$$

where the normal and tangential contributions to the contact force vector $\mathbf{f}^{(1)}_c$ are computed according to (30)-(31).

**Remark 1.** Note that all Lagrange multipliers are defined on the slave side contact boundaries, ie, point, line, and surface contact constraints all live on the same body. Thus, compared to classical mortar methods for pure surface contact, the choice of slave and master body is not arbitrary anymore. For example, when a vertex of the master body contacts a slave surface, the nonpenetration constraints cannot be enforced exactly at this position. A dynamic change of slave and master body definition depending on the current contact situation would solve this issue. However, the realization of such a dynamic slave/master scheme goes beyond the scope of this contribution and remains topic of future research.

## 4 FINITE ELEMENT DISCRETIZATION

For the spatial discretization of the considered frictional contact problem using finite elements, the finite-dimensional subsets $\mathbf{U}^{(l)}_h$ and $\mathbf{V}^{(l)}_h$, which represent approximations of the continuous solution spaces $U^{(l)}$ and $V^{(l)}$ are introduced. In the following, we focus on 3-dimensional (3D) first-order (hexahedral and tetrahedral) Lagrangian finite elements, and thus, the contact surface discretization may consist of three-node triangular elements and of four-node quadrilateral elements. Accordingly, the slave and master geometry and displacement interpolation is given as

$$\mathbf{x}^{(1)}_h \bigg|_{\Gamma^{(1)}} = \sum_{k=1}^{n^{(1)}} N^{(1)}_k \mathbf{x}^{(1)}_k, \quad \mathbf{x}^{(2)}_h \bigg|_{\Gamma^{(1)}} = \sum_{l=1}^{n^{(2)}} N^{(2)}_l \mathbf{x}^{(2)}_l,$$

$$\mathbf{u}^{(1)}_h \bigg|_{\Gamma^{(1)}} = \sum_{k=1}^{n^{(1)}} N^{(1)}_k \mathbf{d}^{(1)}_k, \quad \mathbf{u}^{(2)}_h \bigg|_{\Gamma^{(2)}} = \sum_{l=1}^{n^{(2)}} N^{(2)}_l \mathbf{d}^{(2)}_l.$$

Here, $n^{(1)}$ and $n^{(2)}$ represent the number of nodes on the discrete slave contact surface $\Gamma^{(1)}_{ch}$ and on the discrete master contact surface $\Gamma^{(2)}_{ch}$, respectively. The discrete nodal positions and discrete nodal displacements are given by $\mathbf{x}^{(1)}_k, \mathbf{x}^{(2)}_l, \mathbf{d}^{(1)}_k$, and $\mathbf{d}^{(2)}_l$. Based on the usually employed finite element parameter space for two-dimensional surfaces $\xi^{(l)} = (\xi^{(l)}, \eta^{(l)})$, 

the shape functions $N_k^{(1)}$ and $N_j^{(2)}$ are defined. These shape functions are naturally derived from the underlying bulk shape functions.

The nodal normal vectors are of utmost importance for the formulation of a nonsmooth contact framework, since they define the local direction in which the contact force acts. Thus, contact kinematics are strongly influenced by the way the normal vectors are defined. However, we do not aim here at giving a comprehensive solution for all the problems arising from defining nodal normals between the two arbitrary geometries. The fact that already the simple vertex-to-vertex contact scenario can lead to various problems in defining suitable normal directions illustrates the complexity of this topic, see the works of Bao and Zhao.40,41 In contrast, robust numerical approximations of rather classical CPPs are discussed. Here, the idea is to project a physical point onto $C^1$-continuous geometries, see the aforementioned work.25 Specifically, three different types of CPPs of a point onto a $C^1$-continuous surface, edge, and a point are performed. From an algorithmic perspective, these procedures are denoted as node-to-surface projection, node-to-line projection, and node-to-node projection and are explained in Sections 4.1, 4.2, and 4.3. The resulting nodal normals $n^{(1)}$ on the slave side are then interpolated by the already introduced displacement shape functions $N_k$ via

$$n^{(1)} = \sum_{k=1}^{p} N_k^{(1)} n_k^{(1)}. \quad (44)$$

This results in a $C^0$-continuous field of normals, and our procedure can be interpreted as a numerical smoothing of the normal field to guarantee robust contact projection and evaluation algorithms.

### 4.1 Node-to-surface projection

The classical CPP of a node onto a surface is realized by projecting the slave node $j$ along the master side normal $n^{(2)}$ onto the discretized master surface. This procedure can be stated as follows:

$$\alpha n^{(2)}(\tilde{x}^{(2)}) + \sum_{l=1}^{p} N_l^{(2)} x_l^{(2)} = x_j^{(1)}. \quad (45)$$

Here, the sought-after quantities are the scalar valued distance $\alpha$ between the slave point $x_j$ and the master surface, and the projection point coordinates in the two-dimensional master parameter space $\tilde{x}^{(2)}$. The projection in (45) is non-linear in terms of the unknowns $\alpha$ and $\tilde{x}^{(2)}$ and can be solved with a local Newton-Raphson scheme. Since this procedure assumes to project a point onto a $C^1$-continuous surface, the typical first-order finite element approximation does not guarantee solvability of this projection. To overcome well-known problems arising from this CPP, such as degenerated cases of crossing normals or nonuniqueness of the CPP, the master surface nodal normal field $n^{(2)}$ is formulated based on a $C^0$-continuous field of normals, viz,

$$n^{(2)}(\tilde{x}^{(2)}) = \sum_{l=1}^{p} N_l^{(2)} (\tilde{x}^{(2)}) n_l^{(2)}, \quad (46)$$

with the master side displacement shape functions $N_l^{(2)}$. As aforementioned for the slave side, this can be interpreted as numerical smoothing procedure without changing the actual finite element geometry representation. The nodal normal vectors $n_k^{(2)}$ are based on an averaging procedure in order to create a unique normal at each master node (see Figure 6).

**FIGURE 6** Nodally averaged normal vector $n_j$ at node $j$ with 4 adjacent elements. Element normal vectors are exemplarily visualized for elements $e2$ and $e4$. 
This procedure has been suggested for two-dimensional contact scenarios in the work of Yang et al.\textsuperscript{11} and was used for 3D applications in the work of Popp et al.\textsuperscript{9} Thus, it is only briefly explained in the following. The outward pointing unit normal vectors \( \mathbf{n}_{j,ei} \) of the adjacent master elements \( ei \) at master node \( j \) are employed to create the unique master normal \( \mathbf{n}_j \) via

\[
\mathbf{n}_j = \sum_{i=1}^{n_{adj}^i} \mathbf{n}_{j,ei} / \left\| \sum_{i=1}^{n_{adj}^i} \mathbf{n}_{j,ei} \right\|. \tag{47}
\]

Herein, the number of adjacent elements is denoted as \( n_{adj}^i \). Finally, the slave side nodal normal vector \( \mathbf{n}_j^{(s)} \) at node \( j \) is defined as

\[
\mathbf{n}_j^{(s)} = - \mathbf{n}_j^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right). \tag{48}
\]

### 4.2 Node-to-line projection

The closest distance between a slave node and a master edge is computed with a node-to-line projection. The unit normal vector at slave node \( j \) reads

\[
\mathbf{n}_j = \frac{x_j^{(1)} - \sum_{i=1}^{n_{adj}^i} N_i^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) x_i^{(2)}} {\left\| x_j^{(1)} - \sum_{i=1}^{n_{adj}^i} N_i^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) x_i^{(2)} \right\|}, \tag{49}
\]

with the corresponding master point being defined by the line coordinate \( \xi^{(2)} \) in 1-dimensional parameter space. Note that the expression in (49) does not guarantee that the unit normal is pointing in outward direction of the slave body. Thus, an auxiliary slave normal vector \( \mathbf{n}_{j,aux} \) has to be computed to determine the sign of the slave nodal normal. Therefore, \( \mathbf{n}_j^{(s)} \) is defined as nodally averaged normal vector of all adjacent slave elements, which is a similar procedure as in (47) but performed on the slave side. If the angle between \( \mathbf{n}_j^{(1)} \) and \( \mathbf{n}_j \) is larger than 90\(^\circ\), the sign of \( \mathbf{n}_j \) has to be switched. The idea of a signed normal was already developed in the work of Belytschko et al.\textsuperscript{42} The master parameter space coordinate \( \xi^{(2)} \) can be computed by solving the following scalar projection equation:

\[
\tau^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) \left[ x_j^{(1)} - \sum_{i=1}^{n_{adj}^i} N_i^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) x_i^{(2)} \right] = 0, \tag{50}
\]

with \( \xi^{(2)} \) being the only unknown. Like for the node-to-surface projection, this nonlinear equation can be solved with a local Newton-Raphson scheme. To allow for a robust iteration process and unique solution, a pseudo \( C^1 \)-continuous curve in space is created by construction of a nodal tangent field, ie,

\[
\tau^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) = \sum_{k=1}^{n_{adj}^k} N_k^{(2)} \left( \frac{\xi^{(2)}}{\eta^{(2)}} \right) \tau_k^{(2)}. \tag{51}
\]

This field is again interpolated by the master displacement shape functions \( N_k^{(2)} \). The tangent interpolation in (51) requires a unique tangent definition at each master node, and thus, the normal averaging procedure in (47) is also employed for the tangents, yielding

\[
\tau_j = \frac{\sum_{i=1}^{n_{adj}^i} \tau_{j,ei}} {\left\| \sum_{i=1}^{n_{adj}^i} \tau_{j,ei} \right\|}. \tag{52}
\]

### 4.3 Node-to-node projection

The unit nodal normal vector resulting from a node-to-node projection is given by the difference vector of the spatial nodal positions scaled to unit length. For a slave node \( j \) and a master node \( k \), the slave nodal normal reads

\[
\mathbf{n}_j^* = \frac{x_j^{(1)} - x_k^{(2)}} {\left\| x_j^{(1)} - x_k^{(2)} \right\|}. \tag{53}
\]
Like in (49), the expression in (53) does not guarantee to produce an outward unit normal vector. Thus, the direction of the nodal normal vector $n_j$ has to be assessed by comparing with an auxiliary slave normal vector $n_{j,\text{aux}}$ (see Section 4.2).

### 4.4 CPPs with multiple solutions

The projections in Sections 4.1, 4.2, and 4.3 generally guarantee for a locally unique solution, since they are based on nodal averaged normals and tangents on the master side. However, multiple local solutions could occur as illustrated for two-dimensional setups in Figure 7.

To overcome this problem, the past trajectory of the considered slave node is used to decide which projection is physically more reasonable. Therefore, a trajectory vector $p_j$ for the $j$th node is created via

$$p_j = x^{(1)}_{j,n+1} - x^{(1)}_{j,n}. \quad \text{(54)}$$

Here, $x^{(1)}_{j,n+1}$ is the current spatial coordinate, and $x^{(1)}_{j,n}$ is the spatial coordinate of the last converged time step. To decide which nodal normal should be employed, the angles between the trajectory vector and the considered normals are calculated. Furthermore, the angles between the negative trajectory vector and the nodal normals are computed. Eventually, the nodal normal candidate that encloses the smallest angle with $p$ or $-p$ is utilized as normal vector for the computation of the contact terms. This procedure is also shown in Figure 7. In the left part in Figure 7, three different solutions for the CPP are available. The normal $n^{(1)}_c$ represents the solution with the largest distance from slave to master surface, but it encloses the smallest angle with $p$, and thus, it is chosen to be the slave normal. In the right part in Figure 7, the penetration of a slave node is visualized, which could occur within the predictor step of a dynamic contact analysis. Here, the normal $n^{(1)}_c$ encloses the smallest angle with $-p$, and again, it is employed as physically reasonable choice of the slave normal.

### 4.5 Line-to-line projection

For the evaluation of contact between two crossing edges, it is necessary to detect the pair of points that minimizes the distance between the edges. For this purpose, a CPP between two lines is introduced. In the spatially discretized setup, an edge is represented by one-dimensional line elements. Since only first-order interpolations are considered, each line element consists of two nodes. The CPP between two line elements is realized by the following two conditions:

$$\begin{align*}
\tau^{(1)} \left( \xi^{(1)}_{\text{s}} \right) \left[ \sum_{k=1}^{n^{(1)}} N^{(1)}_k \left( \xi^{(1)}_{\text{s}} \right) x^{(1)}_k - \sum_{l=2}^{n^{(2)}} N^{(2)}_l \left( \xi^{(2)}_{\text{s}} \right) x^{(2)}_l \right] &= 0, \\
\tau^{(2)} \left( \xi^{(2)}_{\text{s}} \right) \left[ \sum_{k=1}^{n^{(1)}} N^{(1)}_k \left( \xi^{(1)}_{\text{s}} \right) x^{(1)}_k - \sum_{l=2}^{n^{(2)}} N^{(2)}_l \left( \xi^{(2)}_{\text{s}} \right) x^{(2)}_l \right] &= 0. \quad \text{(55)}
\end{align*}$$

These conditions enforce the distance vector between the closest points to be orthogonal to the corresponding tangents on both the slave side and the master side (see Figure 8). Similar procedures are employed for CPPs in the context of beam-to-beam contact scenarios, see the work of Meier et al.43

In (55), the slave tangent $\tau^{(1)}(\xi^{(1)}_{\text{s}})$ and the master tangent $\tau^{(2)}(\xi^{(2)}_{\text{s}})$ are computed according to (51) and (52), and thus, they depend on the parameter space coordinates $\xi^{(1)}_{\text{s}}$ and $\xi^{(2)}_{\text{s}}$. These parameter space coordinates represent the unknowns
FIGURE 8 Closest point projection between 2 arbitrarily oriented line elements for edge-to-edge contact

in (55), which are computed by a local Newton-Raphson scheme. The resulting spatial points are denoted as \( \hat{x}_x^{(1)} \) and \( \hat{x}_x^{(2)} \). Consequently, the normal can easily be defined as the cross product of the tangents

\[
\hat{n}_x^{(1)} = \frac{\mathbf{\tau}^{(1)}(\hat{\xi}_x^{(1)}) \times \mathbf{\tau}^{(2)}(\hat{\xi}_x^{(2)})}{\|\mathbf{\tau}^{(1)}(\hat{\xi}_x^{(1)}) \times \mathbf{\tau}^{(2)}(\hat{\xi}_x^{(2)})\|},
\]

and its orientation is once again determined by comparing with a suitable auxiliary slave normal vector (see Section 4.2).

5 | CONTACT EVALUATION

In the following section, suitable algorithms to numerically evaluate the point, line, and surface contact contributions are explained.

5.1 | Point contact

First, the numerical evaluation of point contact is considered. As stated in Section 3, there are 2 strategies to enforce the point contact scenario depending on the involved geometrical entities. Real point contact with vertices being involved is realized via the vector-valued Lagrange multiplier \( \lambda_* \), whereas the point contact that occurs due to crossing edges is treated by a penalty regularization.

5.1.1 | Lagrange multiplier approach

For the Lagrange multiplier constraint enforcement, the discrete counterpart to the vector \( \lambda_* \) is required. It is based on the discrete Lagrange multiplier space \( \mathcal{V}_{\lambda,h} \) being an approximation of \( \mathcal{V}_{\lambda} \). The notation for the discrete point Lagrange multiplier reads

\[
\lambda_{*,h} = \sum_{j=1}^{n_*} \Lambda_j \lambda_{*,j}.
\]

In (57), the shape functions \( \Lambda_j \) of the point Lagrange multiplier interpolation reduce to impulse functions being one at the nodes of physical vertices \( n_* \) and zero at all other points, ie,

\[
\Lambda_j = \begin{cases} 1 & \text{at } x_j, \\ 0 & \text{else}. \end{cases}
\]

Therefore, no interpolation functions are necessary between these points. The use of these Lagrange multipliers at slave vertices can easily be interpreted as the well-known node-to-segment formulation for point contact, see, for example, other works. Nevertheless, in the following, computational details on the numerical evaluation are briefly given. When inserting the introduced finite element discretizations (42) and (57) into the contact virtual work corresponding to the
point contact contribution in (37), the point contact matrices $D_\star \in \mathbb{R}^{3n_\star \times 3n_\star}$ and $M_\star \in \mathbb{R}^{3n_\star \times 3n_\star}$ can be computed by merging the nodal blocks

$$
D_\star[j, k] = D_{\star,jk} l_3 = \Lambda_j N^{(1)}_k l_1 = l_1, \quad j = 1, \ldots, n_\star, \quad k = 1, \ldots, n_\star^{(1)},
$$

$$
M_\star[j, l] = M_{\star, jl} l_3 = \Lambda_j \left( N^{(2)}_l \chi_h \right) l_1, \quad j = 1, \ldots, n_\star, \quad l = 1, \ldots, n_\star^{(2)}.
$$

Herein, $l_1 \in \mathbb{R}^{3\times 3}$ is the identity, and $\chi_h : \mathcal{T}^{(1)}_{ch} \to \mathcal{T}^{(2)}_{ch}$ represents a suitable discrete approximation of the mapping $\chi$ between the contact sides, see, eg, Dickopf and Krause and Puso for more details. Discretization of the nonpenetration constraint in (38) yields the discrete gap function $g_{\star, j}$ at each node $j$, ie,

$$
g_{\star, j} = g_{n, h} = n_j \left[ \hat{x}^{(2)} \left( x_j^{(1)} \right) - x_j^{(1)} \right], \quad j = 1, \ldots, n_\star.
$$

Here, $\hat{x}^{(2)}(x_j^{(1)})$ is the discrete point on the master contact interface side that results from the projection of the slave node position $x_j$, whereas $n_j$ is the discrete nodal normal at node $j$. The discrete relative tangential velocity $(v_{\star, \tau, \text{rel}})_j$ at node $j$ yields

$$(v_{\star, \tau, \text{rel}})_j = (l_3 - n_j \otimes n_j) : \left[ \sum_{l=1}^{n_\star} M_{\star, jl} x_l^{(2)} - l_1 x_j^{(2)} \right]
$$

$$
= (l_3 - n_j \otimes n_j) : \sum_{l=1}^{n_\star} M_{\star, jl} x_l^{(2)} \quad j = 1, \ldots, n_\star^{(1)},
$$

with the time derivative $(\dot{\cdot})$ being shifted from the nodal positions to the contact matrices to guarantee the satisfaction of the fundamental requirement of frame indifference, see the work of Puso and Laursen. Finally, the algorithm to evaluate the point contact contributions for one pair of vertex node and possibly contacting master element reads:

**Algorithm 1**

1. Project the slave node $x_j^{(1)}$ that corresponds to the vertex along its unit normal $n_j^{(1)}$ onto the master element to obtain the projected position on the master element $\hat{x}^{(2)}(x_j^{(1)})$.
2. Evaluate the contact matrices (59) and (60), gap function (61), and relative tangential velocity (62) at these points.

### 5.1.2 Penalty approach

The penalty regularization of the point contact scenario resulting from edge-to-edge contact is considered in the following. Since the nonparallel edge-to-edge scenario results in contact points that are generally not coincident with finite element nodes, stability requirements for the Lagrange multipliers at these points are hardly predictable. Additionally, a point Lagrange multiplier would be located in the interior of the support of the already defined edge Lagrange multipliers. Thus, from an engineering point of view, these situations could be described being as overconstrained. Therefore, the exact (pointwise) enforcement of the contact constraints on $\gamma_{\star, h}$ is relaxed via a penalty regularization.

**Remark 2**. Note that the scenario of nonparallel edges being in contact is the only situation in the entire nonsmooth contact algorithm, where a penalty regularization is needed. To the best of the authors' knowledge, no suitable Lagrange multiplier space can be a priori constructed for such a scenario, and consequently, the penalty approach cannot be avoided at this point.

By inserting the spatial discretization (42) into (41), the discrete penalty force vector of crossing edges results in

$$
f_{\star} = f_{\star}^{(1)} - f_{\star}^{(2)},
$$

with the discrete slave force vector $f_{\star}^{(1)}$ and the discrete master force vector $f_{\star}^{(2)}$. These can be computed by merging the nodal vectors

$$
f_{\star}^{(1)}[k] = f_{\text{pen}} N^{(1)}_k \left( \hat{x}_\star^{(1)} \right), \quad k = 1, \ldots, n_\star^{(1)},
$$

$$
f_{\star}^{(2)}[l] = f_{\text{pen}} N^{(2)}_l \left( \hat{x}_\star^{(2)} \right), \quad l = 1, \ldots, n_\star^{(2)}.
$$
Here, the expression in (41) reduces to a pointwise evaluation at the parameter space coordinates $\hat{\xi}_x^{(1)}$ and $\hat{\xi}_x^{(2)}$. These points represent the parameter space counterparts to the points in physical space $\hat{x}_x^{(1)}$ and $\hat{x}_x^{(2)}$ at which the closest distance between two line elements can be measured (see Section 4.5). Generally, these points are not coincident with finite element nodes and thus they have to be computed via a CPP between two line elements. In (64) and (65), the discrete penalty force vector $f_{pen}$ can be split into its normal part $f_{pen,n}$ and its tangential part $f_{pen,\tau}$. The normal force can be obtained by inserting the finite element discretization into (30), i.e.,

$$f_{pen,n} = \begin{cases} \epsilon_n(-g_{x,n})\hat{n}_x^{(1)} & \text{if } g_{x,n} \leq 0 \\ 0 & \text{if } g_{x,n} > 0 \end{cases}.$$  \hspace{1cm} (66)

Herein, $g_{x,n}$ is the discrete gap function between $\hat{x}_x^{(1)}$ and $\hat{x}_x^{(2)}$ and $\hat{n}_x^{(1)}$ is the unit normal vector defined along the connecting line between $\hat{x}_x^{(1)}$ and $\hat{x}_x^{(2)}$ but pointing in the outward direction of the slave body (see again Figure 8). For defining the discrete frictional penalty force, the discrete relative tangential velocity at $\hat{x}_x^{(1)}$ has to be defined as

$$v_{x,\tau,rel} = (I_3 - \hat{n}_x^{(1)} \otimes \hat{n}_x^{(1)}) \cdot \left[ \sum_{i=1}^{n_{cl}} \left( \frac{v_x^{(2)}}{v_x^{(1)}} \right) x^{(1)}_i - \sum_{k=1}^{n_{cl}} \left( \frac{v_x^{(1)}}{v_x^{(2)}} \right) x^{(2)}_k \right].$$  \hspace{1cm} (67)

Again, the time derivative is shifted to the discrete interpolation, which guarantees frame indifference. The Lie derivative in (31) is defined as

$$\mathcal{L}f_{pen,\tau} = (I_3 - \hat{n}_x^{(1)} \otimes \hat{n}_x^{(1)}) \dot{f}_{pen,\tau}.$$  \hspace{1cm} (68)

This expression contains only material time derivatives of the penalty force itself, and no time derivatives of base vectors are present. Thus, the Lie derivative in (68) is frame indifferent. For the calculation of the Coulomb frictional forces at the edge-to-edge contact points, a trial state-return map strategy is employed, which is an algorithmic time stepping procedure, see the works of Yang et al\(^ {11} \) and Laursen.\(^ {47} \) Here, a trial state is computed by assuming a perfect stick state during the time increment $\Delta t$

$$f_{pen,\tau_{n+1}} = f_{pen,\tau_n} - \epsilon_{\tau}v_{x,\tau,rel}. \hspace{1cm} (69)$$

The lower index $n$ is the time step counter. With this trial force at hand, the final tangential force can be computed with

$$f_{pen,\tau_{n+1}} = \begin{cases} f_{pen,\tau_{n+1}} \text{ if } f_{pen,\tau_{n+1}} - \epsilon_{\tau}f_{pen,n} \leq 0 \\ \| f_{pen,\tau_{n+1}} \| - \epsilon_{\tau}f_{pen,n} \text{ if } f_{pen,\tau_{n+1}} - \epsilon_{\tau}f_{pen,n} > 0 \end{cases}. \hspace{1cm} (70)$$

Here, the first case represents the perfect stick situation, which was assumed for the trial state, and thus, the final tangential force is identical to the trial force. The second case is the slip state where the final force has the absolute value of the Coulomb friction limit but points in the direction of the trial force. To sum up, the discrete penalty forces for the nonparallel edge-to-edge contact setting are no independent unknowns but can rather be expressed in terms of the discrete nodal displacements.

The algorithm for the nonparallel edge-to-edge contact is summarized in the following for a pair of 2 line elements.

**Algorithm 2**

1. Check if the line elements are parallel: If yes, no point contact of crossing edges will occur, otherwise continue.
2. Compute the points $\hat{x}_x^{(1)}$ and $\hat{x}_x^{(2)}$ with their corresponding parameter space coordinates $\hat{\xi}_x^{(1)}$ and $\hat{\xi}_x^{(2)}$. If the parameter space coordinates are outside of the defined intervals of their line elements, i.e., $\hat{\xi}_x^{(1)} < -1$ or $\hat{\xi}_x^{(2)} > 1$, the two line elements do not represent an edge-to-edge contact pair and the algorithm is completed.
3. Compute the normal part of penalty force vector $f_{pen,n}$ with (66) and the tangential force $f_{pen,\tau}$ with (70).
4. If the penalty regularization is active and the force vector is nonzero, compute the slave and master side force vectors $f_{x}^{(1)}$ and $f_{x}^{(2)}$ via (64) and (65).
5.2 Line contact

The discretization and numerical evaluation of line contact is considered next. All possible contact scenarios that result in line contact are treated with a Lagrange multiplier approach for constraint enforcement. Thus, a discrete counterpart of the line Lagrange multiplier must be introduced, which is based on the subset $\mathcal{M}_{l,h}$ being an approximation of the continuous space $\mathcal{M}_l$. The interpolation of the discrete line Lagrange multipliers reads

$$\lambda_{l,h} = \sum_{j=1}^{n_l^{(1)}} \Theta_j \lambda_{l,j}.$$  \hfill (71)

The shape functions $\Theta_j$ are based on the finite element parameter space for 1-dimensional curves $\xi$. The discrete line Lagrange multipliers are carried by the nodes $n_l^{(1)}$, which are defined on physical slave edges except for the nodes $n_s^{(1)}$ attached to vertices. Basically, there are two different types of Lagrange multiplier interpolation. First, the so-called standard shape functions can be employed, which are identical to the displacement interpolation of a two-node line element. Second, shape functions based on a biorthogonality condition can be utilized, which are also commonly known as dual shape functions. These dual shape functions are very advantageous, since they allow for a computationally efficient condensation procedure of the discrete Lagrange multipliers. More details on dual shape functions for contact elements with 1-dimensional parameter space can be found in the works of Popp et al.\(^7\) and Wohlmuth.\(^{48}\) If a line element is connected to a vertex, it would now carry one discrete line Lagrange multiplier and one discrete point Lagrange multiplier. Thus, the partition of unity would not be guaranteed anymore. In order to guarantee partition of unity for these elements, the line Lagrange multiplier shape functions have to be modified in the vicinity of the vertex node (see Figure 9). Here, modification of the shape function of the line Lagrange multiplier yields a constant interpolation to the point Lagrange multiplier. The modified shape functions are denoted by $\tilde{\Theta}_j$. This modification is also applicable for dual shape functions as shown in Figure 10. It should be pointed out that such modifications are well-established in mortar finite element methods in the context of Dirichlet boundary conditions at slave nodes or so-called crosspoints, which arise when multiple mortar subdomains meet at one point, see the aforementioned work\(^{48}\) and the work of Puso and Laursen.\(^{49}\)

By discretizing the contact virtual work related to line contact, the 2 slave side line contact matrices $D_{l} \in \mathbb{R}^{3n_l^{(1)} \times 3n_l^{(1)}}$ and $D_{l,*} \in \mathbb{R}^{3n_l^{(1)} \times 3n_s^{(1)}}$ arise

$$D_{l}[j,k] = D_{l,j,k} l_3 = \int_{\gamma_{l,h}} \Theta_j N_k^{(1)} dL l_3, \quad j = 1, \ldots, n_l^{(1)}, \ k = 1, \ldots, n_l^{(1)},$$  \hfill (72)

$$D_{l,*}[j,k] = D_{l,*j,k} l_3 = \int_{\gamma_{l,h}} \Theta_j N_k^{(1)} dL l_3, \quad j = 1, \ldots, n_l^{(1)}, \ k = 1, \ldots, n_s^{(1)}.$$  \hfill (73)

FIGURE 9 Modification of line Lagrange multiplier interpolation for standard shape functions due to the presence of a point Lagrange multiplier: unmodified shape function (left) and modified shape function (right). The point Lagrange multiplier is visualized as 1 impulse [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 10 Modification of line Lagrange multiplier interpolation for dual shape functions due to the presence of a point Lagrange multiplier: unmodified shape function (left) and modified shape function (right). The point Lagrange multiplier is visualized as 1 impulse [Colour figure can be viewed at wileyonlinelibrary.com]
Here, $D_{\parallel}$ couples the line Lagrange multipliers with the edge displacements and $D_{\ast}$ couples the line Lagrange multipliers with the displacements of vertex nodes. Both matrices can be assembled to the matrix $D_\ast \in \mathbb{R}^{3n_1(1) \times 3n_1(1)}$

$$D_\ast = [D_\parallel \ D_{\ast}] \quad \text{(74)}$$

which allows for an easier notation later on. In addition, the master side line contact matrix $M_\ast \in \mathbb{R}^{3n_1(1) \times 3n_1(2)}$ reads

$$M_{[j,l]} = M_{\ast,j} l_3 = \int_{\gamma_{\ast}^{(1)}} \Theta_j \left( N_l^{(2)} \alpha_{\chi h} \right) dL \quad j = 1, \ldots, n_1^{(1)} \quad l = 1, \ldots, n_1^{(2)} \quad \text{(75)}$$

These slave and master matrices can readily be interpreted as mortar matrices, since they result from an integration of a shape function product over the Lagrange multiplier support. Thus, they have mass matrix characteristics. When inserting the finite element discretization (42) and (71) into the normal part of the constraint equations for line contact (39), the discrete weighted gap $\tilde{g}_{\ast,j}$ at node $j$ for line contact emerges

$$\tilde{g}_{\ast,j} = \int_{\gamma_{\ast}^{(1)}} \Phi_j g_{\ast,h} dL \quad j = 1, \ldots, n_1^{(1)} \quad \text{(76)}$$

Additionally, the weighted relative tangential velocity $(\mathbf{v}_{\ast,rel})_j$ for line contact follows from discretizing the weak frictional sliding constraint in (39), viz,

$$(\mathbf{v}_{\ast,rel})_j = (l_3 - n_j \otimes n_j) \cdot \left[ \sum_{l=1}^{n_1(2)} M_{[j,l]} x_l^{(2)} - \sum_{k=1}^{n_1(1)} D_{[j,k]} x_k^{(1)} \right] \quad \text{(77)}$$

Again, frame indifference is achieved by formulating $(\mathbf{v}_{\ast,rel})_j$ in terms of time derivatives of the mortar matrices, see the aforementioned work.\textsuperscript{46}

In contrast to the point contact formulation in Section 5.1, a numerical integration procedure has to be carried out to evaluate the mortar matrices in (73) and (75) and the kinematic quantities (76) and (77). Since the mortar matrix $M_\ast$, the weighted gap $\tilde{g}_{\ast,j}$ and the weighted relative tangential velocity $(\mathbf{v}_{\ast,rel})_j$ all require an integration over the slave side line contact boundary $\gamma_{\ast}^{(1)}$, with integrands containing quantities from both sides, an exact evaluation cannot be achieved by standard Gauss quadrature rules simply being applied on each slave line element. This is due to the generally non-matching meshes that result from arbitrary line contact situations in the finite deformation regime. To overcome this problem, a so-called segment-based integration scheme is employed, which is based on the idea of preventing all possible discontinuities in the integrands by creating smooth integrable segments. This idea was first outlined for classical segment-to-segment contact formulations\textsuperscript{23,50} and then applied in the context of mortar formulations.\textsuperscript{49,51} Here, the basic principle is adopted for the line contact integration. In order to create line segments that contain only $C^1$-continuous integrands in (75), (76), and (77), the nodes of a considered slave line element and a master element are projected onto an auxiliary plane. Then, a line clipping algorithm is applied to determine the part of the line element that is located within the master element or the master element edges. The whole procedure is visualized in Figure 11. Additionally, the evaluation process is given in the following algorithm.

*FIGURE 11* Main steps of the segment-based integration scheme for the line contact algorithm. Construct an auxiliary plane (left), project slave, and master nodes onto the auxiliary plane (middle) and perform line clipping to identify line segments, for which the numerical integration is then performed (right)
Algorithm 3

1. Construct an auxiliary plane for numerical integration based on the master element center $\mathbf{x}_0^{(2)}$ and the corresponding element normal vector $\mathbf{n}_0^{(2)}$.
2. Project all $n_e^{(2)}$ master element nodes $\mathbf{x}_l^{(2)}, l = 1, \ldots, n_e^{(2)}$ along $\mathbf{n}_0^{(2)}$ onto the auxiliary plane to create the auxiliary master nodes $\tilde{x}_l^{(2)}$.
3. Project all $n_e^{(1)}$ slave line element nodes $\mathbf{x}_k^{(1)}, k = 1, \ldots, n_e^{(1)}$ along their nodal normal $\mathbf{n}_k^{(1)}$ onto the auxiliary plane to create the auxiliary slave nodes $\tilde{x}_k^{(1)}$.
4. Perform line clipping in the auxiliary plane in order to find the overlapping line segment of projected slave nodes. Adequate line clipping algorithms can, eg, be found in the work of Foley.\textsuperscript{52}
5. Define suitable integration points on the created line segment and find their counterparts on the slave and master element by an inverse mapping.
6. Perform numerical integration of the mortar matrices (73) and (75), the weighted gap (76) and the weighted relative tangential velocity (77).

5.3 Surface contact

Finally, the discretization of the surface contact is realized by introducing a mortar finite element approximation of the surface Lagrange multiplier. It is based on the discrete Lagrange multiplier subset $\mathcal{M}_{\text{o},h}$, which is an approximation of $\mathcal{M}_o$. The discretization of the surface Lagrange multiplier reads

$$ \lambda_{\text{o},h} = \sum_{j=1}^{n_\lambda} \Phi_j \lambda_{\text{o},j}, \quad (78) $$

with the shape functions $\Phi_j$ being based on the finite element parameter space for two-dimensional surfaces $\xi^{(i)} = (\xi^{(i)}, \eta^{(i)})$. Again, the shape functions $\Phi_j$ can be chosen as standard shape functions or dual shape functions based on a biorthogonality condition. Dual shape functions for two-dimensional surface can be found in the works of Popp et al\textsuperscript{8} and Wohlmuth.\textsuperscript{8,53} As for the interpolation of the discrete line Lagrange multipliers (71), the shape functions $\Phi_j$ have to be modified in the case of elements that are, at the same time, attached to different types of discrete Lagrange multipliers, ie, line or point Lagrange multipliers. This is necessary to guarantee partition of unity. In contrast to the line Lagrange multiplier interpolation $\Theta_j$, which is based on a one-dimensional parameter space, this modification becomes more complex for the surface shape functions $\Phi_j$. Thus, a general procedure based on a transformation of shape functions is defined for this modification. Starting point for deriving a suitable shape function transformation, which guarantees partition of unity, is a surface element with $n_e$ nodes. Out of this $n_e$ nodes, it is assumed that $\bar{n}_e$ nodes carry discrete line Lagrange multipliers or point Lagrange multipliers. Thus, a transformation coefficient $\zeta$ can be defined as

$$ \zeta = (n_e - \bar{n}_e)^{-1}, \quad (79) $$

It becomes obvious that the transformation is only valid when at least one node carries no other discrete Lagrange multiplier than a surface Lagrange multiplier, ie, when $\bar{n}_e < n_e$. The final transformation is exemplified for a four-node surface element with the first two nodes carrying surface Lagrange multipliers

$$ \left[ \begin{array}{c} \tilde{\Phi}_1 \\ \tilde{\Phi}_2 \\ \tilde{\Phi}_3 \end{array} \right] = \left[ \begin{array}{ccc} 1 & \zeta & \zeta \\ 0 & \zeta & \zeta \\ 0 & 0 & 0 \end{array} \right] \left[ \begin{array}{c} \Phi_1 \\ \Phi_2 \\ \Phi_3 \end{array} \right], \quad (80) $$

with the transformed shape functions $\tilde{\Phi}_j$ and the element transformation matrix $\mathbf{T}_e$. The transformation coefficient for the example given in (80) is obviously $\zeta = 0.5$, since $n_e = 4$ and $\bar{n}_e = 2$. Applying the transformation procedure to different shape functions, ie, three-node linear surface shape functions, is absolutely straightforward. The shape function modification is exemplarily shown in Figure 12. When applying the transformation scheme to the line Lagrange multiplier interpolation $\Theta_j$, we would also formally obtain the constant interpolation of the shape function that has already intuitively been given in Figure 9.
When inserting the finite element discretization for the displacement fields \( \mathbf{u} \) and the surface Lagrange multiplier \( \lambda \) into the surface contact virtual work contribution in (37), the three slave side mortar matrices for surface contact \( \mathbf{D}_{oo} \in \mathbb{R}^{3n^o \times 3n^o} \), \( \mathbf{D}_{oi} \in \mathbb{R}^{3n^o \times 3n^i} \), and \( \mathbf{D}_{os} \in \mathbb{R}^{3n^o \times 3n^s} \) can be computed, viz,

\[
\mathbf{D}_{oo}[j,k] = \mathbf{D}_{oo,jk} \mathbf{l}_3 = \int_{\gamma_{ch}^{(1)}} \Phi_j \lambda^{(1)}_k \, d\gamma \mathbf{l}_3, \quad j = 1, \ldots, n^o, \quad k = 1, \ldots, n^o, \tag{81}
\]

\[
\mathbf{D}_{oi}[j,k] = \mathbf{D}_{oi,jk} \mathbf{l}_3 = \int_{\gamma_{ch}^{(1)}} \Phi_j \lambda^{(1)}_k \, d\gamma \mathbf{l}_3, \quad j = 1, \ldots, n^o, \quad k = 1, \ldots, n^i, \tag{82}
\]

\[
\mathbf{D}_{os}[j,k] = \mathbf{D}_{os,jk} \mathbf{l}_3 = \int_{\gamma_{ch}^{(1)}} \Phi_j \lambda^{(1)}_k \, d\gamma \mathbf{l}_3, \quad j = 1, \ldots, n^o, \quad k = 1, \ldots, n^s. \tag{83}
\]

These matrices represent the coupling of the surface Lagrange multipliers to surface, edge, and vertex nodes, respectively. The complete slave side mortar matrix reads

\[
\mathbf{D}_o = [\mathbf{D}_{oo} \quad \mathbf{D}_{oi} \quad \mathbf{D}_{os}]. \tag{84}
\]

Furthermore, the master side mortar matrix \( \mathbf{M}_o \in \mathbb{R}^{3n^o \times 3n^o} \) is given as

\[
\mathbf{M}_o[j,l] = \mathbf{M}_{oj,l} \mathbf{l}_3 = \int_{\gamma_{ch}^{(1)}} \Phi_j \lambda^{(2)}_l \, d\gamma \mathbf{l}_3, \quad j = 1, \ldots, n^o, \quad l = 1, \ldots, n^o. \tag{85}
\]

The discrete counterpart of the gap function for surface contact is also a mortar-typical weighted gap, since it is integrated over the boundary \( \gamma_{ch}^{(1)} \), via

\[
\bar{g}_{o,j} = \int_{\gamma_{ch}^{(1)}} \Phi_j g_{ch} \, dA \quad j = 1, \ldots, n^o. \tag{86}
\]

Additionally, the weighted relative tangential velocity for surface contact is given as

\[
\left( \mathbf{v}_{o,r,rel} \right)_j = \left( \mathbf{l}_3 - \mathbf{n}_j \otimes \mathbf{n}_j \right) \cdot \left[ \sum_{l=1}^{n^o} \mathbf{M}_o[j,l] \mathbf{x}_{l}^{(1)} - \sum_{k=1}^{n^o} \mathbf{D}_o[j,k] \mathbf{x}_{k}^{(1)} \right]. \tag{87}
\]

The evaluation of these discrete quantities again requires an accurate numerical integration procedure. Here, the well-known segment-based integration scheme for surface mortar methods is employed (see Figure 13). This segmentation scheme was already investigated extensively in the literature and can be found, for example, in the aforementioned works\(^{49}\) and our other work.\(^{35}\) For the sake of completeness, the segment-based integration scheme is briefly recapitulated in the following algorithm.
Algorithm 4

1. Construct an auxiliary plane for numerical integration based on the slave element center $x_{0}^{(1)}$ and the corresponding element normal vector $n_{0}^{(1)}$.
2. Project all $n_{e}^{(1)}$ slave element nodes $x_{k}^{(1)}$, $k = 1, \ldots, n_{e}^{(1)}$ along $n_{0}^{(1)}$ onto the auxiliary plane to create the auxiliary slave nodes $\tilde{x}_{k}^{(1)}$.
3. Project all $n_{e}^{(2)}$ master element nodes $x_{l}^{(2)}$, $l = 1, \ldots, n_{e}^{(2)}$ along $n_{0}^{(1)}$ onto the auxiliary plane to create the auxiliary master nodes $\tilde{x}_{l}^{(2)}$.
4. Perform polygon clipping in the auxiliary plane to find the overlapping region of projected slave and master element. Adequate polygon clipping algorithms can, e.g., be found in the work of Foley.52
5. Perform a decomposition of the clip polygon into triangular subdomains, which will be used for numerical integration and therefore called integration cells.
6. Define suitable integration points on the triangular integration cells and find their counterparts on the slave and master element by an inverse mapping.
7. Perform numerical integration of the mortar matrices (81) and (85), the weighted gap (86), and the weighted relative tangential velocity (87).

6 | COMBINED FORMULATION: GLOBAL SOLUTION SCHEME

In the following section, the semidiscrete problem setup will be stated for the combined point, line, and surface contact scenarios that is characteristic for nonsmooth geometries in 3D. Moreover, the global solution scheme is briefly outlined and our computationally efficient condensation procedure based on dual shape functions for the Lagrange multipliers is explained. Finally, a simple postprocessing scheme for the interface tractions is given.

6.1 | Combined semidiscrete formulation

When combining all previously presented contact contributions, the semidiscrete balance of linear momentum that results from mortar finite element discretization reads

$$r := K_{\text{mass}} \ddot{d} + K_{\text{damp}} \dot{d} + f_{\text{int}}(d) - f_{\text{ext}} + f_{c}(d, \lambda) = 0,$$

(88)
with the mass matrix \( K_{\text{mass}} \) and the damping matrix \( K_{\text{damp}} \), which is based on the widely used Rayleigh model for viscous damping. Furthermore, the internal and external force vectors are denoted as \( f_{\text{int}}(d) \) and \( f_{\text{ext}} \). The contact contribution \( f_c(d, \lambda) \) is split into the Lagrange multiplier force vector denoted as \( f_c(d, \lambda) \) and the penalty force vector for contact of crossing edges \( f_\chi(d) \), which was already defined in (63), ie,

\[
f_c(d, \lambda) = f_c(d, \lambda) + f_\chi(d) .
\]

(89)

The Lagrange multiplier force vector is defined via the global slave and master side mortar matrices \( D \in \mathbb{R}^{3n(1) \times 3n(2)} \) and \( M \in \mathbb{R}^{3n(2) \times 3n(2)} \)

\[
D = [D_+ \ D_0 \ D_\circ], \quad M = [M_+ \ M_0 \ M_\circ] .
\]

(90)

With these matrices, the global contact force vector due to Lagrange multiplier–based contact reads

\[
f_\lambda = [0 - M \ D]^T \lambda ,
\]

(91)

with the global Lagrange multiplier vector

\[
\lambda = [\lambda_+ \ \lambda_\circ \ \lambda_\circ]^T.
\]

(92)

Obviously, all discrete Lagrange multipliers for point, line, and surface contact are contained in \( \lambda \). The inequality constraints for normal and tangential contact are generally stated in semidiscrete form as

\[
g_j \geq 0, \quad \lambda_{n,j} \geq 0, \quad \lambda_{n,j} g_j = 0, \quad j = 1, \ldots, n^{(1)} .
\]

(93)

\[
\Phi_j := \| (\lambda_{\tau,j} ) \| - \| (\lambda_n) \| \leq 0 ,
\]

\[
(\mathbf{v}_{\text{rel},j}) + \beta_j (\lambda_{\tau,j}) = 0, \quad \beta_j \geq 0 , \quad \Phi_j \beta_j = 0, \quad j = 1, \ldots, n^{(1)} .
\]

(94)

The semidiscrete problem formulation given in (88)-(94) is particularly elegant, since we no longer have to distinguish between point, line, and surface contact. For example, in (93) and (94), the gap \( g_j \), the relative tangential velocity \( \mathbf{v}_{\text{rel},j} \), the complementarity parameter \( \beta_j \), and the Lagrange multiplier \( \lambda \) have different physical interpretations for vertex, edge, and surface contact, respectively. Although it contains point, line, and surface contact formulations, the semidiscrete problem formulation remains as simple and compact as for pure surface contact, see, eg, Popp et al.

### 6.2 Semismooth Newton method

The inequality constraints in (93) and (94) introduce additional nonlinearities and nonsmoothness into the global problem formulation, since they divide the set of all slave nodes into the three a priori unknown sets of stick, slip, and inactive nodes. To resolve these nonlinearities, a primal-dual active set strategy is employed as solution algorithm. For this purpose, the set of all slave nodes \( S \) is split into an active set \( A \) for nodes currently being active and an inactive set \( I \) for nodes, which are currently not in contact, regardless whether the respective Lagrange multiplier belongs to the point, line, or surface contact sets. Additionally, the set of all active nodes \( A \) is split into a slip set \( S_\circ \) and a stick set \( S_\circ \) to treat frictional contact. To solve the fully discretized system of nonlinear equations within each time step, the primal-dual active set strategy is reinterpreted as a semismooth Newton method, see the works of Christensen et al and Hintermüller et al. To this end, nodal NCP functions for nonpenetration and frictional sliding are introduced. These are nonsmooth equality constraints but equivalently express the inequality constraints in (93) and (94). The complementarity functions for the normal contact \( C_{n,j} \) and the frictional sliding \( C_{\tau,j} \) for the node \( n_j \) read

\[
C_{n,j}(\lambda_j, d) = \lambda_{n,j} - \max(0, \lambda_{n,j} - c_n g_{n,j}),
\]

(95)

\[
C_{\tau,j}(\lambda_j, d) = \max \left( \frac{\| (\lambda_{\tau,j} ) \|}{\| (\lambda_n) \|}, \| \lambda_{\tau,j} + c_{\tau} u_{\text{rel},j} \| \right) - \frac{\| (\lambda_n) \|}{\| (\lambda_n) \|} \max(0, \lambda_{\tau,j} - c_n g_{n,j}) (\lambda_{\tau,j} + c_{\tau} u_{\text{rel},j}),
\]

\[
c_n, c_\tau > 0 ,
\]

(96)

with the two complementarity parameters \( c_n \) and \( c_\tau \), see other works. The relative slip increment \( u_{\text{rel}} \) directly arises from multiplication of the relative tangential velocity \( v_{\text{rel}} \) with the time increment \( \Delta t \). The distinction between the active set \( A \), inactive set \( I \), slip set \( S_\circ \) and stick set \( S_\circ \) is implicitly contained in (95) and (96), and updates are carried out after each semismooth Newton step. Exact constraint enforcement is achieved if both NCP functions are equal to zero, ie

\[
C_{n,j}(\lambda_j, d) = 0, \quad C_{\tau,j}(\lambda_j, d) = 0 .
\]

(97)
Note, that the parameters $c_n$ as well as $c_l$ do not influence the accuracy of results but may control the convergence behavior. The parameter $c_n$ is typically chosen to be at the order of magnitude of Young’s modulus $E$ of the involved bodies as suggested in the work of Hüeber and Wohlmuth. The tangential parameter $c_l$ balances the different scales of the tangential part of the Lagrange multiplier and the relative tangential slip increment. Having in mind that the Lagrange multipliers as well as the slip increment are of different units for point, line, and surface contact, $c_l$ has different interpretations for these contact scenarios. Consequently, a clever choice of $c_l$ depends on the current load, deformation, and contact status. However, for all our numerical examples we could simply use a constant value for $c_l$ for all slave nodes without recognizing any deterioration of convergence or even convergence problems.

**Remark 3.** Strictly speaking, the nodewise decoupled enforcement of the contact constraints in (93) and (94) is only valid for a diagonal slave side contact matrix $D$ as would be the case for dual mortar formulations of pure surface contact. This cannot entirely be achieved for the combined contact formulation presented in this contribution, due to off-diagonal blocks $D_{nn}$, $D_{*n}$ and $D_{***}$. Still, the main diagonal blocks $D_{nn}$, $D_{ll}$, and $D_{*l}$ can be brought to diagonal form by employing dual shape functions, and thus, most of the nodes are indeed decoupled. We would like to point out that a fully consistent and rigorous variational formulation would require coupled NCP functions containing all coupled slave nodes or suitable lumping techniques, see the work of Blum et al. Nevertheless, nodewise decoupled NCP functions are utilized throughout this contribution without any negative influence on our numerical results.

Eventually, the global problem formulation to be solved consists of Equations (88), (95), and (96). Herein, the nodal sets $A$, $I$, $S_C$, and $S_T$ are updated after each semismooth Newton iteration.

### 6.3 Algebraic representation

In this section, an algebraic representation of the linearized system to be solved within each semismooth Newton step is provided. The resulting system of equations is of saddle-point type and looks in an abstract form as follows:

$$
\begin{pmatrix}
K_{NN} & K_{NM} & K_{NS} & 0 \\
K_{MN} & K_{MM} & K_{MS} & -M^T \\
K_{SN} & K_{SM} & K_{SS} & D^T \\
0 & C_M & C_S & C_λ
\end{pmatrix}
\begin{pmatrix}
\Delta d_N \\
\Delta d_M \\
\Delta d_S \\
\Delta λ
\end{pmatrix}
= -\begin{pmatrix}
r_N \\
r_M \\
r_S \\
r_λ
\end{pmatrix}
\tag{98}
$$

The system of equations in (98) is of increased system size compared to classical structural mechanics problems, since both displacements and Lagrange multipliers show up as primary unknowns. Thus, the solution vector contains increments of discrete displacements $\Delta d$ and Lagrange multipliers $\Delta λ$. The discrete global vector of Lagrange multipliers contains point, line, and surface Lagrange multipliers as stated in (92). Furthermore, the displacement unknowns are split into the three sets of bulk (interior) nodes $(·)_N$, slave nodes $(·)_S$, and master nodes $(·)_M$. The matrix blocks denoted with $K$ contain terms from the linearization of the internal force vector $f_{int}$, and from time discretization of the inertia and damping forces. The upper tilde symbol $(·)$ indicates additional stiffness contributions from the linearized contact force vector $f_c$. The matrix blocks denoted with $C$ represent the linearization of the complementarity functions in (95) and (96). One of the main advantages of dual mortar methods in the context of pure surface contact is the nodewise decoupling of the discrete slave side contact virtual work contribution, which results in a diagonal slave mortar matrix $D$ and allows for a computationally efficient elimination of the additional Lagrange multiplier unknowns, see other works. For this purpose, the third row of (98) is utilized to express the Lagrange multiplier increment in terms of the displacement increments

$$
\Delta λ = -D^{-T}(K_{NN}Δd_N + K_{NM}Δd_M + K_{SS}Δd_S - r_S).
\tag{99}
$$

As aforementioned, inverting the diagonal matrix $D$ is of negligible computational cost for dual mortar formulations in the case of purely surface-based contact. In this contribution, however, not all slave nodes are completely decoupled due to the 3 different sets of Lagrange multipliers (point, line, and surface contact). Therefore, the global slave side mortar matrix exhibits the following block structure:

$$
D = \begin{bmatrix}
D_{nn} & D_{nl} & D_{*n} \\
0 & D_{ll} & D_{*l} \\
0 & 0 & D_{***}
\end{bmatrix}
\tag{100}
$$
It becomes obvious that $D$ is an upper triangular matrix, and its inversion is not as trivial anymore as for classical dual mortar methods. Nevertheless, its inversion is still possible at moderate computational costs as follows:

$$D^{-1} = \begin{bmatrix} D_{oo}^{-1} & 0 & 0 \\ 0 & D_{ii}^{-1} & 0 \\ 0 & 0 & D_{ss}^{-1} \end{bmatrix} \begin{bmatrix} I & -D_{oi} & D_{oi}^{-1} \\ 0 & I & -D_{si}D_{ss}^{-1} \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} D_{oo} & D_{oi}D_{ii}^{-1} \\ D_{oi}D_{ii}^{-1} & D_{ii} \end{bmatrix} + \begin{bmatrix} D_{\ast o}D_{oi}D_{ii}^{-1} \end{bmatrix}.$$

(101)

Herein, only the main diagonal blocks $D_{oo}, D_{ii},$ and $D_{ss}$ have to be inverted. By employing dual shape functions, these main diagonal blocks are still of diagonal shape, and thus, inverting the global $D$ matrix is still rather efficient. By expressing the discrete Lagrange multiplier unknowns in terms of displacement unknowns as stated in (99), the final condensed system of equations becomes

$$\begin{pmatrix} K_{NN} & K_{NM} & K_{NS} \\ K_{MN} + P^T K_{SN} & K_{MM} + P^T K_{SM} & K_{MS} + P^T K_{SS} \\ C_s D^{-1} K_{SN} & C_s + C_r D^{-1} K_{SM} & C_s + C_r D^{-1} K_{SS} \end{pmatrix} \begin{pmatrix} \Delta d_N \\ \Delta d_M \end{pmatrix} = \begin{pmatrix} r_N \\ r_M + P^T r_S \end{pmatrix},$$

(102)

with the well-known mortar projection operator

$$P = D^{-1} M.$$

(103)

The final system of equations in (102) is of constant system size and the only remaining degrees of freedom are the displacement unknowns. All information on the point, line, and surface contact enforcement is included in the modified system matrix and no saddle-point structure occurs anymore. The discrete Lagrange multipliers can be obtained through a simple postprocessing step based on (99).

### 6.4 Postprocessing of interface tractions

Since the discrete Lagrange multiplier unknowns contained in the global vector $\lambda$ in (92) are utilized to enforce different contact scenarios, they also have different physical interpretations. Concretely, the point Lagrange multipliers $\lambda_\ast$ represent discrete point forces, the line Lagrange multipliers $\lambda$ represent line loads, ie, a force divided by a distance quantity, and the surface Lagrange multipliers $\lambda_s$ represent classical interface tractions, ie, a force divided by an area quantity. In order to evaluate and assess the accuracy of computational results, a uniform interface traction quantity is very helpful for postprocessing. Thus, the discrete nodal forces (in a finite element sense) acting on the slave side are considered first. They result from multiplying the global vector of Lagrange multipliers $\lambda$ with the transpose of the slave side mortar matrix $D$ and adding the penalty force contributions from the contact of crossing edges

$$f_c^{(1)} = D^T \lambda + f_c^{(1)}.$$

(104)

This vector contains discrete forces acting on slave nodes. In order to compute discrete tractions for postprocessing, the global slave side force vector $f_c^{(1)}$ is scaled by the diagonal matrix $A$ that contains support area information of the slave nodes. It can be computed by assembling the nodal blocks

$$A[j, k] = A_{jk} = \delta_{jk} \int_{V_c^{(1)}} N_j^{(1)} d \gamma I_s, \quad j, k = 1, \ldots, n^{(1)},$$

(105)

with $\delta_{jk}$ being the well-known Kronecker delta. The effective slave side interface tractions containing the effect of point, line, and surface contact, then follow as

$$t^{(1)} = A^{-1} f_c^{(1)}.$$

(106)

This postprocessing quantity will be used for all our numerical examples.

**Remark 4.** The area matrix $A$ is basically a slave side mortar matrix $D_s$ with surface Lagrange multipliers being defined on all slave nodes including edge nodes and vertex nodes. By employing dual shape functions, $D_s$ automatically becomes a diagonal matrix and thus identical to $A$.

**Remark 5.** The scaling presented in (106) is also applicable for the master side. The master side tractions can be computed by employing the discrete master force vector $f_c^{(2)}$ and performing the scaling with an area matrix that is integrated over the master surface.
In this section, the developed algorithm that has been implemented in our in-house C++-code is validated with numerical examples. The first one is a well-known patch test for surface contact, which is employed to demonstrate that the developed all-entity contact algorithm does not affect the solution quality of pure surface-to-surface contact. Second, an edge-to-surface contact scenario is investigated, and the mortar-based line contact algorithm is compared to a node-to-segment algorithm and a mortar-based surface contact algorithm. Afterward, the special case of contacting edges is analyzed, and a nonparallel edge-to-edge example is given to demonstrate the performance of the penalty regularization. The correct transition between point, line, and surface contact is demonstrated in detail in a bending plate example. Finally, an example from implicit dynamics, i.e., a falling coin, is investigated with respect to robustness and conservation of linear and angular momentum.

### 7.1 Surface contact: patch tests

The first example is a simple patch test for a surface contact scenario, which is investigated to show the ability of the proposed method to represent a constant stress state across nonmatching discretizations at the contact interface. It is well-known that mortar contact formulations are able to successfully pass this test setup, whereas classical node-to-segment formulations would fail, see the works of El-Abbasi and Bathe and Taylor and Papadopoulos. However, the method that has been introduced in this contribution modifies the mortar contact formulation at vertices and edges of the contact boundary, and thus, the patch test has to be revisited to demonstrate that these modifications have no negative influence on the solution accuracy as compared with pure surface contact. The test setup consists of a large block with dimensions $10 \times 10 \times 4$ and a small block with dimensions $5 \times 5 \times 4$. The larger block is completely supported at its lower surface and its upper surface acts as master contact side. The smaller block lies on top of the larger one and acts as slave body. The employed finite element meshes are shown in Figure 14. The nodes attached to vertices carry point Lagrange multipliers, the nodes on edges carry line Lagrange multipliers and all other slave nodes are subject to surface Lagrange multipliers. The upper surface of the slave body and the noncontact part of the upper surface of the lower body are loaded with the constant pressure $p = -1.0$ in Z-direction. The employed material model for both bodies is based on a compressible neo-Hookean law with Young’s modulus $E = 1000$ and Poisson’s ratio $\nu = 0.0$. In addition, frictionless contact is assumed for the simulation. The resulting displacements and Cauchy stresses are shown in Figure 14. It can be seen that the contact patch test requirements are perfectly fulfilled, i.e., the test is passed to machine precision. In addition, the resulting Lagrange multiplier values are visualized in the left part in Figure 15. Here, only the four Lagrange multiplier vectors of the inner surface nodes have noteworthy nonzero values. This is due to the surface Lagrange multipliers being able to represent the constant stress state within the contact interface and, thus, to completely fulfill the contact constraints. Consequently, the point and the line Lagrange multipliers do not significantly contribute to the contact virtual work. Instead, their contact status can be described as the limit case, where the gap values are zero but no noteworthy nonzero Lagrange multiplier values occur. Numerically, this could lead to problems due to an arbitrarily changing contact status of the vertex nodes and the edge nodes for this example, whereas the constraint residual and the structural residual converge perfectly. Therefore, convergence behavior of the Lagrange multiplier increment and the gap function are tracked and changes in the active set are ignored as convergence criterion when both quantities simultaneously approach zero.
However, the Lagrange multiplier solution in the left part in Figure 15 cannot be interpreted as interface traction since the shape function modification in (80) has been applied to the surface Lagrange multiplier shape functions. Taking into account the postprocessing procedure explained in Section 6.4, a representative solution for the contact traction can be derived, which is visualized in the right part in Figure 15. There, the expected constant stress state at each slave node can be observed.

7.2 | Edge-to-surface contact

The next example is introduced to demonstrate the ability of the proposed contact algorithm to represent a constant stress state for edge-contact situations, i.e., it can be interpreted as an edge-to-surface contact patch test. The example consists of a rigid plate that is completely fixed and an elastic cube. The edge length of the cube is $l = 2$, and its material model is of neo-Hookean type with Young's modulus $E = 22.5 \cdot 10^5$ and Poisson's ratio $\nu = 0.0$. It is rotated by 45° twice around two different axes, such that its contact edge is equal the diagonal of the fixed plate. The cube acts as slave body and the plate as master body. During the entire simulation, inertia effects and damping are neglected. The initial distance between the bodies is $d = 2.29 \cdot 10^{-2}$, and the cube is pressed against the plate with a total prescribed displacement at its upper surfaces of $d_{\text{max}} = 0.2$. This displacement boundary condition is applied within 12 quasi-static load steps. This setup is calculated with 3 different contact algorithms. First, the proposed algorithm with its combination of point, line, and surface Lagrange multipliers. Second, with a classical mortar contact algorithm and, finally, with a classical node-to-segment formulation. The resulting displacement solutions are shown in Figure 16. Here, the left part shows the solution for the proposed contact algorithm, which successfully enforces the nonpenetration conditions and leads to a physically correct displacement state. The right part shows a solution computed with a classical node-to-segment algorithm, which also shows a reasonable displacement state. The classical mortar formulation in the middle in Figure 16 obviously produces a large penetration, and contact is only detected very late. This is due to the surface weighted gap function inherent to the classical mortar formulation, see the work of Popp et al.\textsuperscript{8} In Figure 17, the interface tractions are visualized. From this, it can be further deduced that the proposed algorithm with its line Lagrange multipliers perfectly passes the patch test by producing a constant stress state, which is to be expected for this test setup. Again, the visualized stress state is
based on the postprocessing procedure from Section 6.4. The only discrete Lagrange multiplier with a nonzero value is the line Lagrange multiplier at the middle node of the contacting edge. The vertex Lagrange multipliers again exhibit the limit case, where the gap functions are zero but no noteworthy nonzero value for the point Lagrange multiplier arises. Consequently, the entire set of contact constraints are consistently enforced with only one discrete line Lagrange multiplier. In contrast, the classical mortar algorithm produces smaller stresses, since the predicted penetration is far from the physically meaningful state of being zero. Finally, the node-to-segment algorithm, while yielding plausible results from a qualitative point of view, is not able to produce a constant stress state, which is, of course, a well-known deficiency of this type of contact discretization.

7.3 Parallel edge-to-edge contact

The next example is a parallel edge-to-edge contact situation of two elastic cubes. Here, the robustness of the proposed numerical evaluation of the line contact algorithm shall be demonstrated. The material model for both cubes is identical to the elastic cube from the edge-contact example in Section 7.2. Both cubes are rotated by 45° around their individual X-axis such that their edges are perfectly parallel (see Figure 18). The cubes have identical dimensions of 1 × 1 × 1, and trilinear hexahedral elements are employed for the spatial discretization. The finite element meshes are nonmatching at the contacting edges as visualized in Figure 18. The upper block is defined as the slave side and the lower body represents the master side. The lower cube is supported at its lower surfaces and the upper cube is subjected to a prescribed motion at its upper surfaces. Their initial distance is \( d = 0.083 \) and the total prescribed displacement in negative Z-direction is \( d_{\text{max}} = 0.166 \), which is enforced within 50 quasi-static load steps. The resulting contact tractions and the deformed meshes are again shown in Figure 18. The proposed contact algorithm yields perfectly identical contact tractions at all
slave nodes. This is due to the highly accurate segment-based integration scheme explained in Section 5.2. Furthermore, the introduced definition for the nodal normal field and the handling of the CPPs lead to a perfect edge-to-edge contact scenario, which can again be interpreted as a special kind of (edge-to-edge) contact patch test. It should be pointed out, however, that the solution of this example represents somewhat of an academic limit case and is therefore rather sensitive with respect to nodal normal definitions and other numerical evaluation procedures.

### 7.4 Nonparallel edge-to-edge contact

In the following, a scenario of nonparallel edges being in frictional contact is investigated. Again, two cubes with identical dimensions of $1 \times 1 \times 1$ are considered with the same material properties as in the previous examples, i.e., the material model is of neo-Hookean type with Young's modulus $E = 22.5 \cdot 10^5$ and Poisson's ratio $\nu = 0.0$. Both cubes are discretized equally with trilinear hexahedral elements, as shown in Figure 19. The upper cube is rotated by 45° around the X-axis, and the lower cube is rotated by 45° around the Y-axis. The initial distance between the two bodies is $d_i = 0.01$. The lower block is completely fixed at its two lower surfaces and acts as master body for the contact description. Consequently, the upper block is defined as slave body. The two top surfaces of the upper block are subjected to a prescribed motion $\mathbf{d}_{\text{dpc}} = [d_x, d_y, d_z]$, which is split into two different motions. First, in the time interval $0 < t \leq 5$, the displacements in X- and Y-direction are fixed, i.e., $d_x = d_y = 0$, and the displacements in Z-direction are defined by $d_z = -0.012t$. Afterward, in the time interval $5 < t \leq 25$, the displacements in Z-direction are fixed at $d_z = -0.06$, and the displacements in X- and Y-direction are defined as $d_x = d_y = 0.015(t - 5)$. The time step size is chosen as $\Delta t = 0.1$. The resulting point contact of the nonparallel edge-to-edge scenario is realized with a penalty regularization as explained in Section 5.1.2. The required penalty parameters are defined to be at the order of the Young's modulus, i.e., $\epsilon_n = \epsilon_t = 22.5 \cdot 10^5$, and the Coulomb friction coefficient is chosen as $\mu = 0.9$.

The resulting slave side normal contact forces $\mathbf{f}_{n, \text{nc}}^{(1)}$ and the deformation state are visualized for $t = 5$ in Figure 19. The contact point is perfectly in the middle of an edge (line) element, and according to (64), the contact force due to the penalization of the penetration is equally distributed between the adjacent nodes. The tangential contact forces are shown in Figure 20. As expected, they point in the opposite direction of the tangential movement and are consistently split between the two involved nodes of an edge (line) element. In the middle part in Figure 20, it can be seen that the contact point is located near the left involved node, and thus, its share of the overall tangential force is much larger than for the other involved node. The final state of the simulation is shown in the right part in Figure 20. Here, the contact point has moved from the initial slave line element to its left neighbor, and consequently, the outer left node now gets a share of the contact force. This transition from one line element to another only works in a robust manner when $C^1$-continuous nodal tangent fields are used. These fields are based on unique tangent definitions at each node (see (51) and (52)).

Finally, the conservation of linear and angular momentum is investigated. According to Puso and Laursen,\textsuperscript{46} conservation of linear and angular momentum can be formulated as the conservation requirement of interface forces and moments acting on the slave and master side, respectively. Thus, the resulting plots for the absolute interface forces and moments

![FIGURE 19](Nonparallel edge-to-edge contact: initial setting (left) and deformed state with normal contact force at $t = 5$ (right))
in this numerical example are given in Figure 21. It can be seen that the slave and master sides behave identically, which leads to an excellent balance of forces and moments at the contact interface. The interface moment only takes nonzero values after time step 50, when the tangential movement begins. The kink arising in all curves at approximately time step 185 occurs due to the transition of the contact point from one line element to another. A detailed view on the conservation of forces and moments is provided in Figure 22. In the left subfigure, the relative error of the sum of forces from the slave and master side with respect to the slave side force is plotted for contact with friction and frictionless contact. It can be seen that the force and thus the linear momentum is exactly conserved for both scenarios, ie, an accuracy up to machine precision is reached. The right plot in Figure 22 shows the same error calculation for the interface moments.

FIGURE 20  Deformed state and tangential contact forces for nonparallel edge-to-edge contact: initial setting (left), solution at step 100 (middle), and solution at step 250 (right)

FIGURE 21  Absolute values of interface forces (left) and moments (right) for frictional contact of nonparallel edges. Master quantities are defined to be negative for visualization [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 22  Relative errors of interface forces (left) and moments (right) for the contact of nonparallel edges. Comparison of contact with friction (red) and contact without friction (blue) [Colour figure can be viewed at wileyonlinelibrary.com]
It can be seen that the conservation of interface moment and consequently angular momentum is only guaranteed for contact without frictional effects. For the simulation without friction, again, an accuracy up to machine precision can be achieved. However, the relative error for contact with friction seems to be significant at the beginning of the simulation. Yet, as shown in the right part in Figure 21, the absolute value of the interface moment for the slave and master side is practically zero at the beginning of the simulation. Thus, the large relative error that can be observed in Figure 22 until time step 50 is not relevant in practice. As soon as the absolute value of the interface moments increases after time step 50, the relative error decreases toward circa 0.5%, which is acceptable from an engineering point of view. It should also be noted that with an increasing penalty parameter $\epsilon_n$ and for finer meshes, the accuracy in conservation of interface moments, i.e., angular momentum, can be further improved.

7.5 | Bending plate

The next example is utilized to demonstrate the consistent transition between point, line, and surface contact. To this end, an elastic plate is pressed against a rigid foundation. The problem setup is visualized in Figure 23. The plate is meshed
with $70 \times 70 \times 3$ trilinear hexahedral elements with enhanced assumed strain (EAS) element technology. The employed material model is of neo-Hookean type with Young’s modulus $E = 6.25 \times 10^7$ and Poisson’s ratio $\nu = 0.0$. The initial distance between the bodies at their closest points is $d = 0.1$. The nodes at edge $A$, which point in thickness direction of the plate, are only allowed to move in $Z$-direction; and the nodes at edge $B$ are subjected to a prescribed total motion of $d_{\text{max}} = 0.72$ in negative $Z$-direction, which is enforced within 35 steps. All frictional and inertia effects are neglected within this simulation. Obviously, the first contact occurs at the vertex node at the lower end of edge $A$. The very first steps involving contact are shown in detail in Figure 24, where the contact tractions are computed according to (106). Specifically, it can be observed that point contact becomes active at the vertex node in step 5. Until step 7, point contact of the vertex node yields the highest stress concentrations during the entire simulation. In step 8, the active contact set increases with the adjacent edge nodes becoming active. Consequently, the contact tractions decrease due to the larger overall contact zone. Solution step 9 is not visualized, but the vertex node and the two edge nodes remain active. In load step 10, the vertex node eventually becomes inactive and the corresponding point Lagrange multiplier takes on a zero value. Nevertheless, the contact tractions still keep their maximum at the vertex node due to the two active edge nodes and their modified shape functions (see Figure 10). The absolute value of the tractions further decreases since the stresses are continuously shifted from the vertex node to the edge nodes. While only being of qualitative nature, this result nevertheless nicely demonstrates the ability of the proposed algorithm to robustly change between point contact formulation and line contact formulation without any heuristic transition parameter.

Until solution step 20, the number of active edge nodes increases and the active sectors separate from the vertex node (see Figure 25). Then, in step 21, the first active surface nodes occur, and the active surface area completely connects the two active edge sectors in step 25 (see again Figure 25). Interestingly, however, the two edge nodes connected to the vertex node on edge $A$ are still in contact and the highest contact tractions still occur at this region. During the following steps, the region of active surface nodes continuously moves toward edge $B$, and the maximum surface stress values increase.

**FIGURE 25** Contact tractions for the bending plate example
Moreover, the number of active edge nodes reduces, and the contact tractions at the edge nodes and at the vertex node further decrease. This further underlines the robust and consistent transition between point, line, and surface contact.

### 7.6 Plate on plate

With the following example, it is demonstrated that the developed contact formulation allows for capturing complex frictional effects in the finite deformation realm. For this purpose, two elastic plates are considered (see Figures 26 and 27). The material model for both plates is of neo-Hookean type with Poisson’s ratio of \( \nu = 0.3 \). Young’s modulus for the lower plate is chosen as \( E = 5 \cdot 10^5 \) and Young’s modulus for the upper plate is \( E = 7 \cdot 10^5 \). A quasi-static simulation is performed, and thus, inertia effects are not considered. The dimensions of the plates and their orientation can be extracted from Figures 26 and 27. During the entire simulation, the lower plate is completely fixed at its left face side \( A \), and the upper plate is subjected to a prescribed displacement \( \mathbf{d}_{\text{dbc}} = [d_x, d_y, d_z] \) at its right face side \( B \). In the time interval \( 0 \leq t < 6 \), the displacement component \( d_z \) is linearly decreased from 0 to \(-0.75\), and all other displacement components of \( \mathbf{d}_{\text{dbc}} \) remain at zero. Afterward, in the time interval \( 6 \leq t < 18 \), the displacement component \( d_x \) is linearly decreased from 0 to \(-0.6\), the component \( d_z \) is kept constant and \( d_y = 0 \). Thus, the overall simulation time is \( T = 18.0 \). The time step size is defined as \( \Delta t = 0.05 \), and consequently, 360 time steps have to be computed. Spatial discretization is realized with trilinear hexahedral elements (hex8) with EAS element technology. All in all, 4620 elements are employed. With regard to contact interaction, the upper plate is defined as the slave body and the lower plate acts as master body. Frictional effects are described by a coefficient of friction \( \mu = 0.4 \). Due to the tilted upper (slave) plate, the resulting contact interaction is of edge-to-surface type, which is governed by our line contact algorithm. In addition, two vertices of the slave body are also in contact with the master plate, which represent an additional point contact scenario also enforced with Lagrange multipliers. Shortly after the considered simulation time, the contact scenario would change to surface contact.

The resulting normal contact stresses at time \( t = 6.0 \) are visualized in Figure 28. It can be seen that the stress level is comparably low in the middle of the contact line and drastically increases toward the two vertices. These effects near the vertices in flat-to-flat contact interactions are well-known as contact singularities and have already been investigated in several publications, see, for example, the works of Comninou\(^{63}\) and Ciavarella et al.\(^{64}\) Our novel all entity contact algorithm allows for robustly representing the expected stress state. In addition, the relatively sharp transitions between the low stress level in the middle of the contact line and the contact singularities at the vertices are accurately captured.
by the algorithm. The tangential traction due to frictional effects at the end of the simulation, i.e., $t = 18.0$, are shown in Figure 29. Therein, a top view on the two plates can be seen. As expected, the corresponding frictional stresses show the same profile as the normal contact stresses in Figure 28. Finally, the $L^2$-norms of the resulting normal force vector and tangential force vector are plotted for the entire simulation in Figure 30. Both forces linearly increase in the time interval $0 \leq t < 6$, which corresponds to the lowering of the upper plate through the prescribed displacement component $d_z$. Subsequently, frictional sliding is initiated by the given displacement component $d_x$, which leads to nearly constant forces in the time interval $6 \leq t < 18$. Slight oscillations occur due to the nonsmooth geometry approximation inherent to first-order (hex8) elements.

All in all, the results of this example clearly demonstrate that the new all entity contact algorithm is also applicable to frictional contact scenarios.
7.7 Falling coin

The final example demonstrates the conservation properties of the Lagrange multiplier contact algorithms developed in this contribution in the context of transient dynamics. The example consists of an elastic coin (flat cylinder) and an elastic foundation (see Figures 31 and 32). The employed material model for both bodies is of St. Venant-Kirchhoff type. The material properties of the coin are defined with Young’s modulus being \( E = 1 \cdot 10^5 \), Poisson’s ratio being \( \nu = 0.0 \), and the density being \( \rho_0 = 0.3 \). The properties of the foundation are the same except for Young’s modulus, which is defined as \( E = 4 \cdot 10^3 \). The dimensions of the bodies are shown in Figure 31. During the entire simulation, the coin is subjected to a constant (gravitational) body force \( \hat{b} = -700 \) in negative Z-direction. The edges of the lower surface of the foundation are completely fixed during the simulation. The employed finite element discretization is also shown in Figures 31 and 32. All in all, 14 248 trilinear hexahedral elements (hex8) with EAS element technology are employed. For the simulation, inertia effects are considered and implicit time integration is done with a generalized-\( \alpha \) scheme, see the work of Chung and Hulbert. The so-called spectral radius is chosen as \( \rho_\infty = 0.95 \), which introduces slight numerical dissipation. The overall simulation time is \( T = 0.055 \), and the time step is defined to be \( \Delta t = 5 \cdot 10^{-4} \). The contact scenario is defined with the coin being the slave body and the foundation being the master body, respectively. Contact without any frictional effects is assumed, and line and surface Lagrange multipliers are introduced at the edges and the surfaces of the coin.

The resulting deformation of the coin and the foundation is visualized for characteristic time steps in Figure 33. Herein, the deformation corresponding to the first impact is shown on the top left part. This impact is resolved entirely by the line Lagrange multipliers. It introduces a rotation of the coin, which then leads to the next contact situation being dominated by the surface Lagrange multipliers (see the top right part in Figure 33). The bottom left part in Figure 33 illustrates nicely the elastic wave traveling through the foundation after the first impact. The final deformation at the end of the simulation is shown in the bottom right corner in Figure 33.

The mentioned activation and deactivation of the line- and surface-based Lagrange multipliers is additionally shown in Figure 34. Therein, active edge nodes correspond to discrete line Lagrange multipliers. It can be seen that at most points in time contact interaction is actually dominated by the line Lagrange multipliers, whereas only very few situations, such as the one illustrated in the top right and bottom left corners in Figure 33, are characterized by surface contact. Thus, it can be stated that the overall robustness of the simulation is caused by the newly developed segment-based integration scheme for the line contact from Section 5.2.

Again, investigations concerning conservation properties are based on the work of Puso and Laursen. Thus, conservation of linear momentum is achieved, when the sum of contact forces that act on the slave and master side vanishes. In
analogy, conservation of angular momentum is achieved, when the sum of interface moments due to contact forces that act on the slave and master side vanishes. For this investigation, the interface forces and moments of slave and master sides are visualized in Figure 35. It can be seen that the absolute values of force and moment of the slave and master side behave identically at first sight and correspond to the impact situations characterized by the number of active nodes in Figure 34. However, an in-depth investigation of the conservation properties requires a closer look at the relative error of the sums with respect to the slave quantity. The corresponding results are plotted in Figure 36. Conservation of linear momentum is guaranteed for the developed contact algorithms, since the slave and master forces balance perfectly. This conservation is achieved up to machine precision. Conservation of angular momentum is not guaranteed, since the interface moments do not balance. However, the obtained error is very small (max. 0.025%) and thus can be considered negligible from an engineering point of view. Note that conservation of energy cannot be guaranteed with the presented algorithm, since adequate time integration schemes that resolve the discontinuities of the interface velocities in the event of an impact are required for this purpose, see the works of Laursen and Chawla\textsuperscript{66} and Laursen and Love.\textsuperscript{67} These time integration schemes are not employed and are not in the focus of this contribution.
FIGURE 35  Falling coin example: absolute values of interface forces (left) and moments (right) for investigation of the conservation of linear and angular momentum. Master quantities are defined to be negative for better visualization [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 36  Relative errors for balance of interface forces (left) and moments (right) for the falling coin example

8 | CONCLUSION

In this contribution, a combined framework for frictional point contact, line contact, and surface contact based on dual mortar finite element methods has been developed. Three sets of Lagrange multipliers have been introduced for accurately and consistently representing point, line, and surface contact; and all required numerical procedures for projection and integration have been described. To switch between these three Lagrange multiplier sets, no heuristic transition parameters are required. Quite in contrast, the transition between point, line, and surface contact arises implicitly and automatically from the underlying variationally consistent problem formulation. High numerical efficiency was assured by performing a static condensation procedure to eliminate the additional Lagrange multiplier degrees of freedom at negligible computational cost. In order to achieve the desired robustness in all possible contact scenarios, a penalty regularization had to be utilized for the special and rather rare contact scenario of nonparallel edges. Our algorithms have been demonstrated to be applicable to very general 3D problems, while discretization has been restricted to first-order Lagrangian finite elements.

Future work will concentrate on the substitution of the remaining penalty regularization by a Lagrange multiplier approach with proper solution space. Additionally, an extension to second-order Lagrangian finite elements is of high practical interest and, thus, already a topic of current research.

ACKNOWLEDGEMENTS

This work was supported by the German Federal Ministry of Economic Affairs and Energy through project 20T1306 in cooperation with Rolls Royce Deutschland Ltd & Co KG under contract T004.017.000. The authors gratefully acknowledge this support.
ORCID

A. Popp http://orcid.org/0000-0002-8820-466X

REFERENCES


**How to cite this article:** Farah P, Wall WA, Popp A. A mortar finite element approach for point, line, and surface contact. *Int J Numer Methods Eng*. 2018;114:255–291. [https://doi.org/10.1002/nme.5743](https://doi.org/10.1002/nme.5743)