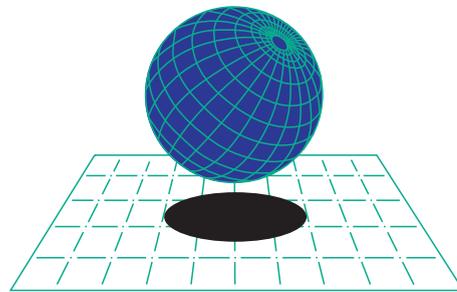


# Master of Science in Computational Mechanics



**CIMNE<sup>R</sup>**

---

## Master Thesis

### Contact formulations for 2D Simulation

---

*Student :* Sylvain VIROT  
*Supervisors :* Prof. Djordje PERIĆ  
Dr. Wulf G. DETTMER



**Swansea University**  
**Prifysgol Abertawe**

May 31<sup>st</sup>, 2012

## Abstract

Modern challenging mechanical problems are involving many sources of non-linearities and contact is one of them. Being able to tackle them individually in a consistent way is necessary before treating problems that gather several difficulties.

This report proposes the development and the implementation of two methods to manage contact mechanics in a Newton-Raphson solved mechanical formulation. Node to segment and Mortar method are described from their energy formulation to force vectors and stiffness matrices, using both Lagrange multipliers and Penalty methods to enforce the contact conditions. The solutions procedures are detailed alongside practical difficulties, explaining the different options available and the one chosen.

To deal with dynamics, the contact management method is formulated into a time-integration scheme and its consistency is discussed. Finally, a Hertz contact theoretical solution is used to validate the contact formulation and some examples are detailed.

## Declarations and Statements

### Declaration

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed .....

Date.....

### Statement 1

This dissertation is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed .....

Date .....

### Statement 2

I hereby give my consent for my dissertation, if relevant and accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .....

Date .....

---

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivations . . . . .	1
1.2	Objectives . . . . .	1
<b>2</b>	<b>Continuous contact mechanics</b>	<b>3</b>
2.1	Framework . . . . .	3
2.2	Gap function . . . . .	4
2.3	Friction and tangential gap . . . . .	6
2.4	Energy formulation . . . . .	7
<b>3</b>	<b>Discretisation and Implementation</b>	<b>10</b>
3.1	Node to Segment Formulation . . . . .	10
3.1.1	Contact detection . . . . .	10
3.1.2	Force and Stiffness computation . . . . .	11
3.1.3	Tangent force predictor/corrector . . . . .	21
3.1.4	Solution procedure and Convergence . . . . .	22
3.2	Mortar method . . . . .	23
3.2.1	Force and Stiffness computation . . . . .	24
3.2.2	Contact detection . . . . .	26
3.2.3	Solution procedure and Convergence . . . . .	27
3.2.4	Stability . . . . .	27
3.3	Time integration . . . . .	28
<b>4</b>	<b>Applications</b>	<b>30</b>
4.1	Validation: Hertz contact . . . . .	30
4.1.1	Frictionless compression . . . . .	31
4.1.2	Slide with friction . . . . .	33
4.2	Sliding block . . . . .	36
4.3	Bouncing balls . . . . .	37
4.3.1	Full elastic ball . . . . .	37
4.3.2	Pressurised ball (Tennis ball) . . . . .	39
4.4	Sliding elastic beam . . . . .	43
4.5	Active rolling wheel . . . . .	46
4.6	Full finger . . . . .	47
4.7	Gears . . . . .	49
4.8	Pseudo-realistic case: the bike . . . . .	51
<b>5</b>	<b>Conclusion</b>	<b>52</b>

## List of Figures

1	Examples of Contact Surfaces . . . . .	3
2	Gap function evaluation . . . . .	6
3	Node To Segment contact detection . . . . .	11
4	Forces equilibrium, expressed as external forces . . . . .	13
5	Mortar geometrical parameters . . . . .	25
6	Contact oscillations . . . . .	29
7	Hertz contact : Cylinder/Plane . . . . .	30
8	Hertz Stresses . . . . .	32
9	$\sigma_{xx}$ around the contact interface . . . . .	34
10	$\sigma_{yy}$ around the contact interface . . . . .	34
11	$\sigma_{zz}$ around the contact interface . . . . .	35
12	$\sigma_{xy}$ around the contact interface . . . . .	35
13	Sliding Block: Problem statement . . . . .	36
14	Sliding Block: NTS and Mortar results . . . . .	36
15	Bouncing ball: free oscillations after one bounce, $dt = 0.25ms$ . . . . .	38
16	Bouncing ball: Von Mises Stresses through time . . . . .	38
17	Bouncing ball: free oscillations von Mises Stresses after one bounce $dt = 0.25 ms$ . . . . .	38
18	Bouncing ball: kinetic and potential energies evolution ( $mJ$ vs. $ms$ ) . . . . .	39
19	Softer ball: kinetic and potential energies evolution ( $mJ$ vs. $ms$ ) . . . . .	40
20	Tennis ball: mesh . . . . .	40
21	Empty tennis ball: not bouncing . . . . .	40
22	Tennis ball: von Mises Stresses during contact . . . . .	41
23	Sliding elastic beam: problem statement . . . . .	43
24	Sliding elastic beam: mortar contact issue . . . . .	44
25	Sliding elastic beam: mortar issue solved . . . . .	44
26	Sliding elastic beam: node to segment with friction evolution . . . . .	45
27	Rolling wheel: problem statement . . . . .	46
28	Wheel: rolling on the ground, small couple . . . . .	46
29	Wheel: sliding on the ground, large couple . . . . .	47
30	Fingertip: model and mesh . . . . .	47
31	Fingertip: Global Von Mises Stresses . . . . .	48
32	Fingertip: Under skin Von Mises Stresses . . . . .	48
33	Fingertip: Skin motion . . . . .	48
34	Fingertip: stress evolution in the sensors zone . . . . .	49
35	Gears: model and mesh . . . . .	50
36	Gears: Von Mises Stresses . . . . .	50
37	Bike: Model description . . . . .	51
38	Bike: animation . . . . .	52

## Acknowledgements

I would like to thank my supervisors, Prof. Djordje Perić and Dr. Wulf G. Dettmer for their help and guidance. Dr. Dettmer gave me a strong base to implement my work by sharing his solver. He also made himself very available and it was a pleasure to be able to discuss and debate about the technical aspects of my work. Prof. Perić kindly directed my work by bringing numerous references, leaving me time to focus on the technical aspects.

Many thanks to Dr. Deniz Somer who also helped me understand the solver code structure and gave me the background for my code.

I am deeply grateful to Manon how brought me so much throughout the year.

## Definitions & Abbreviations

$a$  and  $b$  are dummy mathematical objects

NTS	Node to Segment
LM	Lagrange multiplier
GP	Gauss point
$\mathbf{a}$	Vector $a$
$\mathcal{S}^m$	Master surface
$\mathcal{S}^s$	Slave surface
$d(a, b)$	Distance between $a$ and $b$
$\theta$	Curvilinear abscissa
$g_N$	Normal gap function
$g_T$	Tangential gap function
$\mu$	Friction coefficient
$\mathring{\mathcal{S}}^s$	Interior of $\mathcal{S}^s$
$\Pi$	Energy
$\delta a$	Total virtual variations of $a$
$\Delta a$	Total variations of $a$
$\mathbf{K}$	Stiffness matrix
$\lambda_N$	Lagrange multiplier associated with Normal direction
$\lambda_T$	Lagrange multiplier associated with Tangential direction
$\xi$	Curvilinear abscissa along the master segment
$\zeta$	Curvilinear abscissa along the slave segment

# 1 Introduction

## 1.1 Motivations

Contact is omnipresent in real-life mechanics. Even though it can be replaced by particular boundary conditions in simple simulations, many multibody non-linear applications rely on its proper representation. Indeed, it plays a key role in many systems like engines (bearings, gearboxes, camshaft,...), tires (road/tire contact, tire/rim adhesion) or biological mechanisms like haptics (sense of touch) so, being able to simulate it properly opens a large field where numerics can help improve those systems.

Though, contact mechanics is not easy to solve as its complexity lies both in the spacial discontinuity of the normal contact constrain and the non-conservativity of the tangential one (friction).

This work is a first encounter with contact mechanics implementation and aims to outline and comment the different possible solution procedures. Thus, it is focused on 2D solid mechanics where the geometry is easily manipulable and the computation cheap. Nevertheless, some performance considerations will be discussed. Also, the formulation will be developed keeping in mind that it will be implemented into a full Newton-Raphson solution procedure.

## 1.2 Objectives

The aim is to have an algorithm that prevents two designated surfaces to interpenetrate. The objective is not to have an a priori way to prevent interpenetration

but a method that, during the resolution procedure, imposes conditions to achieve, once converged, the right contact forces. So, the process will act like a corrector applied on the solid solution procedure, after each iteration.

This document also aims to compare different contact formulation (Node to segment and Mortar) and different numerical enforcement (Penalty and Lagrange multipliers). Their consistent implementation into a time-integration scheme will be developed.

However, this work does not aim to develop all the mathematics that would be necessary to fully define and justify the operations involved (no use of Sobolev spaces or distributions even if it would be needed). Instead, the reader will be encouraged to read more specialised literature.

## 2 Continuous contact mechanics

### 2.1 Framework

The theory presented in this document is inspired by the formulation developed by Wriggers<sup>1</sup>. The main ideas will be summed up, corrected and extended.

The most common way of dealing with contact is to break the symmetry of the problem by distinguishing the two surfaces in contact. One is used as a geometrical reference and called ‘Master’ (denoted  $\mathcal{S}^m$ ), the other is the ‘Slave’ (denoted  $\mathcal{S}^s$ ). The slave is split into individual entities depending on the formulation, and can be considered in contact or not with the master. Each of these entity can be either in contact or not in contact, independently from the other ones, with the master surface. The contact enforcement is said to be local (see Figure 1).

The symmetry of the problem can be recovered by defining two contacts, where the second one swaps the master with the slave. Nevertheless, this is breaking a stability condition as shown, according to Puso and Laursen<sup>2</sup>, by Brezzi and Fortin<sup>3</sup>.

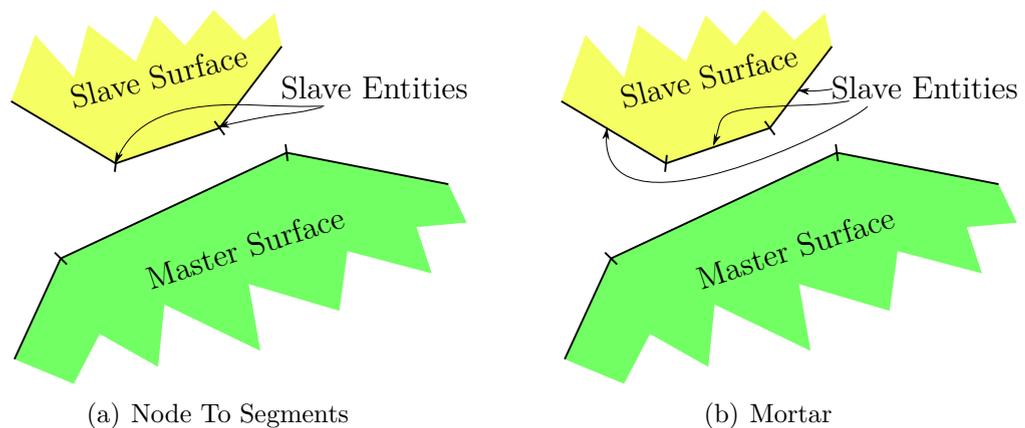


Figure 1: Examples of Contact Surfaces

As explained in the introduction, the method is a corrector to the solid solver : it checks that two surfaces are not penetrating each other and, if they do, imposes forces to prevent this penetration. This means that the following theory is considering a certain penetration and its aim is to find the forces that cancels it.

## 2.2 Gap function

To deal with contact, the notion of distance between the two potentially in contact interfaces must be detailed. To do so, a function, called Gap function and denoted  $g_N$  is defined along the slave surface  $\mathcal{S}^s$ . This function gives, for each point of the slave surface, its distance to the master with a sign, depending on the position regarding to the orientation of the slave surface.

The general definition of the distances are:

$$\forall \mathbf{x}_s \in \mathcal{S}^s, \forall \mathbf{x}_m \in \mathcal{S}^m, \quad d(\mathbf{x}_s, \mathbf{x}_m) = \|\mathbf{x}_m - \mathbf{x}_s\|_2 \quad (1)$$

$$\forall \mathbf{x}_s \in \mathcal{S}^s, \quad d(\mathbf{x}_s, \mathcal{S}^m) = \min_{\mathbf{x}_m \in \mathcal{S}^m} d(\mathbf{x}_s, \mathbf{x}_m) = \min_{\mathbf{x}_m \in \mathcal{S}^m} \|\mathbf{x}_m - \mathbf{x}_s\|_2 \quad (2)$$

Parametrising the master surface:

$$\forall \mathbf{x}_s \in \mathcal{S}^s, \quad d(\mathbf{x}_s, \mathcal{S}^m) = \min_{\theta \in [0;1]} \|\mathbf{x}_m(\theta) - \mathbf{x}_s\|_2 \quad (\text{P1})$$

To find the distance, this minimisation problem must be solved. Thus, the regularity of the master surface  $\theta \mapsto \mathbf{x}_m(\theta)$  is important. For a  $C^1$  interface, the distance  $d(\mathbf{x}_s, \mathcal{S}^m)$  ( $\mathbf{x}_s$  being fixed and not on the interface) is also  $C^1$  so its extrema in  $]0; 1[$  can be obtained by finding the zeros of the derivative of  $d_{\mathbf{x}_s} : \theta \mapsto \|\mathbf{x}_m(\theta) - \mathbf{x}_s\|_2$ .

It gives:

$$d'_{\mathbf{x}_s}(\theta) = \frac{1}{\|\mathbf{x}_m(\theta) - \mathbf{x}_s\|_2} (\mathbf{x}_m(\theta) - \mathbf{x}_s) \cdot \left( \frac{d\mathbf{x}_m}{d\theta}(\theta) - \frac{d\mathbf{x}_s}{d\theta}(\theta) \right) \quad (3)$$

So,

$$d'_{\mathbf{x}_s}(\theta) = 0 \quad \Rightarrow \quad \underbrace{(\mathbf{x}_m(\theta) - \mathbf{x}_s)}_{\text{Projection direction}} \cdot \underbrace{\left( \frac{d\mathbf{x}_m}{d\theta}(\theta) \right)}_{\text{Tangent to } S^m \text{ at } \theta} = 0 \quad (4)$$

The last expression is also valid if the slave node  $\mathbf{x}_s$  is on the interface.

This formulation shows that all the extrema (of  $d_{\mathbf{x}_s}$  in  $]0; 1[$ ) are obtained when the projection direction is orthogonal to the projection surface tangent for an interior point of the projection surface.

The minimisation problem implies: (this problem is not equivalent to the minimisation one)

$$\text{find } \theta_0 \in ]0; 1[ / \quad (\mathbf{x}_m(\theta_0) - \mathbf{x}_s) \cdot \left( \frac{d\mathbf{x}_m}{d\theta}(\theta_0) \right) = 0 \quad (\text{P2})$$

From the set of solutions of (P2), the minimum must be extracted and compared with the distance from the extremities of the interface ( $\theta = 0$  or  $\theta = 1$ ). This way, the maxima of  $d_{\mathbf{x}_s}$  are always removed and the final solution is truly the minimum of  $d_{\mathbf{x}_s}$  on  $[0; 1]$ .

Once the distance is computed, the sign must be determined. The normal of the surface is compared with the projection direction. If the minimum is at  $\theta = 0$  or  $\theta = 1$  no sign can be easily determined. However, it often corresponds to non-interesting cases (see Figure 2(a)) and the contact can be ignored in this case.

To sum-up,  $g_N$  can be precisely defined :

$$g_N : \left( \begin{array}{l} \mathcal{S}^s \rightarrow \mathbb{R} \\ \mathbf{x}_s \mapsto g_N(\mathbf{x}_s) \end{array} \right) \quad (5)$$

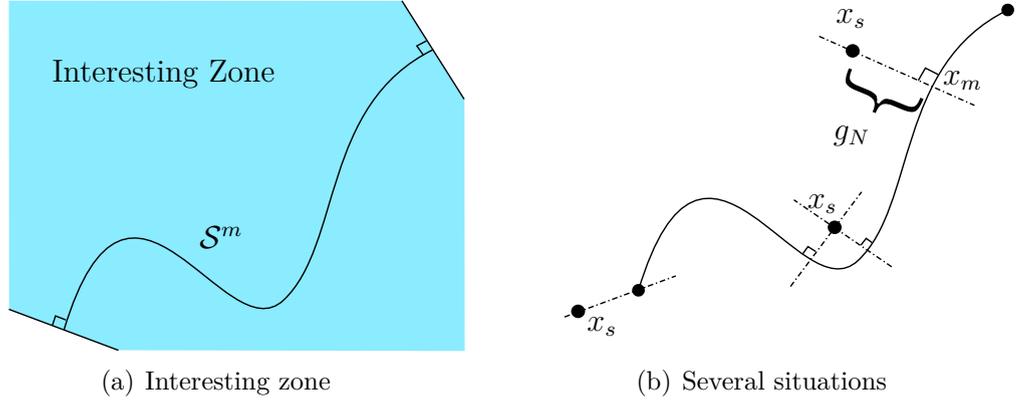


Figure 2: Gap function evaluation

The procedure to compute  $g_N(\mathbf{x}_s)$  is:

Find the solutions  $\{\mathbf{x}_m(\theta_i)\}_i$  of (P2)

Determine the  $\theta_{min} \in \{\{\theta_i\}_i, 0, 1\}$  such that  $d(\mathbf{x}_m(\theta_{min}), \mathcal{S}^m)$  is minimum

If  $\theta_{min} = 0$  or  $1$ , return  $g_N(\mathbf{x}_s) = d(\mathbf{x}_m(\theta_{min}), \mathcal{S}^m)$  (positive gap, no contact)

Compute the normal of the interface at  $\theta_{min} : \mathbf{n}(\theta_{min}) = \left( \frac{d\mathbf{x}_m}{d\theta}(\theta_{min}) \right)^\perp$

Return  $g_N(\mathbf{x}_s) = (\mathbf{n}(\theta_{min}) \cdot (\mathbf{x}_m(\theta_{min}) - \mathbf{x}_s)) d(\mathbf{x}_m(\theta_{min}), \mathcal{S}^m)$

For less regular master surfaces (piecewise  $C^1$  in our implementation), the method is applied on each segment where the interface is  $C^1$ . The extreme cases ( $\theta = 0$  or  $\theta = 1$ ) need to be examined depending on the contact formulation chosen.

## 2.3 Friction and tangential gap

In nature, dry friction lead to either stick two surfaces or dissipate energy along two sliding surfaces. The formulation is simple but needs to be reworked to fit into a discrete force/displacement (potentially quasi-static) finite element procedure.

$$F_t \leq \mu F_n \text{ and } \mathbf{F} \cdot \mathbf{V} < 0 \quad (6)$$

A paradox arise from those equations when trying to apply in quasi-static. Indeed, there is no time dependence in quasi-static formulation so, neither velocity nor dissipation have a meaning. Thus, in quasi-statics, the contact must be "stick" type in the converged step. However, when not converged, the contact point can "slide" to the right position.

The stick case is conservative and so, easy to formulate, while the energy dissipation is transformed into an internal variable (cumulated sliding gap or cumulated tangential force) to fit the Force/Displacement formulation. The dissipation itself comes from the fact that the nodal force is opposed to the sliding displacement, producing a negative input work. From the numerical point of view, this is similar to what exists in plasticity. This algorithm will be described later because it is related to the method used to discretise the surfaces.

Similarly to the normal component, the slave might not be at its physical position (denoted  $\mathbf{x}_s^c$ ), so, a "tangential gap function" (denoted  $g_T$ ) is defined for each point  $\mathbf{x}_s$  of the slave surface  $\mathcal{S}^s$  which is in contact. This function measures the distance between the projection of  $\mathbf{x}_s$  on  $\mathcal{S}^m$  and the point  $\mathbf{x}_s^c$  along the interface  $\mathcal{S}^m$ . In stick case, this point  $\mathbf{x}_s^c$  is either at the point of first contact or at the point where slip ended. In slip case, this point follows the projection of  $\mathbf{x}_s$  on  $\mathcal{S}^m$ . So, in slip case,  $g_T = 0$ .

## 2.4 Energy formulation

The total energy linked to the contact is defined by integrating local contributions along the contact interface. As the gap functions are defined on the slave  $\mathcal{S}^s$ , it's where the integral is computed. The contact energy is divided into two contributions,

the normal and the tangential ones. Also, the energy depends on the enforcement method. The curvilinear abscissa along the slave surface is denoted  $\xi$ .

For penalty method (P):

$$\Pi_N^P = \frac{1}{2} \int_{S^s} \varepsilon_N(\xi) (g_N(\xi))^2 d\xi \quad \Pi_T^P = \frac{1}{2} \int_{S^s} \varepsilon_T(\xi) (g_T(\xi))^2 d\xi \quad (7)$$

where  $\varepsilon_N$  and  $\varepsilon_T$  are the "penalty parameters" that need to be tweaked.

For Lagrange multipliers method (LM):

$$\Pi_N^{LM} = \int_{S^s} \lambda_N(\xi) g_N(\xi) d\xi \quad \Pi_T^{LM} = \int_{S^s} \lambda_T(\xi) g_T(\xi) d\xi \quad (8)$$

where  $\lambda_N$  and  $\lambda_T$  are the Lagrange multipliers. They are extra spacial variables.

Choosing the formulation for both normal and tangential contacts lead to the contact energy ( $\square$  being either  $P$  or  $LM$ ):

$$\Pi_{\text{contact}} = \Pi_N^{\square} + \Pi_T^{\square} \quad (9)$$

This energy needs to be minimised to find equilibrium so its differential is computed:

For penalty method (\* being either  $N$  or  $T$ ):

$$\delta \Pi_*^P = \int_{S^s} \varepsilon_*(\xi) g_*(\xi) \delta g_*(\xi) d\xi + \int_{S^s} \varepsilon_*(\xi) (g_*(\xi))^2 \delta d\xi \quad (10)$$

And for Lagrange multiplier method (\* being either  $N$  or  $T$ ):

$$\delta \Pi_*^{LM} = \int_{S^s} \left( \lambda_*(\xi) \delta g_*(\xi) + \delta \lambda_*(\xi) g_*(\xi) \right) d\xi + \int_{S^s} \lambda_*(\xi) g_*(\xi) \delta d\xi \quad (11)$$

Both can be generalised as (\* being either  $N$  or  $T$  and  $\square$  being either  $P$  or  $LM$ ):

$$\begin{aligned}\Pi_*^\square &= \int_{S^s} p_*^\square(\xi) g_*(\xi) d\xi \\ \delta\Pi_*^\square &= \int_{S^s} \left( p_*^\square(\xi) \delta g_*(\xi) + \delta p_*^\square(\xi) g_*(\xi) \right) d\xi + \int_{S^s} p_*^\square(\xi) g_*(\xi) \delta d\xi \quad (12)\end{aligned}$$

where  $p_*^P(\xi) = \frac{1}{2} \varepsilon_*(\xi) g_*(\xi)$  and  $p_*^{LM}(\xi) = \lambda_*$

## 3 Discretisation and Implementation

### 3.1 Node to Segment Formulation

Node to Segment is the most naïve formulation. Here, the slaves entities are the nodes of the slave surface. This document is also restricted to geometrically linear finite elements so the master surface is made from segments, so it is piecewise  $C^1$ .

#### 3.1.1 Contact detection

The gap function is computed at each slave node and its value is used to determine whether the node is in contact. As the Master surface is only piecewise  $C^1$  ( $C^1$  on each segment between two master nodes), the contact detection is applied on each segment separately. The first condition for contact is that the projection of the slave node is on the master (i.e. the slave is in the "interesting zone" of the master (see Figure 2(a))) and that the gap function is negative. Moreover, to allow closed body contact, a maximum penetration need to be defined.

To take all of that into account, a "Buffer" zone is defined along the Master surface: the contact is activated if and only if the slave node is in that zone (see Figure 3(a)). Nevertheless, due to the non-regularity of the Master surface, blind zones appear when the surface is concave. Wriggers<sup>4</sup> proposes a "Node to Node" formulation for those cases but it has not been used. Instead, a small overlapping is implemented by extending the effective length of the Master surface segments (see Figure 3(b)).

Once the contact is established, the gap is virtually null so it can't be used to check if the contact is lost. Instead, the contact is lost if the contact force is pulling the contact surfaces together.

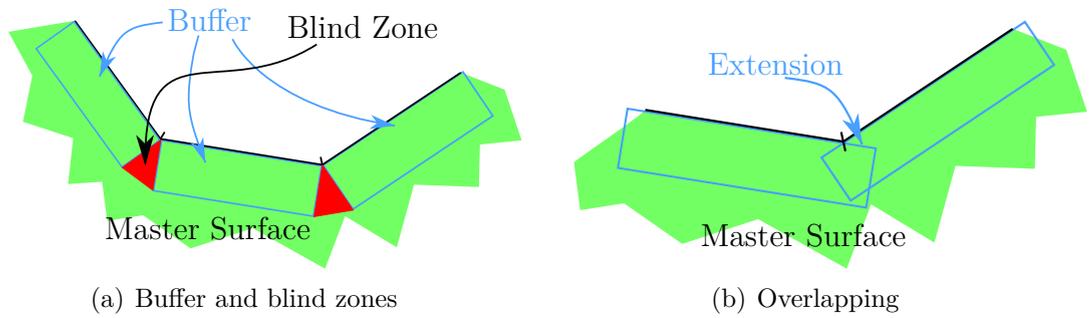


Figure 3: Node To Segment contact detection

### 3.1.2 Force and Stiffness computation

To be solved, the energy formulation (12) needs to be discretised. As the aim is to use a Newton-Raphson solver, both a force vector and a stiffness matrix need to be computed. Unlike Wriggers<sup>1</sup>, this document develops the four cases that arise from the choice of the contact enforcement method (Penalty or Lagrange multipliers) both for normal and tangential force.

The discretisation of (12) starts by the choice of quadrature. Node to segment method consists, by definition, in choosing a collocation method weighted with contact areas to evaluate the integral. From (9) and (12), the total energy variation

can be discretised (\* being either  $P$  or  $LM$  and  $\square$  being either  $P$  or  $LM$ ):

$$\begin{aligned}
 \delta\Pi_{\text{contact}} &= \delta\Pi_N^{\square} + \delta\Pi_T^{\square} \\
 &= \int_{S^s} \left( p_N^{\square}(\xi) \delta g_N(\xi) + \delta p_N^{\square}(\xi) g_N(\xi) \right) d\xi + \int_{S^s} \left( p_T^{\square}(\xi) \delta g_T(\xi) + \delta p_T^{\square}(\xi) g_T(\xi) \right) d\xi \\
 &\quad + \int_{S^s} p_N^{\square}(\xi) g_N(\xi) \delta d\xi + \int_{S^s} p_T^{\square}(\xi) g_T(\xi) \delta d\xi \\
 &\simeq \sum_{\substack{s \text{ being} \\ \text{slave nodes}}} \left( p_N^s A^s \delta g_N^s + p_T^s A^s \delta g_T^s + \delta p_N^s A^s g_N^s + \delta p_T^s A^s g_T^s \right)
 \end{aligned} \tag{13}$$

where  $A^s$  is the contact area associated with slave node  $s$ .  $A^s$  is quite artificial and does not always have a physical meaning. It is considered constant. From now, its parameter is joined with the pressure to give a force ( $p_*$  is now a force). This area is hidden either in the penalty parameter or in the Lagrange multiplier.

Indeed, for the contact to be satisfied at each slave node, each term of the sum needs to be null, so:

$$\delta\Pi_s = p_N^s \delta g_N^s + p_T^s \delta g_T^s + \delta p_N^s g_N^s + \delta p_T^s g_T^s = 0 \tag{14}$$

The forces can be extracted using the fact that they derive from the potential (except for slip case where dissipation is present, see Section 3.1.3):

$$T_* = \frac{\delta\Pi_s}{\delta g_*^s} = p_*^s + g_*^s \frac{\delta p_*^s}{\delta g_*^s} = \begin{cases} \varepsilon_* g_*^s & \text{for Penalty} \\ \lambda_*^s & \text{for Lagrange multipliers} \end{cases} \tag{15}$$

## Geometrical Representation

On Figure 4, the slave node (denoted  $\mathbf{x}_s$ ) is isolated from the body it belongs to in order to clarify the picture. The segment of the master surface in contact

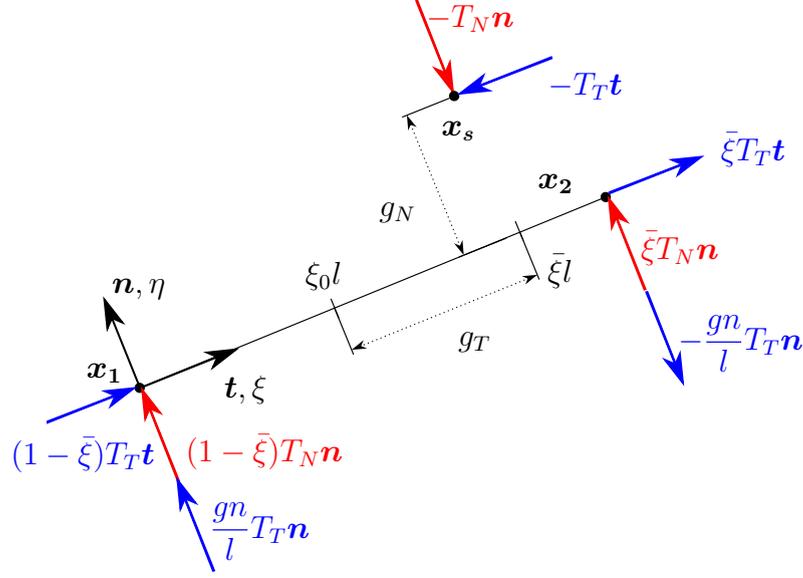


Figure 4: Forces equilibrium, expressed as external forces

with  $\mathbf{x}_s$ , (denoted  $[\mathbf{x}_1, \mathbf{x}_2]$  of length  $l$ ) is the only one represented. On the drawn configuration, the slave node is considered "in contact", and so, in the body. Note that, unlike Wriggers<sup>5</sup>, the interface is orientated clockwise so, the direct orthogonal to the tangent is the inner normal. This changes the sign of  $g_N$ . Moreover, there is an error in the definition of  $\bar{\xi}$  that Wriggers<sup>5</sup> does : he defined  $\bar{\xi} = (\mathbf{x}_s - \mathbf{x}_1) \cdot \mathbf{t}$  before using it like  $\dots (1 - \bar{\xi}) \dots$  which is not homogeneous. The correct definition (which is used here) would be :  $\bar{\xi} = \frac{1}{l} (\mathbf{x}_s - \mathbf{x}_1) \cdot \mathbf{t}$  where  $\mathbf{t}$  is the unit vector along the interface.

Here,  $\xi l$  is the curvilinear abscissa of  $\mathbf{x}_s^c$ .

The equilibrium is obtained by considering the force applied on  $\mathbf{x}_s$  and deducing the forces on  $\mathbf{x}_1$  and  $\mathbf{x}_2$  that minimise the deformation of the interface. This is done by considering that  $\mathbf{x}_s, \mathbf{x}_1, \mathbf{x}_2$  is a rigid body, except for the tangential force (the system is hyperstatic in the  $\mathbf{t}$  direction) which arise from linear elasticity.

For notation convenience, displacements (more exactly, degrees of freedom, including non-physical like Lagrange multipliers) and forces will be put into a vector

(in fact, a vector of vectors):

$$\mathbf{X} = \begin{Bmatrix} \mathbf{x} \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \lambda \end{Bmatrix} \quad \mathbf{F} = \begin{Bmatrix} \mathbf{f}_s \\ \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_\lambda \end{Bmatrix} \quad (16)$$

Thus, several "base" vectors can be defined to help forming the future developments:

$$\mathbf{N} = \begin{Bmatrix} \mathbf{n} \\ -(1 - \bar{\xi})\mathbf{n} \\ -\bar{\xi}\mathbf{n} \\ \mathbf{0} \end{Bmatrix} \quad \mathbf{N}_0 = \begin{Bmatrix} \mathbf{0} \\ -\mathbf{n} \\ \mathbf{n} \\ \mathbf{0} \end{Bmatrix} \quad \mathbf{T} = \begin{Bmatrix} \mathbf{t} \\ -(1 - \bar{\xi})\mathbf{t} \\ -\bar{\xi}\mathbf{t} \\ \mathbf{0} \end{Bmatrix} \quad \mathbf{T}_0 = \begin{Bmatrix} \mathbf{0} \\ -\mathbf{t} \\ \mathbf{t} \\ \mathbf{0} \end{Bmatrix} \quad (17)$$

$$\mathbf{\Lambda}_N = \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ (1) \\ (0) \end{Bmatrix} \quad \mathbf{\Lambda}_T = \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ (0) \\ (1) \end{Bmatrix} \quad \text{where } \mathbf{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \text{and } \boldsymbol{\lambda} = \begin{pmatrix} \lambda_N \\ \lambda_T \end{pmatrix}$$

To compute the energy variation from (14), some simple geometry and differentiation is used to obtain:

$$\begin{aligned} \delta g_N &= \delta \mathbf{X}^T \mathbf{N} & \delta g_T &= \delta \mathbf{X}^T \left( \mathbf{T} + \frac{g_N}{l} \mathbf{N}_0 + \frac{g_T}{l} \mathbf{T}_0 \right) \\ \delta p_N &= \delta \mathbf{X}^T \mathbf{\Lambda}_N & \delta p_T &= \delta \mathbf{X}^T \mathbf{\Lambda}_T \end{aligned} \quad (18)$$

Thus, the contact force (which is the residual of the Newton procedure) can be expressed as:

$$\mathbf{F} = T_N \mathbf{N} + T_T \left( \mathbf{T} + \frac{g_N}{l} \mathbf{N}_0 + \frac{g_T}{l} \mathbf{T}_0 \right) + \underbrace{g_N \mathbf{\Lambda}_N + g_T \mathbf{\Lambda}_T}_{\text{only if LM method}} \quad (19)$$

keeping in mind that for slip case,  $g_T = 0$  (see end of Section 2.3).

To satisfy (14) numerically, a Newton-Raphson procedure is used, so, it must be differentiated (a second time) with respect to the nodal displacements ( $\Delta$  symbol

will be used).

$$\Delta p_N \delta g_N + p_N \Delta \delta g_N + \Delta p_T \delta g_T + p_T \Delta \delta g_T + \delta p_N \Delta g_N + \delta p_T \Delta g_T + \Delta \delta p_N g_N + \Delta \delta p_T g_T = 0 \quad (20)$$

note that  $\Delta \delta \lambda_* = 0$ .

Equation (20) can be written in terms of forces by regrouping some terms together for the penalty case:

$$\begin{aligned} \delta g_* \Delta p_* + \Delta g_* \delta p_* &= 2 \frac{1}{2} \varepsilon_* \Delta g_* \delta g_* = \Delta T_* \delta g_* \\ p_* \Delta \delta g_* + \Delta \delta p_* g_* &= 2 \frac{1}{2} \varepsilon_* g_* \Delta \delta g_* = T_* \Delta \delta g_* \end{aligned} \quad (21)$$

Now, if by convention  $\delta T_* = 0$  for penalty method, (20) can be rewritten:

$$\Delta T_N \delta g_N + T_N \Delta \delta g_N + \Delta T_T \delta g_T + T_T \Delta \delta g_T + \delta T_N \Delta g_N + \delta T_T \Delta g_T = 0 \quad (22)$$

This is compatible with what Wriggers<sup>1</sup> writes with too few explanations about notations.

Obviously, formulas from (18) are still valid if  $\delta$  is replaced by  $\Delta$ . More expressions can be added:

$$\begin{aligned}
 \Delta \delta g_N &= \delta \mathbf{X}^T \underbrace{\left( -\frac{1}{l} (\mathbf{N}_0 \mathbf{T}^T + \mathbf{T} \mathbf{N}_0^T + \frac{g_N}{l} \mathbf{N}_0 \mathbf{N}_0^T) \right)}_{\mathbf{T}_\Delta} \Delta \mathbf{X} \\
 \Delta \delta g_T &= \begin{cases} \text{For stick:} \\ \delta \mathbf{X}^T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T) - \frac{g_T}{l^2} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \Delta \mathbf{X} \\ \text{For slip:} \\ \delta \mathbf{X}^T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2\mathbf{T}_0 \mathbf{N}_0^T) \right) \Delta \mathbf{X} \end{cases}
 \end{aligned} \tag{23}$$

The difference between stick and slip is due to the fact that slip case considers  $g_T = 0$  and so,  $\bar{\xi} = \xi_0$  which is constant. This is not a physical approximation but really a definition, that's why there is a difference between considering it before the differentiation and after.

The only unknown are the  $\Delta T_*$  which depend on the formulations so each case needs to be considered separately. From Section 3.1.3, the tangential force in slip case is  $T_T = \mu T_N \text{sgn}(T_T^{\text{trial}})$  so, in this case,  $\Delta T_T = \mu \Delta T_N \text{sgn}(T_T^{\text{trial}})$ .

To simplify the notations, the stiffness matrices will be denoted  $\mathbf{K}_{\nabla}^{\square, \diamond}$  where  $\square$  is  $P$  or  $LM$  depending on the method used for normal force,  $\diamond$  is  $P$  or  $LM$  depending on tangential method and  $\nabla$  is  $N$  for normal contribution, Slip for tangential slip, Stick for tangential stick or  $LM$  for the contribution associated with the Lagrange multipliers. For example,  $\mathbf{K}_{Stick}^{LM, P}$  is the stiffness contribution for the tangential force in stick case when the normal contact is solved by Lagrange multipliers and the tangential one by penalty method.

**Penalty for both:** This formulation uses  $T_N = \varepsilon_N g_N$  and  $T_T^{\text{trial}} = \varepsilon_T g_T$  so, (22) becomes :

$$\varepsilon_N \Delta g_N \delta g_N + \varepsilon_N g_N \Delta \delta g_N + \begin{cases} \varepsilon_T \Delta g_T \\ \mu \varepsilon_N \Delta g_N \operatorname{sgn}(\varepsilon_T g_T) \end{cases} \delta g_T + \varepsilon_T g_T \Delta \delta g_T + 0 + 0 = 0 \quad (24)$$

Which can be written in terms of  $\delta \mathbf{X}$  and  $\Delta \mathbf{X}$ :

Normal contribution (2 first terms):

$$\delta \mathbf{X}^T \underbrace{\left( \varepsilon_N \mathbf{N} \mathbf{N}^T - \frac{\varepsilon_N g_N}{l} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T + \frac{g_N}{l} \mathbf{N}_0 \mathbf{N}_0^T) \right)}_{\mathbf{K}_N^{P,P}} \Delta \mathbf{X}^T \quad (25)$$

Tangential contribution (2 last terms) in stick case:

$$\begin{aligned} \delta \mathbf{X}^T \left[ \varepsilon_T \mathbf{T}_\Delta \mathbf{T}_\Delta^T + \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T) \right. \right. \\ \left. \left. - \frac{g_T}{l^2} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \\ \Rightarrow \mathbf{K}_{\text{Stick}}^{P,P} = \varepsilon_T \left[ \mathbf{T}_\Delta \mathbf{T}_\Delta^T + \frac{g_T}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \frac{g_T}{l} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T)) \right] \end{aligned} \quad (26)$$

which is correct and the same than in Wriggers<sup>6</sup> whereas Wriggers<sup>1</sup> is wrong (bad factorisation).

Tangential contribution (2 last terms) in slip case:

$$\delta \mathbf{X}^T \left[ \mu \varepsilon_N \operatorname{sgn}(\varepsilon_T g_T) \mathbf{T}_\Delta \mathbf{N}^T + \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \quad (27)$$

$$\begin{aligned} \Rightarrow \mathbf{K}_{\text{Slip}}^{P,P} = \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) \\ + \mu \varepsilon_N \operatorname{sgn}(\varepsilon_T g_T) \mathbf{T}_\Delta \mathbf{N}^T \end{aligned} \quad (28)$$

**Lagrange multiplier for normal force and penalty for tangential:** This formulation uses  $T_N = \lambda_N$  and  $T_T^{\text{trial}} = \varepsilon_T g_T$  so, (22) becomes :

$$\Delta\lambda_N\delta g_N + \lambda_N\Delta\delta g_N + \begin{cases} \varepsilon_T\Delta g_T \\ \mu\Delta\lambda_N \operatorname{sgn}(\varepsilon_T g_T) \end{cases} \delta g_T + \varepsilon_T g_T \Delta\delta g_T + \delta\lambda_N\Delta g_N + 0 = 0 \quad (29)$$

Which can be written in terms of  $\delta\mathbf{X}$  and  $\Delta\mathbf{X}$ :

Normal contribution (2 first terms):

$$\delta\mathbf{X}^T \underbrace{\left( \mathbf{N}\boldsymbol{\Lambda}_N^T - \frac{\lambda_N}{l} (\mathbf{N}_0\mathbf{T}^T + \mathbf{T}\mathbf{N}_0^T + \frac{g_N}{l} \mathbf{N}_0\mathbf{N}_0^T) \right)}_{\mathbf{K}_N^{LM,P}} \Delta\mathbf{X}^T \quad (30)$$

Tangential contribution (2 next terms) in stick case:

$$\begin{aligned} & \delta\mathbf{X}^T \left[ \varepsilon_T \mathbf{T}_\Delta \mathbf{T}_\Delta^T + \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T) \right. \right. \\ & \quad \left. \left. - \frac{g_T}{l^2} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta\mathbf{X}^T \\ \Rightarrow \mathbf{K}_{\text{Stick}}^{LM,P} &= \varepsilon_T \left[ \mathbf{T}_\Delta \mathbf{T}_\Delta^T + \frac{g_T}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \frac{g_T}{l} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T)) \right] \end{aligned} \quad (31)$$

Tangential contribution (2 next terms) in slip case:

$$\begin{aligned} & \delta\mathbf{X}^T \left[ \mu \operatorname{sgn}(\varepsilon_T g_T) \mathbf{T}_\Delta \boldsymbol{\Lambda}_N^T + \right. \\ & \quad \left. \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2\mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta\mathbf{X}^T \quad (32) \\ \Rightarrow \mathbf{K}_{\text{Slip}}^{LM,P} &= \varepsilon_T g_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2\mathbf{T}_0 \mathbf{N}_0^T) \right) \\ & \quad + \mu \operatorname{sgn}(\varepsilon_T g_T) \mathbf{T}_\Delta \boldsymbol{\Lambda}_N^T \quad (33) \end{aligned}$$

Lagrange multipliers contribution (last term):

$$\delta\mathbf{X}^T \boldsymbol{\Lambda}_N \Delta\mathbf{X}^T \mathbf{N} \quad (34)$$

$$\Rightarrow \mathbf{K}_{LM}^{LM,P} = \boldsymbol{\Lambda}_N \mathbf{N}^T \quad (35)$$

**Penalty for normal force and Lagrange multiplier for tangential:** This formulation uses  $T_N = \varepsilon_N g_N$  and  $T_T^{\text{trial}} = \lambda_T$  so, (22) becomes :

$$\varepsilon_N \Delta g_N \delta g_N + \varepsilon_N g_N \Delta \delta g_N + \begin{cases} \Delta \lambda_T \\ \mu \varepsilon_N \Delta g_N \operatorname{sgn}(\lambda_T) \end{cases} \delta g_T + \lambda_T \Delta \delta g_T + 0 + \delta \lambda_T \Delta g_T = 0 \quad (36)$$

Which can be written in terms of  $\delta \mathbf{X}$  and  $\Delta \mathbf{X}$ :

Normal contribution (2 first terms):

$$\delta \mathbf{X}^T \underbrace{\left( \varepsilon_N \mathbf{N} \mathbf{N}^T - \frac{\varepsilon_N g_N}{l} (\mathbf{N}_0 \mathbf{T}^T + \mathbf{T} \mathbf{N}_0^T + \frac{g_N}{l} \mathbf{N}_0 \mathbf{N}_0^T) \right)}_{\mathbf{K}_N^{P,LM}} \Delta \mathbf{X}^T \quad (37)$$

Tangential contribution (2 next terms) in stick case:

$$\begin{aligned} \delta \mathbf{X}^T \left[ \Lambda_T \mathbf{T}_\Delta^T + \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T) \right. \right. \\ \left. \left. - \frac{g_T}{l^2} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \\ \Rightarrow \mathbf{K}_{\text{Stick}}^{P,LM} = \Lambda_T \mathbf{T}_\Delta^T + \frac{\lambda_T}{l} \left( \mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \frac{g_T}{l} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \end{aligned} \quad (38)$$

Tangential contribution (2 next terms) in slip case:

$$\delta \mathbf{X}^T \left[ \mu \varepsilon_N \operatorname{sgn}(\lambda_T) \mathbf{T}_\Delta \mathbf{T}_\Delta^T + \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \quad (39)$$

$$\begin{aligned} \Rightarrow \mathbf{K}_{\text{Slip}}^{P,LM} = \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) \\ + \mu \varepsilon_N \operatorname{sgn}(\lambda_T) \mathbf{T}_\Delta \mathbf{T}_\Delta^T \end{aligned} \quad (40)$$

Lagrange multipliers contribution (last term):

$$\delta \mathbf{X}^T \boldsymbol{\Lambda}_T \Delta \mathbf{X}^T \mathbf{T}_\Delta \quad (41)$$

$$\Rightarrow \mathbf{K}_{LM}^{P,LM} = \boldsymbol{\Lambda}_N \mathbf{T}_\Delta^T \quad (42)$$

**Lagrange multiplier for both:** This formulation uses  $T_N = \lambda_N$  and  $T_T^{\text{trial}} = \lambda_T$  so, (22) becomes :

$$\Delta \lambda_N \delta g_N + \lambda_N \Delta \delta g_N + \begin{cases} \Delta \lambda_T \\ \mu \Delta \lambda_N \operatorname{sgn}(\lambda_T) \end{cases} \delta g_T + \lambda_T \Delta \delta g_T + \delta \lambda_N \Delta g_N + \delta \lambda_T \Delta g_N = 0 \quad (43)$$

Which can be written in terms of  $\delta \mathbf{X}$  and  $\Delta \mathbf{X}$ :

Normal contribution (2 first terms):

$$\delta \mathbf{X}^T \underbrace{\left( \mathbf{N} \boldsymbol{\Lambda}_N^T - \frac{\lambda_N}{l} (\mathbf{N}_0 \mathbf{T}^T + \mathbf{T} \mathbf{N}_0^T + \frac{g_N}{l} \mathbf{N}_0 \mathbf{N}_0^T) \right)}_{\mathbf{K}_N^{LM,LM}} \Delta \mathbf{X}^T \quad (44)$$

Tangential contribution (2 next terms) in stick case:

$$\begin{aligned} & \delta \mathbf{X}^T \left[ \boldsymbol{\Lambda}_T \mathbf{T}_\Delta^T + \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T) \right. \right. \\ & \quad \left. \left. - \frac{g_T}{l^2} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \\ \Rightarrow \mathbf{K}_{\text{Stick}}^{LM,LM} &= \boldsymbol{\Lambda}_T \mathbf{T}_\Delta^T + \frac{\lambda_T}{l} \left( \mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \frac{g_T}{l} (\mathbf{N}_0 \mathbf{N}_0^T) - \frac{g_N}{l} (\mathbf{N}_0 \mathbf{T}_0^T + \mathbf{T}_0 \mathbf{N}_0^T) \right) \end{aligned} \quad (45)$$

Tangential contribution (2 next terms) in slip case:

$$\delta \mathbf{X}^T \left[ \mu \operatorname{sgn}(\lambda_T) \mathbf{T}_\Delta \boldsymbol{\Lambda}_N^T + \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) \right] \Delta \mathbf{X}^T \quad (46)$$

$$\Rightarrow \mathbf{K}_{\text{Slip}}^{LM,LM} = \lambda_T \left( \frac{1}{l} (\mathbf{N}_s \mathbf{N}_0^T + \mathbf{N}_0 \mathbf{N}_s^T - \mathbf{T}_0 \mathbf{T}_s^T) - \frac{g_N}{l^2} (\mathbf{N}_0 \mathbf{T}_0^T + 2 \mathbf{T}_0 \mathbf{N}_0^T) \right) + \mu \operatorname{sgn}(\lambda_T) \mathbf{T}_\Delta \boldsymbol{\Lambda}_N^T \quad (47)$$

Lagrange multipliers contribution (2 last terms):

$$\delta \mathbf{X}^T \boldsymbol{\Lambda}_N \Delta \mathbf{X}^T \mathbf{N} + \delta \mathbf{X}^T \boldsymbol{\Lambda}_T \Delta \mathbf{X}^T \mathbf{T}_\Delta \quad (48)$$

$$\Rightarrow \mathbf{K}_{LM}^{LM,LM} = \boldsymbol{\Lambda}_N \mathbf{N}^T + \boldsymbol{\Lambda}_T \mathbf{T}_\Delta^T \quad (49)$$

Finally, the global stiffness is computed by adding the different stiffnesses depending on the method chosen and the contact state. Note that there is very few differences between the cases and they can be combined to be simpler to implement.

### 3.1.3 Tangent force predictor/corrector

When the contact is sliding, energy is dissipated and the tangential force can not be obtained by differentiating the energy. Unlike 1D plasticity, the contact does not need to keep a record of the history because sliding gaps (equivalent to plastic deformations) are naturally cumulated in the displacements. So, at each iteration, if the slave is in contact, a trial tangential force  $T_T^{\text{trial}}$  is directly computed from the tangential gap or thanks to Lagrange multiplier:

$$T_T^{\text{trial}} = \varepsilon_T g_T \quad \text{or} \quad T_T^{\text{trial}} = \lambda_T \quad (50)$$

This force is compared with the maximum admissible according to the Coulomb law and corrects it if necessary:

$\text{if }  T_T^{\text{trial}}  \leq \mu T_N$ $\text{Stick case, } T_T = T_T^{\text{trial}}$ $\text{else}$ $\text{Slip case, } T_T = \mu T_N \text{sgn}(T_T^{\text{trial}})$
---

### 3.1.4 Solution procedure and Convergence

**Contact state** The implementation requires to decide when to update the contact state (check for new contact or check if the contact is lost). The natural way is to do it at each Newton-Raphson iteration, but it leads to oscillations as the solid is not always converging linearly in displacements (the contact is lost because the slave oscillates across the interface). Thus, another solution is to update the contact state only when the solid has converged. This does not prevent all oscillations because the contact state update can lead to enable the contact for a certain node and disable it for its neighbour and, once the solid is converged, the contact state update switches again between those two nodes. An idea is to moreover allow the contact state to change only for one slave at a time. This would probably give very good results but would lead to many solid iterations which can be very expensive. Although, with friction, stable solutions are multiple and this procedure would lead to the wrong one.

The solution that has been selected is to allow new contact at each Newton step but allow contact loss only for nodes that were in contact at previous time-step (this prevents oscillations during the Newton procedure). Once converged, the contact state is checked and there, any contact can be disabled if necessary.

Coupled with a time-step cutting (when not converged after a certain number of iterations) this procedure seems quite good.

**Stick at first iteration** Just after the time-step update, the first Newton-Raphson iteration is initialised with the displacement of the boundary nodes which displacement is controlled and thus, the stiffness is computed. After this iteration, all the nodes are moved according to the stiffness. Potentially, many nodes will move with a quite large amplitude during this first step, so, the choice has been made to force to stick in order to have a regular stiffness and fix the nodes in contact.

**Contact oscillations around master nodes** Due to the non-regularity of the master surface, some oscillations can appear when a slave is around a master node. This is due to the fact that the converged state on both edges is inside the body (concave surface) so, the slave tries to reach it and is considered in contact with the other edge. And the same happens with the second edge. The solution would be to detect such an oscillation and stick the slave on the master. As detecting this is not obvious and has not been a major objective, the choice has been made to allow only one change of segment by time-step.

## 3.2 Mortar method

Mortar method uses surfaces as slave entities and the contact is satisfied in a weak way, allowing local interpenetration. In our case, the slave surfaces are the segments of the slave contact interface but with quadratic geometric interpolation, it would be a parametrised curve. The enforcement will only be developed for Lagrange multipliers. Most of the theory will not be detailed here and is picked

from Fischer and Wriggers<sup>7</sup> because too little time was available to develop it more.

### 3.2.1 Force and Stiffness computation

Starting from the total energy variation integral (9), a standard Gauss-Legendre quadrature is used with  $N_{GP}$  Gauss points:

$$\begin{aligned}
 \delta\Pi_{\text{contact}} &= \delta\Pi_N^{LM} \\
 &= \int_{\mathcal{S}^s} \left( p_N^{LM}(\xi) \delta g_N(\xi) + \delta p_N^{LM}(\xi) g_N(\xi) \right) d\xi + \int_{\mathcal{S}^s} p_N^{LM}(\xi) g_N(\xi) \delta d\xi \\
 &= \int_{\mathcal{S}^s} \left( \lambda_N(\xi) \delta g_N(\xi) + \delta \lambda_N(\xi) g_N(\xi) \right) d\xi + \int_{\mathcal{S}^s} \lambda_N(\xi) g_N(\xi) \delta d\xi \quad (51) \\
 &\simeq \sum_{\substack{\text{s being} \\ \text{slave segments}}} \sum_{i=1}^{i=N_{GP}} \omega_i \left( \lambda_N^s A^s \delta g_N^s + \delta \lambda_N^s A^s g_N^s + \lambda_N^s \delta A^s g_N^s \right) (x_i)
 \end{aligned}$$

As for Node to segment, the integration area associated to each Gauss point is put into the Lagrange multiplier. Also, the contact condition is enforced on each entity so each term of the sum needs to be null.

$$\delta\Pi_{\text{contact}}^s = \sum_{i=1}^{i=N_{GP}} \omega_i \left( \lambda_N^s(x_i) \delta g_N^s(x_i) + \delta \lambda_N^s(x_i) g_N^s(x_i) \right) \quad (52)$$

The formulation can be expressed at each Gauss-point and summed afterwards:

$$\delta\Pi_{\text{contact}}^{GP_i} = \lambda_N^s(x_i) A^s(x_i) \delta g_N^s(x_i) + \delta \lambda_N^s(x_i) A^s(x_i) g_N^s(x_i) + \lambda_N^s(x_i) \delta A^s(x_i) g_N^s(x_i) \quad (53)$$

Taking the derivative relatively to each degree of freedom is not straight forward as the Gauss-points positions are interpolated from the nodal ones, so, the first step is to define and use the shape functions for each field. Denoting  $\xi$  the curvilinear abscissa along the master surface and  $\zeta$  the one along the slave, the standard linear

shape functions that interpolate both geometry and displacements are:

$$\begin{aligned}
 N_1^s(\zeta) &= 1 - \zeta & \text{and} & & N_2^s(\zeta) &= \zeta \\
 N_1^m(\xi) &= 1 - \xi & \text{and} & & N_2^m(\xi) &= \xi
 \end{aligned}
 \tag{54}$$

To interpolate the Lagrange multipliers field, two kinds of linear shape functions have been tried, the standard ones and special ones as advised by Wohlmuth<sup>8</sup>:

$$\begin{aligned}
 \phi_1^s(\zeta) &= 1 - \zeta & \text{and} & & \phi_2^s(\zeta) &= \zeta \\
 \phi_1^s(\zeta) &= 2 - 3\zeta & \text{and} & & \phi_2^s(\zeta) &= -1 + 3\zeta
 \end{aligned}
 \tag{55}$$

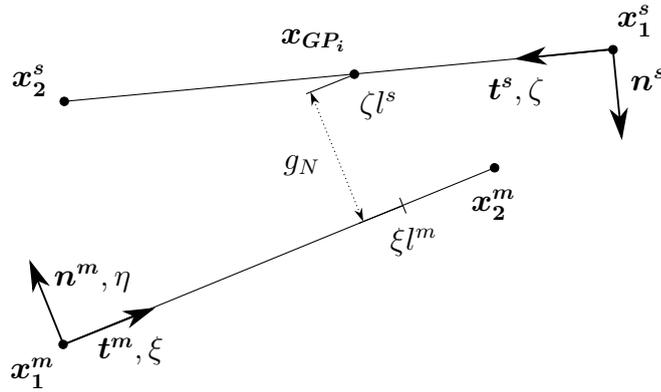


Figure 5: Mortar geometrical parameters

Again, the slave Gauss-point  $x_i$  is isolated from the body and everything is evaluated at this point so, from now on, all the variables are computed from this point of view.

After some geometry and variational calculus, and using the following vectors,

$$\mathbf{X} = \begin{Bmatrix} \mathbf{x}_1^s \\ \mathbf{x}_2^s \\ \mathbf{x}_1^m \\ \mathbf{x}_2^m \\ \lambda \end{Bmatrix} \quad \mathbf{F} = \begin{Bmatrix} \mathbf{f}_1^s \\ \mathbf{f}_2^s \\ \mathbf{f}_1^m \\ \mathbf{f}_2^m \\ \mathbf{f}_\lambda \end{Bmatrix}
 \tag{56}$$

$$\begin{aligned}
 \Phi_N &= \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \phi_1^s(\zeta) \\ \phi_2^s(\zeta) \end{Bmatrix} & N_N &= \begin{Bmatrix} -N_1^s(\zeta)\mathbf{n}^m \\ -N_2^s(\zeta)\mathbf{n}^m \\ N_1^s(\zeta)\mathbf{n}^m \\ N_2^s(\zeta)\mathbf{n}^m \\ \mathbf{0} \end{Bmatrix} & L_T &= \begin{Bmatrix} -\mathbf{t}^s \\ \mathbf{t}^s \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \\
 N_A &= \begin{Bmatrix} N_1^s(\zeta)\mathbf{t}^m \\ N_2^s(\zeta)\mathbf{t}^m \\ -N_1^s(\zeta)\mathbf{t}^m \\ -N_2^s(\zeta)\mathbf{t}^m \\ \mathbf{0} \end{Bmatrix} & L_N &= \begin{Bmatrix} -\mathbf{n}^s \\ \mathbf{n}^s \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} & N &= \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\mathbf{n}^s \\ \mathbf{n}^s \\ \mathbf{0} \end{Bmatrix}
 \end{aligned} \tag{57}$$

everything can be put into matrix form and grouped to form the force vector (residual) and the stiffness matrix:

$$\mathbf{F} = \omega (\lambda_{GP} l^s \mathbf{N}_N + g_N l^s \Phi_N + g_N \mathbf{L}_N) \tag{58}$$

$$\begin{aligned}
 \mathbf{K} &= \omega \left( l^s \Phi_N \mathbf{N}_N^T + g_N \Phi_N \mathbf{L}_T^T + \lambda_{GP} \mathbf{L}_T \mathbf{N}_N^T \right. \\
 &\quad \left. + g_N \mathbf{L}_T \Phi_N^T + \frac{g_N \lambda_{GP}}{l^s} \mathbf{L}_N \mathbf{L}_N^T + l^s \mathbf{N}_N \Phi_N^T \right. \\
 &\quad \left. + \frac{\lambda_{GP} l^s}{l^m} \left( \mathbf{N} \mathbf{N}_A^T - \frac{g_N}{l^s} \mathbf{N} \mathbf{N}^T + \mathbf{N}_A \mathbf{N}^T \right) + \lambda_{GP} \mathbf{N}_N \mathbf{L}_T^T \right)
 \end{aligned} \tag{59}$$

where  $\lambda_{GP} = \lambda_1 \phi_1^s(\zeta) + \lambda_2 \phi_2^s(\zeta)$

Both are valid for a single Gauss-point but, to obtain the total forces/stiffness associated with a certain slave element, those contributions need to be added (one contribution per Gauss-point). The problem is that the master surface corresponding to each Gauss-point can be different (even for Gauss-point from the same slave) and so, it is not just a sum but a kind of assembling process as the  $x_1^m$  and  $x_2^m$  of each Gauss-point can be different.

### 3.2.2 Contact detection

As the contact is imposed in an integral way, the contact detection needs to be consistently defined. When the slave is not in contact, the gap function corresponding

is:

$$g_N^s = \sum_{i=1}^{i=N_{GP}} \omega_i g_N(x_i) \quad (60)$$

This gap is checked in order to determine if the slave is considered in contact or not.

As for Node to Segment (Section 3.1.1),  $g_N^s$  is virtually null when the contact is enforced so it can't be used to determine if the contact is lost. The contact force is used instead of the gap:

$$f_N^s = \sum_{i=1}^{i=N_{GP}} \omega_i \lambda_N(x_i) \quad (61)$$

### 3.2.3 Solution procedure and Convergence

Fischer and Wriggers<sup>7</sup> advise to check for contact before the Newton-Raphson procedure. This leads to converged states where the contact state is wrong (some slaves are sucking the contact surfaces together). To avoid this issue, the contact state is checked at each Newton-Raphson iteration.

Also, on the same slave segment, some Gauss-points can be such that they can't be projected onto a master (no gap can be computed). In that case, the Gauss-point contribution is just ignored.

### 3.2.4 Stability

Defined this way, the formulation does not fulfil the stability condition described by Brezzi and Fortin<sup>3</sup>. To do so, Wohlmuth<sup>8</sup> advises to interpolate the Lagrange multipliers with constant shape function for the slave elements that are at the limit of the contact. In other words, the slave segments that have one neighbour not in contact should use only one Gauss-point. This is neither easy to implement nor computationally cheap so it has not been experimented.

### 3.3 Time integration

To use the contact implementation in dynamics, the contact algorithm must be consistently introduced into the time integration method. As the solver that has been used implements the generalised- $\alpha$  method, its behaviour with contact constrain is studied. To simplify the explanations, this will be done for the Node to Segment formulation.

Basically, the generalised- $\alpha$  method for time integration consists in evaluating the equation at an intermediate time-step (denoted  $t_\alpha$ ) and get back the solution at  $t_{n+1}$  by extrapolation. In the context of Newton-Raphson solution procedure, it means that the residual and stiffness are computed using a displacement (or any other degree of freedom) that is interpolated between the  $t_n$  configuration and  $t_{n+1}$  one. So, by applying this method to enforce the contact, it gives the following issue: if there is no contact at  $t_n$  but a contact at  $t_\alpha$ , the constrain is such that there is no interpenetration at  $t_\alpha$  but, as the displacements are extrapolated, there is penetration at  $t_{n+1}$ . At the next step, the contact will be satisfied at the new  $t_\alpha$  and after extrapolation, the node will be sent out of the body.

The contact is effectively enforced at  $\alpha$  timesteps but in-between, the slave oscillates across the master surface, creating non-physical mechanical excitement.

To prevent this phenomenon, the contact force and residual are computed using the  $t_{n+1}$  state even if the solid solver uses the  $\alpha$  configuration.

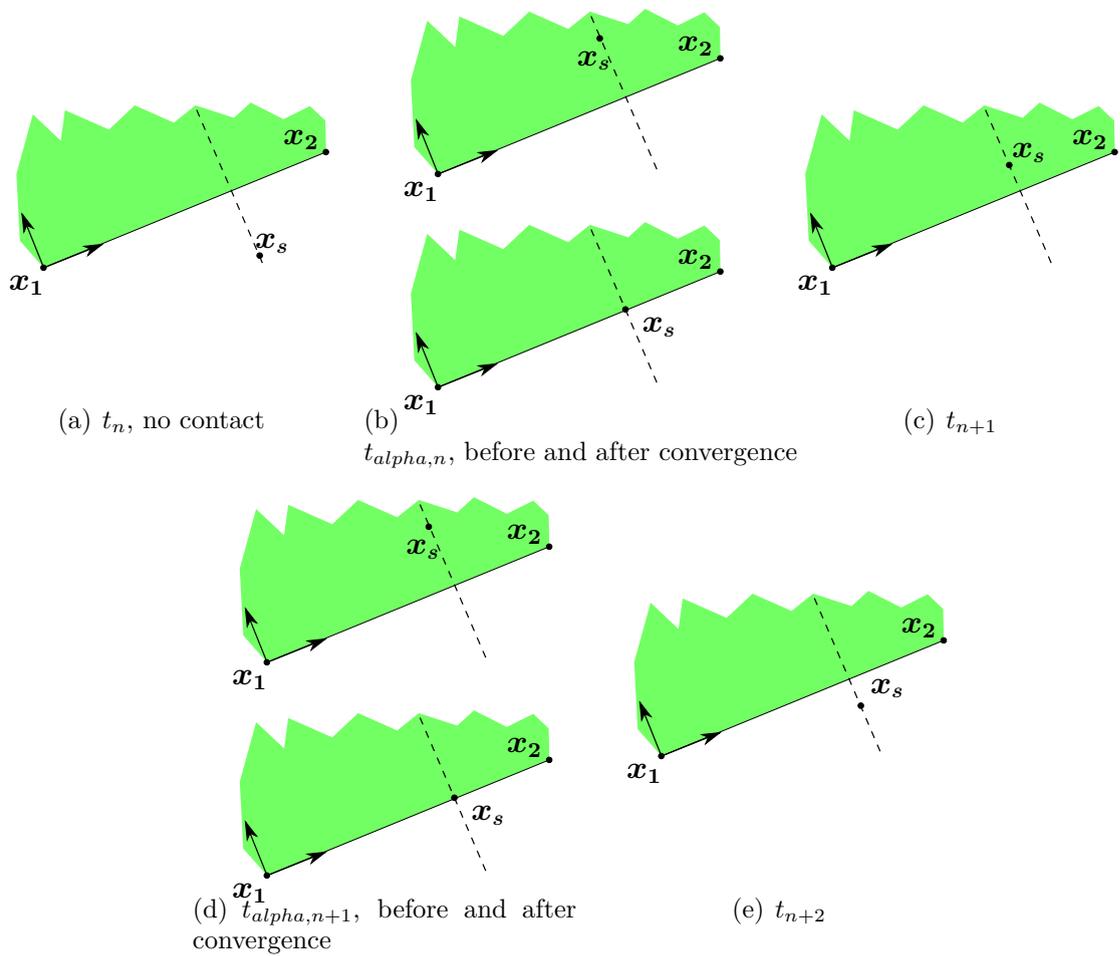


Figure 6: Contact oscillations

## 4 Applications

All the contour-plots are presenting Von Mises Stresses.

### 4.1 Validation: Hertz contact

Due to the unavoidable non-linearity of contact mechanics, analytical solutions are hard to derive. Hertz contact has been developed for a long time from Hertz<sup>9</sup> to Smith<sup>10</sup> by considering small deformations and interpenetration. The following example will be based on what Boresi and Schmidt<sup>11</sup> present.

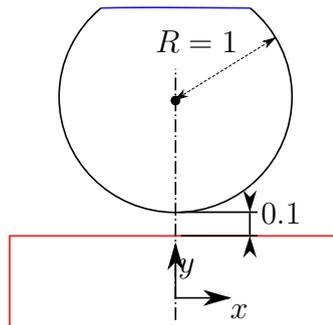


Figure 7: Hertz contact : Cylinder/Plane

The cylinder of diameter 1 mm is moved onto a rigid block (red) by translating its top boundary (blue) in order to achieve a 0.005 mm interpenetration. The material of the cylinder is a standard steel with  $\nu = 0.29$  and  $E = 210$  MPa. A 2D model is used with plane strain.

#### 4.1.1 Frictionless compression

In this case, no friction is considered. Boresi and Schmidt<sup>11</sup> propose an expression for the stresses on the  $\mathbf{y}$  axis:

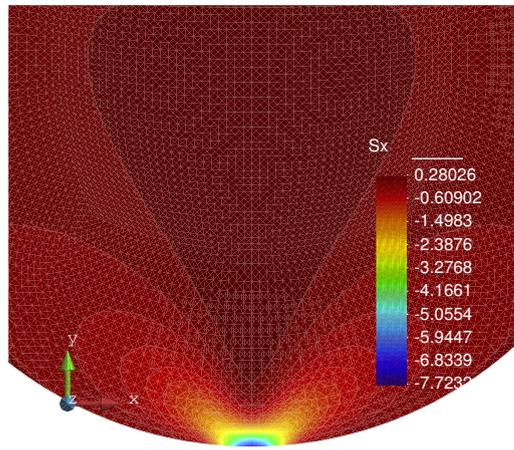
$$\begin{aligned}\sigma_{xx} &= -\frac{b}{\Delta} \left( \frac{\left( \sqrt{1 + \left(\frac{z}{b}\right)^2} - \frac{z}{b} \right)^2}{\sqrt{1 + \left(\frac{z}{b}\right)^2}} \right) \\ \sigma_{xx} &= -\frac{b}{\Delta} \left( \frac{1}{\sqrt{1 + \left(\frac{z}{b}\right)^2}} \right) \\ \sigma_{zz} &= -2\nu \frac{b}{\Delta} \left( \sqrt{1 + \left(\frac{z}{b}\right)^2} - \frac{z}{b} \right)\end{aligned}\tag{62}$$

where  $2b$  is the contact area computed using geometrical consideration by  $b = \sqrt{\frac{2p\Delta}{\pi}}$ ,  $\Delta = 2R \frac{(1-\nu^2)}{E}$  and  $p$  is the total force applied, measured by the solver.

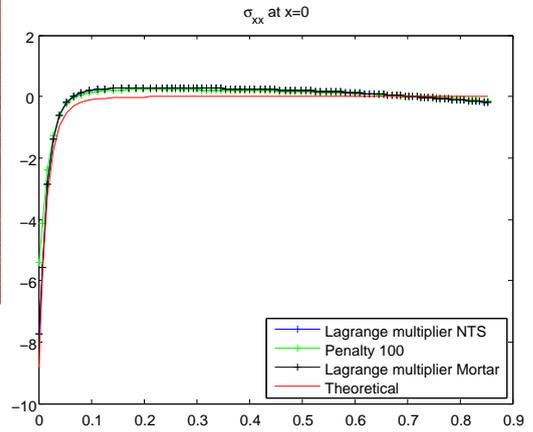
This case has been done for node to segment with Lagrange multiplier, node to segment with penalty ( $\varepsilon_N = 100N/mm$ ) and mortar with Lagrange multiplier.

Figure 8 shows the stress distribution (obtained using node to segment with Lagrange multiplier method) in the cylinder. For each stress and each method, the evolution along the  $\mathbf{y}$  axis is compared with the theoretical value.

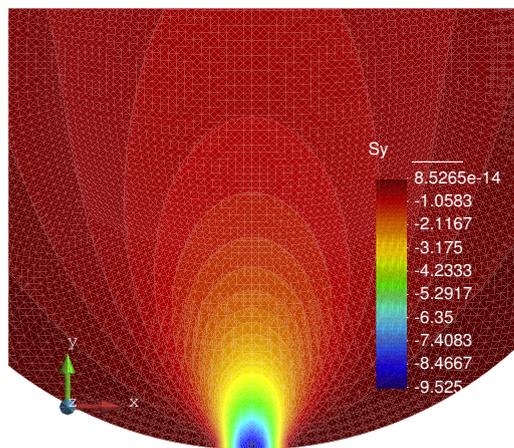
Node to segment with Lagrange multiplier and mortar (with 3 Gauss points) with Lagrange multiplier are giving exactly the same results in this simple case while node to segment with Penalty has a softer behaviour. Note that the theoretical curve is computed using (62) and measuring  $p$  using node to segment with Lagrange multiplier simulation.



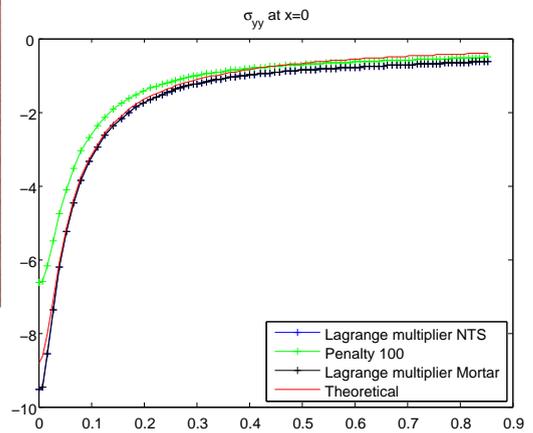
(a)  $\sigma_{xx}$



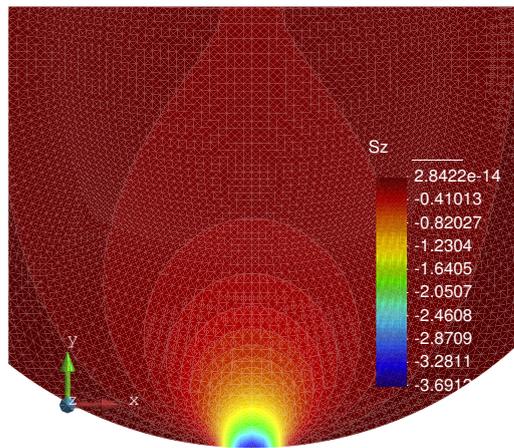
(b)  $\sigma_{xx}$  along  $x = 0$



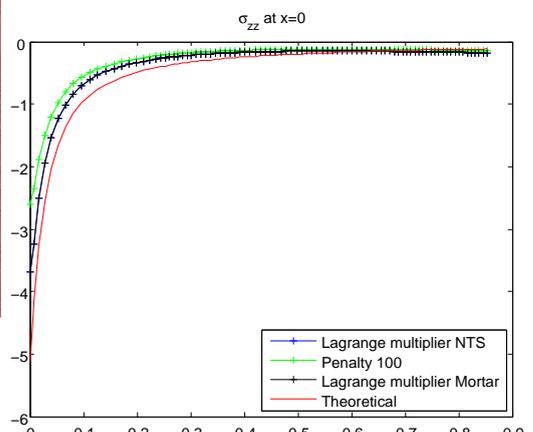
(c)  $\sigma_{yy}$



(d)  $\sigma_{yy}$  along  $x = 0$



(e)  $\sigma_{zz}$



(f)  $\sigma_{zz}$  along  $x = 0$

Figure 8: Hertz Stresses

#### 4.1.2 Slide with friction

In this case, a friction coefficient  $\mu = 0.333$  is considered. In addition to the compression, the cylinder is sheared (translation of the blue boundary) to have sliding. Boresi and Schmidt<sup>11</sup> also propose a theoretical solution for this case (a slight correction has been done):

$$\begin{aligned}\sigma_{xx} &= -\frac{b}{\pi\Delta} \left( y \left( \frac{b^2 + 2y^2 + 2x^2}{b} \phi_1 - \frac{2\pi}{b} - 3x\phi_2 \right) \right. \\ &\quad \left. + \mu \left( (2x^2 - 2b^2 - 3y^2)\phi_2 + \frac{2\pi x}{b} + 2(b^2 - x^2 - y^2)\frac{x}{b}\phi_1 \right) \right) \\ \sigma_{yy} &= -\frac{b}{\pi\Delta} \left( y(b\phi_1 - x\phi_2) + \mu y^2\phi_2 \right) \\ \sigma_{zz} &= -\frac{2\nu b}{\pi\Delta} \left( y \left( \frac{b^2 + x^2 + y^2}{b} \phi_1 - \frac{\pi}{b} - 2x\phi_2 \right) \right. \\ &\quad \left. + \mu \left( (x^2 - b^2 - y^2)\phi_2 + \frac{\pi x}{b} + (b^2 - x^2 - y^2)\frac{x}{b}\phi_1 \right) \right) \\ \sigma_{xy} &= -\frac{b}{\pi\Delta} \left( y^2\phi_2 + \mu \left( (b^2 + 2x^2 + 2y^2)\frac{y}{b}\phi_1 - 2\pi\frac{y}{b} - 3xy\phi_2 \right) \right)\end{aligned}$$

with

$$\begin{aligned}\phi_1 &= \frac{\pi(M + N)}{MN\sqrt{2MN + 2x^2 + 2y^2 - 2b^2}} & \phi_2 &= \frac{\pi(M - N)}{MN\sqrt{2MN + 2x^2 + 2y^2 - 2b^2}} \\ M &= \sqrt{(b + x)^2 + y^2} & N &= \sqrt{(b - x)^2 + y^2}\end{aligned}$$

Figures 9 to 12 present the results (the contourplots are obtained using node to segment and Lagrange multipliers for both normal and tangential forces) and compare it with the theory. A large penalty parameter  $\varepsilon_T = 1000N/mm$  has been chosen to avoid errors.

Considering the big approximations done by the theoretical Hertz contact, the results are close enough to be considered valid. Moreover, the lack of mesh refinement around the contact zone may lead to rough contact stress computation.

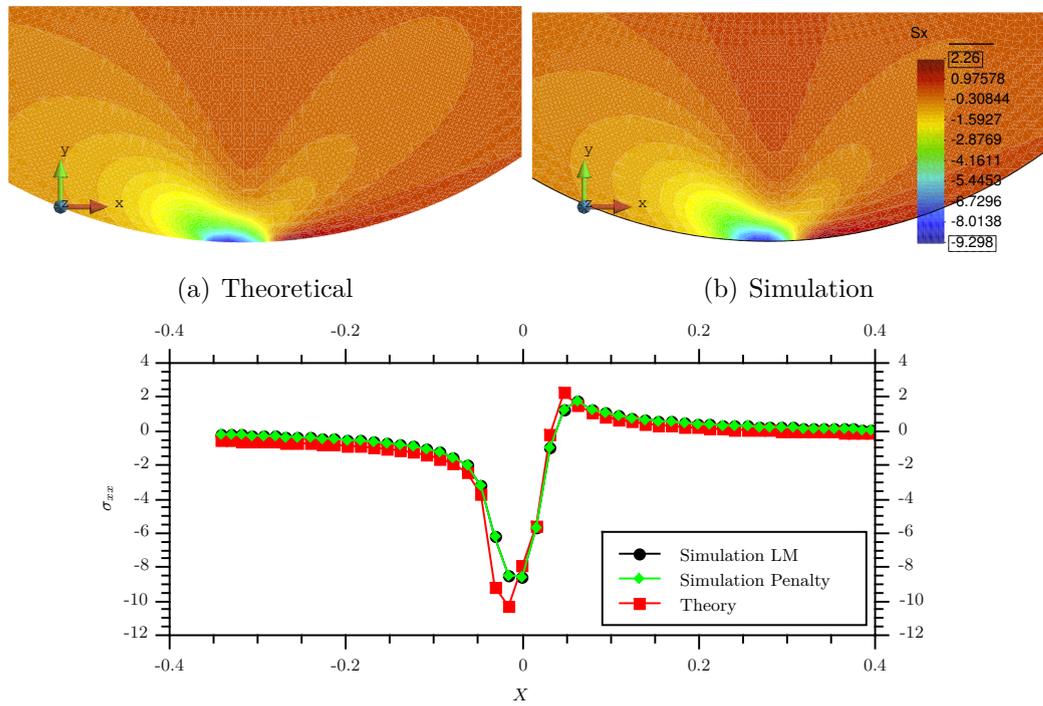


Figure 9:  $\sigma_{xx}$  around the contact interface

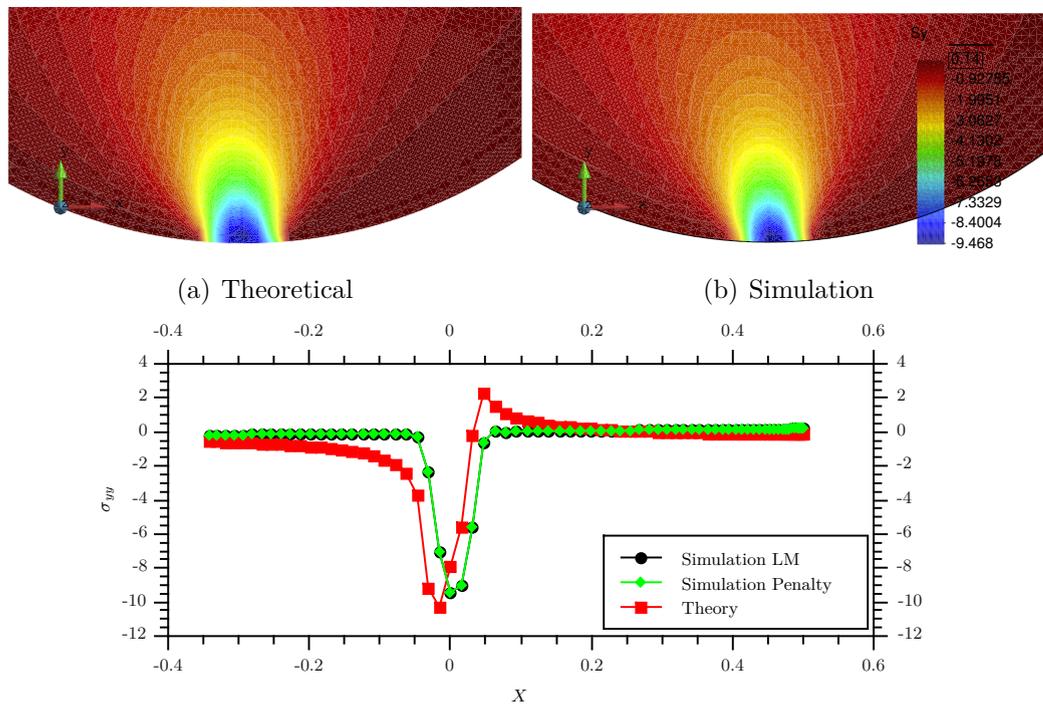


Figure 10:  $\sigma_{yy}$  around the contact interface

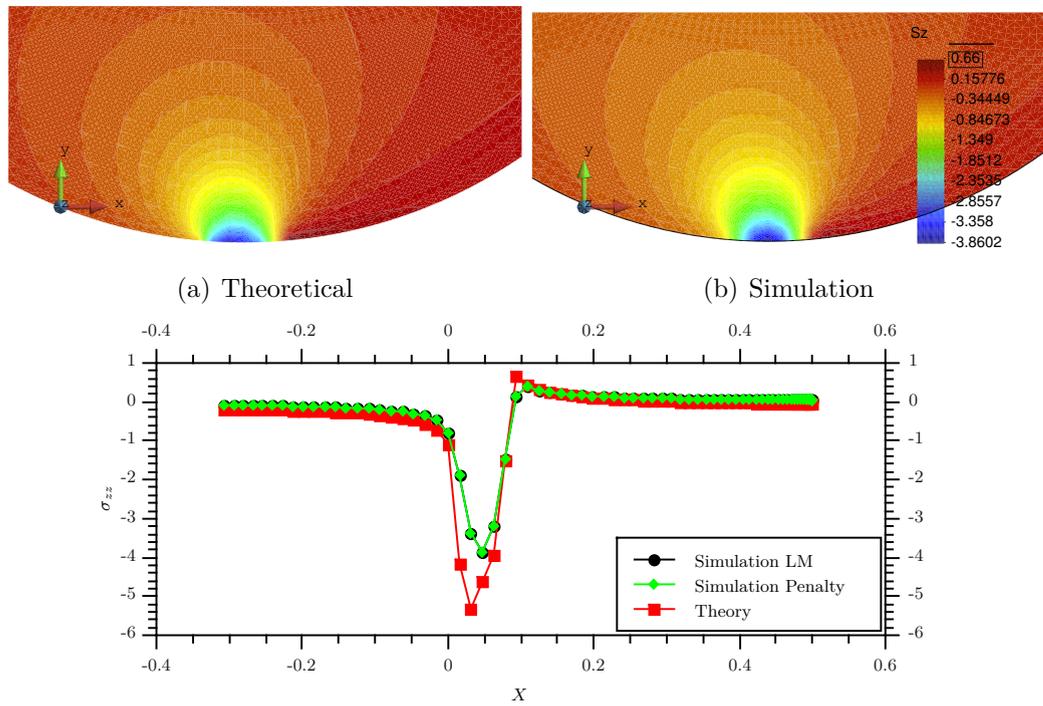


Figure 11:  $\sigma_{zz}$  around the contact interface

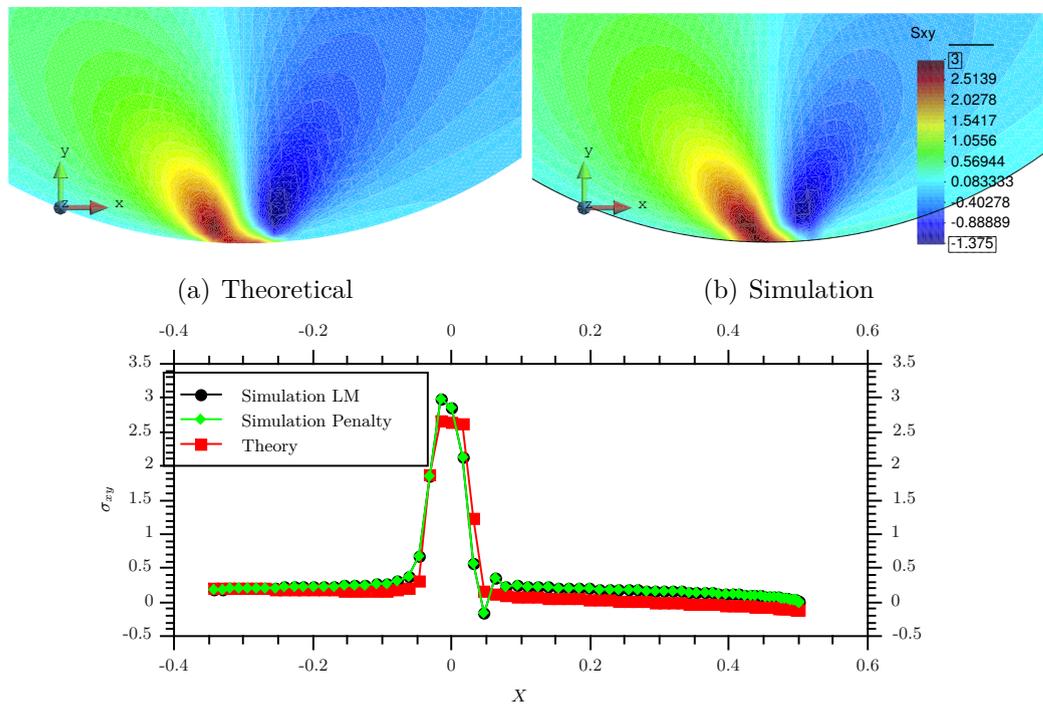


Figure 12:  $\sigma_{xy}$  around the contact interface

## 4.2 Sliding block

This problem is proposed by Fischer and Wriggers<sup>7</sup>. To simplify, the length are written in  $m$  and the pressure in  $Pa$ . The black boundary is fixed while the red one is first moved down  $0.2m$  and slid  $2m$  (without friction) to the left. The master surface is the green one and the yellow is the slave. The Poisson ratio is  $\nu = 0$  and the small object is  $2.1 \cdot 10^3$  stiffer than the big one.

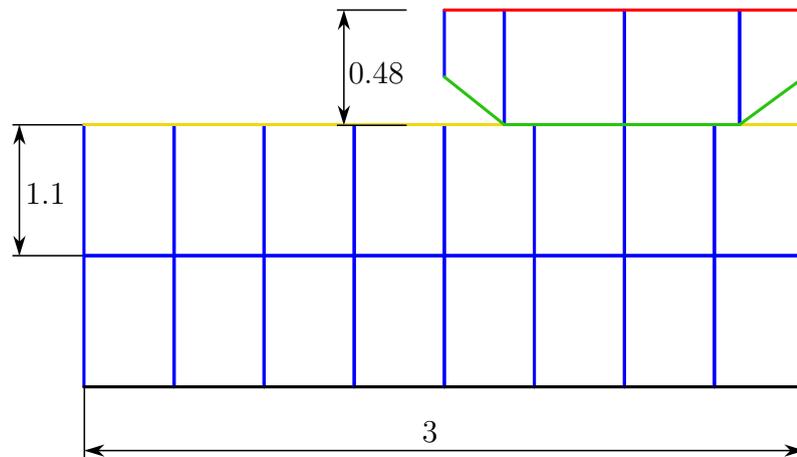
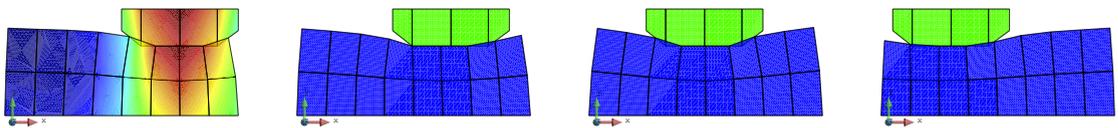


Figure 13: Sliding Block: Problem statement

Node to segment (with Lagrange Multiplier)



Mortar

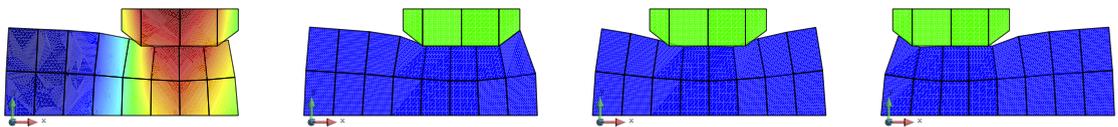


Figure 14: Sliding Block: NTS and Mortar results

As expected, the node to segment formulation prevents the slave nodes to go through the master surface but master nodes can go through the slave surface. Mortar formulation gives some more tricky results. Even though this behaviour has not been emphasised by Fischer and Wriggers<sup>7</sup>, the slave segments are trying to have masters parallel to them. This gives the impression that some of the slave are pulling the master surface. Indeed, some of the slave nodes are acting this way but this is not an issue as long as the overall action of a slave segment (the weighted sum of the contributions of each Gauss point) is pushing the master away.

This is not shown on the figures but the use of the non-standard shape functions (proposed by Wohlmuth<sup>8</sup>) to interpolate the Lagrange multipliers gives a better convergence.

### 4.3 Bouncing balls

Those examples involves dynamics. The principle is simply to drop an elastic ball of  $60mm$  diameter on a solid ground from  $300mm$ . A standard nearly incompressible rubber is considered (nearly incompressible Neo-Hooke) with a density  $\rho = 1.1 \cdot 10^{-3}g/mm^3$ , a bulk modulus  $K = 2500MPa$  and a shear modulus  $\mu = 0.6MPa$ . The weight is simulated with a body force of  $-1 \cdot 10^{-2}N/kg/mm^3$ . The time integration is performed with a generalised- $\alpha$  method and no numerical damping.

#### 4.3.1 Full elastic ball

In this example, the ball is an homogeneous of rubber block. The ball is discretised with a structured mesh using 300 quadrangles elements.

During the simulation, the solver needs to split time-steps after the contact in order to capture the mechanical oscillation of the ball (down to  $0.05ms$  timesteps).

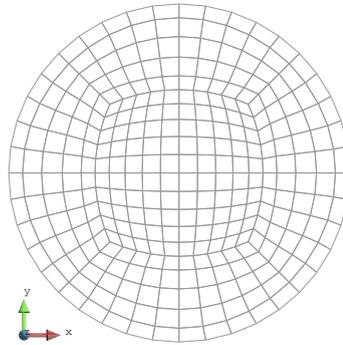


Figure 15: Bouncing ball: free oscillations after one bounce,  $dt = 0.25ms$

This is due to the large stiffness of the material that implies high frequencies vibration modes. Thus, the computation is expensive and has not been carried for more bounces.

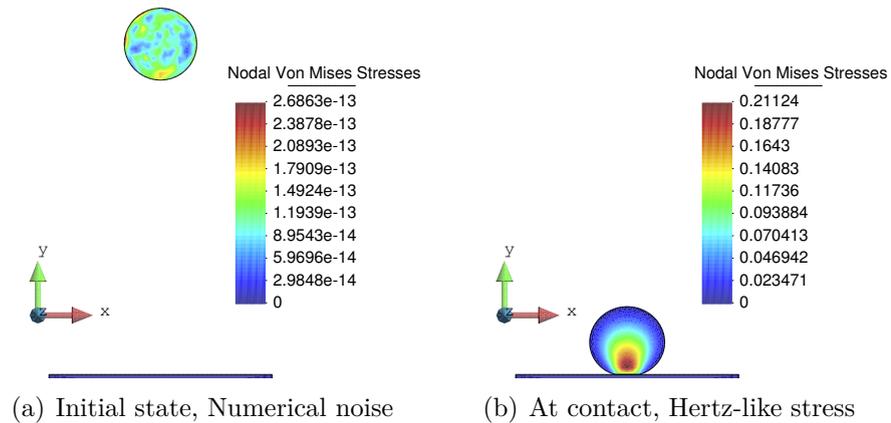


Figure 16: Bouncing ball: Von Mises Stresses through time

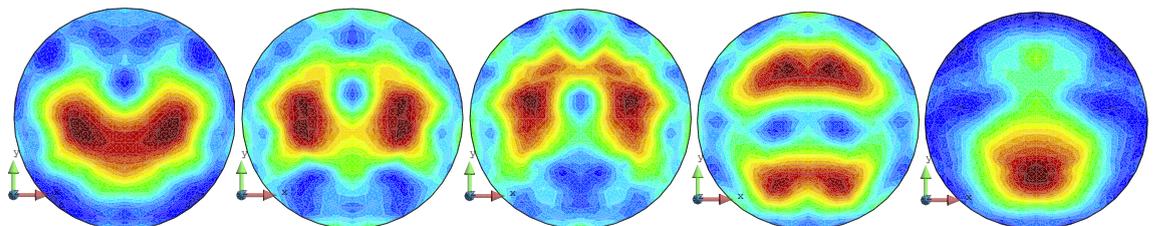


Figure 17: Bouncing ball: free oscillations von Mises Stresses after one bounce  $dt = 0.25ms$

Both mortar method and node to segment work on this case. An interesting comparison would be to check if the energy is conserved. Unfortunately, the solver

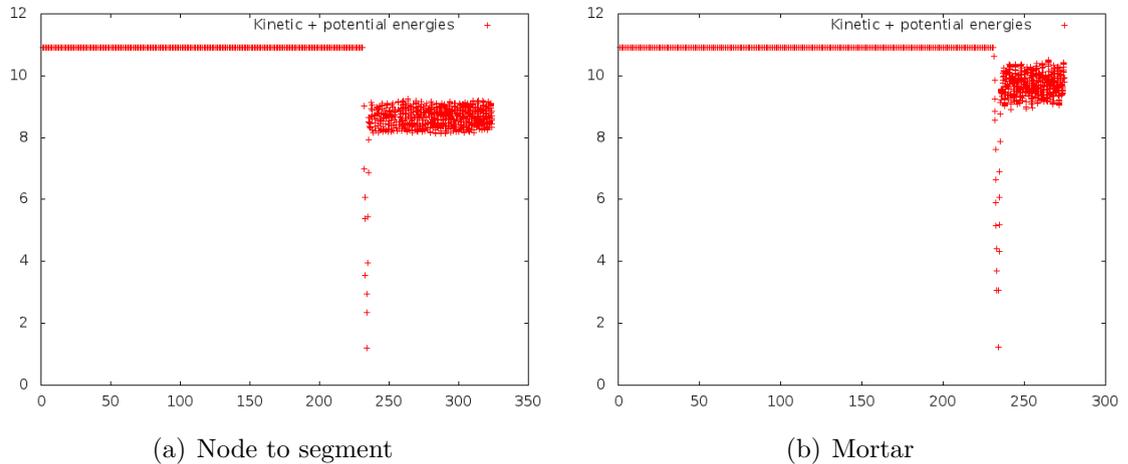


Figure 18: Bouncing ball: kinetic and potential energies evolution ( $mJ$  vs.  $ms$ )

does not provide a way to compute the elastic energy so only the kinetic and potential energies can be plotted on Figure 18. After the first bounce around  $240\text{ ms}$ , the energy is not constant because of the self oscillations which transfer mass and potential energy across the ball. However, the energy after the bounce is higher for the mortar method. This can have two explanations: either the node to segment method dissipated more energy than the mortar method, or it transfers more energy into the ball than the mortar method.

To be able to study more easily the ball, the rubber has been replaced by a softer material with  $K = 0.1\text{MPa}$ , all the other parameters being unchanged. It give quite similar results but the energy after the bounce is less noisy:

Again, the lack of strain energy measurement does not allow to compare the methods properly.

### 4.3.2 Pressurised ball (Tennis ball)

This is another interesting problem where the ball is no more homogeneous but empty and filled with constant pressure gas. This roughly models a tennis ball. The  $3\text{mm}$ -thick ball is modelled with 640 quadrangles. The aim is to look at the influence

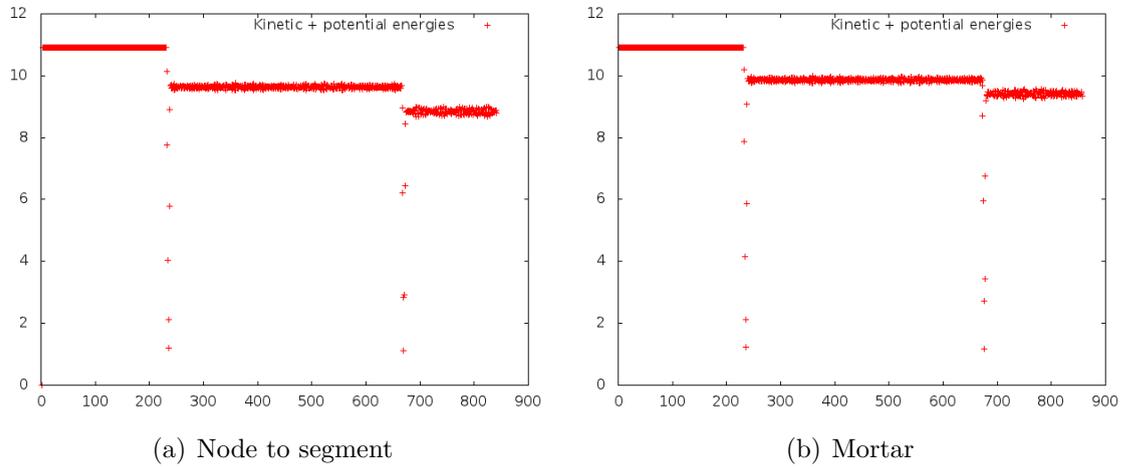


Figure 19: Softer ball: kinetic and potential energies evolution ( $mJ$  vs.  $ms$ )

of having a constant pressure inside the ball as one can argue that the gas-dynamics that happens inside is the only factor that makes the ball bounce.

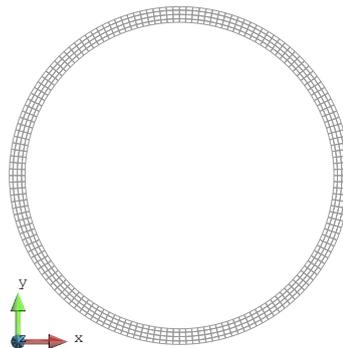


Figure 20: Tennis ball: mesh

First with nothing inside the ball (more exactly, no pressure difference between the exterior and the interior), the result is shown Figure 21 The empty ball is not

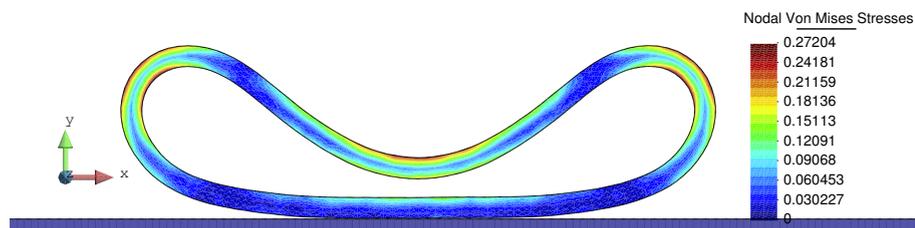


Figure 21: Empty tennis ball: not bouncing

bouncing and collapses under its own weight.

The inner pressure is chosen at 10 bars (real tennis balls pressure is between 2 and 10 bars). The first effect is to slightly inflate the ball. After the contact, the timestep is cut in order to capture the high-frequency waves travelling along the ball. Figure 22 shows the deformations of the pressurised tennis ball around

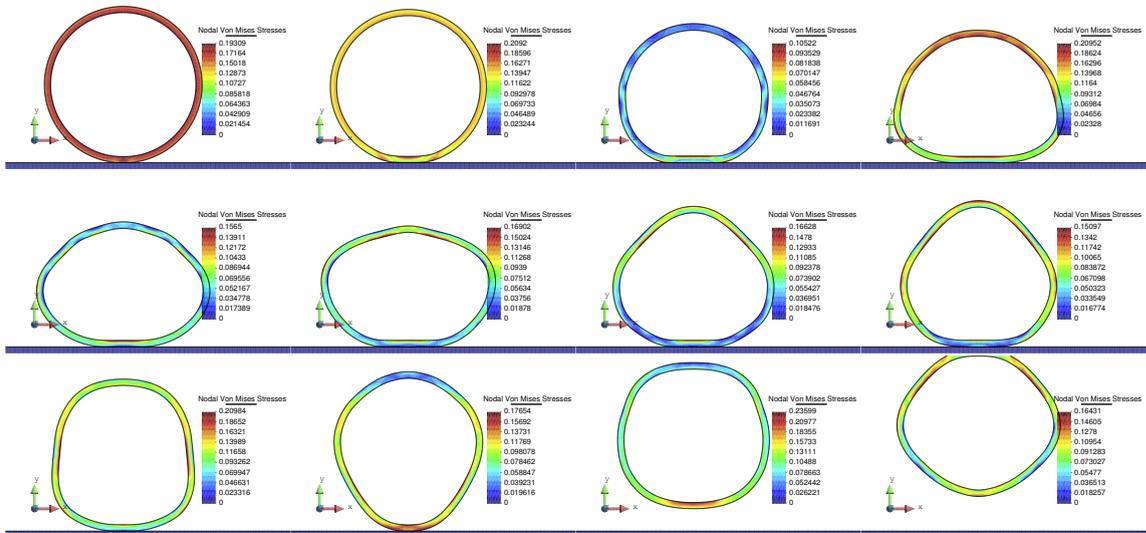


Figure 22: Tennis ball: von Mises Stresses during contact

the contact. First, the contact is just established. Next, the contact area grows and waves are propagating along the ball. The second line of images starts with a configuration where the top of the ball keeps falling while the bottom is bouncing up. At this stage, a dynamical approach for the inner fluid would probably change the behaviour. Finally, the ball travels upward while oscillating.

This simulation is quite computationally expensive and has only be done using node to segment formulation with Lagrange multipliers and in frictionless case.

This example shows the importance of the inner pressure of the ball. Nevertheless, the model is not accurate and does not consider either the fluid dynamics inside the ball or the fabric glued on top of real tennis balls.

## 4.4 Sliding elastic beam

This example has been designed to test the compliance of the contact formulation as the contact surfaces are not naturally parallel (all the previous examples involves curves contact surfaces, insuring that the contact surfaces are parallel). The black

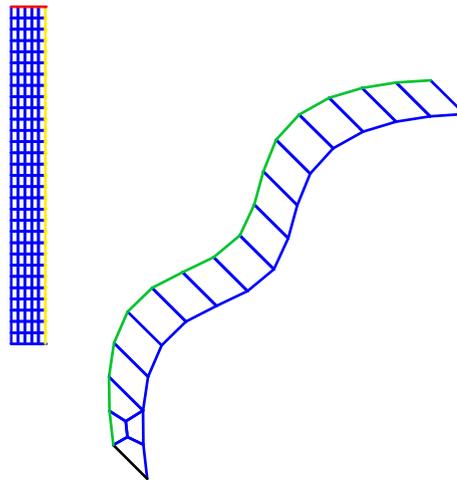


Figure 23: Sliding elastic beam: problem statement

face is fixed and the red one is translated to the right. The master surface is green while the slave is yellow. The slid beam has a 10 times smaller bulk modulus and 20 times smaller shear modulus that the curved one.

**Mortar case** This example should be solvable with mortar method if no friction is considered. But, when the simulation is started using this method, it fails when the first contact should happen. Indeed, by looking at each Newton-Raphon step, it can be seen that the contact is gained and converges but the converged state is not stable as the slave element needs to suck the surface to converge. This is due to the fact that achieving a null integral of the gap function requires at least one Gauss point to pull the master. The amount necessary to obtain equilibrium is too large.

Figure 24 first shows why the contact is considered active : the yellow dot being the

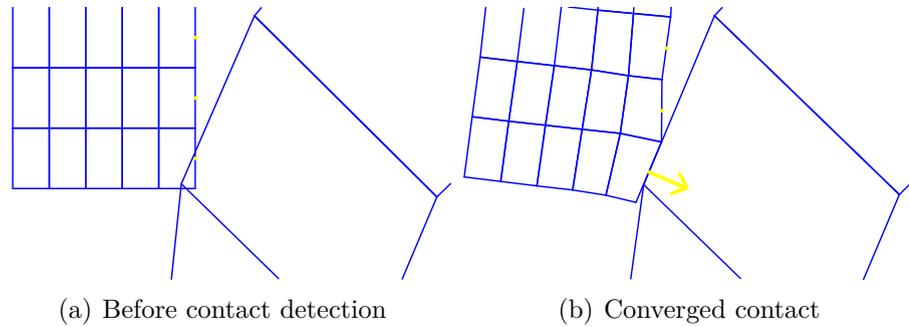


Figure 24: Sliding elastic beam: mortar contact issue

middle of the slave segment, the integral of the gap along it is such that the contact is detected. Once converged, the yellow vector is the total force on the slave surface which is clearly pulling the surfaces together.

However, if this first element in contact is forced to use only one Gauss point (which is equivalent to node to segment with the center of the segment as node), the contact works. Next the second element in contact is its neighbour (that can have several Gauss points) which is now (nearly) parallel to master and converges fine. However, when the first element is losing contact, the solver diverges.

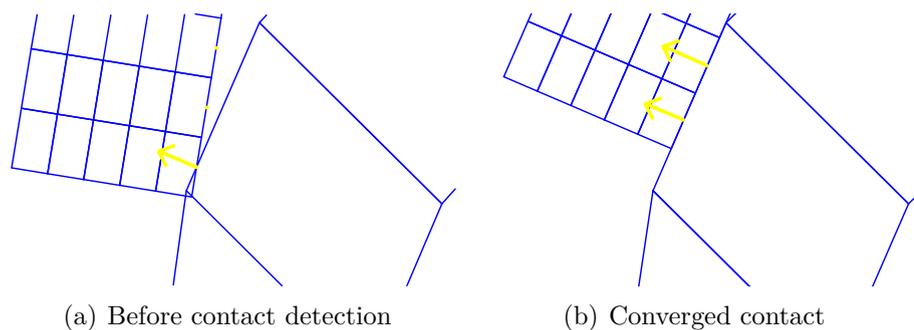


Figure 25: Sliding elastic beam: mortar issue solved

Figure 25 shows that if the first element is integrated with one Gauss point, the second mortar element, which is integrated with 2 or more Gauss points is converging to a valid state.

This is a good example of a non fulfilled stability condition and its consequences.

Obviously, here, the element is manually forced to use one Gauss point so the problem can't be solved but at least, it shows that an automatic enforcement would solve this kind of issues.

**Node to segment case** Using node to segment, the case can be solved even with friction ( $\mu = 0.5$ ). To fully test the procedure, Lagrange multiplier method has been used for both normal and tangential forces as it is the most accurate but less stable solution procedure.

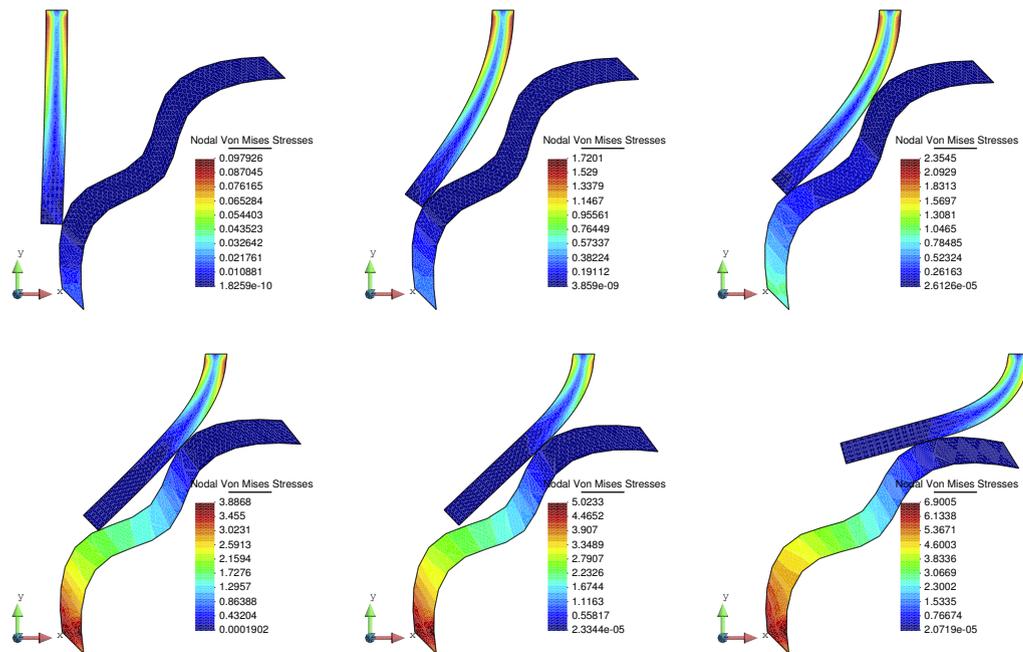


Figure 26: Sliding elastic beam: node to segment with friction evolution

Figure 26 shows the evolution of the case across time. First, the contact is gained and the beam is bended. Next, a second contact point is established and the curved object is deformed. Finally, the first contact point is lost and both solids are bended.

## 4.5 Active rolling wheel

This example is involving dynamics. It work thanks to friction so only node to segment methods will be used.

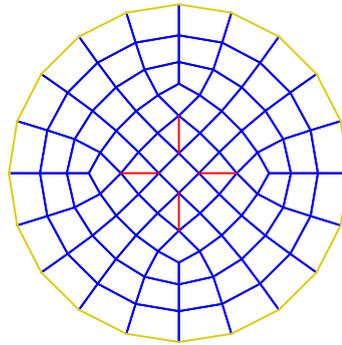


Figure 27: Rolling wheel: problem statement

The wheel is subjected to gravity. The green surface is the fixed master. The yellow is the slave. The red elements are pressure elements, they impose a constant pressure in their transversal direction which allow the wheel to spin (they are oriented such that it creates a moment that leads the wheel). The friction coefficient is  $\mu = 0.1$ . Figure 28 presents the case when the wheel is moved by a small couple : it is rolling

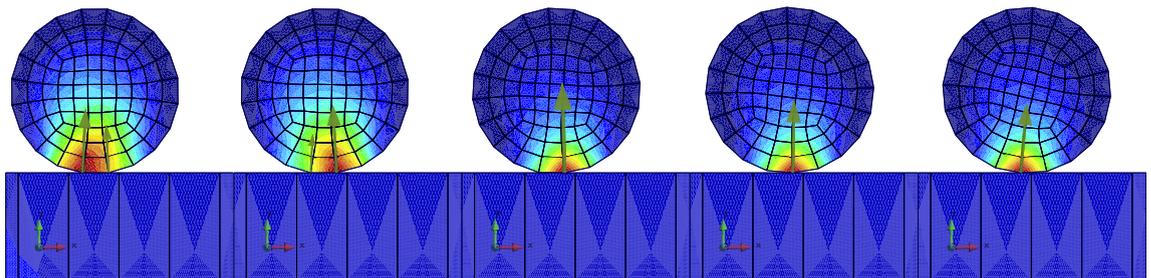


Figure 28: Wheel: rolling on the ground, small couple

on the ground, so at every instant, the wheel rotates around the contact point. The contact point is fixed relatively to the ground.

Figure 29 presents the case when the wheel is moved by a large couple : the leading force is too large and the contact does not stick. The wheel is sliding on the ground. All the friction forces are at their maximum (10% of the normal force).

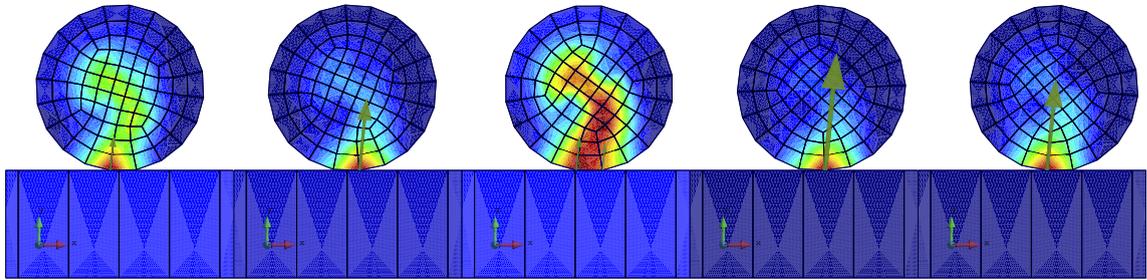


Figure 29: Wheel: sliding on the ground, large couple

## 4.6 Full finger

This case has been studied in the context of the FP7 NanoBioTouch project. The full work has been done by Dr. Deniz Somer using his own implementation. He designed the model that has been used for this work. Thus, it's a good way to test the methods on a more complicated case.

The aim is to study the mechanics of touch and surfaces recognition. To do so, a 2D model of a fingertip cut is slid along a ridged surface. The finger also has ridges that represent fingerprints. The simulation is performed in dynamics using the generalised  $\alpha$  method with damping to help the convergence. On Figure 30, each

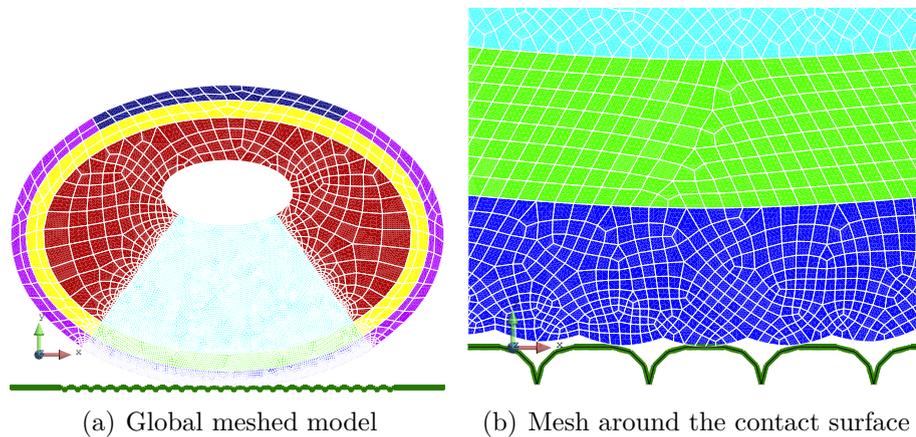


Figure 30: Fingertip: model and mesh

color represents a material but they will not be detailed here. The study is done under constant compression force which is applied on the white solid representing the bone. The horizontal motion is led by the imposed displacement of the bone.

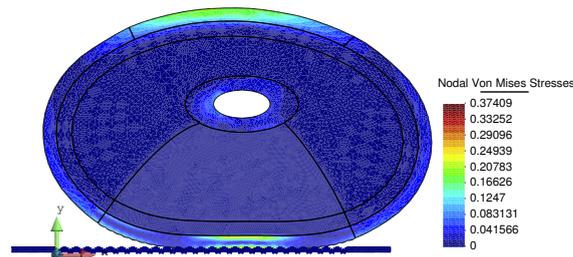


Figure 31: Fingertip: Global Von Mises Stresses

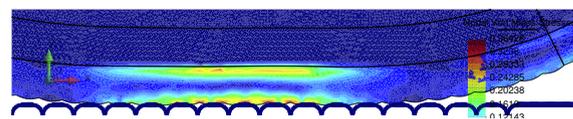


Figure 32: Fingertip: Under skin Von Mises Stresses

When looking at the animated simulation, the first remark concerns the non-regularity of the motion of the skin. Due to ridges inter-locking, the motion happens in fits and starts. Figure 33 shows the step-shaped evolution of the position of one

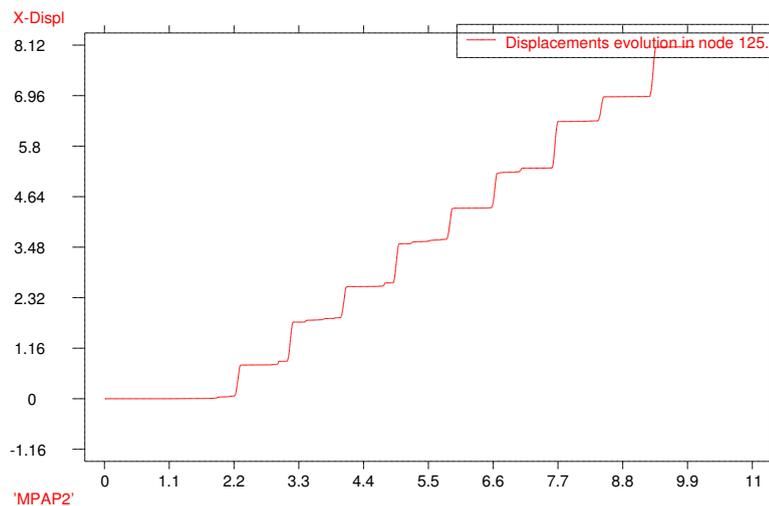


Figure 33: Fingertip: Skin motion

point of the skin.

Also, by looking into biological fingertip descriptions (like Wikipedia<sup>12</sup>), it can be learned that some pressure sensors are located between the epidermis and the dermis, where stress variation can be observed on Figure 32. Figure 34 shows that a

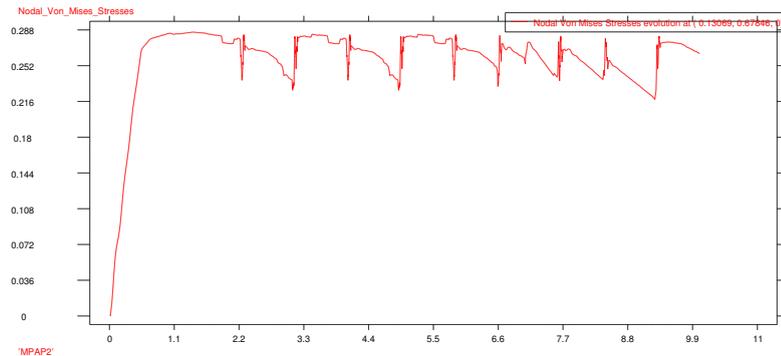


Figure 34: Fingertip: stress evolution in the sensors zone

pattern is repeated a certain frequency. The variation in frequency could be related to non-physical dynamical effects (The finger is probably bouncing at a very low frequency so the actual contact pressure decreases).

Obviously, this is just an application to the methods developed before and a complete analysis would require more simulations and tests. Nevertheless, some sensible results are arising.

## 4.7 Gears

Another quite common problem in mechanics concerns energy transmission through gears. A simple model of gears has been drawn and discretised Figure 36. The gears are made from a very stiff core and softer teeth. Their center is fixed in translation (this is why the core needs to be stiffer) and both gear are driven by pressure elements (the same used with the wheel, Section 4.3.2). Also, both wheels have the same constant couple applied in opposite direction but, as the large wheel is twice as big as the small one, the system is not balanced and so, rotates (the small gear

turns clockwise). The first interesting result is the stress distribution that shows the

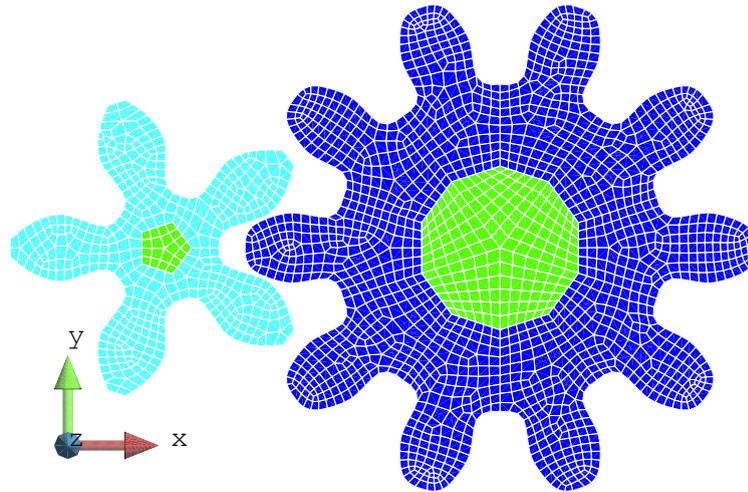


Figure 35: Gears: model and mesh

effort transmission at start-up.

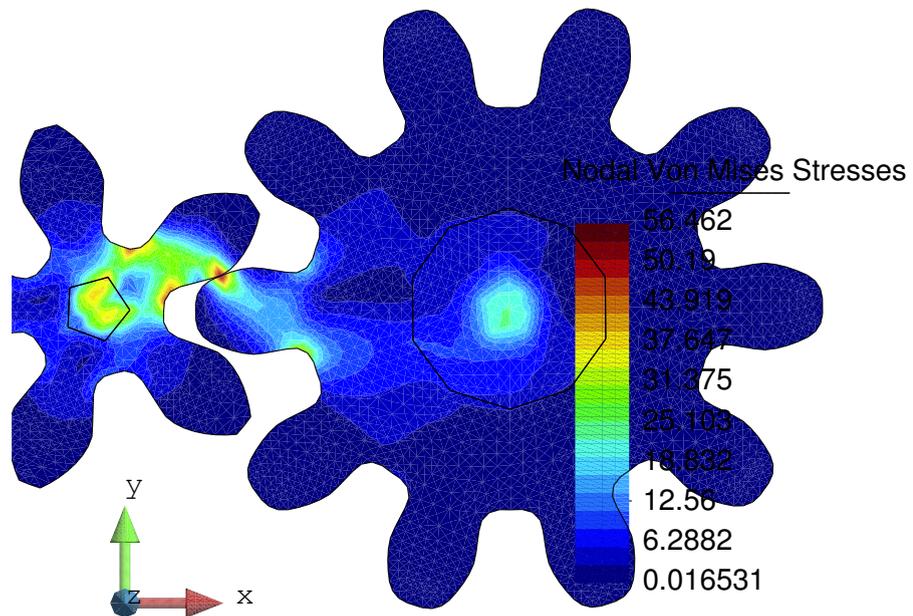


Figure 36: Gears: Von Mises Stresses

This case has been run for a long period of time (thousands of rotations) which shows that the solver is able to handle very large displacements and rotations.

## 4.8 Pseudo-realistic case: the bike

This example has been put together to show how contact mechanics can be combined with dynamics to give a pseudo-realistic simulation of a complex system. The wheels (in green) are similar to the one studied Section 4.3.2. They are in

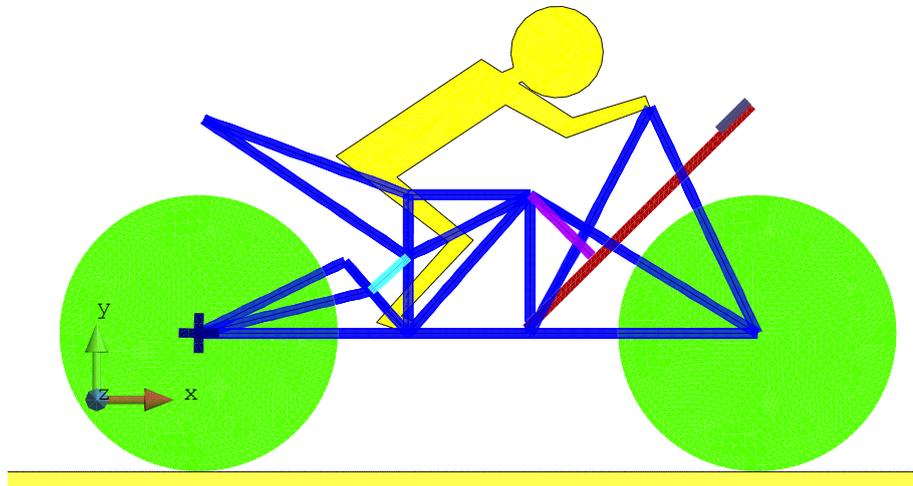


Figure 37: Bike: Model description

frictional contact with the ground with  $\mu = 0.9$ . The rear wheel is moved in rotation by pressure elements. The dark blue lines are trusses that make the frame and the clear blue one is a spring. The yellow pilot is attached to the frame by its feet and hands while in frictional contact with the saddle ( $\mu = 0.2$ ). The red structure is a brake, that can rotate when solicited by the purple pressure element. It comes in frictional contact with the front wheel to actually stop its rotation.

First, the bike bounces and stabilises on the ground. Next the rear wheel rotation is activated. After a while, the power is cut and the bike keeps going on its inertia. Next, the break is activated and the bike does a ‘stoppie’.

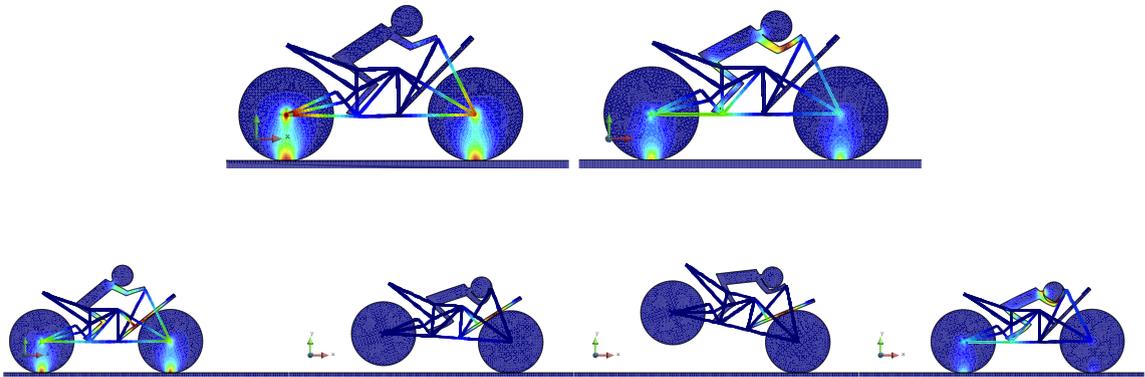


Figure 38: Bike: animation

## 5 Conclusion

From energy formulation to implementation and tests, two contact formulations have been described. Node to segment method has been developed with different condition enforcement methods. Together with Lagrange multipliers, it offers a reliable contact formulation.

Mortar method is tackling the problem from another angle which seems less robust. However, a more advanced analysis of its mathematical background could lead to a smarter technique, more compliant with coarse mesh discretisations.

A large variety of examples have been exposed, and all have been solved with success.

## References

- [1] Peter Wriggers. *Computational Contact Mechanics*. Springer-Verlag, second edition, 2006. ISBN 9783642069048. URL <http://www.springer.com/materials/mechanics/book/978-3-540-32608-3>.
- [2] Michael A. Puso and Tod A. Laursen. A mortar segment-to-segment frictional contact method for large deformations. *Computer Methods in Applied Mechanics and Engineering*, 193(45–47):4891 – 4913, 2004. ISSN 0045-7825. doi: 10.1016/j.cma.2004.06.001. URL <http://www.sciencedirect.com/science/article/pii/S0045782504002518>.
- [3] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*. Springer series in computational mathematics. Springer-Verlag, 1991. ISBN 9780387975825. URL <http://books.google.fr/books?id=yYwZAQAAIAAJ>.
- [4] Peter Wriggers. Matrices for node-to-node (ntv) elements. In *Computational Contact Mechanics*, pages 251–252. Springer-Verlag, second edition, 2006.
- [5] Peter Wriggers. Two-dimensional node-to-segment contact discretization. In *Computational Contact Mechanics*, pages 226–235. Springer-Verlag, second edition, 2006.
- [6] Peter Wriggers. *Computational Contact Mechanics*. John Wiley & Sons, first edition, 2002. ISBN 9780471496809. URL <http://books.google.co.uk/books?id=8e9eNEpvVdYC>.
- [7] K. A. Fischer and P. Wriggers. Frictionless 2d contact formulations for finite deformations based on the mortar method. *Computational Mechanics*, 36:226–244, 2005. ISSN 0178-7675. URL <http://dx.doi.org/10.1007/s00466-005-0660-y>. 10.1007/s00466-005-0660-y.

- [8] Barbara I. Wohlmuth. A mortar finite element method using dual spaces for the lagrange multiplier. *SIAM Journal on Numerical Analysis*, 38(3):989–1012, 2000. doi: 10.1137/S0036142999350929. URL <http://link.aip.org/link/?SNA/38/989/1>.
- [9] Heinrich Hertz. Über die Berührung fester elastischer Körper. *Journal für die reine und angewandte Mathematik*, 92:156–171, 1881.
- [10] C. K. Smith, J. O. and Liu. Stresses due to tangential and normal loads on an elastic solid with application to some contact stress problems. *Journal of Applied Mechanics*, 20:157–166, 1953.
- [11] A.P. Boresi and R.J. Schmidt. *Advanced mechanics of materials*. John Wiley & Sons, 2003. ISBN 9780471438816. URL <http://books.google.co.uk/books?id=eLMeAQAAIAAJ>.
- [12] Wikipedia. Mechanoreceptor — Wikipedia, the free encyclopedia, 2012. URL <http://en.wikipedia.org/wiki/Mechanoreceptor>.