UCRL-JRNL-200667

A Mortar Segment-to-Segment Frictional Contact Method for Large Deformations

M. Puso, T. Laursen

October 30, 2003

Computer Methods in Applied Mechanics and Engineering

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

A Mortar Segment-to-Segment Frictional Contact Method for Large Deformations [*]

As submitted to the journal

Computer Methods in Applied Mechanics and Engineering

Michael A. Puso

The University of California
Lawrence Livermore National Laboratory
P.O. Box 808
Livermore, CA 94550 USA
(925)422-8198, (925)423-4097 FAX
email: puso@llnl.gov

Tod A. Laursen

Computational Mechanics Laboratory
Department of Civil and Environmental Engineering
Duke University
Durham N.C. 27708 USA
email: laursen@duke.edu

Keywords: finite elements, mortar method, contact, large deformation, friction

[*] Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

A Mortar Segment-to-Segment Frictional Contact Method for Large Deformations

Keywords: finite elements, mortar method, contact, large deformation, friction

ABSTRACT

Contact modeling is still one of the most difficult aspects of nonlinear implicit structural analysis. Most 3D contact algorithms employed today use node-on-segment approaches for contacting dissimilar meshes. Two pass node-on-segment contact approaches have the well known deficiency of locking due to over constraint. Furthermore, node-on-segment approaches suffer when individual nodes slide out of contact at contact surface boundaries or when contacting nodes slide from facet to facet. This causes jumps in the contact forces due to the discrete nature of the constraint enforcement and difficulties in convergence for implicit solution techniques.

In a previous work [1], we developed a segment-to-segment contact approach based on the mortar method that was applicable to large deformation mechanics. The approach proved extremely robust since it eliminated the overconstraint which caused “locking” and provided smooth force variations in large sliding. Here, we extend this previous approach in [1] to treat frictional contact problems. The proposed approach is then applied to several challenging frictional contact problems which demonstrate its effectiveness.

1 INTRODUCTION

The simulation of mechanically contacting surfaces, which interact via unilateral constraints, is still one of the most challenging tasks in computational mechanics. The most prominent algorithm for doing flexible body contact problems where there is large sliding is the node-on-segment contact algorithms developed by Hallquist [2]. The node-on-segment approach enforces contact using collocation such that nodes on the “slave side” must not penetrate their opposing “master side” segments (facets). This can be done in a single pass algorithm such that only the nodes on the slave side are considered and nodes on the master side are free to penetrate the slave side. Because the single pass method does not solve the contact patch test [3], it is typically not applied when flexible bodies are contacting. The two pass method is most often chosen for flexible body contact, since it will solve the patch test in 2D and in certain (but not all) 3D mesh configurations, particularly when low order elements are used. The two pass method precludes penetration of the slave side nodes into the master side segments and master side nodes into the slave side segments. For higher order elements such as quadratics, the two pass algorithm is not as effective since it no longer satisfies patch tests. Although this node-on-segment method has solved many problems, as many analysts would attest, the method is still not very robust. This lack of robustness is particularly evident when implicit solution procedures are used to solve the discretized nonlinear equilibrium equations.

The four main numerical issues that affect the robustness of the two pass node-on-segment approach are listed as follows

1. Locking or over-constraint.
2. Non-smooth contact surfaces cause jumps in contact forces when slave nodes slide between adjacent master segments.
3. The discrete constraints cause jumps when slave nodes slide off the boundaries of the master surface. This occurs even for flat surfaces.
4. Inequality equations necessitate determination of active and inactive constraints (e.g. gaps opening and closing, stick-slip, etc.)

The proposed segment-to-segment approach deals with the first three issues listed above. The fourth issue is still open and many computational approaches in mathematical programming have been put forth for handling inequality constraints within the context of contact. Here, an augmented Lagrange approach was applied for handling the inequality constraints in the solution of the contact problems demonstrated in Section 5. It is seen from these numerical examples that the proposed method appears to be quite successful in dealing with the first three issues.

Issues one through three, as they pertain to two pass node-on-segment, can be discussed in more detail to establish the need for the methods we propose. First, in general, the two pass node-on-segment approaches are prone to “lock”. It is straightforward to show that the method precludes a given line of nodes in contact from curving (bending) as shown in Figure 3 when all constraints are active due to sufficiently high pressures. The two pass method fails the

Babuska-Brezzi condition [4] and ill conditioning and poor convergence behavior are typical manifestations of the over-constraint.

Next, in the typical node-on-segment approach, the measure of penetration is given by the slave node gap which is computed by the closest point projection of the slave node onto the segmented master surface. The normal contact force is then chosen to be coaxial with the master segment normal computed at the closest point. The frictional contact force is in the tangent plane associated with the contact normal. Consequently, the non-smooth surface of the low order elements causes jumps in the contact force on the slave node as it transitions from one master segment to the adjacent segment (Figure 1). These jumps in forces can cause spurious chatter as two contact surfaces slide relative to each other and convergence difficulties are often encountered. Contact surface smoothing methods for 2D [5, 6] and for 3D [7, 8, 9] smoothly interpolate the opposing surfaces eliminating this non-smooth behavior, but are considerably more complicated than the classical node-on-segment approach.

Finally, when slave nodes reach the end of a defined master surface and slide off, a jump in the contact force typically occurs. This jump is non-physical and often results in convergence difficulties. Contact surface smoothing does nothing to preclude this pathology.

Node-on-segment approaches use collocation to integrate the contact virtual work such that no explicit representation of contact pressure is involved, just contact forces. By contrast, methods that integrate the contact virtual work using numerical quadrature employ some interpolated contact pressure. Here, as in [10], these methods will be denoted as segment-to-segment approaches since they effectively project a segment on one side of the contact surface onto an adjacent segment of opposing side. A number of different approaches have been employed and several of the more notable are mentioned. For the most part, the methods differ in the way they interpolate the contact pressure and how they formulate the projection. The first instances of segment-to-segment approaches were seen in [11], [3] and [12] and were implemented as penalty methods. These were presented for two dimensional cases and were not always stable in their Lagrange multiplier implementations. The mortar methods originally proposed in [13] for mesh tying, will not lock since they are LBB stable [4]. Several versions of the mortar method have been adapted to handle the contact problem [14, 15, 16] but so far all these methods have only considered two-dimensional linear kinematics, which significantly simplifies the development of the contact formulation. In [1], the authors extend the previous segment-to-segment, mortar contact methods to treat large deformation kinematics for curved 3D surfaces.

In this work, a numerically objective slip definition within the context of segment-to-segment contact is proposed. With this in hand, the method in [1] is extended to treat frictional contact problems.

The following is an outline for this paper. Section 2 introduces the notation used throughout this manuscript and presents the discrete forms of the virtual work for the segment-to-segment approach. Section 3 describes the discrete form of the contact constraints. In particular, the new slip rate definition is presented. Section 4 presents details involving the contact normal and frictional force calculation and demonstrates linear momentum conservation. In Section 5, numerical examples are presented which demonstrate the robustness of the proposed scheme. Section 6 discusses the novel aspects and costs of the proposed approach. Appendix A details the integrations scheme and Appendices B-D provide additional details involving linearization of the proposed formulation.

2 Contact Mechanics for Segment-to-Segment Approach

The precise discrete forms for the frictional contact virtual work within the proposed segment-to-segment context are given in this section. The normal contact aspects are identical to the work in [1], whereas the definition for the slip velocity and frictional traction are novel to this work. For conciseness, only two contacting bodies will be considered throughout the presentation.

2.1 Contact Virtual Work

The motion of two bodies with reference placement Ω^i $(i = 1, 2)$ is characterized by the deformations $\boldsymbol{\varphi}^i : \Omega^i[0, T] \to \mathbb{R}^3$. Material particles of each solid are identified with the reference coordinate $\boldsymbol{X}^i \in \Omega^i$, and we let $\boldsymbol{x}^i = \boldsymbol{\varphi}^i(\boldsymbol{X}^i, t)$ be the current placement of these material particles $\boldsymbol{X}^i$ at time $t \in [0, T]$. The portion of the boundary in contact is given by γ_c where $\gamma_c = \gamma^1 \cap \gamma^2$ and $\gamma^i = \boldsymbol{\varphi}^i(\partial\Omega^i)$. In the finite element discrete form, the spatial configuration of the surface mesh can be parameterized via the appropriate node numbering in the following way

$$\boldsymbol{x}^i = \sum_{A=1}^{n^i} N_A^i(\boldsymbol{\xi}^i)\boldsymbol{x}_A^i \quad \text{where} \quad \boldsymbol{x}^i, \boldsymbol{x}_A^i \in \gamma^i \tag{1}$$

where n^i are the number of nodes defined on the surface mesh of $\boldsymbol{\gamma}^i$ and the shape functions $N_A^i : \boldsymbol{\xi}^i \in \square \to \mathbb{R}$ considered here are bilinear for trilinear hexahedral meshes and/or triangular for tetrahedral meshes. It will be convenient to define a vector of contact surface nodal coordinates $\boldsymbol{\Phi}$ as follows

$$\boldsymbol{\Phi} = \left\{ \begin{array}{c} \boldsymbol{x}_1^1 \\ \vdots \\ \boldsymbol{x}_{n^1}^1 \\ \boldsymbol{x}_1^2 \\ \vdots \\ \boldsymbol{x}_{n^2}^2 \end{array} \right\} \tag{2}$$

The contact virtual work for two deformable bodies constrained at a common interface γ is written

$$\int_\gamma \boldsymbol{t} \cdot \left[\delta\boldsymbol{x}^1(\tilde{\boldsymbol{\xi}}^1) - \delta\boldsymbol{x}^2(\tilde{\boldsymbol{\xi}}^2) \right] d\gamma = 0 \tag{3}$$

$$\text{where} \quad \gamma = \gamma(\boldsymbol{\xi}, \boldsymbol{\Phi}), \quad \tilde{\boldsymbol{\xi}}^i : \gamma \to \{\boldsymbol{\xi}^i \,|\, \boldsymbol{\xi}^i \in \square\},$$

and $\boldsymbol{t}$ is the contact traction to be defined later. In the continuum version of (3), seperate surface parameterizations $\boldsymbol{\xi}^i, (i = 1, 2)$ are mapped to a common interface parametrization γ pointwise in a natural way. In the discrete implementation of (3), the surface $\gamma = \gamma(\boldsymbol{\xi}, \boldsymbol{\Phi})$

is a specifically defined surface for integration formed from the nodal coordinates $\boldsymbol{x}_A^i$ and parameterized, piecewise, by $\boldsymbol{\xi} \in \square$. The mapping $\tilde{\boldsymbol{\xi}}^i : \gamma \to \square$ in (3) is used to map points on the "intermediate" surface γ back to the parent domains $\boldsymbol{\xi}^i$ defined by (1). The definition of γ is not unique. In the 2D segment-to-segment formulations [11, 3, 12], γ is computed by some averaging of the meshes on γ^1 and γ^2 whereas in [16], a separate mesh is supplied by the analyst to define γ for small displacement contact. The typical mortar method description specifies that the non-mortar (slave) surface is specified as γ^1 such that $\gamma \equiv \gamma^1$ and γ^2 is the mortar (master) surface. In that context, the variable of integration is $\boldsymbol{x}^1$ and $\boldsymbol{\xi} = \boldsymbol{\xi}^1$. The extension to 3D curved surfaces is not so simple. In the proposed implementation, we instead assume that $\gamma \approx \gamma^1$, such that γ is composed of piecewise flat segments based on the non-mortar segments. The exact definition for γ and the mapping $\tilde{\boldsymbol{\xi}}^i$ is given in Appendix A.

The contact traction can be seperated into normal $\boldsymbol{t}^N$ and frictional $\boldsymbol{t}^F$ parts where $\boldsymbol{\nu}$ is some surface normal such that

$$\begin{gathered} \boldsymbol{t} = \boldsymbol{t}^N + \boldsymbol{t}^F \\ \boldsymbol{t}^N = p\,\boldsymbol{\nu} \qquad \boldsymbol{t}^F \cdot \boldsymbol{\nu} = 0 \end{gathered} \tag{4}$$

The typical mortar method interpolates tractions on the non-mortar side $(i = 1)$ using the equivalent displacement field interpolation space. In this way, normal and frictional tractions based on this mortar scheme can then be defined

$$\boldsymbol{t}^N = \sum_{A=1}^{n^1} N_A^1(\boldsymbol{\xi}^1) p_A \boldsymbol{\nu}_A \qquad \boldsymbol{t}^F = \sum_{A=1}^{n^1} N_A^1(\boldsymbol{\xi}^1) \boldsymbol{t}_A \tag{5}$$

where p_A is a nodal defined pressure, $\boldsymbol{\nu}_A$ is a nodally averaged normal (c.f. Figure 1 and Appendix B) and $\boldsymbol{t}_A$ is the nodally defined frictional force where $\boldsymbol{t}_A \cdot \boldsymbol{\nu}_A = 0$. Although a nodally averaged normal $\boldsymbol{\nu}_A$ isn't necessary in the implementation, it was found to provide smoother sliding behavior compared to other alternatives. To put it simply, the key to the mortar interpolation is that it accommodates a stable solution by enforcing an appropriate number of constraints n^1. By contrast, the two pass node-on-segment approach has $n^1 + n^2$ constraints and is over-constrained.

Although the procedure is not followed in this work, mortar methods will often modify (5) by employing constant pressure interpolation on the outer segments of the contact surface definition in the event that additional constraints (other contact surface definitions, mesh tying, boundary conditions etc.) are applied to these "cross point" nodes. This procedure avoids potential over-constraint that would be evidenced by too many Lagrange multipliers existing at the surface intersections. At any rate, substituting (5) in the virtual work statement (3) yields the discrete contact virtual work statement in terms of normal contact pressure and frictional traction

$$\sum_{A=1}^{n^1} p_A \delta \hat{g}_A + \boldsymbol{t}_A \cdot \delta \boldsymbol{s}_A = 0 \tag{6}$$

where

$$\delta \hat{g}_A = \boldsymbol{\nu}_A \cdot \sum_{B,C}^{n^1,n^2} (n^1_{AB} \delta \boldsymbol{x}^1_B - n^2_{AC} \delta \boldsymbol{x}^2_C), \tag{7}$$

$$\delta \boldsymbol{s}_A = (\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A) \sum_{B,C}^{n^1,n^2} (n^1_{AB} \delta \boldsymbol{x}^1_B - n^2_{AC} \delta \boldsymbol{x}^2_C) \tag{8}$$

and

$$n^i_{AB} = \int_\gamma N^1_A(\tilde{\boldsymbol{\xi}}^1) N^i_B(\tilde{\boldsymbol{\xi}}^i)\, d\gamma \tag{9}$$

Note that a notational abuse was made in (7) and (8) such that the $n^1_{AB} \delta \boldsymbol{x}^1_B$ term is not in fact summed over $C = 1, n^2$. Likewise, the last terms in (7) and (8) are not summed over $B = 1, n^1$. This notation is used for convenience and will be used throughout in similar contexts. The details involved with the integration of (9) are included in Appendix A.

3 Contact Constraints

As presented in [1], the discrete normal contact gap g_A defined

$$g_A = \boldsymbol{\nu}_A \cdot \boldsymbol{g}_A \tag{10}$$

$$\boldsymbol{g}_A = \sum_{B,C}^{n^1,n^2} (n^1_{AB}\boldsymbol{x}^1_B - n^2_{AC}\boldsymbol{x}^2_C) \tag{11}$$

and pressure p_A are the basis for the discrete form of the Kuhn-Tucker conditions defined on the non-mortar nodes A

$$g_A \leq 0, \quad p_A \geq 0, \quad p_A g_A = 0 \quad \forall\, A = 1, n^1 \tag{12}$$

As was discussed in [1], $\delta\hat{g}_A \neq \delta g_A$, where $\delta\hat{g}_A$ is given by (7) and δg_A is the directional derivative of (10). This choice simplifies the implementation, but since it means that the virtual work expression is not a variation of a quadratic, the ramifications of this choice include a loss of symmetry and a lack of angular momentum conservation [*].

It is clear that an incremental slip definition akin to (8)

$$\Delta\boldsymbol{s}_A = (\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A) \sum_{B,C}^{n^1,n^2} (n^1_{AB}\Delta\boldsymbol{x}^1_B - n^2_{AC}\Delta\boldsymbol{x}^2_C) \tag{13}$$

where $\Delta\boldsymbol{x}^i_A = \boldsymbol{x}^i_A(t_{n+1}) - \boldsymbol{x}^i_A(t_n)$ would not be invariant to rigid body rotations. Such invariance is quantified in the current context by requiring that

$$\Delta\boldsymbol{s}_A = 0 \quad \forall \quad \boldsymbol{\varphi}(\boldsymbol{X}, t_{n+1}) - \boldsymbol{\varphi}(\boldsymbol{X}, t_n) \in \text{SO(3)} \tag{14}$$

The consequence of this lack of invariance would be that spinning bodies would not slip and stick properly if (13) were used to quantify slip. To remedy this, an alternative slip increment is derived based on the time derivative of the gap "vector" $\boldsymbol{g}_A$ defined in (11)

$$\dot{\boldsymbol{g}}_A = \sum_{B,C}^{n^1,n^2} [n^1_{AB}\dot{\boldsymbol{x}}^1_B - n^2_{AC}\dot{\boldsymbol{x}}^2_C + \dot{n}^1_{AB}\boldsymbol{x}^1_B - \dot{n}^2_{AC}\boldsymbol{x}^2_C] \tag{15}$$

[*] Angular momentum conservation could easily be recovered by adding the appropriate $\delta\boldsymbol{\nu}_A$ term in (7) that would appear in the directional derivative of (10).

which can be written in an incremental form

$$\boldsymbol{g}_A(t_{n+1}) - \boldsymbol{g}_A(t_n) \approx \sum_{B,C}^{n^1,n^2} [n^1_{AB}\Delta\boldsymbol{x}^1_B - n^2_{AC}\Delta\boldsymbol{x}^2_C + \\ (n^1_{AB}(t_{n+1}) - n^1_{AB}(t_n))\,\boldsymbol{x}^1_B(t_n) - (n^2_{AC}(t_{n+1}) - n^2_{AC}(t_n))\,\boldsymbol{x}^2_C(t_n)] \tag{16}$$

In the event of flat surfaces in persistent contact, $\boldsymbol{g}_A(t_{n+1}) = \boldsymbol{g}_A(t_n) = 0$ [†], and (16) can be rewritten

$$\sum_{B,C}^{n^1,n^2} n^1_{AB}\Delta\boldsymbol{x}^1_B - n^2_{AC}\Delta\boldsymbol{x}^2_C = -[(n^1_{AB}(t_{n+1}) - n^1_{AB}(t_n))\,\boldsymbol{x}^1_B(t_n) - \\ (n^2_{AC}(t_{n+1}) - n^2_{AC}(t_n))\,\boldsymbol{x}^2_C(t_n)] \tag{17}$$

The left hand side of (17) is similar to the non-invariant slip increment given in (13) whereas the right hand side of (17) is invariant to rigid body rotations since $n^i_{AB}(t_{n+1}) - n^i_{AB}(t_n) = 0$ during such rotations. Conseqently, the right hand side is adopted for the slip increment $\Delta\boldsymbol{s}_A$ now defined

$$\Delta\boldsymbol{s}_A = (\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A) \sum_{B,C}^{n^1,n^2} [(n^1_{AB}(t_{n+1}) - n^1_{AB}(t_n))\,\boldsymbol{x}^1_B(t_n) - \\ (n^2_{AC}(t_{n+1}) - n^2_{AC}(t_n))\,\boldsymbol{x}^2_C(t_n)] \tag{18}$$

Using (18), the discrete Coloumb frictional Kuhn Tucker constraints are then given

$$\phi_A = \mu\, p_A - ||\boldsymbol{t}_A|| \geq 0, \quad \Delta\boldsymbol{s}_A = \zeta_A\,\boldsymbol{t}_A, \quad \zeta_A \geq 0, \quad \phi_A\,\zeta_A = 0 \quad \forall\, A = 1, n^1 \tag{19}$$

where μ is the Coulomb coefficient of friction.

4 Contact Forces

This section presents details involving contact force calculation and momentum conservation.

[†] $\boldsymbol{g}_A = 0$ is enforced for flat surfaces but not necessarily curved surfaces

4.1 Modified Form of Gap Calculation

The form of (18) requires the storage (or recomputation) of the weights $n^i_{AB}(t_n)$ to be used in calculations at time t_{n+1}. To avoid this, (18) can be rewritten

$$\Delta \boldsymbol{s}_A = (\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A) \left[\sum_{B,C}^{n^1,n^2} [n^1_{AB}(t_{n+1})\, \boldsymbol{x}^1_B(t_n) - n^2_{AC}(t_{n+1})\, \boldsymbol{x}^2_C(t_n)] - \boldsymbol{g}_A(t_n) \right] \tag{20}$$

where now only the time t_n nodal gap vector, $\boldsymbol{g}_A(t_n)$ need be stored.

4.2 Penalty Regularization and Augmented Lagrange Algorithm

The penalty regularized form of contact [17] within an augmented Lagrangian scheme was employed in this work. The algorithm is as follows:

Given the slip increment $\Delta \boldsymbol{s}_A$ and gap g_A from t_n to t_{n+1} and the augmented Lagrange k^{th} iterate for frictional traction $\boldsymbol{t}^k_A$ and pressure p^k_A:

(1) Update normal pressure using a suitable penalty parameter κ to get $k+1$ pressure

$$p^{k+1}_A = p^k_A + \kappa g_A \tag{21}$$

(2) Compute trial frictional traction

$$\boldsymbol{t}^{tr}_A = (\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A) \boldsymbol{t}^k_A + \kappa \Delta \boldsymbol{s}_A \tag{22}$$

(3) Determine contact status and update the $k+1$ iterate of time t_{n+1} traction

$$\text{stick} \quad \mu p^{k+1}_A - ||\boldsymbol{t}^{tr}|| \geq 0 \quad \text{then} \quad \boldsymbol{t}^{k+1}_A = \boldsymbol{t}^{tr}_A \tag{23}$$

$$\text{slip} \quad \mu p^{k+1}_A - ||\boldsymbol{t}^{tr}|| < 0 \quad \text{then} \quad \boldsymbol{t}^{k+1}_A = \frac{\mu p^{k+1}_A}{||\boldsymbol{t}^{tr}_A||} \boldsymbol{t}^{tr}_A \tag{24}$$

It is preferable to rotate the historical traction $\boldsymbol{t}^k$ into the current plane as opposed to projecting it when implementing the augmented Lagrangian algorithm above. Even if this rotation is not done, however, the augmented Lagrange process should still converge to perfect stick when appropriate. Furthermore, a Lagrange multiplier implementation would require that the slip vector be projected onto two axes common to the $\boldsymbol{\nu}_A$ plane so that two independent frictional constraint equations are enforced (as opposed to three equations with one redundancy).

4.3 Contact Forces

The contact forces $\boldsymbol{F}$ result directly from the virtual work equation (8) [‡]

$$\boldsymbol{\Phi} \cdot \boldsymbol{F} = \sum_{A,B,C}^{n^1,n^1,n^2} (p_A \boldsymbol{\nu}_A + \boldsymbol{t}_A)(n^1_{AB} \delta \boldsymbol{x}^1_B - n^2_{AC} \delta \boldsymbol{x}^2_C) \tag{25}$$

as well as equations (21,23,24) which describe the pressure and frictional traction. In view of (25), the contact force on node B on side i due to pressure and traction at node A can be written

$$\boldsymbol{f}^i_{AB} = (-1)^{i-1}(p_A \boldsymbol{\nu}_A + \boldsymbol{t}_A) n^i_{AB} \tag{26}$$

Details involving the linearization of (26) are included in Appendix C.

4.4 Linear Momentum Conservation

Satisfaction of linear momentum is demonstrated by substituting $\delta \boldsymbol{x}^i_A = \boldsymbol{a}$ into the virtual work expression (25) and using (26)

$$\begin{aligned} \delta\boldsymbol{\Phi} \cdot \boldsymbol{F} &= \sum_{A=1}^{n^1} (p_A \boldsymbol{\nu}_A + \boldsymbol{t}_A) \cdot \boldsymbol{a} \sum_{B,C}^{n^1,n^2} (n^1_{AB} - n^2_{AC}) \\ &= -\boldsymbol{a} \cdot (\boldsymbol{F}^1 - \boldsymbol{F}^2) \qquad \text{where} \qquad \boldsymbol{F}^i = \sum_{A,B}^{n^1,n^i} \boldsymbol{f}^i_{AB} \\ &= 0 \end{aligned} \tag{27}$$

Since $\boldsymbol{a}$ is arbitrary, the expression can only be true if the interface forces are equivalent ($\boldsymbol{F}^1 = \boldsymbol{F}^2$). Exact linear momentum conservation requires that the contact weight functions exactly

[‡] Because $\boldsymbol{t}_A$ is in the tangent plane associated with $\boldsymbol{\nu}_A$ by construction, the projection operator in (8) in not necessary in (25).

satisfy the consistency condition

$$\sum_{B,C}^{n^1,n^2}(n^1_{AB}-n^2_{AC})=0 \quad \forall A=1,n^1 \tag{28}$$

Of course (28) is satisfied when the integration in (9) is exact. On the other hand, with trilinear hexahedrals exact integration is not easily achieved since the bilinear shape functions of the segments need to be integrated over arbitrarily shaped domains (Figure 2). For interfaces that are comprised exclusively of linear tetraheda, exact integration is more readily achieved but lower integration rules may be desired for efficiency. The appropriate numerical integration technique guaranteeing (28) is presented in Appendix A.

5 Examples

The example problems were chosen to demonstrate the robustness and accuracy of the proposed approach. In particular, the proposed method effectively addresses issues $1-3$ in the Introduction. The first example, in particular, demonstrates the locking behavior (over-constraint) of the two pass node-on-segment contact method and highlights the effectiveness of the proposed segment-to-segment mortar approach for this problem. Example 2 shows that in large sliding, non-smooth contact surfaces are smoothed out by the proposed method. Example 2 also demonstrates how the discrete constraints of node-on-segment cause problems when nodes slide off the boundaries of surfaces. Single pass contact is a valid approach for flexible-to-rigid (or nearly rigid) contact problems. Example 3 is truly a flexible-to-flexible problem not applicable to single pass contact and again demonstrates superior robustness compared to the non-smooth and smooth node-on-segment approaches.

In the examples, the segment-to-segment approach is compared to the standard node-on-segment approach [2] and the smoothed node-on-segment approach [8]. The augmented Lagrange technique was used in all cases for enforcing the contact constraints.

All results are shown for both cases of non-mortar side specification. The legends in all plots denote the designated non-mortar side in parentheses. Clearly, it is seen that the non-mortar side designation will cause some bias. Neverthess, it appears the bias is not serious when the two sides of the mesh are sufficiently refined.

The examples are all large deformation kinematics and use the neo-Hookean, hyperelastic material ([18], Equation 9.2.3). Here, we specify the Young's modulus E and Poisson's ratio ν in the example problems in lieu of the shear modulus μ and bulk modulus K used in [18].

5.1 Example 1: Beam Bending

This example demonstrates the locking free behavior of the proposed method. Two 10×1 beams ($E = 1, \nu = 0$) (Figure 3) are sandwiched together by an ambient pressure $p = 0.1$. The beam is then bent by applying a moment to the end of the beam. Figures 3(b,c) show the results where a sliding surface is placed between two dissimilarly meshed beams. Figure 3(b) shows the mortar method results where the upper surface was designated as the non-mortar side and contact stick is maintained throughout the deformation. The two pass node-on-segment demonstrates locking in Figure 3(c) since the applied pressure is sufficiently high to force all contact constraints to be active along the interface. Consequently, the interface cannot bend and the beams lock. Figure 4 shows displacement versus moment for the beams analyses. The conforming results compare well with the beams simulated using the proposed method. The node-on-segment analysis diverged after the moment reached 0.02.

5.2 Example 2: Ironing

In this problem, depicted in Figure 5, a cylindrical die $(E = 1000, \nu = 0.499)$ is pressed into the slab $(E = 1, \nu = 0.3)$ and then slid over the surface. The die, with a center initially located 2.5 units from the left end of the slab, travels -1.4 units in the vertical z direction from time $0s - 0.2s$ and then 4 units in the horizontal x direction over the remaining $1.3s$. The slab dimensions are $9 \times 4 \times 3$ and the die is 0.2 thick, 5.2 wide with a radius of 3.

The classical two pass node-on-segment contact could not achieve convergence to any degree on this problem due to locking. The two pass smoothed node-on-segment experienced fewer locking problems. Single pass node-on-segment contact could be applied using tight gap tolerances with the slave side chosen on the flexible slab.

Due to the high Poisson's ratio, $\nu = 0.499$ the slab expands transversely such that it slides off the outside edges of the indenter after sufficient slip is developed. This effect is readily seen in Figures 5 at times 0.6 and 0.8. At the point that nodes begin sliding off the boundary causing jumps in the contact forces, convergence could not be achieved for any of the node-on-segment approaches. The contact forces are shown in Figure 6, where it is seen that the smooth node-on-facet method failed at time 5.5. Because the segment-to-segment approach provides smooth behavior at contact surface boundaries, it can successfully handle the transition of the slab nodes at the overhang. Furthermore, the mortar results in Figure 6 demonstrate little bias as to choice of non-mortar side.

In the event that the Poisson's ratio is lowered to 0.3, the slab doesn't expand as much in the transverse direction, so that no overhang is developed. In this case the single pass node-on-segment can successfully complete the analysis, whereas the two pass algorithm cannot. It is to be noted, however, that a somewhat loose convergence tolerance was necessary for the non-smooth contact analysis since jumps in the contact normal cause a significant amount of chatter as seen in Figure 7. The proposed segment-to-segment approach effectively smooths out the contact forces and compares well with the single pass node-on-segment as seen in Figure 8.

Finally, a 1×1 block was slid across the slab as seen in Figure 8 This problem is particularly difficult since the outside corners experience nearly singular stresses. In this analysis, convergence could only be achieved for the case where the non-mortar side was applied to the block since the slab mesh was too coarse. Convergence could not be achieved using any form of smooth or non-smooth node-on-segment methods to any significant extent. The resulting force for the block simulation is shown in Figure 9 and does show some oscillations. Nevertheless, these oscillations are sufficiently smooth so as not to cause convergence problems.

5.3 Example 3: Spheres in Sphere

In this example, a solid sphere $(R = 0.6, E = 1.0, \nu = 0.3)$ is placed within a hollow sphere $(R_o = 2.0, R_i = 0.7, E = 1.0, \nu = 0)$ and a uniform body force is applied to the solid sphere such that it is pressed into the outer hollow sphere (Figure 10). Figure 10 shows the resulting deformation from simulation where the non-mortar surface was on the coarsely meshed (inner) sphere. Because both spheres are equally flexible, single pass node-on-segment is not an option. Furthermore, the classical two pass node-on-segment approach fails to make it past the first time step unless some form of “trick” such as segment freezing [1] is used. This difficulty arises because the node-on-node contact at the center causes extreme oscillations in contact force. Even with segment freezing, the analysis failed early at time 0.7. The smooth two pass node-on-segment method failed at time 8.3, presumably due to locking. With the non-mortar side on the fine (outer) sphere, the analysis failed at time 22.3 due to extensive deformation of the elements in the outer sphere. The analysis with the non-mortar side on the coarse (inner) sphere fared somewhat better, failing at time 30.3 as seen in Figure 10. Figure 11 shows the displacement of the center of the sphere versus time for the different approaches.

6 Conclusions

In this work, a 3D large deformation implementation of a segment-to-segment frictional contact approach was developed and demonstrated. In particular, a numerically objective slip rate was introduced to incorporate friction into the previous work [1]. The mortar approach was adopted for the Lagrange multiplier interpolation such that a designated mortar and non-mortar surface was used in the implementation. The numerical integration technique used to evaluate the contact surface integrals and the process to linearize the resulting contact forces was presented. The necessity to choose a mortar and non-mortar side is restrictive but eliminates the locking encountered by the two pass node-on-segment approach. Several example problems demonstrated the superior robustness of the proposed approach over classical node-on-segment and smooth node-on-segment approaches. This robustness is attributed to the follow facts:

1. The method does not lock.
2. Contact forces vary smoothly during large sliding.
3. Contact forces do not jump as nodes slide off the contact boundary.

Although the proposed approach is significantly more expensive computationally than the non-smoothed node-on-segment approach, the added robustness and accuracy (i.e. no locking) should justify the expense. In fact, most of the cost involves the linearization, which can likely be approximated in many cases to reduce such costs. For example, neglecting the derivatives of $\boldsymbol{\nu}_A$ and n^i_{AB} in Appendices B and C makes the cost comparable to the node-on-segment approach while successfully solving most problems to practical error tolerances.

7 Appendices

7.1 Appendix A: Numerical Integration

This section focuses on the numerical integration technique used for computing the contact weights in (9).

The proposed scheme is valid for meshes formed by linear tetrahedrals and/or trilinear hexahedrals. Some modification would be required for higher order elements. To integrate (9), the required overlap of shape function A defined over non-mortar segment k and shape function B defined over segment l needs to be identified (Figure 2). This is most easily performed by approximating segment k by a flat segment $\tilde{k}$ using the segment normal at the center of k. This segment represents the surface of integration γ as defined in (3). For each non-mortar segment k, the local node set $\mathcal{S}^1(k)$ defines the segment surface of k via the interpolation $\boldsymbol{x}^1_h = \sum_{B\in\mathcal{S}^1(k)} N^1_B(\boldsymbol{\xi}^1)\boldsymbol{x}^1_B$. Interpolations for N_B are linear and bilinear for tetrahedrals and hexahedral meshes respectively. For mortar segment l, the local node set $\mathcal{S}^2(l)$ defines the segment surface via $\boldsymbol{x}_h^2 = \sum_{B \in \mathcal{S}^2(l)} N_B^2(\boldsymbol{\xi}^2)\boldsymbol{x}_B^2$. The following details the proposed numerical integration algorithm (refer to Figure 12)

1. Loop over non-mortar side elements $k = 1$,n_s^1 (# of segments on non-mortar side).
   (a) Form plane $\boldsymbol{p}$ from the normal $\boldsymbol{n}$ and point $\boldsymbol{x}_0$ at the geometric center of element k.
   (b) Compute points $\tilde{\boldsymbol{x}}_A^1$ by projecting facet k nodes onto plane $\boldsymbol{p}$ to form facet $\tilde{k}$ (see also Appendix D). Facet $\tilde{k}$ admits the parameterization

$$\tilde{\boldsymbol{x}}^1 = \sum_{A \in \mathcal{S}^1(k)} N_A^1(\boldsymbol{\xi}^1)\tilde{\boldsymbol{x}}_A^1 \tag{29}$$

   (c) Loop over mortar side elements $l = 1, n_s^2$ (# of segments on mortar side).
      i. Perform rough screen to determine if facet l is close to facet k.
      ii. If l "far" from k, go to top of mortar element loop and increment l.
      iii. Compute points $\tilde{\boldsymbol{x}}_A^2$ by projecting facet l nodes onto plane $\boldsymbol{p}$ to form facet $\tilde{l}$. Facet $\tilde{l}$ admits the parameterization

$$\tilde{\boldsymbol{x}}^2 = \sum_{A \in \mathcal{S}^2(l)} N_A^2(\boldsymbol{\xi}^2)\tilde{\boldsymbol{x}}_A^2 \tag{30}$$

      iv. Use a clipping algorithm [19] to form polygon formed by $\tilde{k} \cap \tilde{l}$ (Figure 12(b),(c)).
      v. Locate geometric center of polygon and divide polygon into n_p triangular pallets (Figure 12(d)). Each triangle p is parameterized by its vertices $\boldsymbol{x}_{I,p}$ $(I = 1, 3)$ and triangular shape functions N_I within the parent domain $\boldsymbol{\xi} = [\xi_1, \xi_2]^T$ as

$$\boldsymbol{x}_p = \sum_{I=1}^{3} N_I(\boldsymbol{\xi})\boldsymbol{x}_{I,p} \tag{31}$$

      where $N_1 = \xi_1, N_2 = \xi_2, N_3 = 1 - \xi_1 - \xi_2$. The segment in Equation (31) represents the parameterized form for $\gamma = \gamma(\boldsymbol{\xi}, \boldsymbol{\Phi})$ in (3).
      vi. Use Gauss-Radau rules [20] to locate ng Gauss points $\boldsymbol{\xi}_g$ and integration weights w_g for each triangular pallet.
      vii. To compute $\boldsymbol{\xi}_g^1$ and $\boldsymbol{\xi}_g^2$ from $\boldsymbol{x}_p(\boldsymbol{\xi}_g) \in \gamma$, the inverse mapping in (3$_3$) is defined by equating (29,30) with (31) [§]

$$\boldsymbol{x}_p(\boldsymbol{\xi}_g) = \tilde{\boldsymbol{x}}^i(\boldsymbol{\xi}_g^i) \tag{32}$$

[§] As mentioned in the following discussion, $\tilde{k}$ and $\tilde{l}$ are assumed convex such that the inverse exists

viii. Compute contribution $n^{1,p}_{AB(k,l)}$ and $n^{2,p}_{AC(k,l)}$ over pallet p for $i = 1, 2$ for all $A, B \in \mathcal{S}^1(k)$ and $C \in \mathcal{S}^2(l)$

$$n^{1,p}_{AB(k,l)} = A^p \sum_{g=1}^{ng} w_g N^1_A(\boldsymbol{\xi}^1_g) N^1_B(\boldsymbol{\xi}^1_g) \tag{33}$$

$$n^{2,p}_{AC(k,l)} = A^p \sum_{g=1}^{ng} w_g N^1_A(\boldsymbol{\xi}^1_g) N^2_C(\boldsymbol{\xi}^2_g) \tag{34}$$

where A^p is the pallet area calculated from (31).

ix. Assemble contribution $n^{i,p}_{AB(k,l)}$ into coefficient n^i_{AB} by summing contributions over all pallets for all $A, B \in \mathcal{S}^1(k)$ and $C \in \mathcal{S}^2(l)$

$$n^1_{AB} = n^1_{AB} + \sum_{p=1}^{n_p} n^{1,p}_{AB(k,l)}, \quad n^2_{AC} = n^2_{AC} + \sum_{p=1}^{n_p} n^{2,p}_{AC(k,l)} \tag{35}$$

(d) End loop over mortar side elements.

2. End loop over non-mortar side elements.

The above algorithm relies on the fact that the segments must be convex. If k is convex and not overly warped, $\tilde{k}$ will be convex. If the normals for non-mortar and mortar segments k and l are relatively close, the projected mortar segment $\tilde{l}$ will be convex. Consequently, the polygon $\tilde{k} \cap \tilde{l}$ will be convex and the intersection polygon (Figure 12(c)(d)) will have a geometric center located in the polygon. This facilitates the triangularization of the polygon for numerical integration and the inverse mappings in step (c) above.

For interfaces that are exclusively made up of linear tetrahedra, the integrals in (33,34) are quadratic in $\boldsymbol{x}$ such that at least three point triangular integration [20] is necessary over each pallet to exactly integrate (33,34). To solve the patch test exactly, only linear accuracy is necessary in (33,34) [¶] such that only one point triangular integration is required.

Where there are trilinear hexahedra on the interface, the inverse mappings $\tilde{\boldsymbol{\xi}}^i : \boldsymbol{x}_p \rightarrow \square$ are rational polynomial functions in general such that exact integration (33,34) is not achieved with

[¶] Patch test satisfaction only requires that $\int_\gamma N^1_A \, d\gamma$ be integrated exactly.

a standard triangular Gauss integration scheme. If the element segments/facets on sides 1 and 2 are flat and not warped (i.e. they are parallelograms), then the inverse mapping is linear and the arguments in (33,34) are quartic in $\boldsymbol{\xi}$ requiring at least six or seven point integration [20]. To solve the patch test exactly with parallelogram segments, quadratic accuracy is necessary such that at least three point integration is required. A scheme that exactly passes the patch test when the segments are warped yet planar (non-parallelogram) and satisfies exact linear momentum conservation has been devised but is more complicated. Here, seven point integration was used for most example problems and appears to be more than sufficient when compared to higher order nine and thirteen point schemes. Nevertheless, the choice of integration rule is somewhat problem dependent.

7.1.1 Linear Momentum Conservation

The proposed algorithm will *exactly* conserve linear momentum through satisfaction of (28). This is achieved by requiring equivalent numerical integration of the polygons and pallets formed by the intersecting mortar and non-mortar sides. Referring to (33) and (34) and applying the identity $\sum_{B\in\mathcal{S}^i(m)} N^i_B = 1$ yields

$$\sum_{B\in\mathcal{S}^1(k),C\in\mathcal{S}^2(l)} [n^{1,p}_{AB(k,l)} - n^{2,p}_{AC(k,l)}] = A^p \sum_{g=1}^{ng} w_g N^1_A(\boldsymbol{\xi}^1_g) - A^p \sum_{g=1}^{ng} w_g N^1_A(\boldsymbol{\xi}^1_g) = 0 \tag{36}$$

Since this is true for arbitrary A and p, (28) is satisfied for the entire interface. If the non-mortar side is integrated independently of the mortar side, *exact* linear momentum will not necessarily follow in general.

7.2 Appendix B: Average Nodal Normal Calculation

In this section, the method used for computing and linearizing the vertex normal is described. Consider a vertex A on the non-mortar surface with n_A segments attached (Figure 1). The vertex normal is computed by averaging normals from surrounding segments using the formula

$$\boldsymbol{\nu}_A = 1/||\textstyle\sum_{i=j}^{n_A}\boldsymbol{n}_j||\displaystyle\sum_{j=1}^{n_A}\boldsymbol{n}_j \tag{37}$$

where $\boldsymbol{n}_j$ is the normal computed at the corner of each facet j from edge vectors $\mathbf{v}_j$ common to node A, i.e.

$$\boldsymbol{n}_j = \mathbf{v}_j \times \mathbf{v}_{j+1}/||\mathbf{v}_j \times \mathbf{v}_{j+1}|| \tag{38}$$

Edge vectors are computed from the current nodal positions, for example $\mathbf{v}_j = \boldsymbol{x}_B^1 - \boldsymbol{x}_A^1$ referring to Figure 1. Introducing the unnormalized vector $\boldsymbol{n}'_j$ defined:

$$\boldsymbol{n}'_j = \mathbf{v}_j \times \mathbf{v}_{j+1}, \tag{39}$$

the following formula for linearization of (37) results from the chain rule

$$\begin{gathered}\Delta\boldsymbol{\nu}_A = 1/||\textstyle\sum_{i=1}^{n_A}\boldsymbol{n}_i||[\boldsymbol{I} - \boldsymbol{\nu}_A \otimes \boldsymbol{\nu}_A]\displaystyle\sum_{j=1}^{n_A} 1/||\boldsymbol{n}'_j||[\boldsymbol{I} - \boldsymbol{n}_j \otimes \boldsymbol{n}_j]\Delta\boldsymbol{n}'_j \\ \Delta\boldsymbol{n}'_j = [\Delta\mathbf{v}_j \times \mathbf{v}_{j+1} + \mathbf{v}_j \times \Delta\mathbf{v}_{j+1}]\end{gathered} \tag{40}$$

By exploiting the identities

$$\boldsymbol{\omega} \times \boldsymbol{x} = \hat{\mathbf{W}}\boldsymbol{x} \quad \text{where} \quad \hat{\mathbf{W}} = \begin{bmatrix} 0 & -\omega_3 & -\omega_2 \\ \omega_3 & 0 & -\omega_1 \\ \omega_2 & \omega_1 & 0 \end{bmatrix} \tag{41}$$

and $\Delta\mathbf{v}_j = \Delta\boldsymbol{x}_B^1 - \Delta\boldsymbol{x}_A^1$ along with some algebra, Equation (40) can be rewritten in matrix form

$$\Delta\boldsymbol{\nu}_A = \hat{\boldsymbol{W}}_A\Delta\boldsymbol{\Phi}_A \tag{42}$$

where the matrix $\hat{\boldsymbol{W}}_A \in \mathbb{R}_{3\times3*n_A}$ is formulated in terms of edges $\mathbf{v}_j$ attached to the vertex A and $\boldsymbol{\Phi}_A \in \mathbb{R}_{3*n_A}$ is the vector of the coordinates $\boldsymbol{x}^1$ of the nodes forming the edges common to vertex A.

7.3 Appendix C: Linearization of Contact Forces

This section presents an overview of the linearization of the contact force (26). The linearization of pressure p_A involves linearization of (21), (10) and (11). The linearization of the $\boldsymbol{t}_A$ involves the linearization of the slip (20) and traction formulas (22,24). Of course this involves chain rule differentiation that will ultimately result in the derivatives of two key parts: $\Delta\boldsymbol{\nu}_A$ which was treated in Appendix B and Δn^i_{AB} which will be treated here. The remaining portion of the linearization is straightforward and will be omitted.

The linearization of n^i_{AB} is expedited by exploiting the discrete partitions of $n^{i,p}_{AB(k,l)}$ over pallets (35) such that

$$\Delta n^i_{AB} = \sum_{k,l}^{n_s^1, n_s^2} \sum_{p=1}^{n_p} \Delta n^{i,p}_{AB(k,l)} \tag{43}$$

where (k,l) are the overlapping non-mortar and mortar segments as described in the Appendix A. To be brief, the (k,l) subscripts are dropped in what follows with the understanding that each pallet contribution $n^{i,p}_{AB}$ must ultimately be assembled via (35). Using $\boldsymbol{\xi}$ as the variable of integration, the linearization of $n^{i,p}_{AB}$ proceeds

$$\begin{aligned}\Delta n^{i,p}_{AB} &= \Delta \int_{\boldsymbol{\xi}^p \in \square} N^1_A(\tilde{\boldsymbol{\xi}}^1) N^i_B(\tilde{\boldsymbol{\xi}}^i) A^p \, d\boldsymbol{\xi} \\ &= \sum_{g=1}^{ng} [\Delta N^1_A(\boldsymbol{\xi}^1_g) N^i_B(\boldsymbol{\xi}^i_g) + N^1_A(\boldsymbol{\xi}^1_g) \Delta N^i_B(\boldsymbol{\xi}^i_g)] w_g A^p + n^{i,p}_{AB} \Delta A^p / A^p \end{aligned} \tag{44}$$

where, again, the $\boldsymbol{\xi}^i_g$ are the local coordinates. To complete (44), the linearizations ΔN^i_A and ΔA^p are required. To accomplish this, the different parameterizations over the pallet are equated as in (32)

$$\sum_{I=1}^{3} N_I(\boldsymbol{\xi}) \boldsymbol{x}_{I,p} = \sum_{A=1}^{4} N^i_A(\boldsymbol{\xi}^i) \tilde{\boldsymbol{x}}^i_A \tag{45}$$

where local node numbering $A = 1, 4$ is used. Taking the directional derivative of (45) gives

$$\sum_{I=1}^{3} N_I(\boldsymbol{\xi}) \Delta \boldsymbol{x}_{I,p} = \tilde{\boldsymbol{\tau}}^i_\alpha \Delta \xi^{i,\alpha} + \sum_{A=1}^{4} N^i_A(\boldsymbol{\xi}^i) \Delta \tilde{\boldsymbol{x}}^i_A \tag{46}$$

where $\tilde{\boldsymbol{\tau}}^i_\alpha$ denotes the covariant basis at $\boldsymbol{\xi}^i$ associated with the flat plane in the $\alpha = 1, 2$ direction, obtained by taking the partial derivative of (29) or (30) in the direction of coordinate $\xi^{i,\alpha}$. In (46) and what follows, the sum over the direction α (or β) is implied. Rearrangement of (46) to solve for the $\Delta\xi^{i,\alpha}$ gives for each surface i

$$\Delta\xi^{i,\alpha} = {}^i\tilde{m}^{\alpha\beta}\tilde{\boldsymbol{\tau}}^i_\beta \cdot [\sum_{I=1}^{3} N_I(\boldsymbol{\xi})\Delta\boldsymbol{x}_{I,p} - \sum_{A=1}^{4} N^i_A(\boldsymbol{\xi}^i)\Delta\tilde{\boldsymbol{x}}^i_A] \tag{47}$$

where ${}^i\tilde{m}^{\alpha,\beta}$ is the (dual) metric associated with the $\boldsymbol{\tau}^i_\alpha$ basis. In other words, we may invert a 2×2 system to determine the directional derivative of $\boldsymbol{\xi}^i$. With this expression in hand, the variation of a shape functions evaluated at $\boldsymbol{\xi}_g$ in (44) can be written as

$$\Delta N^i_C(\boldsymbol{\xi}^i_g) = N^i_{C,\alpha}(\boldsymbol{\xi}^i_g)\,{}^i\tilde{m}^{\alpha\beta}\tilde{\boldsymbol{\tau}}^i_\beta \cdot [\sum_{I=1}^{3} N_I(\boldsymbol{\xi}_g)\Delta\boldsymbol{x}_{I,p} - \sum_{A=1}^{4} N^i_A(\boldsymbol{\xi}^i_g)\Delta\tilde{\boldsymbol{x}}^i_A] \tag{48}$$

This gives the required directional derivative in terms of the variations of the triangle vertices (the $\boldsymbol{x}_{I,p}$) and the projected facet nodes (the $\tilde{\boldsymbol{x}}^i_A$). To assemble the required stiffness, however, one needs to write the directional derivative in terms of the facet nodes directly. Calculation of these directional derivatives is outlined in Appendix D; here these relationships are indicated schematically via

$$\left\{ \begin{array}{c} \Delta\boldsymbol{x}_{1,p} \\ \Delta\boldsymbol{x}_{2,p} \\ \Delta\boldsymbol{x}_{3,p} \end{array} \right\} = \boldsymbol{D}_p\Delta\boldsymbol{\Phi}_{kl}, \quad \left\{ \begin{array}{c} \Delta\tilde{\boldsymbol{x}}^i_1 \\ \Delta\tilde{\boldsymbol{x}}^i_2 \\ \Delta\tilde{\boldsymbol{x}}^i_3 \\ \Delta\tilde{\boldsymbol{x}}^i_4 \end{array} \right\} = \boldsymbol{E}^i\Delta\boldsymbol{\Phi}_{kl} \quad \text{and} \quad \Delta\boldsymbol{\Phi}_{kl} = \left\{ \begin{array}{c} \Delta\boldsymbol{x}^1_1 \\ \vdots \\ \Delta\boldsymbol{x}^1_4 \\ \Delta\boldsymbol{x}^2_1 \\ \vdots \\ \Delta\boldsymbol{x}^2_4 \end{array} \right\} \tag{49}$$

where $\boldsymbol{D}_p$ is a 9×24 matrix, $\boldsymbol{E}^i$ is a 12×24 matrix and $\Delta\boldsymbol{\Phi}_{kl}$ is the 24-vector of locally numbered incremental displacements of the overlapping non-mortar and mortar segments (k, l) that define the polygon containing pallet p. These operators are useful in deriving a matrix-vector version of (48), which will be expressed in terms of 24-vectors $\boldsymbol{P}^{i,p}_C$, $i = 1, 2$ and $C = 1, \ldots, 4$. Examination of (48) and (49) will show that

$$\Delta N^i_C(\boldsymbol{\xi}^i_g) = \boldsymbol{P}^{i,p^T}{}_C(\boldsymbol{\xi}^i_g)\Delta\boldsymbol{\Phi}_{kl} \quad \text{where} \quad \boldsymbol{P}^{i,p}_C(\boldsymbol{\xi}^i_g) = N^i_{C,\alpha}\,{}^i\tilde{m}^{\alpha\beta}(\boldsymbol{T}^{i,p}_\beta - \tilde{\boldsymbol{T}}^i_\beta) \tag{50}$$

and with the 24-vectors $\boldsymbol{T}^{i,p}_\beta$ and $\tilde{\boldsymbol{T}}^i_\beta$ defined

$$\boldsymbol{T}^{i,p}_\beta = \boldsymbol{D}_p^T \left\{ \begin{array}{c} N_1(\boldsymbol{\xi}_g)\tilde{\boldsymbol{\tau}}^i_\beta \\ N_2(\boldsymbol{\xi}_g)\tilde{\boldsymbol{\tau}}^i_\beta \\ N_3(\boldsymbol{\xi}_g)\tilde{\boldsymbol{\tau}}^i_\beta \end{array} \right\} \quad \text{and} \quad \tilde{\boldsymbol{T}}^i_\beta = \boldsymbol{E}^{iT} \left\{ \begin{array}{c} N^i_1(\boldsymbol{\xi}^i_g)\tilde{\boldsymbol{\tau}}^i_\beta \\ N_2(\boldsymbol{\xi}^i_g)\tilde{\boldsymbol{\tau}}^i_\beta \\ N^i_3(\boldsymbol{\xi}^i_g)\tilde{\boldsymbol{\tau}}^i_\beta \\ N_4(\boldsymbol{\xi}^i_g)\tilde{\boldsymbol{\tau}}^i_\beta \end{array} \right\} \tag{51}$$

Also required for the calculation of (44) is the variation of the facet pallet area, ΔA^p. The area of the pallet triangle is given by

$$A^p = 1/2||(\boldsymbol{x}_{3,p} - \boldsymbol{x}_{1,p}) \times (\boldsymbol{x}_{2,p} - \boldsymbol{x}_{1,p})|| \tag{52}$$

which is readily differentiated to obtain

$$\Delta A^p = 1/2\, \boldsymbol{n} \cdot [(\Delta\boldsymbol{x}_{3,p} - \Delta\boldsymbol{x}_{1,p}) \times (\boldsymbol{x}_{2,p} - \boldsymbol{x}_{1,p}) + (\boldsymbol{x}_{3,p} - \boldsymbol{x}_{1,p}) \times (\Delta\boldsymbol{x}_{2,p} - \Delta\boldsymbol{x}_{1,p})] \tag{53}$$

where $\boldsymbol{n}$ is the unit normal to the projection plane $\boldsymbol{p}$ (Figure 12) formed by non-mortar segment k. Using the identity (41) and the definition for $\Delta\boldsymbol{x}_{I,p}$ in (49) along with some algebra defines a 9-vector $\boldsymbol{J}$ which provides a matrix form for $\Delta A^p/A^p$ i.e.

$$\Delta A^p/A^p = \boldsymbol{J}_p^T \boldsymbol{D}_p\, \Delta\boldsymbol{\Phi}_{kl} \tag{54}$$

Equation (44) can now be written in matrix form using (50) and (54)

$$\Delta n^{i,p}_{AB} = \boldsymbol{P}^{i,p}_{AB}\Delta\boldsymbol{\Phi}_{kl} \tag{55}$$

where $\boldsymbol{P}^{i,p}_{AB}$ is the 24-vector defined

$$\boldsymbol{P}^{i,p}_{AB} = \sum_{g=1}^{ng} [N^i_B(\boldsymbol{\xi}^i_g)\boldsymbol{P}^{1,p}_A(\boldsymbol{\xi}^1_g) + N^1_A(\boldsymbol{\xi}^1_g)\boldsymbol{P}^{i,p}_B(\boldsymbol{\xi}^i_g)]w_g A^p + n^{i,p}_{AB}\boldsymbol{D}_p\boldsymbol{J}_p^T \tag{56}$$

7.4 Appendix D: Compute $\boldsymbol{E}^i$ and $\boldsymbol{D}_p$ for Equation (49)

As described in Appendix A, the non-mortar facet k is used to form a plane for determining the overlap of opposing mortar facets l. Exploiting local node numbering for the facets k and

l, the plane is formed from the geometric center $\boldsymbol{x}_0$ (i.e. $\boldsymbol{\xi}^1 = \boldsymbol{0}$) and center normal $\boldsymbol{n}$ of facet k

$$\boldsymbol{x}_0 = \sum_{A=1}^{4} N_A^1(\boldsymbol{0})\boldsymbol{x}_A^1, \qquad \boldsymbol{n} = \boldsymbol{x},_\xi^1(\boldsymbol{0}) \times \boldsymbol{x},_\eta^2(\boldsymbol{0}) / ||\boldsymbol{x},_\xi^1(\boldsymbol{0}) \times \boldsymbol{x},_\eta^2(\boldsymbol{0})|| \tag{57}$$

where the surface tangents are computed

$$\boldsymbol{x},_\xi^1 = \sum_{A=1}^{4} N,_{\xi A}^1(\boldsymbol{0})\,\boldsymbol{x}_A^1 \qquad \boldsymbol{x},_\eta^1 = \sum_{A=1}^{4} N,_{\eta A}^1(\boldsymbol{0})\,\boldsymbol{x}_A^1 \tag{58}$$

Both non-mortar and mortar segment nodes are then projected onto the plane via

$$\tilde{\boldsymbol{x}}_A^i = \boldsymbol{x}_A^i - (\boldsymbol{x}_A^i - \boldsymbol{x}_0) \cdot \boldsymbol{n}\boldsymbol{n} \tag{59}$$

Linearizing (59) is straightforward such that

$$\begin{aligned} \Delta\tilde{\boldsymbol{x}}_A^i &= [\boldsymbol{I} - \boldsymbol{n} \otimes \boldsymbol{n}]\Delta\boldsymbol{x}_A^i + \boldsymbol{n} \otimes \boldsymbol{n}\Delta\boldsymbol{x}_0 - \boldsymbol{n} \otimes (\boldsymbol{x}_A^i - \boldsymbol{x}_0)\Delta\boldsymbol{n} \\ &= \boldsymbol{E}_A^i \Delta\boldsymbol{\Phi}_{kl} \end{aligned} \tag{60}$$

and $\boldsymbol{E}_A^i$ is a 3×24 matrix computed accordingly from the definition of $\Delta\boldsymbol{\Phi}_{kl}$ in (49_3) and linearization of (57) and (58). The 12×24 matrix $\boldsymbol{E}^i$ is then assembled from the matrices $\boldsymbol{E}_A^i$ as

$$\boldsymbol{E}^i = \left\{ \begin{array}{c} \boldsymbol{E}_1^i \\ \vdots \\ \boldsymbol{E}_4^i \end{array} \right\} \tag{61}$$

The definition for the matrix $\boldsymbol{D}_p$ in (49) and $\boldsymbol{x}_c$ can be written according to Figure 13

$$\left\{ \begin{array}{l} \Delta\boldsymbol{x}_p \\ \Delta\boldsymbol{x}_{p+1} \\ \Delta\boldsymbol{x}_c \end{array} \right\} = \boldsymbol{D}_p \Delta\boldsymbol{\Phi}_{kl}, \qquad \boldsymbol{x}_c = \frac{1}{n_p} \sum_{p=1}^{n_p} \boldsymbol{x}_p \tag{62}$$

The computation of $\boldsymbol{x}_p$ and $\boldsymbol{x}_{p+1}$ depends on how the vertex was formed. From Figure 13(b) it is seen that $\boldsymbol{x}_{p+1} = \tilde{\boldsymbol{x}}_{A+1}^1$ such that rows in $\boldsymbol{E}^1$ are used in the linearization

$$\Delta\boldsymbol{x}_{p+1} = \boldsymbol{E}_{A+1}^1 \Delta\boldsymbol{\Phi}_{kl} \tag{63}$$

as prescribed by (60). Thus, to assemble the 9×24 matrix $\boldsymbol{D}_p$ the second three rows will coincide with the 3×24 matrix $\boldsymbol{E}^1_{A+1}$ i.e.

$$\boldsymbol{D}_p^T = [\boldsymbol{d}_1^p, \boldsymbol{d}_2^p, \boldsymbol{d}_3^p] \;\text{ where }\; \boldsymbol{d}_2^p = \boldsymbol{E}^{1}{}^{T}_{A+1} \tag{64}$$

The situation is more complicated for the point $\boldsymbol{x}_p$ in Figure 13 where $\boldsymbol{x}_p = \boldsymbol{x}_{int}$. The Cyrus-Beck parametric line clipping algorithm described in [19] and recast here is used to find $\boldsymbol{x}_{int}$ from Figure 13 such that

$$\boldsymbol{x}_{int} = \tilde{\boldsymbol{x}}_A^1 - \frac{(\tilde{\boldsymbol{x}}_B^2 - \tilde{\boldsymbol{x}}_A^1) \times (\tilde{\boldsymbol{x}}_{B+1}^2 - \tilde{\boldsymbol{x}}_B^2) \cdot \boldsymbol{n}}{(\tilde{\boldsymbol{x}}_{A+1}^1 - \tilde{\boldsymbol{x}}_A^1) \times (\tilde{\boldsymbol{x}}_{B+1}^2 - \tilde{\boldsymbol{x}}_B^2) \cdot \boldsymbol{n}} \cdot (\tilde{\boldsymbol{x}}_{A+1}^1 - \tilde{\boldsymbol{x}}_A^1) \tag{65}$$

where $\boldsymbol{n}$ is the normal for plane $\boldsymbol{p}$ as in Figure 12. Linearizing (65) is laborious, yet straightforward, with the result used for assembling the first three rows in $\boldsymbol{D}_p$ i.e.

$$\Delta\boldsymbol{x}_{int} = \boldsymbol{E}_{int}\Delta\boldsymbol{\Phi}_{kl} \;\text{ such that }\; \boldsymbol{d}_1^p = \boldsymbol{E}_{int}^T \tag{66}$$

and $\boldsymbol{E}_{int}$ schematically represents the linearization of (65). Finally, from $(62)_2$

$$\boldsymbol{d}_3^p = \frac{1}{n_p}\sum_{p=1}^{n_p} \boldsymbol{d}_1^p \tag{67}$$

When the non-mortar and mortar facet edges overlap (Figure 13(c)), the denominator in (65) is zero and $\boldsymbol{x}_{int}$ is undefined. In this case, the pallet vertex is a segment vertex such then $\boldsymbol{x}_p = \boldsymbol{x}_B^2$ and $\boldsymbol{x}_{p+1} = \boldsymbol{x}_{A+1}^1$ as illustrated in Figure 13(c). The linearization is then handled as in (63) i.e. $\Delta\boldsymbol{x}_p = \boldsymbol{E}_B^2\Delta\boldsymbol{\Phi}_{kl}$.

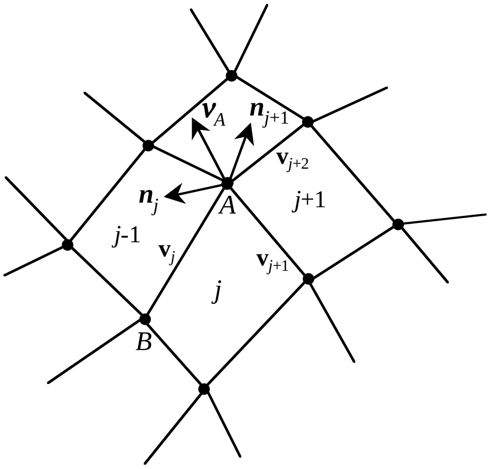


Figure 1: Segments (facets) $j-1, j$ and $j+1$ meet at node A. Contact forces jump since the normal $\boldsymbol{n}_j$ on segment j is not equal to the normal $\boldsymbol{n}_{j+1}$ on segment $j+1$ along common edge. An average vertex normal $\boldsymbol{\nu}_A$ can be computed from $\boldsymbol{n}_j$ and $\boldsymbol{n}_{j+1}$.

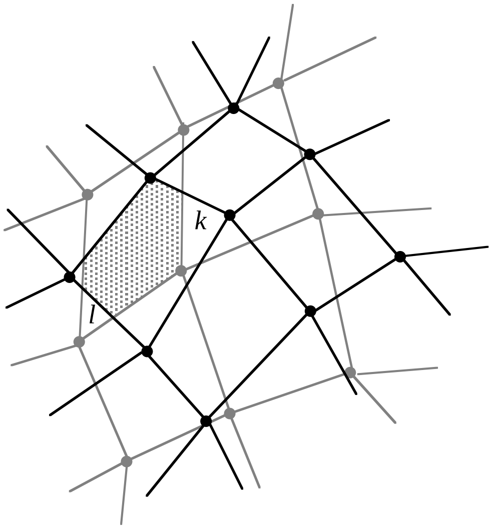
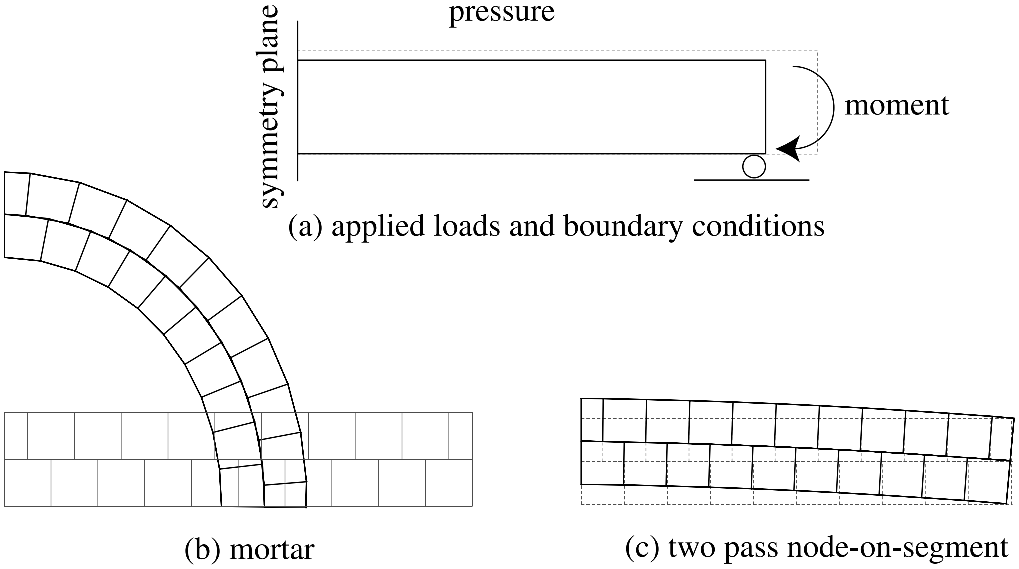


Figure 2: Interfacing meshes such that segment k overlaps l. The domain of overlap shown symbolically as the shaded portion, can have a complicated shape.



(a) applied loads and boundary conditions

(b) mortar

(c) two pass node-on-segment

Figure 3: Sandwiched beams in bending. (a) Beam is subjected to ambient pressure over entire outer surface. Moment is then applied at the end. (b) Results from analysis with the non-mortar surface on the fine (upper) surface. (c) Results illustrating the locking with the dissimilar mesh using node-on-segment contact.

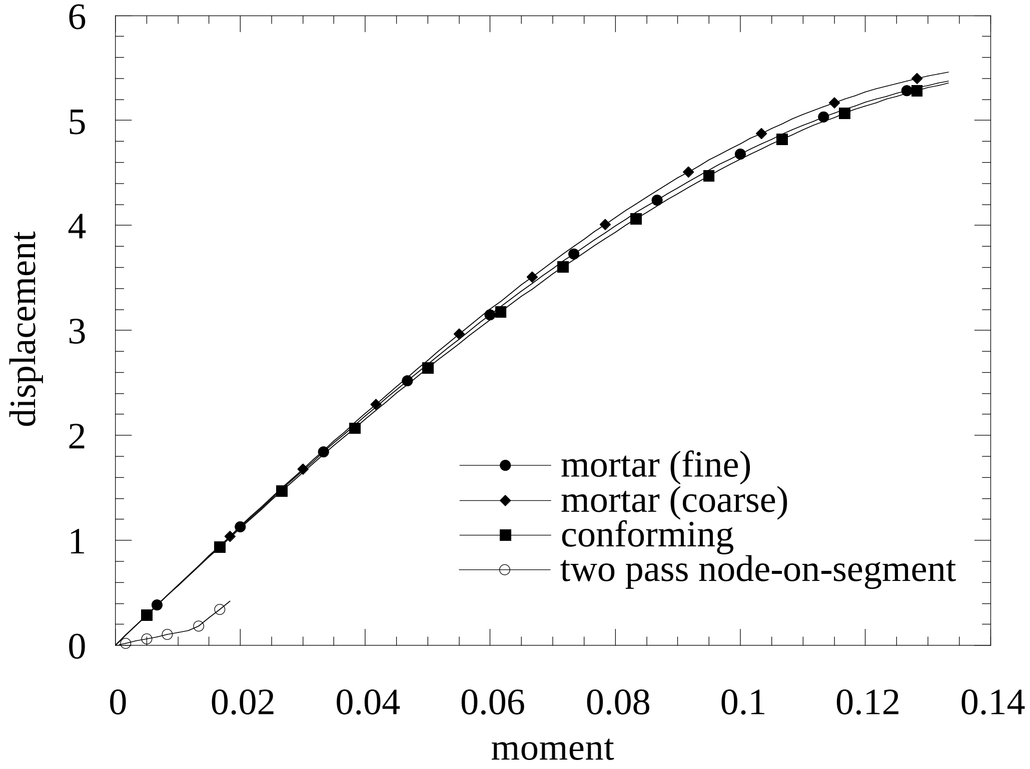
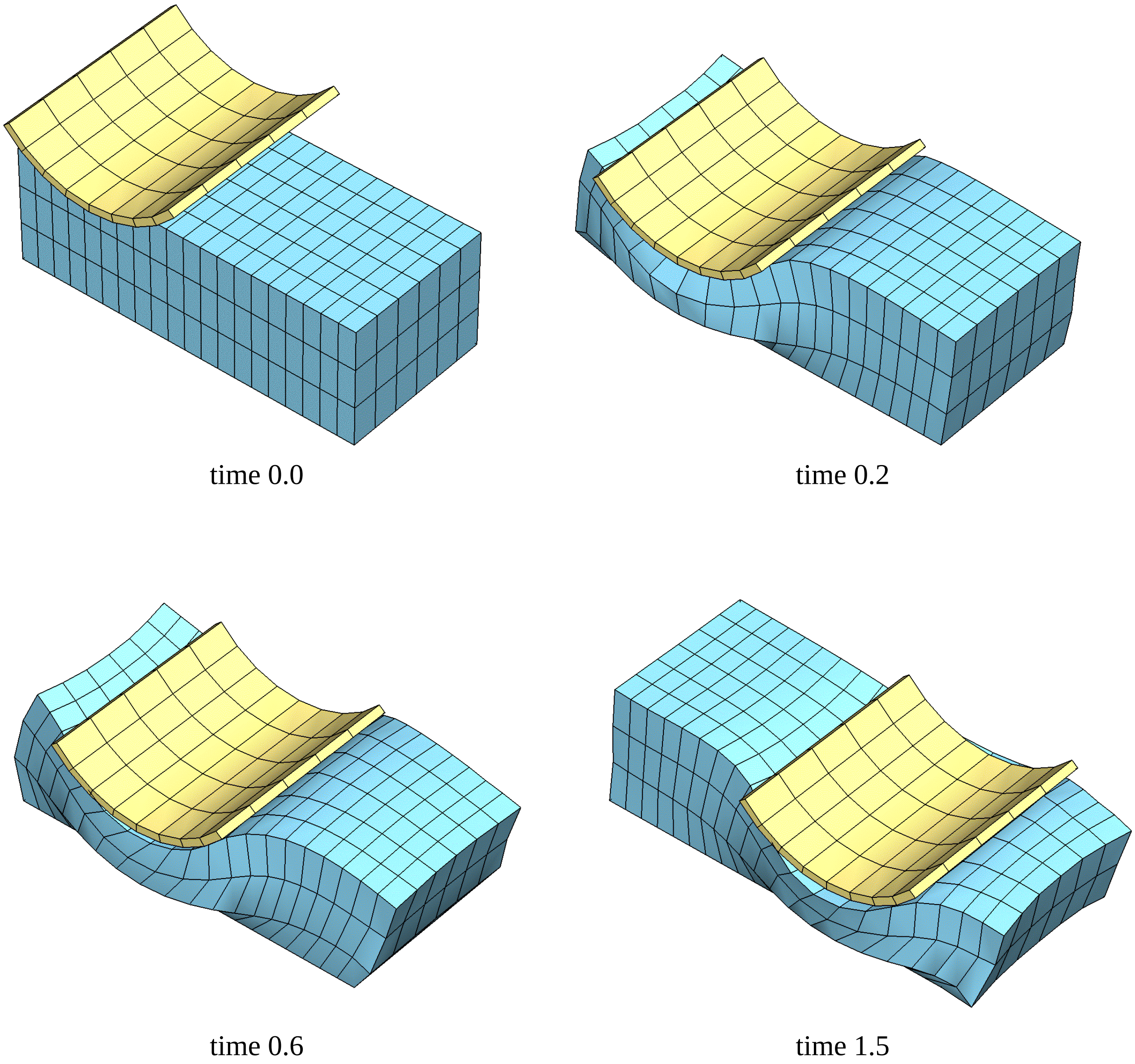


Figure 4: Displacement versus moment from beam bending example. Both mortar results compare well with the conforming mesh result.



time 0.0

time 0.2

time 0.6

time 1.5

Figure 5: Results for four time steps on $\nu = 0.5$ ironing example problem. The die is fully lowered in the z direction at time $t = 0.2$ and then slid in the x direction across the slab over the remaining time. Notice that at $t = 0.2$ that the flexible slab does not overhang the stiff indenter due to stick. As sliding progresses, slip occurs in both horizontal x *and* y directions such that slab edges slide transversely out of the contact area along the y edge.

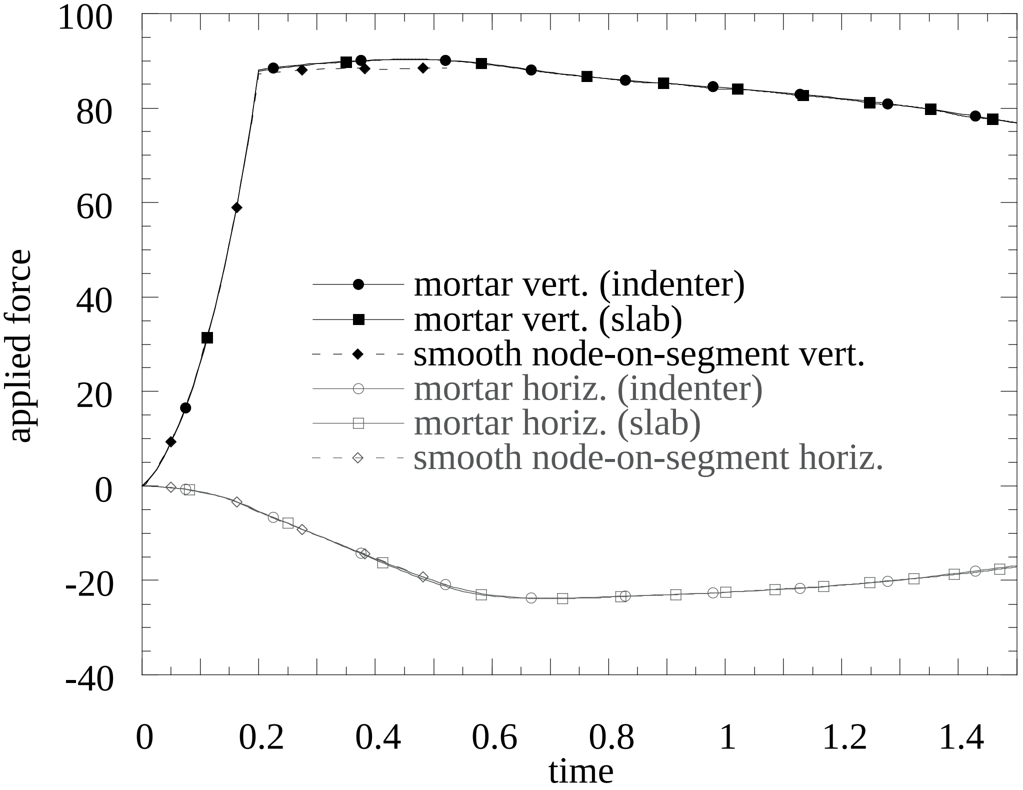


Figure 6: Applied forces versus time from mortar and smooth node-on-segment $\nu = 0.499$ analyses. The smooth node-on-segment analysis diverges at the point where the slab begins to "squirt" outside the y edges of the indenter (c.f. Figure 5).

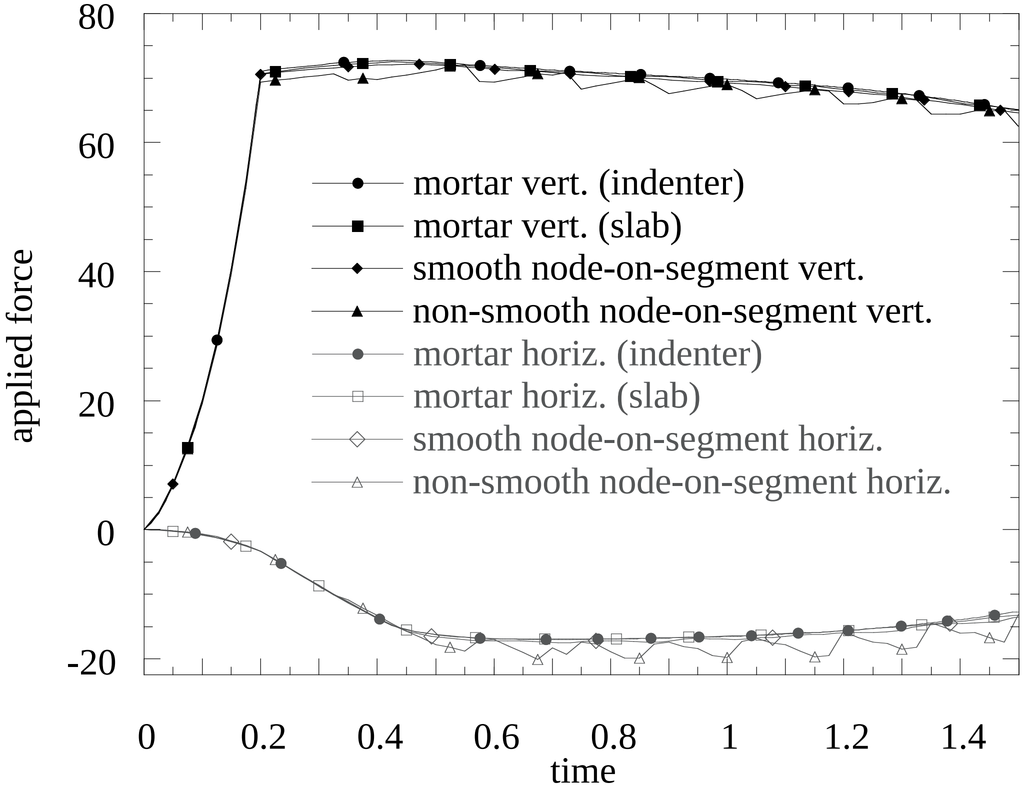
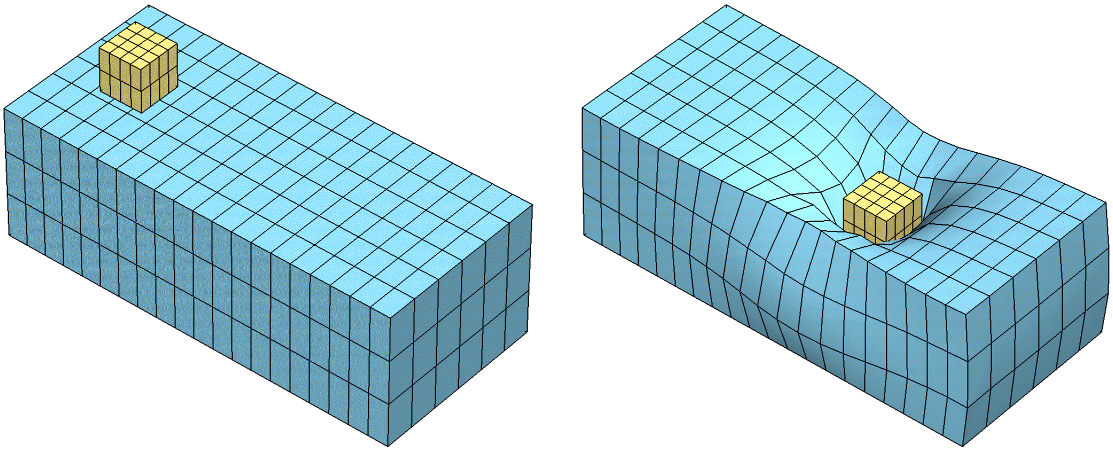


Figure 7: Applied force from mortar and single pass node-on-segment $\nu = 0.3$ analyses. The smooth node-on-segment and mortar approaches compare very well whilst the non-smooth node-on-segment exhibits significant chatter.



time 0.0

time 1.5

Figure 8: Stiff square block is lowered in z direction and dragged across flexible slab.

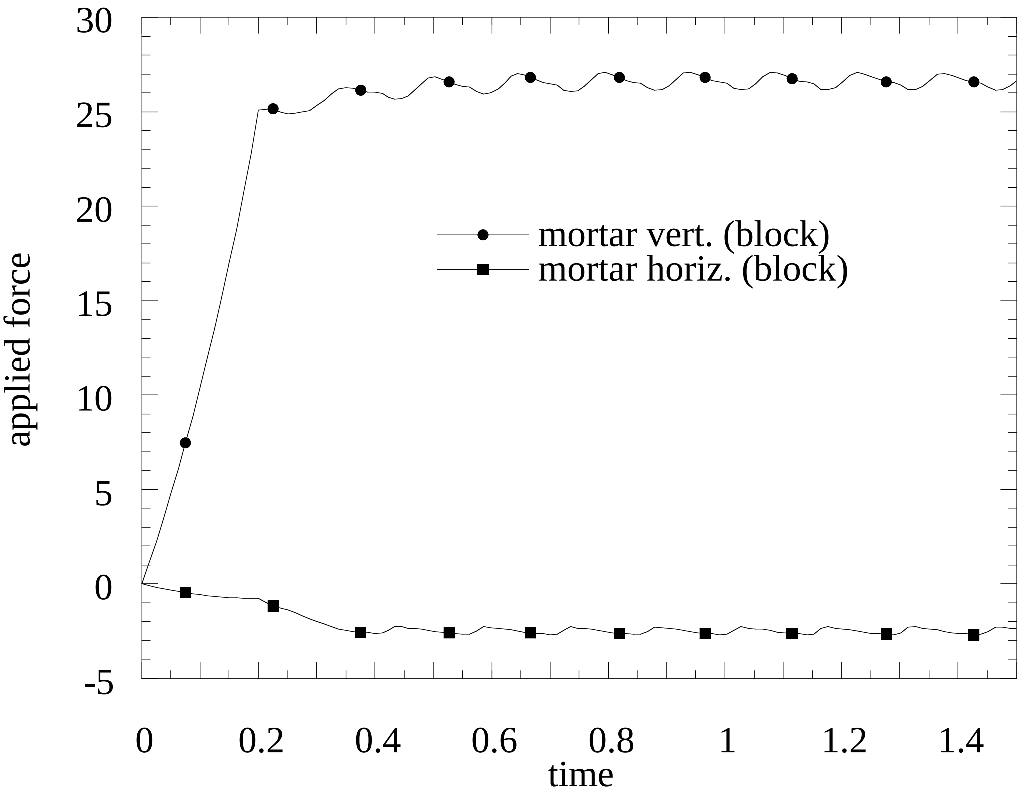
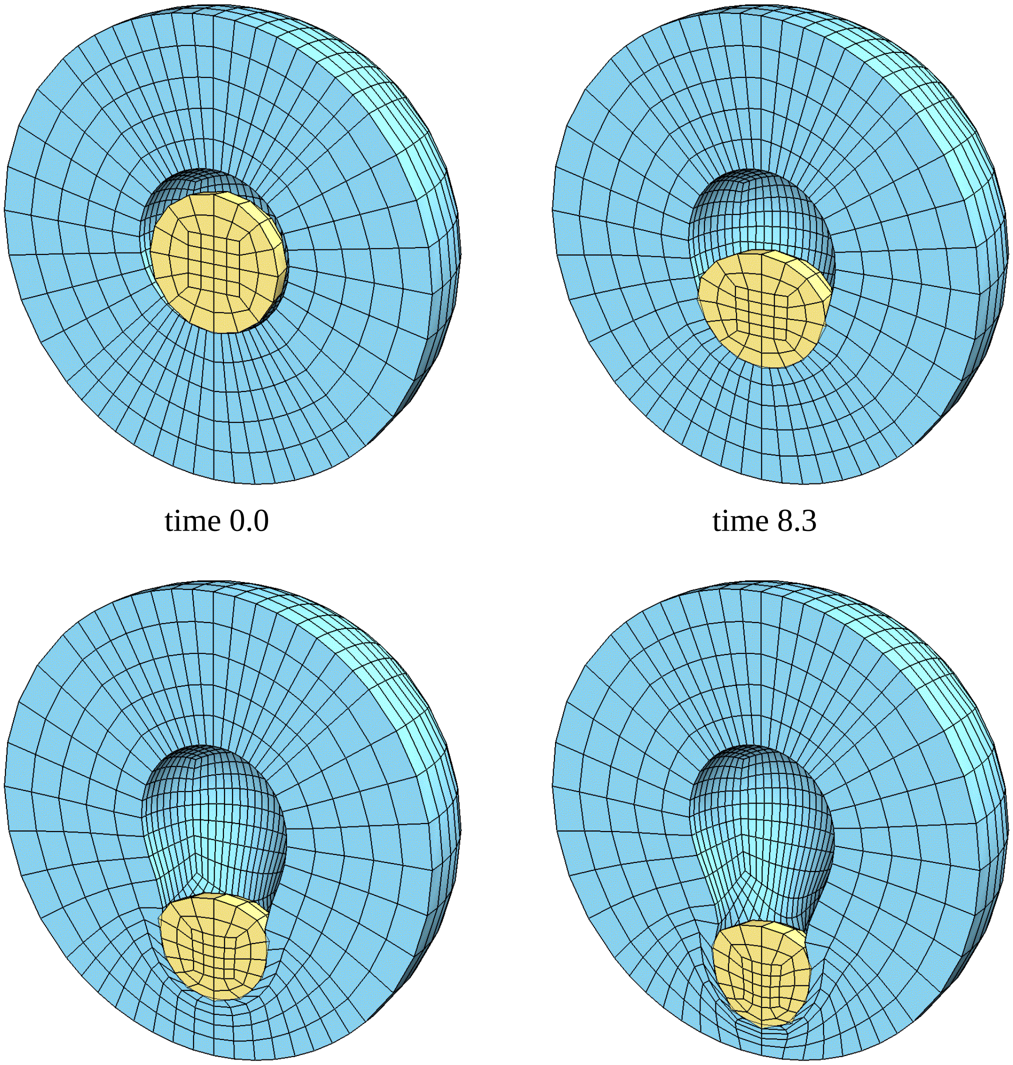


Figure 9: Applied forces resulting from mortar analysis of square block (non-mortar side) and flexible slab (mortar side) shown in 8.



time 0.0

time 8.3

time 22.3

time 30.3

Figure 10: Trajectory of solid sphere pressed against hollow sphere from analysis with coarse (inner) mesh as non-mortar surface.

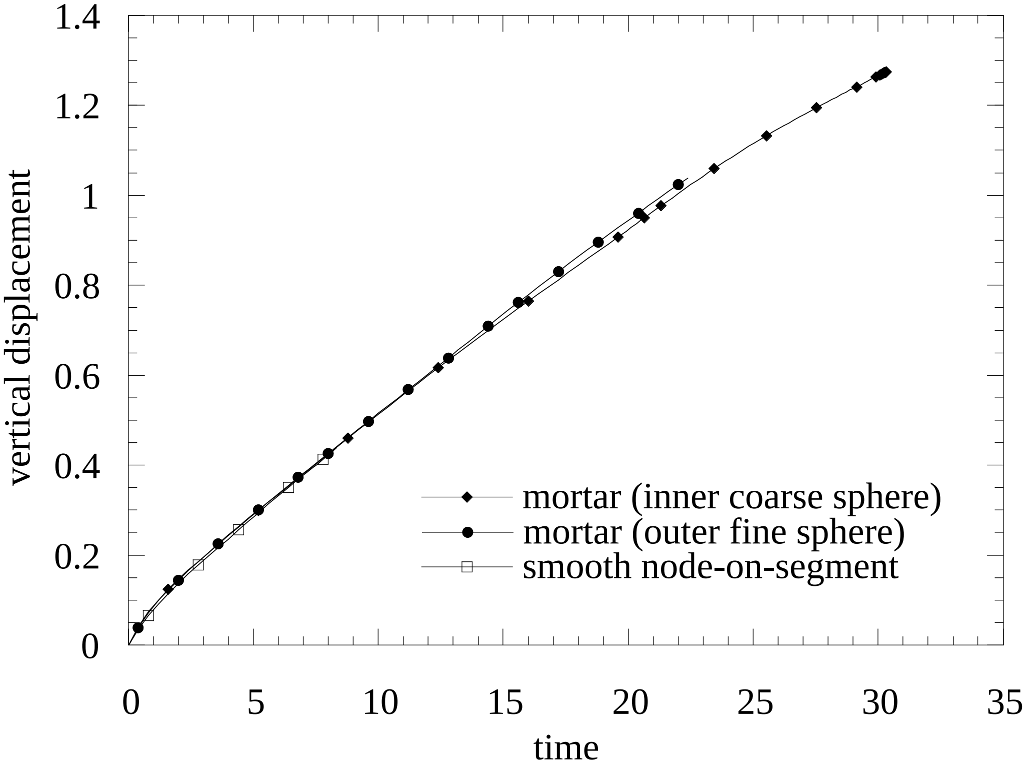


Figure 11: Center nodal vertical displacement versus time for solid sphere pressed into hollow sphere.

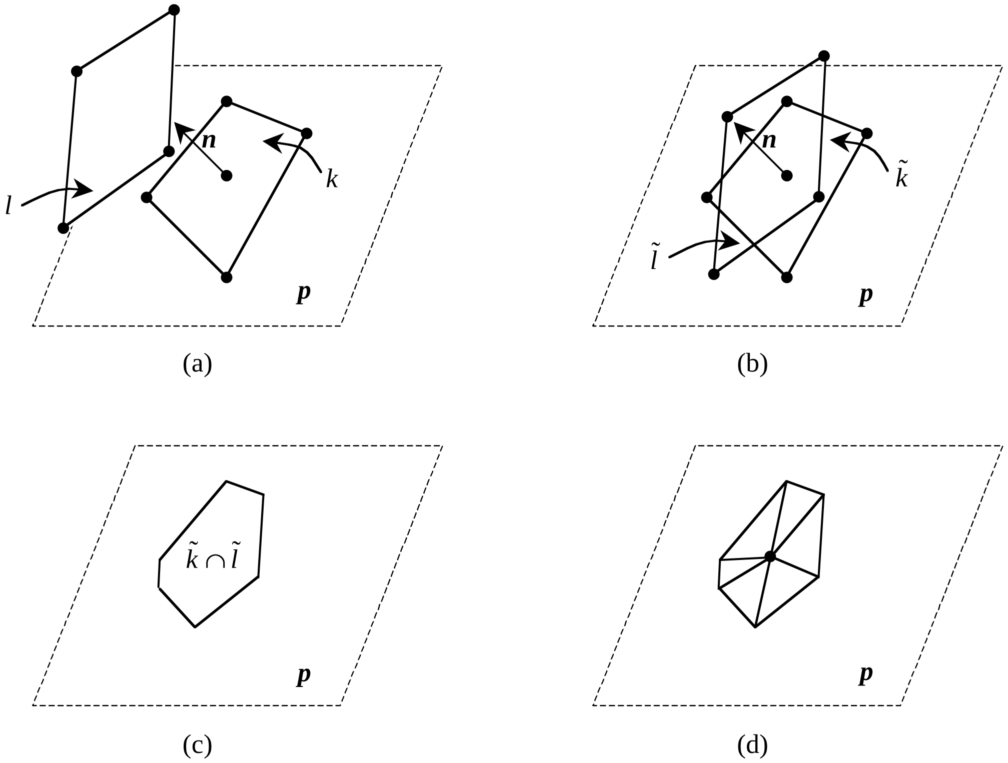


Figure 12: (a) Mortar segment l, non-mortar segment k and plane $\boldsymbol{p}$ formed from the center point and normal $\boldsymbol{n}$ of k. (b) Facet $\tilde{k}$ and $\tilde{l}$ formed by projecting k and l onto plane. (c) Polygon formed from clipping algorithm. (d) Discretization of polygon into n_p triangular pallets.

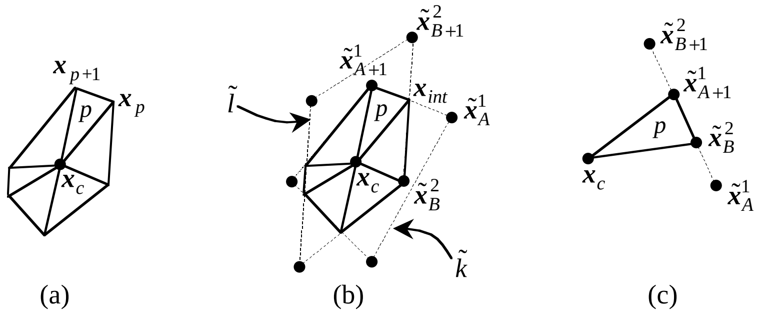

$\boldsymbol{x}_{p+1}$
$\boldsymbol{x}_p$
p
$\boldsymbol{x}_c$
(a)
$\tilde{\boldsymbol{x}}^2_{B+1}$
$\tilde{\boldsymbol{x}}^1_{A+1}$
$\boldsymbol{x}_{int}$
$\tilde{\boldsymbol{x}}^1_A$
$\tilde{l}$
p
$\boldsymbol{x}_c$
$\tilde{\boldsymbol{x}}^2_B$
$\tilde{k}$
(b)
$\tilde{\boldsymbol{x}}^2_{B+1}$
$\tilde{\boldsymbol{x}}^1_{A+1}$
p
$\tilde{\boldsymbol{x}}^2_B$
$\boldsymbol{x}_c$
$\tilde{\boldsymbol{x}}^1_A$
(c)

Figure 13: (a) Pallet p is formed by $\boldsymbol{x}_p$, $\boldsymbol{x}_{p+1}$ and $\boldsymbol{x}_c$ (b) Points on pallet correspond to projected segments $\tilde{k}$ and $\tilde{l}$ such that $\boldsymbol{x}_p = \boldsymbol{x}_{int}$ and $\boldsymbol{x}_{p+1} = \tilde{\boldsymbol{x}}^1_{A+1}$. (c) Pathological edge-on-edge case.

References

[1] M.A. Puso and T.A. Laursen. A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 2003. to appear.

[2] J.O. Hallquist, G.L. Goudreau, and D.J. Benson. Sliding interfaces with contact-impact in large-scale lagrangian computations. *Computer Methods in Applied Mechanics and Engineering*, 51:107–137, 1985.

[3] P. Papadopoulos and R.L. Taylor. A mixed formulation for the finite element solution of contact problems. Technical Report UCB/SEMM Report 90/18, University of California at Berkeley, 1990.

[4] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer Verlag, New York, 1991.

[5] B. Engelmann. Nike2d: User manual. Technical Report UCRL-MA-105413, Lawrence Livermore National Laboratory, University of California, 1991.

[6] V. Padmanabhan and T.A. Laursen. Surface smoothing procedure for large deformation contact analysis. *Finite Elements in Analysis and Design*, 37:173–198, 2001.

[7] P. Wriggers and L. Krstulovic-Opara. On smooth finite element discretization for frictional contact problems. *Zeitschrift Fur Angewandte Mathematik und Mechanik*, 80:77–80, 2000.

[8] M.A. Puso and T.A. Laursen. A 3d contact smoothing algorithm using gregory patches. *International Journal for Numerical Methods in Engineering*, 54:1161–1194, 2002.

[9] T. Belytschko, W.J.T. Daniel, and G. Ventura. A monolithic smoothing-gap algorithm for contact-impact based on the signed distance function. *International Journal for Numerical Methods in Engineering*, 55:101–125, 2002.

[10] G. Zavarise and P. Wriggers. A segment-to-segment contact strategy. *Mathl. Comput. Modelling*, 28:497–515, 1998.

[11] J.C. Simo, P. Wriggers, and R.L. Taylor. A perturbed lagrangian formulation for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 50:163–180, 1985.

[12] P. Papadopoulos and R.L. Taylor. A mixed formulation for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 50:163–180, 1992.

[13] C. Bernardi, Y. Maday, and A.T. Patera. A new nonconforming approach to domain decomposition: The mortar element method. In H. Brezia and J.L. Lions, editors, *Nonlinear Partial Differential Equations and Their Applications*, pages 13–51. Pitman and Wiley, 1992.

[14] F.B. Belgacem, P. Hild, and P. Laborde. Approximation of the unilateral contact problem by the mortar finite element method. *Comptes Rendus De L'Academie Des Sciences*, 324:123–127, 1997.

[15] P. Hild. Numerical implementation of two nonconforming finite element methods for unilateral contact. *Computer Methods in Applied Mechanics and Engineering*, 184:99–123, 2000.

[16] T.W. McDevitt and T.A. Laursen. A mortar-finite element formulation for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 48:1525–1547, 2000.

[17] N. Kikuchi and J.T. Oden. *Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods*. SIAM, Philadelphia, 1988.

[18] J.C. Simo and T.J.R. Hughes. *Computational Inelasticity*. Springer-Verlag, New York, 1997.

[19] J. Foley and et al. *Computer Graphics, Principles and Practice*. Addison-Wesley, Reading, second edition, 1997.

[20] G.R. Cowper. Gaussian quadrature formulas for triangles. *International Journal for Numerical Methods in Engineering*, 7:405–408, 1973.