

# Modelling unilateral frictionless contact using the null-space method and cubic B-Spline interpolation

J. Muñoz\*

*Dep. Applied Mathematics III, LaCàN  
Univ. Politècnica de Catalunya (UPC), Spain*

---

## Abstract

The analysis of unilateral sliding contact in elasticity is equivalent to a minimisation problem subjected to a set of inequality constraints. However, the presence of boundary discontinuities, such as those stemming from the spatial discretisation, appears as a major problem to determine the set of active constraints. This work introduces a smoothing technique of the master surface resorting to cubic B-Spline interpolation, which is  $C^1$  continuous in contact situations between elastic and rigid bodies, and  $G^1$  continuous in elastic-elastic contact problems. The technique is applied in conjunction with the null-space method, where the solution is searched in an unconstrained manifold. The resulting formulation eases the contact transition along the master surface, and recovers the quadratic convergence of the iterative Newton-Raphson process. The robustness of the method is demonstrated using 2D and 3D examples.

*Key words:* contact, splines, smoothing, null-space method, master-slave, Lagrange multipliers.

## 1 Introduction

The modelling of unilateral contact requires the satisfaction of a set of kinematic constraints. These are commonly imposed using penalty methods or a set of Lagrange multipliers (see for instance [31] for a general description or [23] and [1] for specific relevant implementations). While the former use a problem dependent penalty parameter, and leads to potential violation of the constraints, the latter satisfies the constraints exactly but gives rise to an ill-conditioned stiffness matrix which couples force and displacement unknowns. Although the ill-conditioning can be amended resorting to the augmented Lagrangian method, additional variables are still being used, and slower convergence rates may be obtained. We will alternatively use here the null-space method [3], also called master-slave method [12], or minimum set of generalised coordinates [21]. In this method, the constrained problem is projected onto a manifold where no constrained unknowns exist, and therefore, the computation of the Lagrange multipliers is circumvented (the latter can be retrieved in the post-processing, once the converged solution has been obtained). The present paper describes an implementation of the null-space method for frictionless contact problems in statics.

Regarding the contact constraints, they may be formulated in a weak sense, as it is done in the mortar methods [15,26], which leads to segment-on-segment strategies. Alternatively, and as employed here, the impenetrability contact constraints may be imposed at each node, resulting a node-on-segment strategy. Although it is well known that the latter does not in general satisfy the

---

\*

*Email address:* [j.munoz@upc.edu](mailto:j.munoz@upc.edu)

Tel. +34 93 405 41 81, Fax. +34 93 401 18 25 (J. Muñoz).

*URL:* <http://www-lacan.upc.es/munoz> (J. Muñoz).

patch test (the transfer of a constant pressure through a flat surface, which is partially in contact with another body [8]), it is numerically shown here that it allows to model complex contact situations, where contact is progressively affecting points, lines or surfaces.

The essence of the null-space method has been previously applied to the modelling of incompressibility materials [16] and joints in multibody systems [3,12,17,18]. While both types of problems encounter *equality* constraints, the present case deals with unilateral contact, and therefore yields *inequality* constraints. This is an additional source of non-linearity, which in problems with large displacements and deformations, and due to the spatial discretisation, may become critical. Indeed, the set of active constraints may highly vary throughout the analysis, giving rise to intermittent activation of the constraints or so-called “chatter” [25,26]. Different techniques have been proposed in the literature. Among them, we mention (i) mortar methods, which lead to smoother contact conditions [15,26], (ii) design of double iterative loops, one for the non-linear solution and another for the determination of active constraints [5], or (iii) the direct smoothing of the contact surfaces [2,10,14,20,24,32]. The present work focuses on the latter approach. In particular, the chatter is alleviated here by interpolating the contact surfaces with smooth cubic  $C^1$  B-Splines.

General smoothing techniques have been widely applied in computational mechanics (see the early works [6,30]). In contact analysis, usual venues are Hermite patches [20,33], Overhauser segments [9] or cubic Béziars [23]. In contrast, in the present paper, the curve or surface is parametrised with a cubic B-Spline using a global parametrisation throughout all the segments. This route is specially advantageous when using the null-space method, where the positions of the nodes in contact are described using their parametric coordinates. By

using a global parametrisation, the trajectories of the slave nodes on the master surface are fully described with a unique set of parameters. However, the method requires the use of structured discretisations for the master side. Although this is not needed in alternative smoothing techniques such as Gregory patches [24] or subdivision schemes [29], these techniques require special treatment of nodes where the mesh is un-structured, and in many cases, such as Gregory patches, only  $G^1$  continuity is obtained (tangent plane continuity), but not  $C^1$  continuity, even for the undeformed configurations. We show in the numerical examples that the convergence of the solution process is improved when using  $C^1$  continuity.

In addition to improving the mechanical analysis, the description of the contact surfaces as B-Splines furnishes a common frame for the geometrical and mechanical models [11,13], which avoids the usual conversion of geometrical entities into discretised elements. (Interestingly, though, the use of B-Splines in [13] was motivated by the desire to provide an intuitive tool for geometrical design, and not to improve the mechanical analysis.) However, since the merger of geometrical and meshing tools is still in its infancy, the paper includes a methodology that transforms structured surfaces interpolated with finite elements (the output usually obtained after the meshing process), into cubic B-Splines.

Although most of the examples are restricted to *rigid* master surfaces, the paper also includes the extension of the theory to *deformable* boundaries interpolated with cubic B-Splines. This can be achieved by either (i) considering a basis of Béziers functions for the whole continuum that contains the master surface, or (ii) designing elements with mixed interpolation, B-Splines for the boundary and standard Lagrangian interpolation for the interior. The choices (i) and (ii) are similar to those adopted in [11] and [28], respectively, when us-

ing NURBS instead of B-Splines. We have adopted here the former solution, which has been also implemented and tested in two-dimensional numerical tests.

Similar instances of the null-space method have been termed master-slave approach in some previous works [12,16–18]. We change here the terminology in order to avoid confusion between the method, and the master and slave surfaces commonly employed in contact detection. The usual description of the contact conditions with master and slave surfaces (something that it is also exploited here), does not imply the use of the master-slave approach presented in these references. Moreover, while the null-space method is a general method to project constraints in the optimisation context, the master-slave approach constructs a suitable projection by a kinematic reasoning: the contact (slave) point onto the master surface is described with the parametric (or released) coordinates. Such a geometrical partition of coordinates has been employed in the analysis of mechanisms, but to the author’s knowledge, its use in more general frictionless contact situations, as those presented here, is unprecedented. The resulting formulation is specially advantageous and suited for contact surfaces parametrised with B-Splines.

The outline of the paper is as follows. In Section 2 the equilibrium equations of unilateral contact mechanics are recast as a minimisation problem, which in Section 3 is transformed using the null-space method. The construction of the B-Spline domain from a set of finite elements is described in Section 4, and some representative two and three-dimensional examples are shown in Section 5. Finally, Section 6 highlights the novel features of the proposed method and gives some concluding remarks.

## 2 Contact mechanics

Following standard notation [4,31], the initially undeformed and the current deformed configurations of an elastic body are denoted by  $\mathcal{B}_0 \subset \mathbb{R}^{n_{sd}}$  and  $\mathcal{B} \subset \mathbb{R}^{n_{sd}}$ , respectively ( $n_{sd} = 2$  or  $3$  is the number of space dimensions). Let us also introduce the deformation map  $\phi : \mathbf{X} \in \mathcal{B}_0 \rightarrow \mathbf{x} \in \mathcal{B}$ , where  $\mathbf{X}$  and  $\mathbf{x}$  are the material and spatial coordinates in the initial and deformed configurations, respectively.

It is assumed that a standard finite element (FE) discretisation of body  $\mathcal{B}_0$  with the usual Lagrangian shape functions has been performed, and as a result,  $n_c$  nodes are located at the boundary  $\partial B_0$ . To ease the notation, we will assume in this section that the  $\mathcal{B}_0$  is in contact with a rigid body, although considering contact with an elastic body would lead to the similar equations.

The (static) equilibrium of  $\mathcal{B}_0$  in unilateral contact, and considering also for simplicity only external forces stemming from the contacts constraints, can be then derived by minimising the total elastic potential  $\mathcal{V}(\mathbf{x})$  subjected to a set of *inequality* contact constraints, i.e.

$$\begin{aligned} \min_{\mathbf{x} \in V} \mathcal{V}(\mathbf{x}) \\ \text{s.t. } \bar{\Phi}(\mathbf{x}) \geq 0. \end{aligned} \tag{1}$$

The set  $V$  contains all the admissible deformed coordinates, i.e. those that lead to a bounded elastic energy. The vector expression  $\bar{\Phi}(\mathbf{x}) \geq \mathbf{0}$  is the set of the constraints  $\Phi_i(\mathbf{x}_i) \geq 0$ ,  $i = 1, \dots, n_c$ , where the functions  $\Phi_i(\mathbf{x}_i)$  represent the nodal gaps between the potential contact nodes  $\mathbf{x}_i$ , called *slave nodes*, and the boundary of the rigid fixed body, called the *master surface* or *slidesurface* (see Figure 1). The latter is described as a parametric surface

$\varphi(\boldsymbol{\xi}) : \mathbb{R}^{n_{sd}-1} \rightarrow \mathbb{R}^{n_{sd}}$ , with  $\xi_\alpha, \alpha = 1, \dots, n_{sd} - 1$ , the parametric coordinates.

The function  $\Phi_i(\mathbf{x}_i)$  is defined for each slave node  $i$  as,

$$\Phi_i(\mathbf{x}_i) \equiv (\mathbf{x}_i - \mathbf{x}_{\xi_i}) \cdot \mathbf{n}_{\xi_i}. \quad (2)$$

The vector  $\mathbf{x}_{\xi_i} \in \mathbb{R}^{n_{sd}}$  is the projection of node  $\mathbf{x}_i$  onto the master surface  $\varphi(\boldsymbol{\xi})$ , and corresponds to evaluating  $\varphi(\boldsymbol{\xi})$  at the parametric coordinate  $\boldsymbol{\xi}_i$ , i.e.  $\mathbf{x}_{\xi_i} = \varphi(\boldsymbol{\xi}_i)$ . The vector  $\mathbf{n}_{\xi_i} \in \mathbb{R}^{n_{sd}}$  is the outward normal of the master surface at  $\boldsymbol{\xi} = \boldsymbol{\xi}_i$ . The parametric coordinates  $\boldsymbol{\xi}_i$  can be found as the solution of the following minimisation problem:

$$\min_{\boldsymbol{\xi}} \|\mathbf{x}_i - \varphi(\boldsymbol{\xi})\|^2.$$

Note that from this equation, it follows that  $\boldsymbol{\xi}_i$  satisfies the following relation:

$$(\mathbf{x}_i - \mathbf{x}_{\xi_i}) \cdot \nabla_{\boldsymbol{\xi}} \varphi \Big|_{\boldsymbol{\xi}=\boldsymbol{\xi}_i} = 0, \quad (3)$$

with  $[\nabla_{\boldsymbol{\xi}} \varphi]_{ki} = \frac{\partial \varphi(\boldsymbol{\xi})_k}{\partial \xi_i}$ . Figure 1 illustrates the geometrical meaning of the nodal gap  $\Phi_i(\mathbf{x}_i)$  and the terms involved in its definition.

The solution of the constrained minimisation problem in (1) is equivalent to finding a set of spatial coordinates  $\mathbf{x}$  and a vector of Lagrange multipliers  $\bar{\boldsymbol{\lambda}}^T = \{\lambda_1, \dots, \lambda_{n_c}\}$  that satisfies the Karush-Kuhn-Tucker (KKT) conditions [19] of problem (1):

$$\begin{aligned} \nabla \mathcal{V} - (\nabla \bar{\boldsymbol{\Phi}})^T \bar{\boldsymbol{\lambda}} &= \mathbf{0}, \\ \bar{\boldsymbol{\Phi}}(\mathbf{x}) &\geq \mathbf{0}, \\ \bar{\boldsymbol{\lambda}} &\geq \mathbf{0}, \\ \lambda_i(\mathbf{x}) \Phi_i(\mathbf{x}) &= 0, \quad i = 1, \dots, n_c, \end{aligned} \quad (4)$$

where the gradient must be understood with respect to the discretised spatial variables (nodal variables). The first condition can be derived from the fact that the optimal deformed solution minimises the extended Lagrangian  $\mathcal{L}(\mathbf{x}, \bar{\boldsymbol{\lambda}}) = \mathcal{V}(\mathbf{x}) - \bar{\boldsymbol{\lambda}}^T \bar{\boldsymbol{\Phi}}(\mathbf{x})$ . The last equation in (4) is the so-called linear complementary problem [21], which states that  $\lambda_i$  or  $\Phi_i(\mathbf{x})$  may be non-zero, but at least one of the two must be zero. Physically, the Lagrange multipliers  $\lambda_i$  correspond to the normal contact forces that prevent the constraint violation.

For a given set of values  $\mathbf{x}$ , the  $n_c$  *inequality* constraints  $\Phi_i(\mathbf{x}) \geq 0$  are henceforth classified in two sets: the set of *active* constraints  $\mathcal{A}(\mathbf{x})$  and the set of *inactive* constraints  $\mathcal{I}(\mathbf{x})$ , which may be formally expressed as,

$$\mathcal{A}(\mathbf{x}) = \{i | \Phi_i(\mathbf{x}) = 0\},$$

$$\mathcal{I}(\mathbf{x}) = \{i | \Phi_i(\mathbf{x}) > 0\}.$$

During the solution of constrained problem, the sets  $\mathcal{A}(\mathbf{x})$  and  $\mathcal{I}(\mathbf{x})$  must be determined, and actually, in contact problems with large displacements, the sets of active and inactive constraints may highly vary throughout the analysis. In the present work, by smoothing the contact surfaces, we aim to detect and ease the transition of the constraint state, from active to inactive and vice versa.

Assuming that the sets  $\mathcal{A}(\mathbf{x})$  and  $\mathcal{I}(\mathbf{x})$  at the optimal solution are known, the inactive constraints can be ignored when solving (4), i.e. the solution of (4) is also the solution of the following KKT conditions:

$$\nabla\mathcal{V} - (\nabla\Phi(\mathbf{x}))^T\boldsymbol{\lambda} = \mathbf{0}, \quad (5a)$$

$$\Phi(\mathbf{x}) = \mathbf{0}, \quad (5b)$$

$$\boldsymbol{\lambda} \geq \mathbf{0}, \quad (5c)$$

where here, the vectors  $\Phi(\mathbf{x})$  and  $\boldsymbol{\lambda}$  have components  $\Phi_i$  and  $\lambda_i$ , with  $i \in \mathcal{A}(\mathbf{x})$ .

### 3 Null-space method

#### 3.1 Derivation of the projection matrix

The null-space method applied to the present contact problem in (5) consists in projecting equation (5a) onto the manifold  $\mathcal{K} = \{\mathbf{x} \in V | \Phi_i(\mathbf{x}) = 0, i \in \mathcal{A}(\mathbf{x})\}$  and solving the projected unconstrained minimisation problem onto manifold  $\mathcal{K}$ . Note that the solution projected in this set is the same as the solution in the more restrictive “smaller” manifold  $\bar{\mathcal{K}} = \{\mathbf{x} \in V | \Phi_i(\mathbf{x}) \geq 0, i = 1, \dots, n_c\}$ .

In the optimisation context [19], this is usually achieved algebraically by constructing a matrix  $\mathbf{N}$  such that

$$\text{range}(\mathbf{N}) = \text{null}(\nabla\Phi).$$

It then follows that  $\mathbf{N}^T(\nabla\Phi)^T = \mathbf{0}$  or  $(\nabla\Phi)\mathbf{N} = \mathbf{0}$ . Therefore, after multiplying (5a) by  $\mathbf{N}^T$ , the following projected (or *reduced*) system of non-linear equations is obtained:

$$\mathbf{N}^T\nabla\mathcal{V}(\mathbf{x}) = \mathbf{0}, \quad (6)$$

together with the constraints (5b) and (5c).

In the analysis of mechanisms with joints [12,17], the projection operator  $\mathbf{N}$  is constructed following a more kinematic reasoning. A set of release or joint coordinates are introduced, which replace the Cartesian coordinates, reduce the number of degrees of freedom of the system, and permit to obtain unconstrained motion equations. This method is usually called the master-slave approach or generalised minimum set of coordinates. We will borrow these ideas to construct the projection matrix  $\mathbf{N}$ .

For those nodes that are in contact, i.e. those for which  $\Phi_i(\mathbf{x}) = 0$  holds, the infinitesimal parametric coordinates  $\delta\boldsymbol{\xi}$  can be used to describe their Cartesian infinitesimal displacements  $\delta\mathbf{x}$  as follows,

$$\delta\mathbf{x} = \nabla_{\boldsymbol{\xi}}\boldsymbol{\varphi}(\boldsymbol{\xi})\delta\boldsymbol{\xi}.$$

Actually, the tangent operator  $\nabla_{\boldsymbol{\xi}}\boldsymbol{\varphi}$  furnishes the components of the desired null-space matrix  $\mathbf{N}$ . This is stated in the following proposition:

**Proposition 1** *If  $\Phi_i(\mathbf{x}) = 0$  for a given node coordinate  $\mathbf{x}_i$ , then*

$$\left( \nabla\Phi_i(\mathbf{x}) \Big|_{\Phi_i=0, \mathbf{x}=\mathbf{x}_i} \right) \nabla_{\boldsymbol{\xi}}\boldsymbol{\varphi}(\boldsymbol{\xi}) \Big|_{\boldsymbol{\xi}=\boldsymbol{\xi}_i} = 0.$$

To proof this, note that when  $\Phi_i(\mathbf{x}) = 0$ , it follows from expression (2) that  $\mathbf{x}_i = \mathbf{x}_{\xi_i}$ , and therefore, the gradient of  $\Phi_i(\mathbf{x})$  at  $\mathbf{x} = \mathbf{x}_i$  is given by

$$\begin{aligned} \nabla\Phi_i(\mathbf{x}) \Big|_{\Phi_i=0, \mathbf{x}=\mathbf{x}_i} &= \left( \mathbf{n}_{\xi}^T \nabla(\mathbf{x} - \mathbf{x}_{\xi}) + (\mathbf{x} - \mathbf{x}_{\xi})^T \nabla\mathbf{n}_{\xi} \right) \Big|_{\Phi_i=0, \mathbf{x}=\mathbf{x}_i} \\ &= \mathbf{n}_{\xi}^T \nabla(\mathbf{x} - \boldsymbol{\varphi}(\boldsymbol{\xi})) \Big|_{\Phi_i=0, \mathbf{x}=\mathbf{x}_i} \\ &= \mathbf{n}_{\xi_i}^T - \mathbf{n}_{\xi}^T \left( \nabla_{\boldsymbol{\xi}}\boldsymbol{\varphi}(\boldsymbol{\xi}) \frac{\partial\boldsymbol{\xi}}{\partial\mathbf{x}} \right) \Big|_{\Phi_i=0, \mathbf{x}=\mathbf{x}_i} \end{aligned} \quad (7)$$



obtained, which in the present case will be solved iteratively using a Newton-Raphson process. Further details of the solution process are given in Section 3.3.

It is worth pointing out that, indeed, the resulting system of equations in (6) has less unknowns than the initial constrained problem in (5a). Assuming that there are  $n_A$  active constraints, the system in (6) has  $n_A$  equations less than (5a) (one degree of freedom less per active constraint). Also, in the actual implementation, no global matrix  $\mathbf{N}$  is employed. Instead, just the nodal products  $\boldsymbol{\varphi}_i^T(\nabla\mathcal{V}(\mathbf{x}))_i$  are performed for  $i \in \mathcal{A}(\mathbf{x})$ . In this way, a substantial reduction of the additional cost of the projection is achieved.

### 3.2 Setting active and inactive constraints

As mentioned above, the projection matrix given in (8) corresponds to the case where all the constraints are active, which is not usually the case. The sets  $\mathcal{A}(\mathbf{x})$  and  $\mathcal{I}(\mathbf{x})$  are determined at each iteration during the solution process according to the rules indicated in Table 1.

The gaps  $\Phi_i$  may be computed from expression (2), whereas the Lagrange multipliers are obtained from the non-projected equilibrium equations (5a). By gathering the equations conjugate to a slave node  $i$ , assuming  $i \in \mathcal{A}(\mathbf{x})$ , it follows that,

$$(\nabla\mathcal{V}(\mathbf{x}))_i - \lambda_i \nabla\Phi_i = \mathbf{0}. \quad (11)$$

Since the Lagrange multipliers must be calculated for the active constraints, and in this case it has been deduced in Section 3.1 that  $\nabla\Phi_i = \mathbf{n}_{\xi_i}^T$ , the dot-product of (11) by  $\mathbf{n}_{\xi_i}$  gives rise to the following equation:

$$\lambda_i = \mathbf{n}_{\xi_i} \cdot (\nabla \mathcal{V}(\mathbf{x}))_i. \quad (12)$$

This is the component of the reaction force orthogonal to the master surface, which in fact, since only frictionless contact is considered, is the only non-zero component. On the other hand, if an inactive constraint is activated, the Cartesian coordinates are projected to the master surface by solving equation (3).

In the present case, the active-inactive strategy is applied at each iteration, which in turn may modify the structure of matrix  $\mathbf{N}$ . If a certain active constraint  $j \in \mathcal{A}(\mathbf{x})$  becomes inactive, the corresponding term  $\varphi_j$  in matrix  $\mathbf{N}$ , equation (8), is replaced by an identity matrix  $\mathbf{I}_{n_{sd}}$ , and vice versa. Therefore, matrix  $\mathbf{N}$  varies from iteration to iteration not only due to the non-linear character of  $\varphi_i$ , but also due to changes in its structure. Nonetheless, due to the diagonal structure of  $\mathbf{N}$  in (8), this modification may be performed only to the slave nodes after assembling the global system of equations.

It has been numerically experienced that the smoothness of these changes, and thus the achievement of equilibrium, depends in turn on the actual smoothness of the contact surfaces. This fact is further discussed in Section 4 and in the numerical examples.

### 3.3 *Linearisation, solution and update process*

We derive here the stiffness matrix of the discretised and projected non-linear equations in (6). We assume first that the master surface is rigid, i.e. the coordinates of the control points are constant. The extension of the solution process in presence of master surfaces that belong to a deformable body is given in subsection 3.4.

The unknowns of equation  $\mathbf{N}^T \nabla \mathcal{V}(\mathbf{x}) = \mathbf{0}$  are the set of parametric coordinates  $\boldsymbol{\xi}_i$ ,  $i \in \mathcal{A}(\mathbf{x})$ , and the standard Cartesian coordinates  $\mathbf{x}_j$  of the remaining nodes. To ease the forthcoming expressions, both sets of variables are posted together in a global variable  $\mathbf{p}_\xi$ , which in case that all the constraints are active, is given by,

$$\mathbf{p}_\xi^T = \{\boldsymbol{\xi}_1^T, \dots, \boldsymbol{\xi}_{n_c}^T, \mathbf{x}_{n_c+1}^T, \dots, \mathbf{x}_{n_N}^T\}.$$

Vector  $\mathbf{p}_\xi$  contains the minimum number of unknowns. However, its structure changes according to the state of the constraint sets  $\mathcal{A}(\mathbf{x})$  and  $\mathcal{I}(\mathbf{x})$ , in the same manner as the structure of matrix  $\mathbf{N}$ , i.e., whenever  $j \in \mathcal{I}(\mathbf{x})$ , the term  $\boldsymbol{\xi}_j$  in  $\mathbf{p}_\xi$  is replaced by  $\mathbf{x}_j$ , and if  $j \in \mathcal{A}(\mathbf{x})$ ,  $\mathbf{x}_j$  is replaced by  $\boldsymbol{\xi}_j$ . Moreover, by denoting the whole set of Cartesian coordinates by  $\mathbf{p}^T = \{\mathbf{x}_1^T, \dots, \mathbf{x}_{n_N}^T\}$ , the infinitesimal displacements  $\delta \mathbf{p}$  and  $\delta \mathbf{p}_\xi$  are related through

$$\delta \mathbf{p} = \mathbf{N} \delta \mathbf{p}_\xi. \quad (13)$$

The iterative solution of the non-linear equations in (6) with a Newton-Raphson process requires the computation of its linear form, which may be derived as follows:

$$D(\mathbf{N}^T \nabla \mathcal{V}(\mathbf{x})) = \mathbf{N}^T D \nabla \mathcal{V}(\mathbf{x}) + D(\mathbf{N}^T) \nabla \mathcal{V}(\mathbf{x}). \quad (14)$$

By making use of the standard stiffness matrix  $\mathbf{K}_E$  such that  $D \nabla \mathcal{V}(\mathbf{x}) = \mathbf{K}_E D \mathbf{x}$ , and recalling relation (13), which is also valid for the iterative displacements  $D \mathbf{p}$ , the first term in (14) is expressible as,

$$\mathbf{N}^T D \nabla \mathcal{V}(\mathbf{x}) = \mathbf{N}^T \mathbf{K}_E \mathbf{N} D \mathbf{p}_\xi, \quad (15)$$

where  $D \mathbf{p}_\xi$  are the iterative displacements of the parametric (active con-

straints) and Cartesian coordinates (inactive constraints and remaining nodes). The second term in (14) can be computed recalling the expression of  $\varphi_i$  in (10), and by noting that at iteration  $\ell$  and for each node  $i \in \mathcal{A}(\mathbf{x})^\ell$ , the equilibrium equations associated to this node read,

$$\varphi_i^T (\nabla \mathcal{V}(\mathbf{x}))_i = \sum_{k=1}^{n_k} (\mathbf{P}_k \cdot (\nabla \mathcal{V}(\mathbf{x}))_i) \nabla_\xi I_k(\boldsymbol{\xi}). \quad (16)$$

The linearisation of this term, keeping  $\nabla \mathcal{V}(\mathbf{x})$  unchanged, yields

$$\left( \sum_{k=1}^{n_k} (\mathbf{P}_k \cdot (\nabla \mathcal{V}(\mathbf{x}))_i) \nabla_\xi \otimes \nabla_\xi I_k(\boldsymbol{\xi}) \right) D\boldsymbol{\xi}_i = \mathbf{k}_{\varphi_i} D\boldsymbol{\xi}_i. \quad (17)$$

where  $\mathbf{k}_{\varphi_i} = \sum_{k=1}^{n_k} (\mathbf{P}_k \cdot (\nabla \mathcal{V}(\mathbf{x}))_i) \nabla_\xi \otimes \nabla_\xi I_k(\boldsymbol{\xi})$  and  $(\nabla_\xi \otimes \nabla_\xi I_k(\boldsymbol{\xi}))_{ij} = \frac{\partial^2 I_k(\boldsymbol{\xi})}{\partial \xi_i \partial \xi_j}$ . In summary, gathering the results in (15) and (17) we obtain,

$$\frac{\partial}{\partial \mathbf{p}_\xi} (\mathbf{N}^T \nabla \mathcal{V}(\mathbf{x})) = \underbrace{\mathbf{N}^T \mathbf{K}_E \mathbf{N}}_{\mathbf{K}} + \mathbf{K}_N, \quad (18)$$

where  $\mathbf{K}_N$ , arising from the linearisation of matrix  $\mathbf{N}$ , is given by

$$\mathbf{K}_N = \begin{bmatrix} \mathbf{k}_{\varphi_1} & & & & \\ & \ddots & & & \\ & & \mathbf{k}_{\varphi_{n_c}} & & \\ & & & & \mathbf{0}_{n_{sd}(n_N - n_c)} \end{bmatrix}. \quad (19)$$

Therefore, at iteration  $\ell$ , the Newton-Raphson process yields the following system of linear equations:

$$\mathbf{K}^\ell D\mathbf{p}_\xi = - \left( \mathbf{N}^T \nabla \mathcal{V}(\mathbf{x}) \right)^\ell.$$

After solving this equation, the update the Cartesian coordinates of the nodes not in contact is given by  $\mathbf{x}_i^\ell = \mathbf{x}_i^{\ell-1} + D\mathbf{x}_i$ , and the parametric coordinates by  $\boldsymbol{\xi}_i^\ell = \boldsymbol{\xi}_i^{\ell-1} + D\boldsymbol{\xi}_i$ . The Cartesian coordinates of the nodes in contact are updated as  $\mathbf{x}_{\xi_i}^\ell = \boldsymbol{\varphi}(\boldsymbol{\xi}_i^\ell)$ . In this way, it is ensured that the active constraints are not violated, and therefore, condition (5b) can be removed from the KKT conditions. However, condition (5c) may be violated. If  $\lambda_i < 0$ , the constrained is set to the inactive state, as described in section 3.2. Table 2 summarises the solution process in conjunction with the update of the sets  $\mathcal{I}(\mathbf{x})$  and  $\mathcal{A}(\mathbf{x})$ .

### 3.4 Extension to deformable master surfaces

In this subsection we are just concerned with the additional terms arising in the equilibrium equations and in the solution process when the control points coordinates are also unknown and belong to a deformable domain. The way we interpolate the continuum with B-Splines will be detailed in Section 4.3.

The equilibrium equation in (6) stem from the variation of the discretised elastic potential with respect to the nodal coordinates  $\mathbf{p}_\xi$ . However, when the control points coordinates  $\mathbf{P}_k$  are additional degrees of freedom, the minimisation of the total potential, which also depends on  $\mathbf{P}^T = \{\mathbf{P}_1^T \dots \mathbf{P}_K^T\}$ , i.e.  $\mathcal{V}(\mathbf{x}, \mathbf{P})$ , yields in turn additional KKT conditions. More specifically, equation (5) is extended now as follows:

$$\nabla_{\mathbf{x}}\mathcal{V}(\mathbf{x}, \mathbf{P}) - (\nabla_{\mathbf{x}}\boldsymbol{\Phi}(\mathbf{x}, \mathbf{P}))^T\boldsymbol{\lambda} = \mathbf{0}, \quad (20a)$$

$$\nabla_{\mathbf{P}}\mathcal{V}(\mathbf{x}, \mathbf{P}) - (\nabla_{\mathbf{P}}\boldsymbol{\Phi}(\mathbf{x}, \mathbf{P}))^T\boldsymbol{\lambda} = \mathbf{0}, \quad (20b)$$

$$\boldsymbol{\Phi}(\mathbf{x}, \mathbf{P}) = \mathbf{0}, \quad (20c)$$

$$\boldsymbol{\lambda} \geq \mathbf{0}. \quad (20d)$$

Here, the notation  $\nabla_x$  and  $\nabla_P$  denotes the gradient with respect to the nodal variables and  $\mathbf{x}_i$  and the control point coordinates  $\mathbf{P}_k$ , respectively. While the expression for  $\nabla_x \Phi_i$  is the same derived in (7), the term  $\nabla_P \Phi_i$  reads,

$$\nabla_P \Phi_i(\mathbf{x}, \mathbf{P}) = \left[ -I^1(\boldsymbol{\xi}_i) \mathbf{n}_{\xi_i}^T \dots - I^k(\boldsymbol{\xi}_i) \mathbf{n}_{\xi_i}^T \dots - I^{n_k}(\boldsymbol{\xi}_i) \mathbf{n}_{\xi_i}^T \right] = -\mathbf{n}_{\xi_i}^T \mathbf{M}_i^T, \quad (21)$$

where  $\mathbf{M}_i^T = [I^1(\boldsymbol{\xi}_i) \mathbf{I} \dots I^{n_k}(\boldsymbol{\xi}_i) \mathbf{I}]$ . The first equation (20a) is projected in the same manner as it has been done when using rigid master surfaces, i.e.  $\mathbf{N}^T \nabla_x \mathcal{V} = \mathbf{0}$ . The components of this equation associated to each slave node  $\mathbf{x}_i$  read,

$$\nabla_{\xi} \boldsymbol{\varphi} \cdot (\nabla_x \mathcal{V})_i = \mathbf{0}, \quad i = 1, \dots, n_c. \quad (22)$$

On the other hand, since  $\nabla_x \Phi_i = \mathbf{n}_{\xi_i}^T$ , it follows from (20a) that  $\lambda_i = \mathbf{n}_{\xi_i}^T (\nabla_x \mathcal{V})_i$ . Inserting this relation into (20b) we obtain the following equilibrium equation associated to each control point  $\mathbf{P}_k$  on the master surface where a node  $\mathbf{x}_i$  is in contact:

$$(\nabla_P \mathcal{V})_k + I^k(\boldsymbol{\xi}_i) (\mathbf{n}_{\xi_i} \otimes \mathbf{n}_{\xi_i}) (\nabla_x \mathcal{V})_i = \mathbf{0}, \quad k = 1, \dots, n_k. \quad (23)$$

Furthermore, since  $\mathbf{I} = \mathbf{n}_{\xi_i} \otimes \mathbf{n}_{\xi_i} + \sum_{\alpha=1}^2 (\nabla_{\xi} \boldsymbol{\varphi})_{\alpha} \otimes (\nabla_{\xi} \boldsymbol{\varphi})_{\alpha}$ , with  $(\nabla_{\xi} \boldsymbol{\varphi})_{\alpha}$  the  $\alpha$ -th tangent vector to  $\boldsymbol{\varphi}$ , and from (22) we have that  $\sum_{\alpha=1}^2 (\nabla_{\xi} \boldsymbol{\varphi})_{\alpha} \otimes (\nabla_{\xi} \boldsymbol{\varphi})_{\alpha} (\nabla_x \mathcal{V})_i = \mathbf{0}$ , it can be deduced that equations (22) and (23) are equivalent to,

$$\begin{aligned} \nabla_{\xi} \boldsymbol{\varphi} \cdot (\nabla_x \mathcal{V})_i &= \mathbf{0}, \quad i = 1, \dots, n_c. \\ (\nabla_P \mathcal{V})_k + I^k(\boldsymbol{\xi}_i) (\nabla_x \mathcal{V})_i &= \mathbf{0}, \quad k = 1, \dots, n_k. \end{aligned} \quad (24)$$

In summary, from expressions (21) and (24), the projected equilibrium equations for variable control points may be written as,

$$\begin{bmatrix} \mathbf{N} & \mathbf{M} \\ \mathbf{0} & \mathbf{I}_{n_k n_{sd}} \end{bmatrix}^T \nabla \mathcal{V} = \mathbf{0}, \quad (25)$$

where again, we have removed the conditions  $\Phi = \mathbf{0}$  and  $\lambda \geq \mathbf{0}$  due to the fact that we assume that we know the active set of constraints, and only those are included in the constrained problem. The symbol  $\nabla$  denotes now the gradient with respect to the nodal position  $\mathbf{x}_i$  and the control point  $\mathbf{P}_k$ , i.e.  $(\nabla \mathcal{V})^T = [(\nabla_x \mathcal{V})^T (\nabla_P \mathcal{V})^T]$ . Matrix  $\mathbf{M}$  is the assembling of matrices  $\mathbf{M}_i$  for each slave node  $\mathbf{x}_i$ . We remark that equation (25) has been here derived as a constrained minimisation problem, but it can be also obtained resorting to the virtual work principle, and using the following relation between the virtual displacements:

$$\delta \mathbf{p} = \mathbf{N} \delta \mathbf{p}_\xi + \mathbf{M} \delta \mathbf{P}. \quad (26)$$

This is in fact the extension of equation (13) to situations where the control points of the master surface are additional degrees of freedom.

By using expressions in (16), (17) and (26), the linear part of the extended non-linear equations in (25) may be written as,

$$D \begin{bmatrix} \mathbf{N}^T \nabla_x \mathcal{V} \\ \mathbf{M}^T \nabla_x \mathcal{V} + \nabla_P \mathcal{V} \end{bmatrix} = \begin{bmatrix} \mathbf{N}^T \mathbf{K}_E \mathbf{N} + \mathbf{K}_N & \mathbf{K}_M + \mathbf{N}^T \mathbf{K}_E \mathbf{M} \\ \mathbf{K}_M^T + \mathbf{M}^T \mathbf{K}_E \mathbf{N} & \mathbf{M}^T \mathbf{K}_E \mathbf{M} + \mathbf{K}_P \end{bmatrix} \begin{Bmatrix} D\mathbf{p}_\xi \\ D\mathbf{P} \end{Bmatrix} \quad (27a)$$

with

$$[\mathbf{K}_M]_{ik} = \nabla_\xi I_k(\boldsymbol{\xi}) \otimes (\nabla \mathcal{V})_i \quad (27b)$$

and where  $\mathbf{K}_P$  is the elemental stiffness matrix of the elements that are interpolated with B-Spline interpolation functions. As before, the parametric and Cartesian coordinates are updated in the usual manner  $\mathbf{x}_i^\ell = \mathbf{x}_i^{\ell-1} + D\mathbf{x}_i$ ,  $\boldsymbol{\xi}_i^\ell = \boldsymbol{\xi}_i^{\ell-1} + D\boldsymbol{\xi}_i$ , which is in this case also employed for the control points,  $\mathbf{P}_k^\ell = \mathbf{P}_k^{\ell-1} + D\mathbf{P}_k$ .

We note that, as expected in contact problems, the stiffness matrix in (27) couples the degrees of freedom of the slave node  $\mathbf{x}_i$  with those on the master side,  $\mathbf{P}_k$ , and therefore the sparsity of the whole stiffness matrix is modified throughout the contact analysis. However, the steps indicated in Table 2 for the solution process are still valid for the present case with deformable master surfaces.

#### 4 Slidesurface construction with splines

In standard finite elements, only  $C^0$  continuity of the contact surfaces is usually employed. This fact poses some difficulties in achieving a stable active set. In order to improve this pitfall, the master slidesurface is here interpolated with a cubic  $C^1$  B-Spline. As it will be shown in the numerical examples,  $C^1$

continuity permits to preserve the quadratic convergence characteristic of the Newton-Raphson solution process.

Although geometric entities are commonly modelled with B-Splines or NURBS (Non Uniform Rational B-Splines), this geometrical information is usually lost in the pre-processor and replaced by a mesh information. For this reason, we will next describe an algorithm that reconstructs master B-Splines curves (2D) or surfaces (3D) from a set of elements. The resulting interpolation has the same number of elements, but with different number of nodes per element and connectivity pattern.

#### 4.1 B-Spline interpolation

A brief summary of the form and properties of B-Spline interpolation is given. For further details, the reader is referred to [22]. A B-Spline curve is shaped according to  $n + 1$  *control points*  $\mathbf{P}_1, \dots, \mathbf{P}_{n+1}$ . Each control point  $\mathbf{P}_i$  has associated a basis function  $B_{i,p}(\xi)$ , where  $p$  is the order of the basis functions and are defined over a parametric space  $\xi$ ,  $0 \leq \xi \leq 1$ . In the present case cubic B-Splines are employed, and hence  $p = 3$ . The resulting curve is given by the following interpolation scheme:

$$\mathbf{C}(\xi) = \sum_{i=1}^{n+1} B_{i,p}(\xi) \mathbf{P}_i. \quad (28)$$

The support of each basis function is determined from a *knot vector*  $\mathbf{k}$ , which has  $m + 1$  components or *knot values*. It has the following general form:

$$\mathbf{k}^T = \{\underbrace{0, \dots, 0}_{p+1}, \xi_1, \dots, \xi_{m-2p-1}, \underbrace{1, \dots, 1}_{p+1}\}_{m+1}.$$

The intervals  $[\xi_i, \xi_{i+1})$  are called *knot spans*, which are used to define the basis

functions. The latter are the *Bernstein polynomials*, which are constructed according to the following recursive formulae (here, all knot values in  $\mathbf{k}$  must be considered indexed, regardless they are equal or different, 0 or 1):

$$B_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases},$$

$$B_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} B_{i,p-1}(\xi) + \frac{\xi_{i+p-1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} B_{i+1,p-1}(\xi).$$

The norm of the tangents,  $|\mathbf{C}(\xi)'|$ , is usually called *speed*. B-Spline surfaces are constructed in a similar manner resorting to the tensor product of curves. Given a structured patch of control points  $\mathbf{P}_{i,j}$ ,  $i = 1, \dots, n_1+1$ ;  $j = 1, \dots, n_2+1$ , and the basis  $B_{i,p}$  and  $B_{j,q}$ , the interpolating B-Spline surface is expressed as:

$$\mathbf{S}(\xi_1, \xi_2) = \sum_{i=1}^{n_1+1} \sum_{j=1}^{n_2+1} B_{i,p}(\xi_1) B_{j,q}(\xi_2) \mathbf{P}_{i,j}.$$

We will henceforth assume  $p = q = 3$ . The simplest B-Spline curve of order  $p$  is called a Bézier segment, which has the following knot vector:

$$\mathbf{k} = \{\underbrace{0, \dots, 0}_{p+1}, \underbrace{1, \dots, 1}_{p+1}\}. \quad (29)$$

Some of the the relevant properties of B-Splines are,

- P1. Local support:  $B_{i,p}(\xi) = 0$  for  $\xi \notin [\xi_i, \xi_{i+p+1})$ . For  $p = 3$ ,  $B_{i,3}(\xi) = 0$  for  $\xi \notin [\xi_i, \xi_{i+4})$
- P2. For a given span  $[\xi_i, \xi_{i+1})$ , at most  $p + 1$  (here =4) basis functions are non-zero. (This property follows from P1.)

- P3.  $m = n + p + 1$  (Relation between number basis functions (or control points),  $n + 1$ , and dimension of knot vector,  $m + 1$ ).
- P4. The tangents of a B-Spline curve that passes through a point  $\mathbf{P}_k$  have the directions  $\mathbf{P}_k - \mathbf{P}_{k-1}$  and  $\mathbf{P}_{k+1} - \mathbf{P}_k$ .
- P5. Non-negativity:  $B_{i,p}(\xi) \geq 0, \forall i, p$ .

From property P3, it follows that a Bézier segment of order  $p$ , which has the knot vector in (29), has  $p + 1$  control points, in the present case, 4. Also, due to property P5, the resulting B-Spline interpolation yields smooth curves that start and end at the initial and last control point and that approach the interior control points.

#### 4.2 Transforming FE mesh to B-Spline interpolation

Several algorithms that interpolate a set of points and tangents with a B-Spline curve and a prescribed order of continuity (normally no higher than  $C^1$ ) can be found in the literature (see for instance [22] and references therein). It is shown in the numerical examples below that requiring only  $G^1$  continuity (no change in direction of the tangents, but their norm or speed may be discontinuous) may pose some detrimental effects in the Newton-Raphson iterative process, and therefore,  $C^1$  continuous curves will be constructed. It is worth pointing out that given a set of points,  $C^1$  continuity may be achieved using quadratic B-Splines patches. However, their use is not advised for the following reasons. Firstly it is not possible to additionally prescribe the tangents at all the points. Secondly, it has been numerically tested that after using the averaged tangents at the common nodes between elements, and interpolating the end nodes of a set of quadratic elements, the resulting curve differs substantially from the original unmeshed curve, and last, and in contrast to quadratic B-Splines,

cubic B-Splines can handle inflection points and collinearity without special treatment [22].

The following methodology resorts to the algorithm developed in [27], which is also described in [22], p.395-405. This algorithm constructs a cubic  $C^1$  B-Splines from a set of nodes and tangents, and it is here adapted to interpolate a FE mesh. The details about interpolating a slideline are explained next. The interpolation of slidesurfaces uses a similar technique and just the main differences will be commented.

Given  $n_e$  elements, a set of  $n_e$  Bézier segments are constructed first, from where a B-Spline curve is built. The whole process is summarised in the following steps:

1. Compute the averaged tangents at each common node of two adjacent element as  $\bar{\mathbf{t}} = 0.5(\mathbf{t}^a + \mathbf{t}^b)$  (see Figure 2a for an scheme with 2 elements).
2. For each element  $e$ , construct one cubic Bézier segment  $C^e(\xi^e)$ ,  $0 \leq \xi^e \leq 1$  with 4 control points,  $\mathbf{P}_1^e \dots \mathbf{P}_4^e$ , and such that  $|\mathbf{C}^e(0)'| = |\mathbf{C}^e(1/2)'| = |\mathbf{C}^e(1)'|$  (see Figure 2b). These segments go through the end-nodes of each element. In the reduced model with two elements in Figure 2a, this is equivalent to saying  $\mathbf{N}_1 \equiv \mathbf{P}_1^a$ ,  $\mathbf{N}_3 \equiv \mathbf{P}_4^a \equiv \mathbf{P}_1^b$  and  $\mathbf{N}_5 \equiv \mathbf{P}_4^b$ . However, if quadratic finite elements are initially employed, the segment does not pass through the mid-nodes  $\mathbf{N}_2$  and  $\mathbf{N}_4$ . Instead, the averaged tangents  $\bar{\mathbf{t}}$  are the same (in direction) of the Bézier segment. It is shown in [22], p.396-397, that such construction is unique. Due to property P4, the position of the central control points  $\mathbf{P}_2^e$  and  $\mathbf{P}_3^e$  is given by

$$\mathbf{P}_2^e = \mathbf{P}_1^e + \alpha \bar{\mathbf{t}}_1^e \quad ; \quad \mathbf{P}_3^e = \mathbf{P}_4^e - \alpha \bar{\mathbf{t}}_4^e,$$

where  $\bar{\mathbf{t}}_1^e = \bar{\mathbf{t}}_4^{e-1}$  and  $\bar{\mathbf{t}}_4^e$  are the averaged tangents, and  $\alpha$  is the positive solution of the following equation [22]:

$$\alpha^2 a + \alpha b + c = 0$$

with  $a = 16 - |\bar{\mathbf{t}}_4^e + \bar{\mathbf{t}}_1^e|^2$ ,  $b = 12(\mathbf{P}_4^e - \mathbf{P}_1^e) \cdot (\bar{\mathbf{t}}_4^e + \bar{\mathbf{t}}_1^e)$  and  $c = -36|\mathbf{P}_4^e - \mathbf{P}_1^e|^2$ .

3. Compute a set of knot values  $\bar{\xi}^{e+1} = \bar{\xi}^e + 3|\mathbf{P}_2^e - \mathbf{P}_1^e|$ . In this way, the  $n_e$  Bézier segments have unit speed with respect to the intervals  $[\bar{\xi}^e, \bar{\xi}^{e+1}]$ .
4. Remove the internal common nodes,  $\mathbf{P}_4^1 \dots \mathbf{P}_4^{n_e-1}$  (Figure 2c), and use the following knot vector:

$$\mathbf{k} = \{0, 0, 0, 0, \frac{\bar{\xi}_1}{\xi_{n_e}}, \frac{\bar{\xi}_1}{\xi_{n_e}}, \frac{\bar{\xi}_2}{\xi_{n_e}}, \frac{\bar{\xi}_2}{\xi_{n_e}}, \dots, \frac{\bar{\xi}_{n_e-1}}{\xi_{n_e}}, \frac{\bar{\xi}_{n_e-1}}{\xi_{n_e}}, 1, 1, 1, 1\}$$

which has total dimension  $2(n_e+3)$ . Obviously, since  $n = 2n_e+1$ , property P3 holds.

5. Each new element  $e$  is defined by the knot span  $\xi \in [\xi_{2e+4}, \xi_{2e+5})$ , and by 4 control points (which will be also called nodes):  $\mathbf{P}_3^{e-1}$ ,  $\mathbf{P}_2^e$ ,  $\mathbf{P}_3^e$  and  $\mathbf{P}_2^{e+1}$ . A exception to this rule is the first and last element, which have the following nodes (see Figure 2c):

- First element:  $\mathbf{P}_1^1$ ,  $\mathbf{P}_2^1$ ,  $\mathbf{P}_3^1$  and  $\mathbf{P}_2^2$
- Last element:  $\mathbf{P}_3^{n_e-1}$ ,  $\mathbf{P}_2^{n_e}$ ,  $\mathbf{P}_3^{n_e}$  and  $\mathbf{P}_4^{n_e}$

As a result of the steps indicated above, the whole set of elements in the slideline have been re-parametrised via a parameter  $\xi$ ,  $0 \leq \xi \leq 1$ . However, the cubic curve has been defined using a new set of  $n_e$  elements which are defined through the knot spans. In doing this, and due to property P2, each element can be associated to 4 nodes (corresponding to the 4 non-zero basis functions). In contrast to standard meshes with finite elements, the nodal connectivity of the new elements overlaps, i.e. adjacent elements have more than one common node.

A similar  $C^{(1,1)}$  continuous construction can be developed for B-Spline surfaces. In contrast to the curves defined above, which have unit speed, the resulting surfaces have variable speed (for instance, for each  $\xi_2$ , the derivative

$\frac{\partial \mathcal{S}}{\partial \xi_1}$  has a speed equal to the total chord of the surface in the  $\xi_1$  direction, see [22]). Each new element has 16 nodes. Figure 3 shows the resulting connectivity associated to 4 initial quadratic elements.

### 4.3 B-Spline interpolation of elastic domains

In order to allow the elastic deformation of a domain, while preserving the smooth B-Spline interpolation of the domain, two main approaches may be adopted: (i) to fully discretise the domain with B-Spline basis, or (ii) use a mixed interpolation for the elements at the boundaries: standard Lagrangian basis at the interior of such elements, and B-Splines for the boundaries. We have chosen here the first approach for simplicity. Indeed, in two-dimensional problems, it is sufficient to resort to the previous B-Spline construction of the domain (surface reconstruction for two-dimensional problems), and use the B-Spline functions  $I^k(\boldsymbol{\xi})$  for the shape and test functions when formulating the discretised weak form of the elastic problem.

We note that the cubic B-Spline interpolation of the continuum parallels standard cubic FE formulation, and only the basis functions and the connectivity must be modified. Due to the degree of the polynomials, 3 Gauss point quadrature must be employed along each dimension. However, in contrast to FE elements, the parent domain of the B-Spline elements belongs to a unique parent domain  $\boldsymbol{\xi} \in [0, 1]^{n_{sd}}$  for the *whole domain*, and thus the location of the Gauss points of each element is different. Therefore, an additional storage of the elemental shape function values is needed (but no additional computational cost is required). The study of B-Spline interpolation of continua for non-linear elastic problems is out of the scope of this paper, but we recognise that further analyses are necessary in order to fully assess the proposed for-

mulation in contact problems. We have here implemented it to show that the essence of the null-space method and cubic B-Splines can be also combined for contact problems between two elastic bodies.

We shall though comment two disadvantages of the proposed approach in elastic-elastic contact. First, the deformation of the domain, and hence the variation of the control point positions  $\mathbf{P}_k$ , spoils the  $C^1$  continuity achieved in the previous subsection. Nonetheless,  $G^1$  continuity is still maintained, and, as it will be shown in the numerical examples, this is still a non negligible gain with respect to standard FE interpolations. Second, the fact that the control points are in general not exactly located on the master surface prevents the direct implementation of two-pass algorithms (the role of the contactor and target surfaces are shifted in the second pass). Indeed, the control points cannot be used as slave nodes. Alternatively, a set of nodes located on the master surface could be used as slave nodes, where the location of such nodes would in turn depend on the control point coordinates. This venue implies a rather different structure of the null-space method described here, but definitely requires further investigation, specially because two-pass algorithms have been shown to overconstrain and worsen the inf-sup condition in contact problems when using standard FE interpolations [8].

## 5 Numerical tests

### 5.1 Two-spring system

In order to illustrate the effects of using a  $C^1$  function for the master surface, the 2D two-spring system depicted in Figures 4a and 4b is analysed. A prescribed displacement  $\bar{\mathbf{u}}_0 + \boldsymbol{\delta}$  is imposed at the upper nodes of the springs in

one load-step, with  $\bar{\mathbf{u}}_0^T = \{-0.25, -0.1\}$  and  $\boldsymbol{\delta}^T = \{0.01, 0\}$ . The two springs have an elastic potential that is quadratic to their length and with a constant stiffness equal to 1.

In a first set of runs, the master slideline, which is the horizontal line in Figures 4a and 4b, has been modelled with two equal and unequal finite elements, and in a second run the master slideline has been interpolated using a single B-Spline. Convergence is accepted when the  $L_2$ -norm of the residual is lower than the convergence tolerance  $\epsilon = 10E - 10$ . In the notation used here, the residual corresponds to the vector  $\mathbf{N}^T \nabla \mathcal{V}(\mathbf{x})$ .

The solution is a deformed configuration with the common node of the springs located at a distance  $\|\boldsymbol{\delta}\| = 0.01$  off the mid-point of the master slideline. The evolution of the residual is plotted during the Newton-Raphson iterations. In the units of the graph, quadratic convergence is revealed by a slope equal to  $\log 2 \approx 0.3$ . When using equal elements, the FE and B-Spline interpolation yield very similar results, showing the usual quadratic convergence (see Figure 5a). However, the FE interpolation requires one additional iteration when using unequal element sizes (see Figure 5b). Obviously, for the straight master slideline used here, the same results are obtained when using linear and quadratic finite elements.

The convergence problems are much severe when using a curved master slideline, see Figure 6. When attempting to impose only the displacement  $\bar{\mathbf{u}}_0$ , the B-Spline interpolation requires 3 iterations, whereas the FE interpolation with equal elements is unable to converge. In the present implementation, when convergence is not achieved after 10 iterations, the prescribed incremental displacement is halved. If convergence is achieved eventually in less than 10 iterations, the incremental displacement is multiplied by 1.5. Using this

strategy, the interpolation with quadratic elements manages to converge for a prescribed displacement  $\bar{\mathbf{u}} = 1.25\bar{\mathbf{u}}_0$  (see Figure 7a). However, whenever the solution is searched for  $\bar{\mathbf{u}} = \bar{\mathbf{u}}_0$ , no convergence can be achieved. The source of this impossibility can be found in Figure 7b. It reveals the chatter effect: the  $x$ -coordinate of the slave node alternates between two values at each side of the equilibrium position. Although the difference between the values is minimal for the quadratic elements, the discontinuity in the direction (not the speed though), prevent the analysis to converge. Although some artifacts could be thought to palliate the chatter when using finite elements, the use of B-Splines interpolation furnishes already optimal convergence without further treatment.

## 5.2 Contact on elastic beam

In this example we analyse the contact of a single squared linear element rotated 45 degrees, against a deformable casted beam, which is discretised with  $3 \times 6$  elements. Figure 8a depicts the initial geometry, the mesh and the material parameters employed. In this figure,  $\lambda$  and  $\mu$  denote the Lamé constants of a Neo-Hookean hyperelastic material used for both domains, the beam and the square. The latter has a prescribed displacement  $\mathbf{u}_0^T = \{-0.43 - 0.15\}$  imposed at the two top straight lines of the element boundary. The analysis is run using 20 equal displacement increments of  $0.05\mathbf{u}_0$ .

In the present case, the slave node corresponds to the bottom node of the rotated squared, and the master slideline is the top horizontal boundary of the beam, which in this case is elastic, and thus tests the extension of the formulation for elastic-elastic contact. The example has been run using the following three discretisation for the whole beam: (i) linear elements, (ii) quadratic el-

elements and (iii) cubic B-Spline elements. In the three cases, the results yield very similar conclusions to those in the previous example. The linear and quadratic FE interpolation could not achieve converged results after a certain contact transition between elements on the master slideline. Figures 9a and 9b show the last converged deformations when using linear and quadratic elements, respectively (surprisingly, the slave node manages to slide along more elements in the linear case). Figure 8b shows the deformed configuration after the last step when using cubic B-Spline interpolation, which requires between 4 and 7 iterations in all the increments when using a convergence tolerance of  $\epsilon = 1.0E - 10$  (full quadratic convergence was obtained).

### 5.3 *Punch indentation*

A prism with dimensions  $100 \times 100 \times 50$  is modelled and meshed with  $24 \times 24 \times 12$  linear hexahedra. A neo-Hookean hyperelastic material [4] is considered with parameters  $\lambda = \mu = 1$ . The prism is indented with a rigid spherical cap (master surface), which is vertically displaced 5 units (see Figure 10a).

This problem has been analysed in [5] using mortar methods and a double loop strategy: an inner loop for the Newton-Raphson iterations and a second loop for the detection of active and inactive constraints. In the present case, since the contact detection is performed after each update, it can be said that both loops are merged in one single loop.

This example is used to show that quadratic convergence is obtained despite considering one single loop for contact detection and solution process. The problem has been run using a convergence tolerance  $\epsilon = 10E - 15$  and imposing the prescribed displacement of 5 units in one single step and in two steps of 2.5 units each. As Figure 11 shows, quadratic convergence is achieved during

the last iterations. Furthermore, Table 3 also indicates that no variations in the contact state of the constraints are detected during the last 2-4 iterations.

#### 5.4 *Ironing problem*

This example is a slightly modified version of the ironing problem in [24,26]. A cylindrical rigid die with radius 3 slides onto a 9x4x3 prism. The die, which in our analysis is the master surface, is initially in contact with the upper surface of the prism, with its axis located at 2.5 units from the left end of the prism. The die descends 1.4 units in the  $z$  direction during the first 0.2 seconds (a pseudo-time  $\tau$  is considered as a measure of the applied displacements). During the following 3 seconds the die slides at 4 units/1.3 seconds, until it reaches the end of the prism, which occurs at  $\tau = 3.2$  seconds. The prism is considered here a neo-Hookean hyperelastic material with Lamé constants  $\mu = \lambda = 1$ .

The initial geometry and mesh is indicated in Figure 12a, and Figures 12b-12e show the deformation of the prism at pseudo-times  $\tau = 0.2, 1.0, 2.95$  and 3.0. It is worth remarking that between the last two plotted time-steps, the right end of the prism snaps upwards, as the plot of node  $B$  shows. The analysis is run with a constant pseudo-time step size  $\Delta\tau = 0.05$ , which is able to capture the strong discontinuity in the displacements.

#### 5.5 *Falling rod into hemisphere*

This is a 3D problem undergoing large displacements and deformations. A vertical prism with dimensions  $0.2 \times 0.2 \times 1.5$  is located with the longer axis aligned with the  $Z$  axis, and with the centre of the bottom face at point

$\{-0.1, -0.3, 0\}$  (see Figure 14a). A rigid hemisphere with diameter 1.0 is situated below the prism, with its centre at  $\{0, 0, 0\}$ . The prism is modelled using  $2 \times 2 \times 15$  linear hexahedra and as a neo-Hookean hyperelastic material with parameters  $\lambda = \mu = 10$ , whereas the hemisphere is the master surface. Two meshes have been used to discretise the relevant part of the hemisphere (a sector 0.8 units width): a first one using 5 elements and a second one using 4 elements.

The top face of the prism has the displacement prescribed with  $\bar{u}_x = \bar{u}_y = 0$  and incremental displacements in the  $z$  direction equal to  $\Delta\bar{u}_z = -0.05$ . Figure 14a depicts the initial mesh and geometry of the model, and Figures 14b and 14c show the deformed geometry at  $\bar{u}_z = -0.8$  and  $\bar{u}_z = -1.6$ , using the B-Spline interpolation of the hemisphere. Post-processors do not have in general the capability to draw B-Splines, and therefore it is plotted as a set of quadratic elements. Due to this discrepancy between the numerical model and the post-processing, in some of the figures it may appear that the constraints are violated, although it has been verified that these are exactly satisfied.

While the model with B-Spline interpolation could be run without any convergence problems (all the displacement increments took 7 or less iterations), when the master surface was interpolated with quadratic elements the analysis could not converge for the values  $\bar{u}_z = -0.72$  and  $\bar{u}_z = -0.55$  when using 5 and 4 elements per side of the master surface, respectively. This was due to the discontinuities of the tangents at the boundaries of the master elements. Figure 15 plots the parametric coordinates of the nodes initially at  $\{-0.2, -0.2, 0\}$  and at  $\{0, -0.2, 0\}$ , from the instant that these nodes start sliding. It can be indeed deduced from this plot that the difficulties arise when the node slides through the boundaries of the master elements. The dotted lines in the plot indicate the boundaries of the elements in the parametric

space. The trajectory followed by the nodes on the quadratic elements has abrupt changes precisely on these dotted lines. Contrarily, the trajectory of the slave node when using B-Spline interpolation crosses the dotted lines without any difficulty, and although the B-Splines obtained when using 4 and 5 elements per side are different, the trajectories of the slave nodes in the two cases have similar trends.

We remark that during this example, contact situations such as nodes-on-surface, line-on-surface and surface-on-surface are successfully reproduced. Moreover, the method is able to pursue the analysis when the nodes slide off the master surface, as shown in Figure 14c. In this case, a line-search algorithm [4] was also implemented in conjunction with the Newton-Raphson process in order to achieve convergence.

## 6 Conclusions

The essence of the null-space method has been applied to model rigid-elastic and elastic-elastic contact problems. The version of the method presented here inherits the ideas employed in the analysis of mechanisms, namely the master-slave method.

The resulting formulation has the advantage of solving a system with the minimum number of unknowns. This is in contrast to methods with Lagrange multipliers, where the latter must be added to the kinematic variables. The cost of reducing system is minimal and can be performed node by node during the assembling process.

Unilateral contact is specially challenging due to the constant switches in the state of the constraints. It has been numerically demonstrated that by using

a B-Spline interpolation of the slidesurface, the chatter effect is inexistent. This re-parametrisation is specially suited for the null-space method considered here. Indeed, in the B-Spline interpolation the slidesurface is globally parametrised and  $C^1$  continuous for rigid-elastic contact, and  $G^1$  for elastic-elastic contact. In this way, special treatments in element transitions have been avoided, and the quadratic convergence of the Newton-Raphson process has been maintained. In addition, the algorithm is able to model contact situations where the slave nodes slide off the master surface, or where contact is progressively affecting nodes, lines and surfaces.

Pre- and post-processing are not yet at the stage of handling numerical entities with B-Spline interpolation. For this reason, a specific conversion tool has been designed which transforms structured meshes with quadratic elements into B-Splines. The need for structured meshes in the master side is in fact required by this conversion tool, but not by the projection null-space algorithm. We note that since no requirement on the slave side exists, the method accepts remeshing strategies on this side, but not in the master side. In any case, if the exact geometry (control points and knot vectors) is available at the computational level, the need for structured meshes may be relaxed. However, the use of remeshing strategies within B-Spline interpolations, even for structured meshes, is still an ongoing research topic (see some recent developments in the context of NURBS interpolation in [7])

We remark that B-Spline polynomial interpolation may not be sufficient to represent certain rational curves such as circles or ellipses. Nonetheless, non-rational splines have a lower cost and can be integrated exactly, in contrast to NURBS (rational splines) [11]. In the present study non-rational splines have been used in order to demonstrate the advantages of the  $C^1$  continuity at the boundary in certain situations. The modelling of deformable master

surfaces requires to modify the equilibrium equations associated to the control points on the master surface and the linearisation process. However, in the approach adopted here, the direct use of the cubic B-Spline basis functions yields already a formulation that outperforms the standard Lagrangian interpolation in elastic-elastic contact problems. However, the approach, as it has presented here, does not allow to use the boundary interpolated with B-Splines as the slave side. Some suggestions to alleviate this drawback have been proposed, which parallel some of the ideas described in [28] for other physical problems. This venue is as yet left for future work. Finally, we mention that the projection of the motion equations onto an unconstrained sub-manifold is specially advantageous in dynamic analysis. Indeed, by circumventing the contact constraints, their associated infinite eigenfrequencies are also avoided. The extension to dynamic analysis of the smoothing technique presented here remains as yet unexplored.

## 7 Acknowledgements

The author is financially supported by the Spanish Research Program Juan de la Cierva. This support is greatly acknowledged.

## References

- [1] F Armero and E Petőcz. Formulation and analysis of conserving algorithms for frictionless dynamic contact/impact problems. *Comp. Meth. Appl. Mech. Engng.*, 158:269–300, 1998.
- [2] T Belytschko, WJT Daniel, and GK Ventura. A monolithic smoothing-gap algorithm for contact-impact based on the signed distance function. *Int. J.*

- Num. Meth. Engng.*, 55:101–125, 2002.
- [3] P Betsch. The discrete null space method for the energy consistent integration of constrained mechanical systems. Part I: Holonomic constraints. *Comp. Meth. Appl. Mech. Engng.*, 194:5159–5190, 2005.
- [4] J Bonet and RD Wood. *Non-linear continuum mechanics for finite element analysis*. Cambridge University Press, 1997.
- [5] S Brunssen, F Schmid, M Sch afer, and B Wohlmuth. A fast and robust iterative solver for nonlinear contact problems using a primal-dual active set strategy and algebraic multigrid. *Biol. Cell*, 96:471–477, 2004.
- [6] RW Clough and TL Tocher. Finite element stiffness matrices for analysis of plates in bending. In *Proceedings of the Conference on Matrix Methods in Structural Mechanics, Air Force Institute of Technology, Wright Patterson A.F. Base, Ohio, USA, October 1965*.
- [7] J A Cottrell, T J R Hughes, and A Reali. Studies of refinement and continuity in isogeometric structural analysis. *Comp. Meth. Appl. Mech. Engng.*, 196:4160–4183, 2007.
- [8] N El-Abbasi and K J Bathe. Stability and patch test performance of contact discretizations and a new solution algorithm. *Comput. Struct.*, 79:1473–1486, Prentice Hall 2001.
- [9] N El-abbasi, S A Meguid, and A Czekanski. On the modelling of smooth contact surfaces using cubic splines. *Int. J. Num. Meth. Engng.*, 50:953–967, 2001.
- [10] S Hübner and BI Wohlmuth. A primal-dual active set strategy for non-linear multibody contact problems. *Comp. Meth. Appl. Mech. Engng.*, 194:3147–3166, 2005.
- [11] TJR Hughes, JA Cottrell, and Y Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comp. Meth. Appl. Mech. Engng.*, 194:4135–4195, 2005.

- [12] G Jelenić and J J Muñoz. *Multibody dynamics. Computational methods and applications*. Ed: JC García Orden et. al. Computational Methods in Appl. Sc. Springer-Verlag, 2007.
- [13] P Kagan, A Fischer, and P Z Bar-Yoseph. New b-spline finite element approach for geometrical design and mechanical analysis. *Int. J. Num. Meth. Engng.*, 41:435–438, 1998.
- [14] N Kikuchi. A smoothing technique for reduced integration penalty method in contact problems. *Int. J. Num. Meth. Engng.*, 18:343–350, 1982.
- [15] TW McDevitt and TA Laursen. A mortar-finite element formulation for frictional contact problems. *Int. J. Num. Meth. Engng.*, 48:1525–1547, 2000.
- [16] J Muñoz. On the modelling of incompressibility in linear and non-linear elasticity with the master-slave approach. *Int. J. Num. Meth. Engng.*. Accepted.
- [17] J J Muñoz and G Jelenić. Sliding contact conditions using the master-slave approach with application on the geometracally non-linear beams. *Int. J. Solids Struct.*, 41:6963–6992, 2004.
- [18] J J Muñoz, G Jelenić, and M Crisfield. Master-slave approach for the modelling of joints with dependent degrees of freedom in flexible mechanisms. *Comm. Num. Meth. Engng.*, 19:689–702, 2003.
- [19] J Nocedal and S J Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer-Verlag, 1999.
- [20] V Padmanabhan and T A Laursen. Surface smoothing procedure for large deformation contact analysis. *Finite Elem. Anal. Des.*, 37(2):173–198, 2001.
- [21] F Pfeiffer and C Glocker. *Multibody dynamics with unilateral contacts*. Wiley Series in Nonlinear Sc. Wiley & Sons, 1996.
- [22] L Piegl and W Tiller. *The NURBS book*. Monographs in Visual Comm. Springer-Verlag, 2nd edition, 1997.

- [23] G Pietrzak and A Curnier. Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment. *Comp. Meth. Appl. Mech. Engng.*, 177:351–381, 1999.
- [24] M A Puso. An energy and momentum conserving method for rigid–flexible body dynamics. *Int. J. Num. Meth. Engng.*, 53:1393–1414, 2002.
- [25] M A Puso and T A Laursen. A 3D contact smoothing method using Gregory patches. *Int. J. Num. Meth. Engng.*, 54:1161–1194, 2002.
- [26] M A Puso and T A Laursen. A mortar segment-to-segment contact method for large deformation solid mechanics. *Comp. Meth. Appl. Mech. Engng.*, 93:601–629, 2004.
- [27] G Renner. A method of shape description for mechanical engineering practice. *Comput.in Ind.*, 3:137–142, 1982.
- [28] R Sevilla, S Fernández-Méndez, and A Huerta. NURBS-Enhanced Finite Element Method (NEFEM). Submitted, 2007.
- [29] M Stadler and GA Holzapfel. Subdivision schemes for smooth contact surfaces of arbitrary mesh topology in 3D. *Int. J. Num. Meth. Engng.*, 60:1161–1195, 2004.
- [30] B Fraeijns De Veubeke. A conforming finite element for plate bending. *Int. J. Solids Struct.*, 4:95–108, 1968.
- [31] P Wriggers. *Computational Contact Mechanics*. Wiley, 2002.
- [32] P Wriggers and L Krstulovic-Opara. On smooth finite element discretization for frictional contact problems. *J. Appl. Math. Mech. (ZAMM)*, 80:77–80, 2000.
- [33] P Wriggers, L Krstulovic-Opara, and J Korelc. Smooth C1-interpolations for two-dimensional frictional contact problems. *Int. J. Num. Meth. Engng.*, 51:1469–1495, 2001.

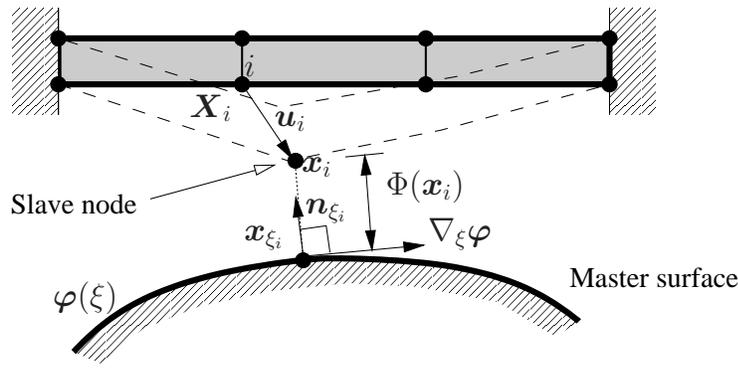


Fig. 1. Illustration of contact constraints.

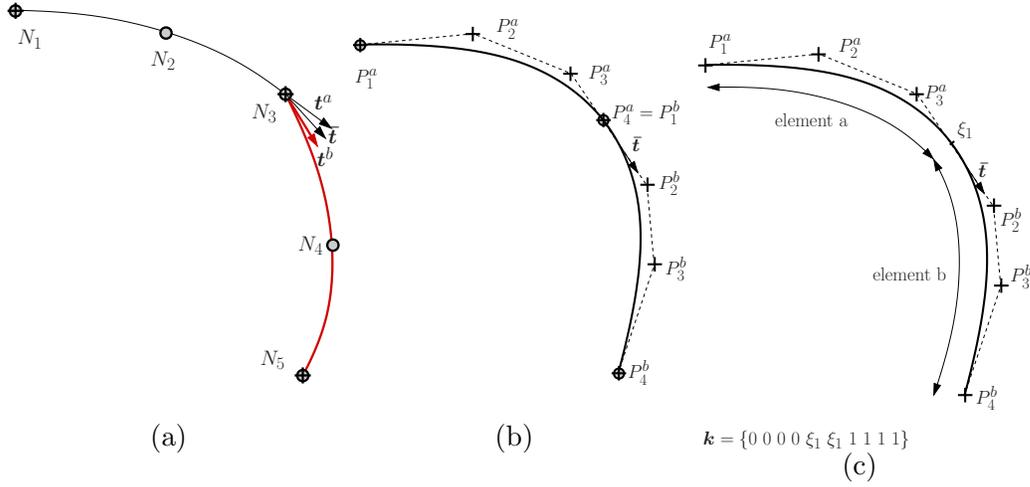


Fig. 2.  $C^1$  spline construction from 2 quadratic elements. Initial elements (a), intermediate control points and B-Spline curve (b), and final control points and knot vector (c).

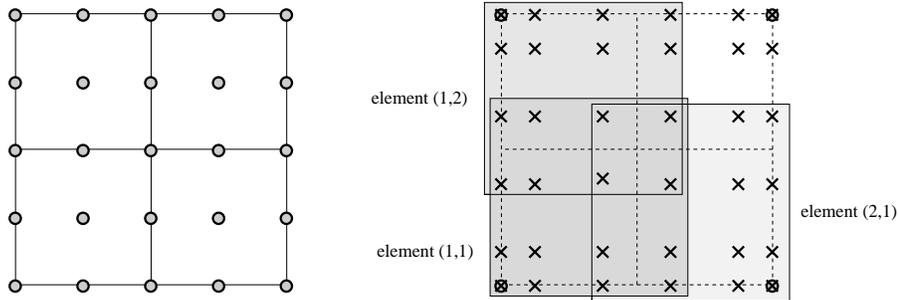


Fig. 3.  $C^1$  B-Spline surface construction from 4 quadratic 9-noded finite elements. Initial elements (a) and 3 of the 4 resulting *spline* elements (b).

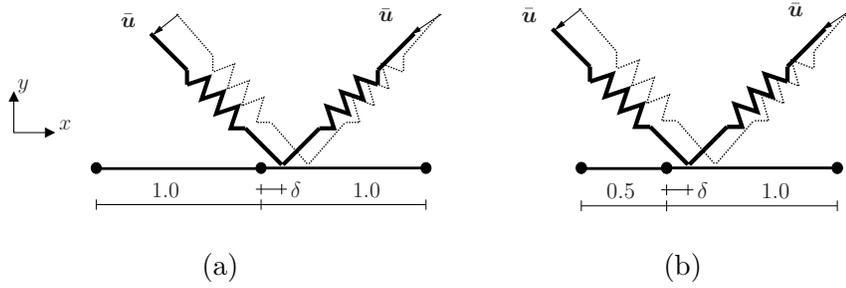


Fig. 4. Two-spring system with equal (a) and unequal (b) elements. Initial position is plotted with thin lines, and final position at equilibrium with thick lines.

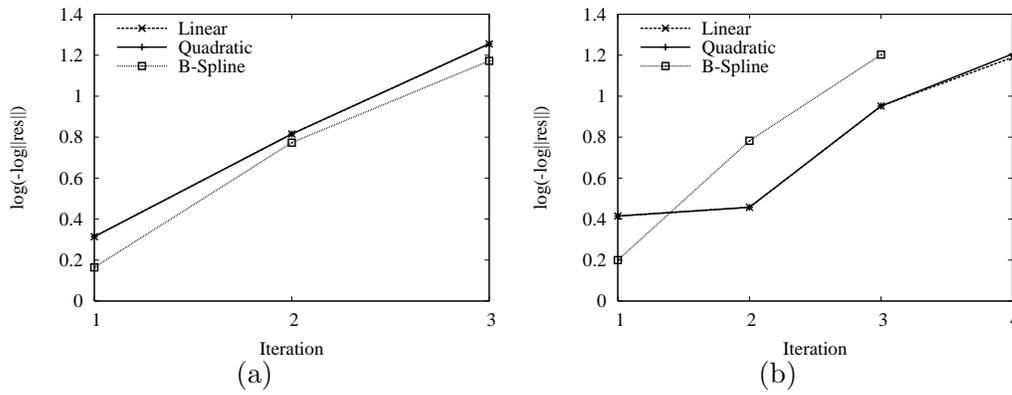


Fig. 5. Two-spring system: Residual norm using equal (a) and unequal (b) master elements.

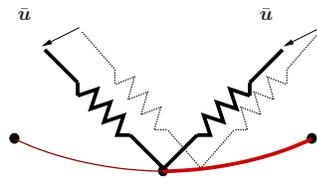
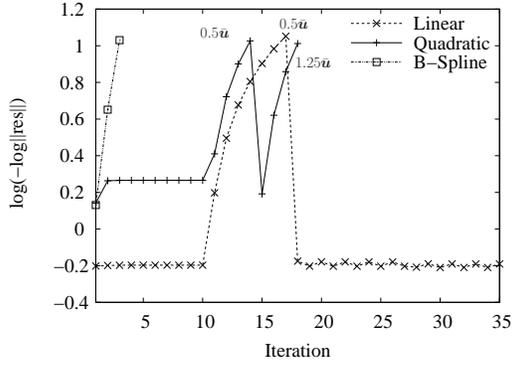
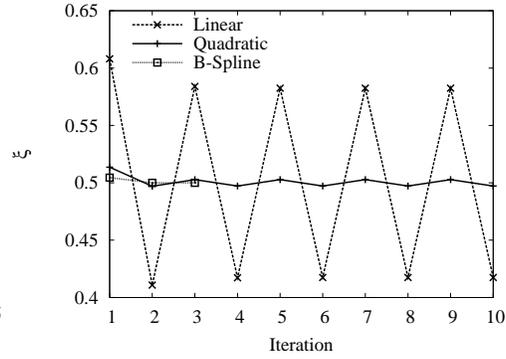


Fig. 6. Two spring system with curved slideline: initial (thin lines) and final deformed configuration (thick lines).

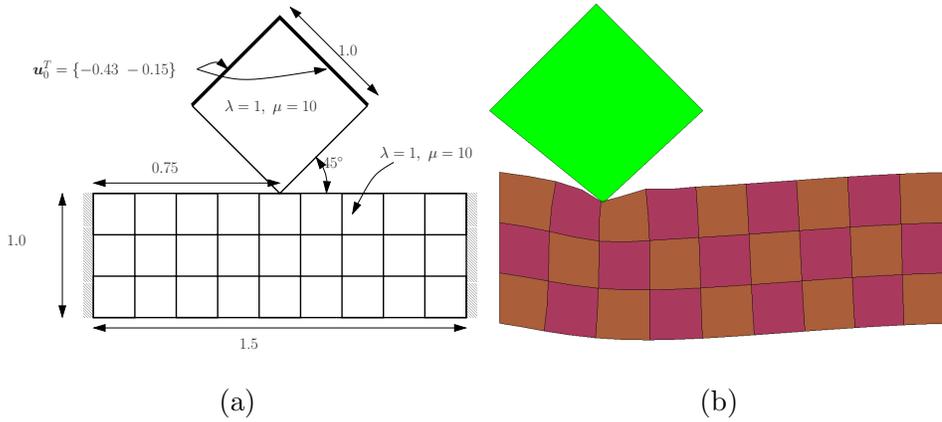


(a)



(b)

Fig. 7. Two spring system with curved slideline: residual norm (a) and parametric coordinate (b) for each iteration of the solution process.



(a)

(b)

Fig. 8. Contact on elastic beam: initial geometry, mesh and material parameters (a), and deformed mesh when using cubic B-Spline interpolation (b).

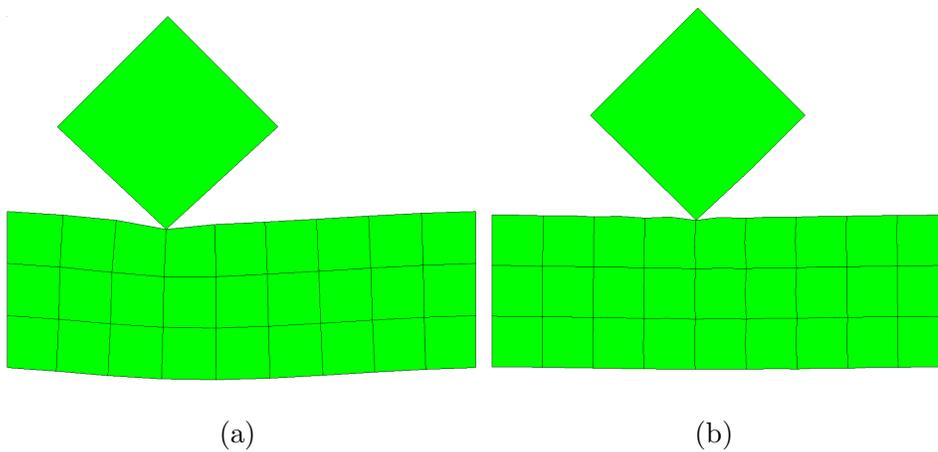


Fig. 9. Contact on elastic beam: last converged deformed mesh when using linear (a) and quadratic (b) elements for the beam.

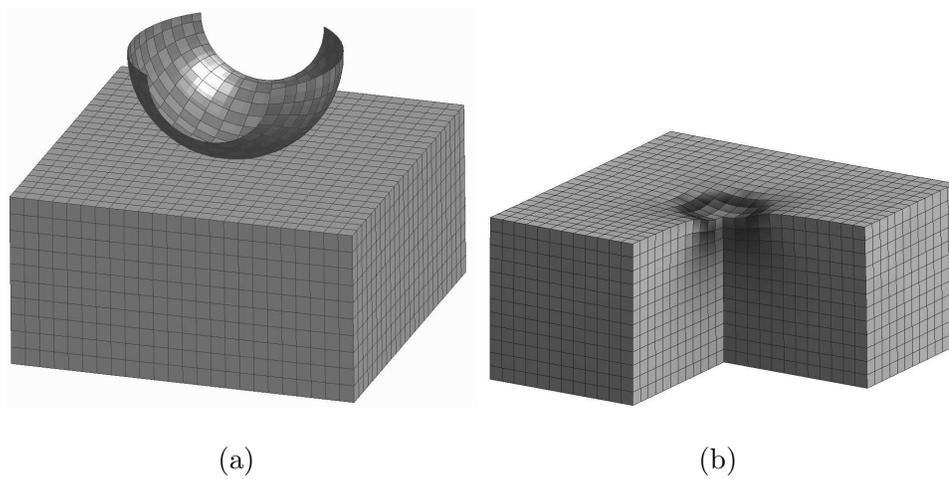


Fig. 10. Punch problem: initial mesh (a) and deformed mesh with Von Mises equivalent stresses (b).

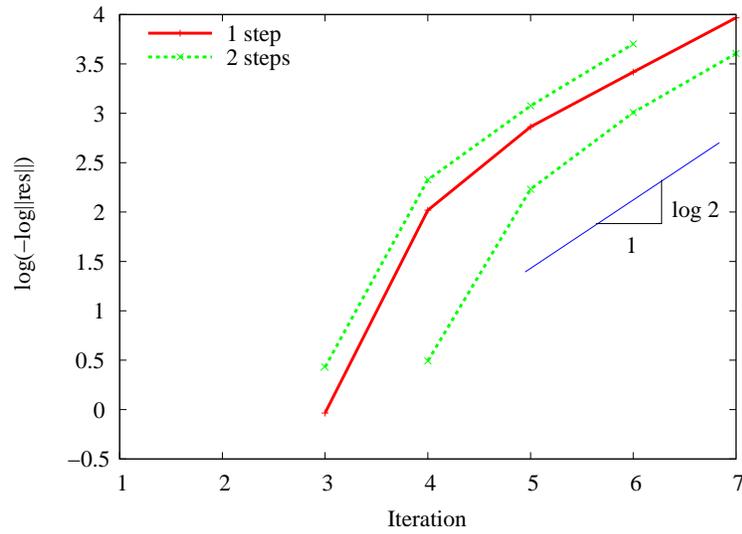


Fig. 11. Punch problem: Evolution of the  $L_2$  norm of the residual when applying 1 single steps or 2 steps (values for which the logarithm is not defined are not plotted).

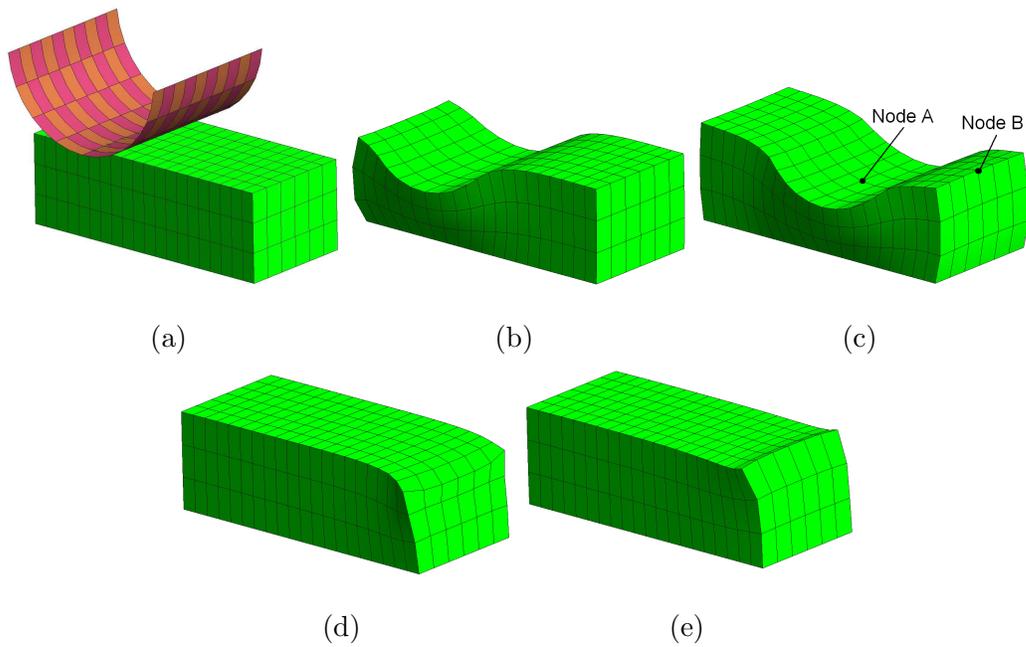


Fig. 12. Ironing problem: Initial geometry (a), and deformations at  $\tau = 0.2$  (b),  $\tau = 1.0$  (c),  $\tau = 2.95$  (d) and  $\tau = 3.0$  (e).

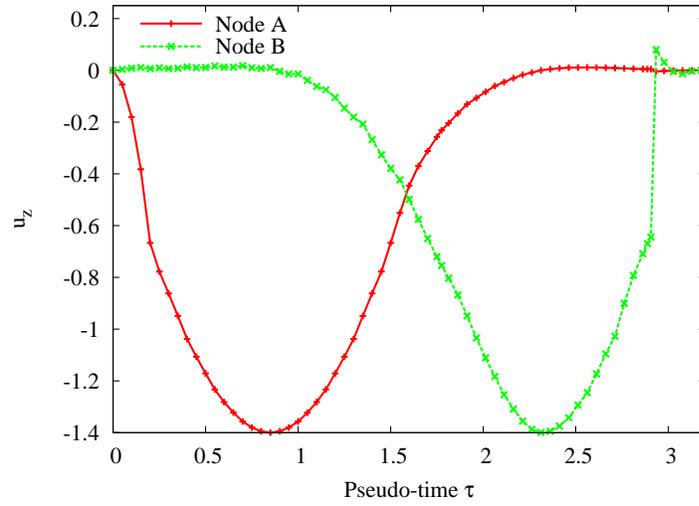


Fig. 13. Ironing problem: Deformation of nodes A and B indicated in Figure 12c.

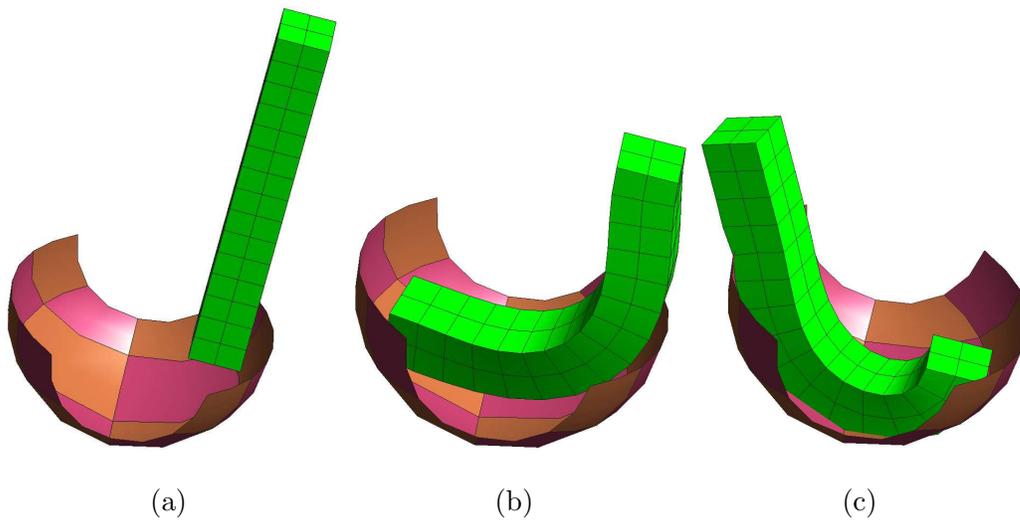


Fig. 14. Falling rod into a hemisphere: initial geometry (a) and two deformed configurations at  $\bar{u}_z = -0.8$  (b) and  $\bar{u}_z = -1.6$  (c).

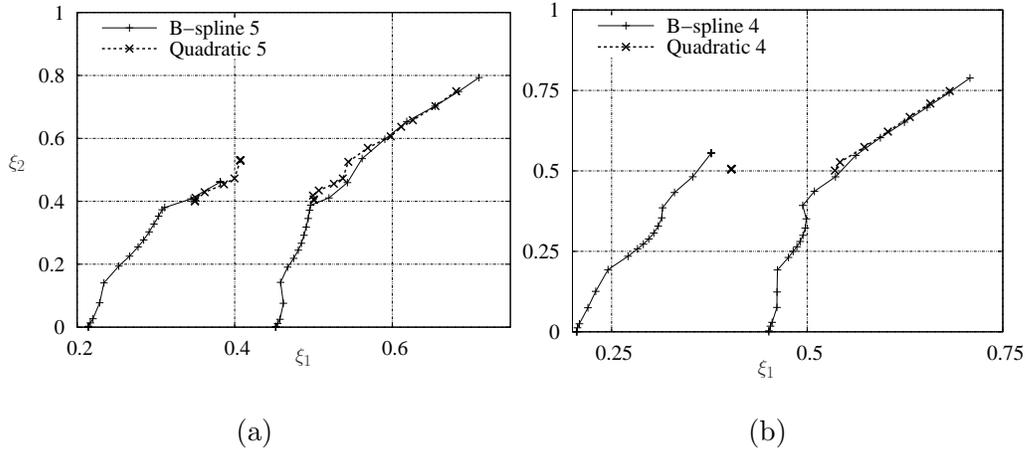


Fig. 15. Falling rod into hemisphere: trajectory in parametric space of the contact points initially at  $\{-0.2, -0.2, 0.0\}$  (curves on the right) and at  $\{0.0, -0.2, 0.0\}$ , (curves on the left). The trajectories of each point are plotted when using quadratic and B-Spline interpolation, and interpolating the hemisphere with 5 (a) and 4 (b) elements in each direction.

$$i \in \mathcal{A}(\mathbf{x}) \left\{ \begin{array}{l} \lambda_i < 0 \text{ Set constraint as inactive.} \\ \lambda_i \geq 0 \text{ Leave constraint as active.} \end{array} \right.$$

$$i \in \mathcal{I}(\mathbf{x}) \left\{ \begin{array}{l} \Phi_i \leq 0 \text{ Set constraint active.} \\ \Phi_i > 0 \text{ Leave constraint as inactive.} \end{array} \right.$$

Table 1

Criteria used to set active and inactive constraints

Iteration  $k$ :

1. Update parametric and Cartesian coordinates as,

$$\boldsymbol{\xi}_i^\ell = \boldsymbol{\xi}_i^{k-1} + D\boldsymbol{\xi} \text{ and } \mathbf{x}_i^\ell = \mathbf{x}_i^{k-1} + D\mathbf{x}_i.$$

2. Compute Cartesian coordinates of slave nodes:  $\mathbf{x}_{\xi_i}^\ell = \boldsymbol{\varphi}(\boldsymbol{\xi}_i^\ell)$ ,  $i \in \mathcal{A}(\mathbf{x})^\ell$ .

3. Compute  $\Phi_i(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_i}$  and  $\lambda_i$ ,  $i = 1, \dots, n_c$ , Eqns. (2) and (12).

4. Update active and inactive constraint sets  $\mathcal{A}(\mathbf{x})^\ell$  and  $\mathcal{I}(\mathbf{x})^\ell$  according to Table 1.

5. If a constraint is activated, project Cartesian coordinates onto master surface by solving Eqn. (3).

6. Built matrix  $\mathbf{N}$  (Eqn. (8)), stiffness matrix  $\mathbf{K}$  (Eqn. (18)), and reduced system of equations (6).

7. Solve system of equations and obtain iterative displacements  $D\mathbf{p}_\xi$ .

8. Check convergence:  $\|\mathbf{N}^T \nabla \mathcal{V}(\mathbf{x})\| < \epsilon$ ?

- Yes: Go to next load step and set  $k = 1$ .
- No: Set  $k = k + 1$  and proceed to next iteration in step 1.

Table 2

Steps performed at each iteration of the solution process

	1 Step		2 Steps			
Iter	$\ res\ $	CC	$\ res\ $	CC	$\ res\ $	CC
1	1.775E+02	3	1.820E+02	7	9.467E+01	-
2	1.136E+01	9	1.232E+01	5	6.918E+01	9
3	3.809E-01	9	2.151E-01	3	5.956E+00	9
4	5.406E-04	6	3.569E-05	-	1.946E-01	8
5	2.500E-08	-	3.988E-10	-	9.015E-05	1
6	5.830E-14	-	2.438E-18	-	1.594E-09	-
7	1.124E-23	-		-	1.068E-16	-

Table 3

Punch problem: Values of the  $L_2$  norm of the residual when applying 1 single steps or 2 steps. CC indicates the number of changes detected in the state of the contact constraints.