

# On solving frictional contact problems: formulations and comparisons of numerical methods.

Vincent Acary and Maurice Brémond and Olivier Huber

**Abstract** Each chapter should be preceded by an abstract (10–15 lines long) that summarizes the content. The abstract will appear *online* at `www.SpringerLink.com` and be available with unrestricted access. This allows unregistered users to read the abstract as a teaser for the complete chapter. As a general rule the abstracts will not appear in the printed version of your book unless it is the style of your particular book or that of the series to which your book belongs.

Please use the 'starred' version of the new Springer `abstract` command for typesetting the text of the online abstracts (cf. source file of this chapter template `abstract`) and include them with the source files of your manuscript. Use the plain `abstract` command if the abstract is also to appear in the printed version of the book.

## Notation

The following notation is used throughout the article: the 2-norm for a function  $g$  is denoted by  $\|g\|$  and for a vector  $x \in \mathbb{R}^n$  by  $\|x\|$ . The index  $\alpha \in \mathbb{N}$  is used to identify the variable pertaining to a single contact. A multivalued mapping  $T: \mathbb{R}^n \rightrightarrows \mathbb{R}^n$  is an operator whose images are sets. The second order cone, also known as Lorentz or ice-cream cone, is defined as  $K_\mu := \{(x, t) \in \mathbb{R} \times \mathbb{R}_+ \mid \|x\| \leq \mu t\}$ ,  $\mu \geq 0$ . By polarity, the dual convex cone to a convex cone  $K$  defined by

---

Vincent Acary  
INRIA, 655, Avenue de l'Europe, 38334 Saint Ismier Cedex, e-mail: `vincent.acary@inria.fr`

Maurice Brémond  
INRIA, 655, Avenue de l'Europe, 38334 Saint Ismier Cedex, e-mail: `maurice.bremond@inria.fr`

Olivier Huber  
e-mail:

$$K^\star = \{x \in \mathbb{R}^n \mid y^\top x \geq 0, \text{ for all } y \in K\}. \quad (1) \quad \{\text{eq:dual-cone}\}$$

The normal cone  $N_K: \mathbb{R}^n \rightrightarrows \mathbb{R}^n$  to a closed convex set  $X$  is the set

$$N_K(x) = \{d \in \mathbb{R}^n \mid d^\top (y - x) \leq 0\} \quad (2)$$

The notation  $0 \leq x \perp y \geq 0$  denotes that  $x \geq 0$ ,  $y \geq 0$  and  $x^\top y = 0$ . A complementarity problem associated with a function  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$  is to find  $x \in \mathbb{R}^n$  such that  $0 \leq F(x) \perp x \geq 0$ . The generalized complementarity problem is given by  $K^\star \ni F(x) \perp x \in K$ , where  $K$  is a closed convex cone. Finite-dimensional Variational Inequality (VI) problems subsumes complementarity problems, system of equations. Solving a  $\text{VI}(X, F)$  is to find  $x \in X$  such that

$$F(x)^\top (y - x) \geq 0 \quad \text{for all } y \in X. \quad (3)$$

It is easy to see this problem is equivalent to solving a *generalized equation*

$$0 \in F(X) + N_X(x). \quad (4)$$

The Euclidean projector on a set  $X$  is denoted by  $P_X$ .

## 1 Introduction

More than thirty years after the pioneering work of (Panagiotopoulos, 1975), (Nečas et al, 1980), (Haslinger, 1983, 1984; Haslinger and Panagiotopoulos, 1984), (Del Piero and Maceri, 1983, 1985), (Katona, 1983), (Chaudhary and Bathe, 1986), (Jean and Moreau, 1987), (Mitsopoulou and Doudoumis, 1988) on numerically solving mechanical problems with contact and friction, there are still active research activities on this subject in the computational mechanics and applied mathematics communities. This can be explain by the fact that problems from mechanical systems with unilateral contact and Coulomb friction are difficult to numerically solve and the mathematical results of convergence of the numerical algorithms are rare and most of these require rather strong assumptions. In this article, we want to give some insights of the advantages and weaknesses of standard solvers found in the literature by comparing them on the large sets of examples coming from the simulation of a wide range of mechanical systems.

### 1.1 Problem statement

In this section, we formulate an abstract, algebraic finite-dimensional frictional contact problem. We cast this problem as a complementarity problem over cones, and

discuss the properties of the latter. We end by presenting some instances with contact and friction phenomenon that fits our problem description.

### Abstract problem

We want to discuss possible numerical solution procedures for the following three-dimensional finite-dimensional frictional contact problem and some of its variants. Let  $n_c \in \mathbb{N}$  be the number of contact points and  $n \in \mathbb{N}$  the number of degrees of freedom of a discrete mechanical system.

The problem data are: a symmetric positive (semi-) definite matrix  $M \in \mathbb{R}^{n \times n}$ , a vector  $f \in \mathbb{R}^n$ , a matrix  $H \in \mathbb{R}^{n \times m}$  with  $m = 3n_c$ , a vector  $w \in \mathbb{R}^m$  and a vector of coefficients of friction  $\mu \in \mathbb{R}^{n_c}$ . The unknowns are two vectors  $v \in \mathbb{R}^n$ , a velocity-like vector and  $r \in \mathbb{R}^m$ , a contact reaction or impulse, solution to

$$\begin{cases} Mv = Hr + f \\ K^* \ni \hat{u} \perp r \in K \end{cases} \quad \text{with} \quad \begin{cases} u := H^\top v + w \\ \hat{u} := u + g(u), \end{cases} \quad (5) \quad \{\text{eq:soccp1-intro}\}$$

where the set  $K$  is the cartesian product of Coulomb's friction cone at each contact, that is

$$K = \prod_{\alpha=1 \dots n_c} K^\alpha = \prod_{\alpha=1 \dots n_c} \{r^\alpha, \|r_T^\alpha\| \leq \mu^\alpha |r_N^\alpha|\} \quad (6) \quad \{\text{eq:CC}\}$$

and  $K^*$  is dual. The function  $g: \mathbb{R}^m \rightarrow \mathbb{R}^m$  is a nonsmooth function defined as

$$g(u) = [[\mu^\alpha \|u_T^\alpha\|, 0, 0]^\top, \alpha = 1 \dots n_c]^\top. \quad (7) \quad \{\text{eq:gg}\}$$

Note that the variable  $u$  and  $\hat{u}$  do not appear as unknowns since they can be directly obtained from  $v$ .

---

#### REDACTION NOTE O.H. 1.1

*I think I got the gist of the following paragraph, but I think we should work on it a bit. For me we discuss approximations in the literature that don't solve the true problem. I believe figure to illustrate the effect of neglecting  $g$  would be useful (for the optimizer).*

*I also think that we should be more explicit that we focus on the reduced form, and also explain that in the rigid case, it may not destroy the sparse structure of the problem.*

---

### A Second Order Cone Complementarity Problem (SOCCP).

From the mathematical programming point of view, the problem appears to be a Second Order Cone Complementarity Problem (SOCCP) (Facchinei and Pang, 2003) which can be generically defined as

$$\begin{cases} y = f(x) \\ K^* \ni y \perp x \in K, \end{cases} \quad (8)$$

where  $K$  is a second order cone. If the nonlinear part of the problem (5) is neglected ( $g(u) = 0$ ), the problem is an associated friction problem with dilatation, and by the way, is a gentle Second Order Cone Linear Complementarity Problem (SOCLCP) with a positive definite matrix  $H^\top M^{-1}H$  (possibly semi-definite). The assumption of an associated frictional law, i.e., a friction law where the local sliding velocity is normal the friction cone differs dramatically from the standard Coulomb friction since it generates a non-vanishing normal velocity when we slide. In other terms, the sliding motion implies the separation of the bodies. When the non-associated character of the friction is taken into account through  $g(u)$ , the problem is non monotone and nonsmooth, and therefore is very hard to solve efficiently. For a given numerical algorithm, it is not so difficult to design mechanical example to run the algorithm into troubles. Proof of convergence of the numerical algorithms are rare and most of these required strong assumptions on either the friction coefficients or full rank assumptions of matrices or operators or the dimension of the space in which the problem is formulated. Among these results, we can cite the Czech school where the coefficient of friction is assumed to be bounded and small. This assumption allows us to use fixed point methods on the convex sub-problems of Tresca friction (friction threshold that does depend on the normal reaction and then transform the cone into a semi-cylinder). We can also mention the results from (Pang and Trinkle, 1996; Stewart and Trinkle, 1996; Anitescu and Potra, 1997) where the friction cone is polyhedral (in 2D or by a faceting process). In that case, if  $w = 0$  or  $w \in \text{im}(H^\top)$ , Lemke's algorithm is able to solve the problem. The question of existence of solutions has been treated in (Klarbring and Pang, 1998; Acary et al, 2011) under similar assumptions but with different techniques. The question of uniqueness remains a difficult problem in the general case.

#### Range of applicability.

We clearly choose to simplify a lot the general problems of formulating the contact problems with friction by avoiding including too much side effects that are themselves interesting but render the study too difficult to carry out in a single article. We choose finite dimensional systems where the time dependency does not appear explicitly.

Nevertheless, we believe that there are a strong interest to study this problem since it appears to be relatively generic in numerous simulations of systems with contact and friction. This problem is indeed at the heart of the simulation of mechanical systems with 3D Coulomb's friction and unilateral constraints in the following cases:

- it might be the result of the time-discretization by event-capturing time-stepping methods or event-detecting (event-driven) techniques of dynamical systems with

friction; the variables are homogeneous to pairs velocity/impulses or acceleration forces

- it might be the result of space-discretization (by FEM for instance) of the elastic quasi-static problems of frictional contact mechanics; in that case, the variables are homogenous to displacements/forces of displacement rate/forces.
- if the system is a dynamical mechanical system composed of solids, the problem is again obtained by a space and time discretization
- if the material follows a nonlinear mechanical bulk behavior, we can use this model after a standard Newton linearization procedure.

For a description of the derivation of such problems in various practical situations we refer to (Laursen, 2003; Wriggers, 2006; Acary and Brogliato, 2008; Acary and Cadoux, 2013).

## ***1.2 Objectives and outline of the article***

In this article, after stating the problem with more details in Section 2, we recall the existence result of (Acary et al, 2011) for the problem in (5) in Section 2.3. In this framework, we briefly present in Section 3 a few alternative formulations of the problem that enable the design of numerical solution procedures : a) finite-dimensional Variational Inequalities(VI) and Quasi-Variational Inequalities(QVI) b) Nonsmooth equations and c) Optimization based formulations.

Right after these formulations, we list some of the most standard algorithms dedicated to one of the previous formulation :

1. the fixed point and projection numerical methods for solving VI are reviewed with a focus on self-adaptive time-step rules (Section 4),
2. the nonsmooth (semi-smooth) Newton methods are described based on the various nonsmooth equations formulations (Section 5),
3. Section 6 is devoted to the presentation of splitting and proximal point techniques,
4. and finally, in Section 7, the Panagiotopoulos alternating optimization technique, the successive approximation technique and the SOCLCP approach are outlined.

Since it is difficult to be exhaustive on the approaches developed in the literature to solve frictional contact problems, we decided to leave out the scope of the article the following approaches:

- the approaches that alter the fundamental assumptions of the 3D Coulomb friction model by faceting the cone as in the pioneering work of (Klarbring, 1986) and followed by (Al-Fahed et al, 1991; Pang and Trinkle, 1996; Stewart and Trinkle, 1996; Anitescu and Potra, 1997; Haslinger et al, 2004), or by convexifying the Coulomb law (associated friction law with normal dilatancy) (Heyn et al, 2013; Tasora and Anitescu, 2013, 2011; Anitescu and Tasora, 2010; Tasora and Anitescu, 2009; Krabbenhoft et al, 2012) or finally by regularizing the friction

law (Kikuchi and Oden, 1988). In the same way, we are discussing recent developments methods for the frictionless case (Morales et al, 2008; Miyamura et al, 2010; Temizer et al, 2014).

- the approaches that are based on domain decomposition and parallel computing. We choose in this article to focus on single domain computation and to skip the discussion about distributed computing mainly for a sake of length of the article. (cite a bit a literature Krause, Koziara, Renouf, Heyn, .... )

Finally, some possible interesting approaches have not been reported. We are thinking mainly to the interior point methods approach (Christensen and Pang, 1998; Miyamura et al, 2010; Kleinert et al, 2014). some basic implementations of such methods do not give satisfactory results. One of reasons is the fact that we were not able to get robustness and efficiency on a large class of problems. As it is reported in (Kleinert et al, 2014; Krabbenhof et al, 2012), it seems that it is needed to alter the friction Coulomb's law by adding regularization or dilatency in the model. In the same spirit, we skip also the comparison for the possibly very promising methods developed in (Heyn et al, 2013; Heyn, 2013) that are based on Krylov subspace and spectral methods. It could be very interesting to bench also these methods on the actual Coulomb friction model, that is to say, in the nonmonotone case. Finally, our preliminary results on the use of direct general SOCP or SOCLCP solvers were not convincing. Indeed, the structure of contact problems (product of a large number of small second order cones) has to be taken into account to get efficiency and unfortunately, these solvers are difficult to adapt to this structure.

Other comparisons articles have already been published in the literature. One of the first comparison study has been done in (Raous et al, 1988) and in (Chabrand et al, 1998). In this work, several formulations are detailed in the bidimensional case (variational inequality, linear complementarity problem (LCP) and augmented Lagrangian formulation) and comparisons of fixed point methods with projection, splitting methods and Lemke's method for solving LCP. Other comparisons have been done on 2D systems in (Mijar and Arora, 2000a,b, 2004a,b). In (Christensen et al, 1998), a very interesting comparison in the three-dimensional case has been carried out which shows the superiority of the semi-smooth Newton methods over the interior point methods. Comparisons on simple multi-body systems composed of kinematic chains can be found in (Mylapilli and Jain, 2017).

The comparison are performed on a large set of examples using performance profiles. **Let us summarize the main conclusion from Section 8: on one hand**, the algorithms based on Newton methods for nonsmooth equations solve quickly the problem when they succeed, but suffer from robustness issues mainly if the matrix  $H$  has not full rank. On the other hand, the iterative methods dedicated to solving variational inequalities are quite robust but with an extremely slow rate of convergence. To sum up, as far as we know there is no option that combines time efficiency and robustness. The set of problems used here are from the FCLIB collection<sup>1</sup>. In

---

<sup>1</sup> <https://frictionalcontactlibrary.github.io/index.html>, which aims at providing many problems to compare algorithms on a fair basis

this work, this collection is solved with the software SICONOS and its component SICONOS/NUMERICS<sup>2</sup>(Acary et al, 2015).

## 2 Description of the 3D frictional contact problems

### 2.1 Signorini's condition and Coulomb's friction.

Let us consider the contact between two bodies  $A \subset \mathbb{R}^3$  and  $B \subset \mathbb{R}^3$  with sufficiently smooth boundaries, as depicted on Figure 1.

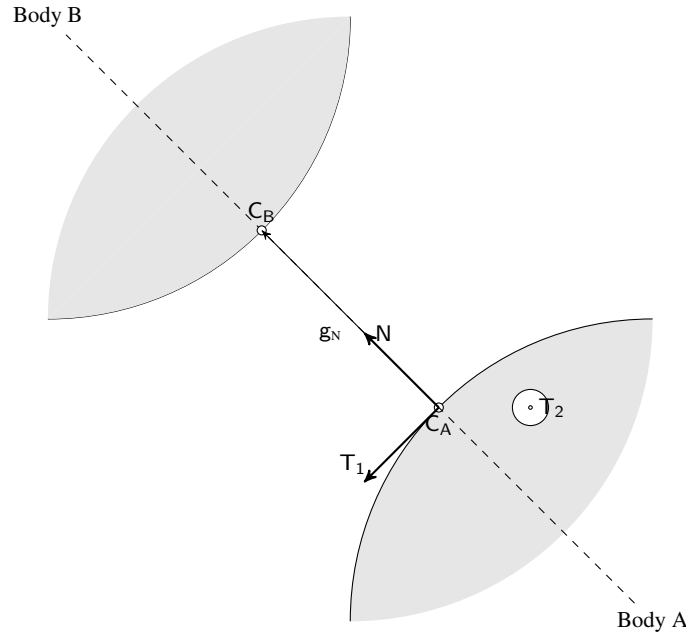


Fig. 1: Contact kinematic

From the body  $A$  “perspective”, the point  $C_A \in \partial A$  is called a *master point to contact*. The choice of this master point  $C_A$  to write the contact condition is crucial in practice and amounts to consistently discretizing the contact surface. The vector  $N$  defines an outward unit normal vector to  $A$  at the point  $C_A$ . With  $T_1, T_2$  two vectors in the plane orthogonal to  $N$ , we can build an orthonormal frame  $(C_A, N, T_1, T_2)$  called the *local frame at contact*. The slave contact point  $C_B \in \partial B$  is defined as the projection of the point  $C_A$  on  $\partial B$  in the direction given by  $N$ . Note that we assume

<sup>2</sup> <http://siconos.gforge.inria.fr>

that such a point exists. The gap function is defined as the signed distance between  $C_A$  and  $C_B$

$$\{eq:gap\} \quad g_N = (C_B - C_A)^\top N. \quad (9)$$

Consider two strictly convex bodies, which are non penetrating, *i.e.*  $A \cap B = \emptyset$ , the master and slave contact points can be chosen as the proximal points of each bodies and the normal vector  $N$  can be written as

$$\{eq:normal\} \quad N = \frac{C_B - C_A}{\|C_B - C_A\|}. \quad (10)$$

The contact force exerted by  $A$  on  $B$  is denoted by  $r \in \mathbb{R}^3$  and is decomposed in the local frame as

$$\{eq:reaction\} \quad r := r_N N + r_{T_1} T_1 + r_{T_2} T_2, \quad \text{with } r_N \in \mathbb{R} \text{ and } r_T := [r_{T_1}, r_{T_2}]^\top \in \mathbb{R}^2. \quad (11)$$

The *Signorini condition* states that

$$\{eq:signo\} \quad 0 \leq g_N \perp r_N \geq 0, \quad (12)$$

and models the unilateral contact. The condition (12), written at the *position level*, can also be defined at the *velocity level*. To this end, the relative velocity  $u \in \mathbb{R}^3$  of the point  $C_B$  with respect to  $C_A$  is also decomposed in the local frame as

$$u := u_N N + u_{T_1} T_1 + u_{T_2} T_2 \quad \text{with } u_N \in \mathbb{R} \text{ and } u_T = [u_{T_1}, u_{T_2}]^\top \in \mathbb{R}^2. \quad (13)$$

At the velocity level, the Signorini condition is written

$$\{eq:signo-velocity\} \quad \begin{cases} 0 \leq u_N \perp r_N \geq 0 & \text{if } g_N \leq 0 \\ r_N = 0 & \text{otherwise.} \end{cases} \quad (14)$$

The Moreau's viability Lemma (Moreau, 1988) ensures that (14) implies (12) if  $g_N \geq 0$  holds in the initial configuration.

Coulomb's friction models the frictional behavior of the contact force law in the tangent plane spanned by  $(T_1, T_2)$ . Let us define the Coulomb friction cone  $K$  which is the isotropic second order cone (Lorentz or ice-cream cone)

$$\{eq:CoulombCone\} \quad K = \{r \in \mathbb{R}^3 \mid \|r_T\| \leq \mu r_N\}, \quad (15)$$

where  $\mu$  is the coefficient of friction. The Coulomb friction states for the sticking case that

$$\{eq:Coulom-stick\} \quad u_T = 0, \quad r \in K, \quad (16)$$

and for the sliding case that

$$\{eq:Coulom-slide\} \quad u_T \neq 0, \quad r \in \partial K, \quad \text{and} \quad \exists \alpha > 0 \text{ such that } r_T = -\alpha u_T. \quad (17)$$

With the Coulomb friction model, there are two relations between  $u_T$  and  $r_T$ . The distinction is based on the value of the relative velocity  $u_T$  between the two bodies.



If  $u_T = 0$  (sticking case), we have  $r_T \leq \mu r_N$ . On the other hand, we get the sliding case.

### Disjunctive formulation of the Signorini-Coulomb model

If we consider the velocity-level Signorini condition (14) together with the Coulomb friction (16)–(17) which is naturally expressed in terms of velocity, we obtain a disjunctive formulation of the frictional contact behavior as

$$\begin{cases} r = 0 & \text{if } g_N > 0 \text{ (no contact)} \\ r = 0, u_N \geq 0 & \text{if } g_N \leq 0 \text{ (take-off)} \\ r \in K, u = 0 & \text{if } g_N \leq 0 \text{ (sticking)} \\ r \in \partial K, u_N = 0, \exists \alpha > 0, u_T = -\alpha r_T & \text{if } g_N \leq 0 \text{ (sliding)} \end{cases} \quad (18) \quad \{\text{eq:contact-disjunctive}\}$$

In the computational practice, the disjunctive formulation is not suitable for solving the Coulomb problem as it suggests the use of enumerative solvers, with an exponential complexity. In the sequel, alternative formulations of the Signorini-Coulomb model suitable for numerical applications are delineated. The core idea is to translate the cases in (18) into complementarity relations.

### Inclusion into normal cones

The Signorini condition (12) and (14), in their complementarity forms can be equivalently written as an inclusion into a normal cone to  $\mathbb{R}_+$

$$-g_N \in N_{\mathbb{R}_+}(r_N) \quad \text{and} \quad -u_N \in N_{\mathbb{R}_+}(r_N), \quad (19) \quad \{\text{eq:signo-inclusion-velo}\}$$

if  $g_N \leq 0$  and  $r_N = 0$  otherwise. An inclusion form of the Coulomb friction for the tangential part can be also proposed: let  $D(\cdot)$  be the Coulomb disk:

$$D(c) := \{x \in \mathbb{R}^2 \mid \|x\| \leq c\}. \quad (20) \quad \{\text{eq:diskR}\}$$

For the Coulomb friction, we get

$$-u_T \in N_{D(\mu r_N)}(r_T). \quad (21) \quad \{\text{eq:Coulomb-inclusion}\}$$

Since  $D(\mu r_N)$  is not a cone, the inclusion (21) is not a complementarity problem, but a variational inequality. The formulation (21) is often related to Moreau's maximum dissipation principle of the frictional behavior:

$$r_T \in \arg \max_{\|z\| \leq \mu r_N} z^\top u_T. \quad (22) \quad \{\text{eq:moreau-maxprinciple}\}$$

This means that the couple  $(r_T, u_T)$  maximizes the energy lost through dissipation.

In (Acary and Brogliato, 2008; Acary et al, 2011), another formulation is proposed inspired by the so-called bipotential (De Saxcé, 1992; De Saxcé and Feng, 1991; Saxcé and Feng, 1998). The goal is to form a complementarity problem out of (19) and (21). To this end, we introduce the modified relative velocity  $\hat{u} \in \mathbb{R}^3$  defined by

$$\hat{u} = u + [\mu \|u_{\text{T}}\|, 0, 0]^{\top}. \quad (23)$$

```
{eq:contact-SOCCP}
```

$$K^* \ni \hat{u} \perp r \in K \quad (24)$$

if  $g_N \leq 0$  and  $r = 0$  otherwise.

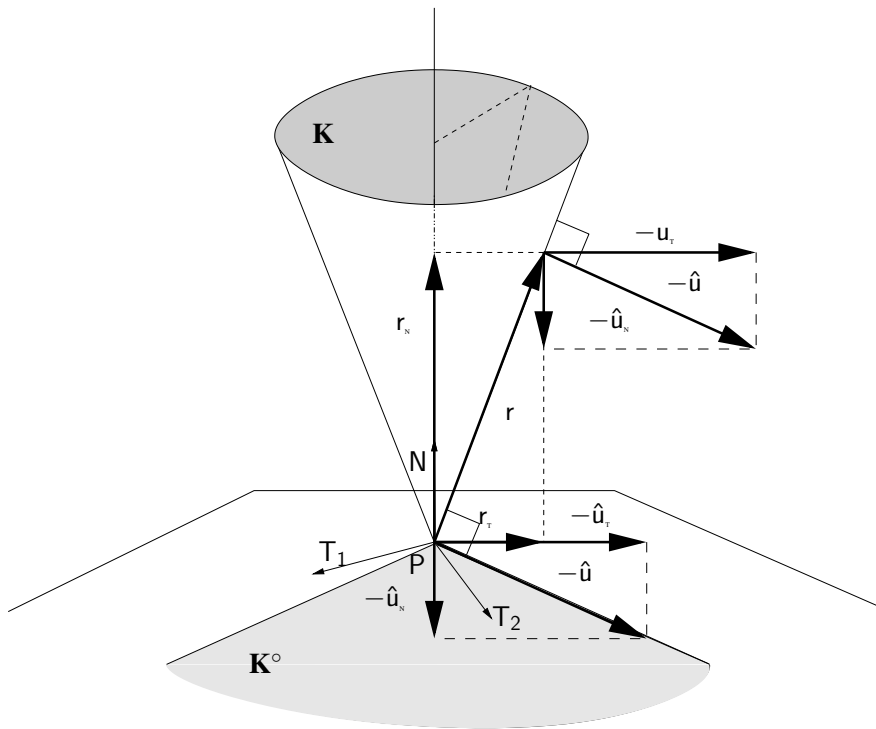


Fig. 2: Coulomb's friction law in the sliding case.

## 2.2 Frictional contact discrete problems

We assume that a finite set of  $n_c$  contact points and their associated local frames has been defined. In general, this task is not straightforward and amounts to correctly discretizing the contact surfaces. For more details, we refer to (Wriggers, 2006; Laursen, 2003). For each contact  $\alpha \in \{1, \dots, n_c\}$ , the local velocity is denoted by  $u^\alpha \in \mathbb{R}^3$ , the normal velocity by  $u_N^\alpha \in \mathbb{R}$  and the tangential velocity by  $u_T^\alpha \in \mathbb{R}^2$  with  $u^\alpha = [u_N^\alpha, (u_T^\alpha)^\top]^\top$ . The vectors  $u, u_N, u_T$  respectively collect all the local velocity  $u = [(u^\alpha)^\top, \alpha = 1 \dots n_c]^\top$ , all the normal velocity  $u_N = [u_N^\alpha, \alpha = 1 \dots n_c]^\top$ , and all the tangential velocity  $u_T = [(u_T^\alpha)^\top, \alpha = 1 \dots n_c]^\top$ . For a contact  $\alpha$ , the modified local velocity, denoted by  $\hat{u}^\alpha$ , is defined by

$$\hat{u}^\alpha = u^\alpha + g^\alpha(u) \quad \text{where} \quad g^\alpha(u) = [\mu^\alpha \|u_T^\alpha\|, 0, 0]^\top. \quad (25) \quad \{\text{eq:modified}\}$$

The vector  $\hat{u}$  and the function  $g$  collect all the modified local velocity at each contact  $\hat{u} = [\hat{u}^\alpha, \alpha = 1 \dots n_c]^\top$  and the function  $g(u) = [[\mu^\alpha \|u_T^\alpha\|, 0, 0]^\top, \alpha = 1 \dots n_c]^\top$ .

For each contact  $\alpha$ , the reaction vector  $r^\alpha \in \mathbb{R}^3$  is also decomposed in its normal part  $r_N^\alpha \in \mathbb{R}$  and the tangential part  $r_T^\alpha \in \mathbb{R}^2$  as  $r^\alpha = [r_N^\alpha, (r_T^\alpha)^\top]^\top$ . The Coulomb friction cone for a contact  $\alpha$  is defined by  $K^\alpha = \{r^\alpha \in \mathbb{R}^3 \mid \|r_T^\alpha\| \leq \mu^\alpha |r_N^\alpha|\}$  and the set  $K^{\alpha,*}$  is its dual. The set  $K$  is the cartesian product of Coulomb's friction cone at each contact, that is

$$K = \prod_{\alpha=1, \dots, n_c} K^\alpha \quad \text{and} \quad K^* \text{ is its dual.} \quad (26) \quad \{\text{eq:CC_bis}\}$$

In this article, we investigate the case where the problem is given in its *reduced* form. We consider that the discretized dynamics are of the form

$$Mv = Hr + f, \quad (27) \quad \{\text{eq:global_dyn}\}$$

with  $M$  an symmetric positive-definite matrix. The local velocities at the point of contact are given by

$$u = H^\top v + w. \quad (28) \quad \{\text{eq:local_v}\}$$

More information on the term  $w$  is given later in this section. The (global) velocities  $v$  can be substitute in (28) by using a Schur-complement technique. This yields

$$u = H^\top M^{-1} Hr + H^\top M^{-1} f + w. \quad (29)$$

Let us define  $W$ , often called the *Delassus matrix*, as

$$W := H^\top M^{-1} H \quad (30) \quad \{\text{eq:Delassus}\}$$

and the vector  $q$  as

$$q := H^\top M^{-1} f + w. \quad (31) \quad \{\text{eq:qq}\}$$

We are now ready to define the mathematical problem that we solve.

**Problem FC (Discrete frictional contact problem).** Given

- a symmetric positive semi-definite matrix  $W \in \mathbb{R}^{m \times m}$ ,
- a vector  $q \in \mathbb{R}^m$ ,
- a vector  $\mu \in \mathbb{R}^{n_c}$  of coefficients of friction,

find a vector  $r \in \mathbb{R}^m$ , such that

$$\{eq:soccp2\} \quad \begin{cases} K^* \ni \hat{u} \perp r \in K \\ u = Wr + q \\ \hat{u} = u + g(u) \end{cases} \quad (32)$$

with  $g(u) = [[\mu^\alpha \|u_\tau^\alpha\|, 0, 0]^\top, \alpha = 1 \dots n_c]^\top$ . An instance of the problem is denoted by  $FC(W, q, \mu)$   $\square$

---

#### REDACTION NOTE V.A. 2.1

*Is the assumption on the symmetry of the matrix  $M$  is mandatory ? I don't think so but it has to be checked carefully in the sequel.*

---

### 2.3 Existence of solutions

The question of the existence of solution for the Problem FC have been studied in (Klarbring and Pang, 1998) and (Acary et al, 2011) with different analysis techniques. The key assumption for existence of solutions in both articles is as follows

$$\{ass\} \quad \exists v \in \mathbb{R}^m : H^\top v + w \in \text{int} K^*, \quad (33)$$

or equivalently

$$\{asseq\} \quad w \in \text{im} H + \text{int} K^*. \quad (34)$$

Under the assumption, the Problem FC have a solution. Therefore, it makes sense to design a procedure to solve the problem. In the sequel, we will compare numerical methods only when this assumption is satisfied.

This assumption is easy verified in numerous applications. For applications in nonsmooth dynamics where the unknown  $v$  is a relative contact velocity, the term  $w$  vanishes if we have only scleronomic constraints. For  $w \in \text{im}(H^\top)$  (and especially  $w = 0$ ), the assumption is trivially satisfied. As it is explained in (Acary and Cadoux, 2013), the term  $w$  has several possible sources. If the constraints are formulated at the velocity level, an input term of  $w$  is given in dynamics by the impact law. In the case of the Newton impact law, it holds that  $w \in \text{im}(H^\top)$ . For other impact law, this is not clear. Another input in  $w$  is given by constraints that depends explicitly on

time. In that case, we can have  $w \notin \text{im}(H^\top)$  and non existence of solutions. If the constraints are written at the position level,  $w$  can be given by initial of terms that comes from velocity discretization. In that cases, the existence is also not ensured.

The assumption is also satisfied **whenever**  $\text{im} H = \mathbb{R}^m$  or in other words if  $H^\top$  has full row rank. Unfortunately, in large number of applications  $H^\top$  is rank deficient. From the mechanical point of view, the rank deficiency of  $H$  and the amount of friction seems to play a fundamental role in the question of the existence (and uniqueness) of solutions. In the numerical comparisons, we will attempt to get a deeper understanding on the role of these assumptions on the convergence of the algorithms. The rank deficiency of  $H$  is related to the number of constraints that are imposed to the system with respect to the number of degrees of freedom in the system. It is closely related to the concept of hyperstaticity in overconstrained systems. In the most favorable cases, it yields indeterminate Lagrange multipliers but also to unfeasible problems and then to the lost of solutions in the worst cases. The second assumption on the amount of friction is also well-known. The frictionless problem is easy to solve if it is feasible. It is clear that large friction coefficients prevent from sliding and therefore increase the degree of hyperstaticity of the system.

### 3 Alternative formulations

In this section, various equivalent formulations of Problem FC are given. Our goal is to show that such problems can be recast into several well-known problems in the mathematical programming and optimization community. These formulations will serve as a basis for numerical solution procedures that we develop in later sections.

#### 3.1 Variational Inequalities (VI) formulations

Let us recall the definition a finite-dimensional VI( $X, F$ ): find  $z \in X$  such that

$$F^\top(z)(y - z) \geq 0 \quad \text{for all } y \in X, \quad (35) \quad \{\text{eq:vi}\}$$

with  $X$  a nonempty subset of  $\mathbb{R}^n$  and  $F$  a mapping from  $\mathbb{R}^n$  into itself. We refer to (Harker and Pang, 1990; Facchinei and Pang, 2003) for the standard theory of finite-dimensional variational inequalities. The easiest way to state equivalent VI formulations of Problem FC is to use the following equivalences:

$$K^* \ni \hat{u} \perp r \in K \iff -\hat{u} \in N_K(r) \iff \hat{u}^\top(s - r) \geq 0, \text{ for all } s \in K. \quad (36) \quad \{\text{eq:SOCCP-1}\}$$

For Problem FC, the following equivalent formulation in VI is directly obtained from

$$-(Wr + q + g(Wr + q)) \in N_K(r). \quad (37) \quad \{\text{eq:inclusion-1}\}$$

The resulting VI is denoted by  $\text{VI}(F_{\text{vi}}, X_{\text{vi}})$  with

$$F_{\text{vi}}(r) := Wr + q + g(Wr + q) \quad \text{and} \quad X_{\text{vi}} := K. \quad (38) \quad \{\text{eq:vi-II}\}$$

Uniqueness properties.

In the general case, it is difficult to prove uniqueness of solutions to (38). If the matrix  $H$  has full rank and the friction coefficients are “small”, a classical argument for the uniqueness of solution of VIs can be satisfied. Note that the full rank hypothesis on  $H$  implies that  $W$  is positive-definite. Therefore, we have  $(x - y)^\top W(x - y) \geq C_W \|x - y\|^2$  with  $C_W > 0$ . Using this relation (38) yields

$$\begin{aligned} (F_{\text{vi}}(x) - F_{\text{vi}}(y))^\top (x - y) &= (x - y)^\top W(x - y) \\ &\quad + \sum_{\alpha=1}^{n_c} \mu^\alpha (x_N^\alpha - y_N^\alpha) [\| [Wx + q]_\tau^\alpha \| - \| [Wy + q]_\tau^\alpha \|] \\ &\geq C_W \|x - y\|^2 + \sum_{\alpha=1}^{n_c} \mu^\alpha (x_N^\alpha - y_N^\alpha) [\| [Wx + q]_\tau^\alpha \| - \| [Wy + q]_\tau^\alpha \|]. \end{aligned} \quad (39)$$

\{\text{eq:mono-II}\}

Note that for small values of the coefficients of friction the first term in the right-hand side dominates the second one. Hence, the mapping  $F_{\text{vi}}$  is strictly monotone and this ensures that the VI has at most one solution (Facchinei and Pang, 2003, Theorem 2.3.3). The fact that  $H$  is full rank also implies that the Assumption (33) for the existence of solutions is trivially satisfied. Hence, there exists a unique solution to the  $\text{VI}(F_{\text{vi}}, X_{\text{vi}})$ .

### 3.2 Quasi-Variational Inequalities (QVI)

Let us recast Problem FC into the QVI framework. A QVI is a generalization of the VI, where the feasible set is allowed to depend on the solution. Let us define this precisely: let  $X$  be a multi-valued mapping  $\mathbb{R}^n \rightrightarrows \mathbb{R}^n$  and let  $F$  be a mapping from  $\mathbb{R}^n$  into itself. The quasi-variational inequality problem, denoted by  $\text{QVI}(X, F)$ , is to find a vector  $z \in X(z)$  such that

$$F^\top(z)(y - z) \geq 0, \quad \forall y \in X(z). \quad (40)$$

\{\text{eq:qvi}\}

The QVI formulation of the frictional contact problems is obtained by considering the inclusions (19) and (21). We get

$$u^\top (s - r) \geq 0, \quad \text{for all } s \in C(\mu, r_N), \quad (41)$$

\{\text{eq:qvi-1}\}

where  $C(\mu, r_N)$  is the Cartesian product of the semi-cylinders of radius  $\mu^\alpha r_N^\alpha$  defined as

$$C(\mu, r_N) := \prod_{\alpha=1}^{n_c} \{s \in \mathbb{R}^3 \mid s_N \geq 0, \|s_\tau\| \leq \mu^\alpha r_N^\alpha\}. \quad (42)$$

\{\text{eq:cylinder}\}

Note that the QVI (41) involves only  $u$  and not  $\hat{u}$ : this is the main interest of this formulation. The price to pay is the dependence on  $r$  of the set  $C(\mu, r_N)$ . Problem FC can be expressed as a QVI by substituting the expression of  $u$ , which yields

$$(Wr + q)^\top (s - r) \geq 0, \text{ for all } s \in C(\mu, r_N). \quad (43) \quad \{\text{eq:qvi-IIbis}\}$$

This expression is compactly rewritten as  $\text{QVI}(F_{\text{qvi}}, X_{\text{qvi}})$ , with

$$F_{\text{qvi}}(r) := Wr + q \quad \text{and} \quad X_{\text{qvi}}(r) := C(\mu, r_N). \quad (44) \quad \{\text{eq:qvi-II}\}$$

Since  $W$  is assumed to be positive semi-definite matrix,  $F_{\text{qvi}}$  is monotone. Thus we get an affine monotone  $\text{QVI}(F_{\text{qvi}}, X_{\text{qvi}})$  for Problem FC.

### 3.3 Nonsmooth Equations

In this section, we expose a classical approach to solving a VI or a QVI, based on a reformulation of the inclusion as an nonsmooth equation. The term nonsmooth equation highlights that the mapping we consider fails to be differentiable. This is the price to pay for this reformulation. We can apply fixed-point and Newton-like algorithms to solve the resulting equation. Given the nonsmooth nature of the problem, applying Newton's method appears challenging, but it can still be done for some reformulations. More precisely for Problem FC, we search for an equation of the type

$$G(r) = 0 \quad (45) \quad \{\text{eq:ne-1}\}$$

where  $G$  is generally only locally Lipschitz continuous. The mapping  $G$  is such that the zeroes ( $r$ ) of (45) are the solutions of (32).

Natural and normal maps for the VI formulations

A general-purpose reformulation of VI is obtained by using the normal and natural maps, see (Facchinei and Pang, 2003) for details. The natural map  $F^{\text{nat}}: \mathbb{R}^n \rightarrow \mathbb{R}^n$  associated with the VI (35) is defined by

$$F^{\text{nat}}(z) := z - P_X(z - F(z)), \quad (46) \quad \{\text{eq:naturalmap}\}$$

where  $P_X$  is the Euclidean projector on the set  $X$ . A well-known result (see (Facchinei and Pang, 2003)) states that the solutions of a VI are related to the zeroes of the natural map :

$$z \text{ solves VI}(X, F) \iff F^{\text{nat}}(z) = 0, \quad (47) \quad \{\text{eq:VI-normalnatural}\}$$

Using (35), it is easy to see that if  $z$  solves  $\text{VI}(X, F)$ , then it is also a solution to  $\text{VI}(X, \rho F)$  for any  $\rho > 0$ . Therefore, we can define a parametric variant of the natural map by

$$F_{\rho}^{\text{nat}}(z) = z - P_X(z - \rho F(z)). \quad (48) \quad \{\text{eq:parametrized-naturalnor}\}$$

The relations given in (47) continue to hold for the parametric mapping. Using those equivalences, the frictional contact problem can be restated as zeroes of nonsmooth functions. With the natural map, Problem FC under the VI form (38) can be reformulated as

$$\{\text{eq:natural-II}\} \quad F_{\text{vi}}^{\text{nat}}(r) := [r - P_K(r - \rho(Wr + q + g(Wr + q)))] = 0. \quad (49)$$

Following the same lines, the normal map may also be used to derive algorithms. The normal map  $F^{\text{nor}} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  is defined by

$$\{\text{eq:normalmap}\} \quad F^{\text{nor}}(x) := F(P_X(x)) + x - P_X(x), \quad (50)$$

and its parametric variant

$$\{\text{eq:normalmap-b}\} \quad F_{\rho}^{\text{nor}}(x) = \rho F(P_X(x)) + x - P_X(x). \quad (51)$$

An equivalent results apply

$$\{\text{eq:VI-normalmap}\} \quad z \text{ solves VI}(X, F) \iff z = P_X(x) \text{ for some } x \text{ such that } F^{\text{nor}}(x) = 0. \quad (52)$$

The normal map based formulation of VI are also obtained in the same way.

In the seminal work of (Sibony, 1970), iterative methods for solving monotone VIs are based on the natural map and fixed point iterations. The role of  $\rho$  is recognized to be very important for the rate of convergence. To improve the methods, Sibony (1970) proposes to use “skewed” projector based on a non-Euclidean metric. Given a positive definite matrix  $R \in \mathbb{R}^{n \times n}$ , a skewed projector  $P_{X,R}$  onto  $X$  is defined as follows:  $z = P_{X,R}(x)$  is the unique solution of the convex programm

$$\{\text{eq:opt-proj-skew}\} \quad \begin{cases} \min \frac{1}{2}(y - x)^{\top} R(y - x), \\ \text{s.t. } y \in X. \end{cases} \quad (53)$$

The skew natural map can be also defined and yield the following nonsmooth equation

$$\{\text{eq:skew-natural-NE}\} \quad F_R^{\text{nat}}(z) = z - P_{X,R}(z - R^{-1}F(z)). \quad (54)$$

The zeros of  $F_R^{\text{nat}}(z)$  are also solution of the VI( $X, F$ ). Considering the skew natural map, we obtain for Problem FC under the VI form (38),

$$\{\text{eq:natural-typeII}\} \quad F_{\text{vi},R}^{\text{nat}}(r) := [r - P_{K,R}(r - R^{-1}(Wr + q + g(Wr + q)))] = 0. \quad (55)$$

The previous case is retrieved by choosing  $R = \rho^{-1}I_{n \times n}$ .

---

### REDACTION NOTE V.A. 3.1

*It can be interesting to use something  $R$  as a preconditionner of the*



problem ?  $R = \text{diag}(W)$  or incomplete LU. Warning,  $W$  is only SPR, so we cannot have  $R = W$ .

---

### Jean–Moreau’s and Alart–Curnier’s functions

Using the alternative inclusions formulations (19)–(21) with a given set of parameters  $\rho_N, \rho_T$  such that

$$\begin{cases} -\rho_N u_N \in N_{\mathbb{R}_+^{n_c}}(r_N), & \rho_N > 0, \\ -\rho_T u_T \in N_{D(\mu, (r_N)_+)}(r_T), & \rho_T > 0, \end{cases} \quad (56) \quad \{\text{eq:inclusion-rhorho}\}$$

we can replace  $P_K$  into  $P_{\mathbb{R}_+^{n_c}}$  and  $P_{D(\mu, (r_N)_+)}$  where

$$D(\mu, (r_N)_+) = \prod_{\alpha=1 \dots n_c} D(\mu^\alpha (r_N^\alpha)_+). \quad (57) \quad \{\text{eq:diskR-prod}\}$$

defines the Cartesian product of the Coulomb disks for each contact. The notation  $x_+$  stands for  $x_+ = \max(0, x)$ . Using this procedure, Jean and Moreau (1987) ; Christensen et al (1998) propose the following nonsmooth equation formulation of the frictional contact condition

$$\begin{cases} r_N - P_{\mathbb{R}_+^{n_c}}(r_N - \rho_N u_N) = 0, \\ r_T - P_{D(\mu, (r_N)_+)}(r_T - \rho_T u_T) = 0. \end{cases} \quad (58) \quad \{\text{eq:Moreau-Jean-1}\}$$

The parameters  $\rho_N, \rho_T$  may be also chosen contact by contact. Problem FC is then reformulated as

$$F_{\text{mj}}(r) := \begin{bmatrix} r_N - P_{\mathbb{R}_+^{n_c}}(r_N - \rho_N(Wr + q)_N) \\ r_T - P_{D(\mu, (r_N)_+)}(r_T - \rho_T(Wr + q)_T) \end{bmatrix} = 0. \quad (59) \quad \{\text{eq:MJ-II}\}$$

In the seminal work of Alart & Curnier (Curnier and Alart, 1988; Alart and Curnier, 1991), the augmented Lagrangian approach is invoked (see Remark 2) to obtain a similar formulation motivated by the development of nonsmooth (or generalized) Newton methods (see Section 5.2). To be accurate, the original Alart–Curnier function is given by

$$\begin{cases} r_N - P_{\mathbb{R}_+^{n_c}}(r_N - \rho_N u_N) = 0, \\ r_T - P_{D(\mu, (r_N - \rho_N u_N)_+)}(r_T - \rho_T u_T) = 0. \end{cases} \quad (60) \quad \{\text{eq:AC-1}\}$$

The difference between (58) and (60) is in the radius of the disk:  $D(\mu, (r_N - \rho_N u_N)_+)$  rather than  $D(\mu, (r_N)_+)$ . Problem FC can be also reformulated as in (59) using (60). This yields

$$F_{\text{ac}}(r) := \begin{bmatrix} r_N - P_{\mathbb{R}_+^{n_c}}(r_N - \rho_N(Wr + q)_N) \\ r_T - P_{D(\mu, (r_N - \rho_N u_N)_+)}(r_T - \rho_T(Wr + q)_T) \end{bmatrix} = 0. \quad (61) \quad \{\text{eq:AC-II}\}$$

*Remark 1.* From the QVI formulation (41), the following nonsmooth equation can also be written

$$r = P_{C(\mu, r_N)}(r - \rho u) \quad (62) \quad \{\text{eq:qvi-proj}\}$$

which corresponds to (58).

*Remark 2.* In the literature of computational mechanics (Curnier and Alart, 1988; Simo and Laursen, 1992; Alart and Curnier, 1991), very similar expressions are obtained using the concept of augmented Lagrangian functions. This concept introduced in the general framework of Optimization by (Hestenes, 1969) and developed and popularized by (Rockafellar, 1974, 1993) is a strong theoretical tool for analyzing existence and regularity of solutions of constrained optimization problems. Its numerical interest is still a subject of intense debate in the mathematical programming community. In the nonconvex nonsmooth context of problems of frictional contact problems, its invocation is not so clear, but it has enabled the design of robust numerical techniques. Nevertheless, it is worth to note that some of these methods appear as variants of the methods developed to solve variational inequalities in other contexts. The method developed by (Simo and Laursen, 1992) is a dedicated version of fixed point with projection for VI (see Section 1) and the method of (Alart and Curnier, 1991) is a tailored version of semi-smooth Newton methods (see Section 5). Nevertheless, the concept of augmented Lagrangian have never been used in the optimization literature for this purpose.

#### Xuewen-Soh-Wanji functions

Following earlier work of (Park and Kwak, 1994) and (Leung et al, 1998), the following function is proposed in (Xuewen et al, 2000) as follows

$$\{\text{eq:XSW-II}\} \quad F_{\text{sw}}(r) := \begin{bmatrix} \min(u_N, r_n) \\ \min(\|u_t\|, \mu r_N - \|r_t\|) = 0 \\ |u_{t_1} r_{t_2} - u_{t_2} r_{t_1} + \max(0, u_{t_1} r_{t_1}) = 0 \end{bmatrix} = 0. \quad (63)$$

In (Xuewen et al, 2000), the system is solved by a generalized Newton method with a line-search procedure.

#### Hüeber–Stadler–Wolhmuth functions

In (Stadler, 2004; Hüeber et al, 2008), and subsequently in (Kozłara and Bićanić, 2008), another function is used to reformulate the problem FC:

$$\{\text{eq:HSW-II}\} \quad F_{\text{hsw}}(r) := \begin{bmatrix} r_N - P_{\mathbb{R}_+^{n_c}}(r_N - \rho_N(Wr + q)_N) \\ \max(\mu(r_N - \rho_N u_N), \|r_t - \rho_t u_t\|) r_t - \mu \max(0, r_n - \rho_N u_N)(r_t - \rho_t u_t) \end{bmatrix} = 0. \quad (64)$$

In (Hüeber et al, 2008), this function is used considering the problem written at the velocity level rather than in (Koziara and Bićanić, 2008) is stated at the velocity level.

### General SOCC-functions

More generally, a large family of reformulations of the SOCCP (24) in terms of equations can be obtained by using a so-called Second Order Cone Complementarity (SOCC) function. Let us consider the following SOCCP over a symmetric cone  $K^* = K$ . A SOCC-function  $\phi$  is defined by

$$K \ni x \perp y \in K \iff \phi(x, y) = 0. \quad (65) \quad \{\text{eq:SOCC-function}\}$$

The frictional contact problem can be written as a SOCCP over symmetric cones by applying the following transformations

$$x = T_x \hat{u} = \begin{bmatrix} \hat{u}_N \\ \mu \hat{u}_T \end{bmatrix} \text{ and } y = T_y r = \begin{bmatrix} \mu r_N \\ r_T \end{bmatrix}. \quad (66) \quad \{\text{eq:SOCC-function1}\}$$

Clearly, the nonsmooth equations of the previous sections provides several examples of SOCC-functions and the natural map offers the simplest one. In (Fukushima et al, 2001), the standard complementarity functions for Nonlinear Complementarity Problems (NCP) such as the celebrated Fischer-Burmeister function are extended to the SOCCP by means of Jordan algebra. Smoothing functions are also given with their Jacobians and they studied their properties in view of the application of Newton's method. For the second order cone, the Jordan algebra can be defined with the following non-associative Jordan product

$$x \cdot y = \begin{bmatrix} x^\top y \\ y_N x_T + x_N y_T \end{bmatrix} \quad (67) \quad \{\text{eq:Jordan-product}\}$$

and the usual componentwise addition  $x + y$ . The vector  $x^2$  denotes  $x \cdot x$  and there exists a unique vector  $x^{1/2} \in K$ , the square root of  $x \in K$ , defined as

$$(x^{1/2})^2 = x^{1/2} \cdot x^{1/2} = x. \quad (68) \quad \{\text{eq:Jordan-sqrt}\}$$

A direct calculation for the SOC in  $\mathbb{R}^3$  yields

$$x^{1/2} = \begin{bmatrix} s \\ x_T \\ \frac{x_T}{2s} \end{bmatrix}, \quad \text{where } s = \sqrt{(x_N + \sqrt{x_N^2 - \|x_T\|^2})/2}. \quad (69) \quad \{\text{eq:Jordan-sqrt2}\}$$

We adopt the convention that  $0^{1/2} = 0$ . The vector  $|x| \in K$  denotes  $(x^2)^{1/2}$ . Thanks to this algebra and its associated operator, the projection onto  $K$  can be written as

$$P_K(x) = \frac{x + |x|}{2}. \quad (70) \quad \{\text{eq:Jordan-projection}\}$$

This formula provides a new expression for the natural map and its associated non-smooth equations. This is exactly what is done in (Hayashi et al, 2005) where the natural map (46) is used together with an expression of the projection operator based on the Jordan algebra calculus. The resulting SOCCP is then solved with a semi-smooth Newton method, and a smoothing parameter can be added.

Most of the calculus in Jordan algebra are based on the spectral decomposition, a basic concept in Jordan algebra, see (Fukushima et al, 2001) for more details. For  $x = (x_N, x_T) \in \mathbb{R} \times \mathbb{R}^2$ , the spectral decomposition is defined by

$$x = \lambda_1 u_1 + \lambda_2 u_2, \quad (71)$$

where  $\lambda_1, \lambda_2 \in \mathbb{R}$  and  $u_1, u_2 \in \mathbb{R}^3$  are the spectral values and the spectral vectors of  $x$  given by

$$\lambda_i = x_N + (-1)^i \|x_T\|, \quad u_i = \begin{cases} \frac{1}{2} \begin{bmatrix} 1 \\ (-1)^i \frac{x_T}{\|x_T\|} \end{bmatrix}, & \text{if } x_T \neq 0 \\ \frac{1}{2} \begin{bmatrix} 1 \\ (-1)^i w \end{bmatrix}, & \text{if } x_T = 0 \end{cases} \quad i = 1, 2 \quad (72)$$

with  $w \in \mathbb{R}^2$  any unit vector. Note that the decomposition is unique whenever  $x_T \neq 0$ . The spectral decomposition enjoys very nice properties that simplifies the computation of basic functions such that

$$\begin{aligned} x^{1/2} &= \sqrt{\lambda_1} u_1 + \sqrt{\lambda_2} u_2, \text{ for any } x \in K, \\ P_K(x) &= \max(0, \lambda_1) u_1 + \max(0, \lambda_2) u_2. \end{aligned} \quad (73)$$

More interestingly, general SOCC-functions can also be extended and smoothed version of this function can be also developed (see (Fukushima et al, 2001) ). Let us start with the Fischer-Burmeister function

$$\phi_{\text{FB}}(x, y) = x + y - (x^2 + y^2)^{1/2}. \quad (74)$$

It can show that the zeroes of  $\phi_{\text{FB}}$  are solutions of the SOCCP (65) using the Jordan algebra associated with  $K$ . Using the spectral decomposition, the Fischer-Burmeister function can be easily computed as

$$\phi_{\text{FB}}(x, y) = x + y - (\sqrt{\bar{\lambda}_1} \bar{u}_1 + \sqrt{\bar{\lambda}_2} \bar{u}_2) \quad (75)$$

where  $\bar{\lambda}_1, \bar{\lambda}_2 \in \mathbb{R}$  and  $\bar{u}_1, \bar{u}_2 \in \mathbb{R}^3$  are the spectral values and the spectral vectors of  $x^2 + y^2$  that is

$$\bar{\lambda}_i = \|x\|^2 + \|y\|^2 + 2(-1)^i \|x_N x_T + y_N y_T\|$$

$$\bar{u}_i = \begin{cases} \frac{1}{2} \begin{bmatrix} 1 \\ (-1)^i \frac{x_N x_T + y_N y_T}{\|x_N x_T + y_N y_T\|} \end{bmatrix}, & \text{if } x_N x_T + y_N y_T \neq 0 \\ \frac{1}{2} \begin{bmatrix} 1 \\ (-1)^i w \end{bmatrix}, & \text{if } x_N x_T + y_N y_T = 0 \end{cases}, i = 1, 2. \quad (76)$$

Finally, Problem FC is then reformulated as

$$F_{\text{FB}}(u, r) := \begin{bmatrix} u - Wr - q \\ \Phi_{\text{FB}} \left( \begin{bmatrix} \mu r_N \\ r_T \end{bmatrix}, \begin{bmatrix} \frac{1}{\mu} (u_N + \mu \|u_T\|) \\ u_T \end{bmatrix} \right) \end{bmatrix} = 0. \quad (77) \quad \{\text{eq:FB-II}\}$$

where the mapping  $\Phi_{\text{FB}} : \mathbb{R}^{3n_c} \times \mathbb{R}^{3n_c} \rightarrow \mathbb{R}^{3n_c}$  is defined as

$$\Phi_{\text{FB}}(x, y) = [(\phi(x^\alpha, y^\alpha), \alpha = 1 \dots n_c)^\top]. \quad (78) \quad \{\text{eq:FB-1}\}$$

### 3.4 Optimization problems

In this section, several optimization-based formulations are proposed. The quest for an efficient optimization formulation of the frictional problem is a hard task. Since the problem is nonsmooth and nonconvex, the use of an associated optimization problem is interesting from the numerical point of view if we want to improve the robustness and the stability of the numerical methods.

A straightforward optimization problem can be written whose cost function to minimize is the scalar product  $r^\top \hat{u}$ . Indeed, this product is always positive and vanishes at the solution. Let us consider this first optimization formulation

$$\begin{cases} \min r^\top \hat{u} = r^\top u + \sum_{\alpha=1}^{n_c} \mu^\alpha r_N^\alpha \|u_T^\alpha\| \\ s.t. \hat{u} \in K^*, \\ r \in K, \end{cases} \quad (79) \quad \{\text{eq:opt-1}\}$$

which amounts to minimizing the DeSaxcé's bipotential function (De Saxcé, 1992) over  $K^* \times K$ . A first simplification can be made by noting that

$$\hat{u} \in K^* \iff u_N \geq 0, \quad (80) \quad \{\text{eq:equiv-cone}\}$$

which leads to

$$\begin{cases} \min r^\top u + \sum_{\alpha=1}^{n_c} \mu^\alpha r_N^\alpha \|u_T^\alpha\| \\ s.t. u_N \geq 0 \\ r \in K. \end{cases} \quad (81) \quad \{\text{eq:opt-2}\}$$

Starting from Problem FC, a direct substitution of  $u = Wr + q$  yields

$$\begin{cases} \min r^\top (Wr + q) + \sum_{\alpha=1}^{n_c} \mu^\alpha r_N^\alpha \|(Wr + q)_T^\alpha\| \\ \text{s.t. } (Wr + q)_N \geq 0, \\ r \in K. \end{cases} \quad (82) \quad \{\text{eq:opt-3}\}$$

which is a nonlinear optimization problem with a nonsmooth and nonconvex cost function. From the numerical point of view this problem may be very difficult and we have to ensure that the cost function have to be zero at the solution which is not guaranteed of some local minima are reached in the minimization process.

Other optimization-based formulations have been proposed in the literature. They are not direct optimization formulation but they try to identify an optimization sub-problem which is well-posed and for which efficient numerical methods are available. Three approaches can be listed in three categories: a) the *alternating optimization* problems, b) the *successive approximation* method and c) the *convex SOCP* approach.

The Panagiotopoulos alternating optimization approach

aims at solving the frictional contact problem by alternatively solving the Signorini condition for a fixed value of the tangential reaction  $r_T$ , and solving the Coulomb friction model for a fixed value of the normal reaction  $r_N$ . Let us split the matrix  $W$  and the vector  $q$  in the following way:

$$\{\text{eq:W-split}\} \quad u = Wr + q \iff \begin{bmatrix} u_N \\ u_T \end{bmatrix} = \begin{bmatrix} W_{NN} & W_{NT} \\ W_{TN} & W_{TT} \end{bmatrix} \begin{bmatrix} r_N \\ r_T \end{bmatrix} + \begin{bmatrix} q_N \\ q_T \end{bmatrix}. \quad (83)$$

Two sub-problems can therefore be identified: the first one is to find  $u_N$  and  $r_N$  such that

$$\{\text{eq:A0-1}\} \quad \begin{cases} u_N = W_{NN}r_N + \tilde{q}_N, \\ 0 \leq u_N \perp r_N \geq 0, \end{cases} \quad (84)$$

where  $\tilde{q}_N = q_N + W_{NT}r_T$ . The second problem is to find  $u_T$  and  $r_T$  such that

$$\{\text{eq:A0-2}\} \quad \begin{cases} u_T = W_{TT}r_T + \tilde{q}_T, \\ -u_T \in N_{D(\mu, \tilde{r}_N)}(r_T), \end{cases} \quad (85)$$

where  $\tilde{r}_N$  is fixed and  $\tilde{q}_T = q_T + W_{TN}r_N$ . Since  $W$  is a symmetric positive semi-definite matrix,  $W_{NN}$  and  $W_{TT}$  are also symmetric semi-definite positive matrices. Therefore, two convex optimization problems can be formulated:

$$\{\text{eq:A0-3}\} \quad \begin{cases} \min \frac{1}{2} r_N^\top W_{NN} r_N + r_N^\top \tilde{q}_N \\ \text{s.t. } r_N \geq 0, \end{cases} \quad (86)$$

and

$$\left\{ \begin{array}{l} \min \frac{1}{2} r_{\text{T}}^{\top} W_{\text{TT}} r_{\text{T}} + r_{\text{T}}^{\top} \tilde{q}_{\text{T}} \\ \text{s.t. } r_{\text{T}} \in D(\mu, \tilde{r}_{\text{N}}). \end{array} \right. \quad (87)$$

This approach has been proposed by (Panagiotopoulos, 1975) for two-dimensional applications in soil foundation computing. It has also been used in other finite element applications in (Barbosa and Feijóo, 1985; Tzaferopoulos, 1993) and studied from the mathematical point of view in (Haslinger and Panagiotopoulos, 1984; Haslinger et al, 1996).

The successive approximation

method identifies a single optimization problem by introducing a function that maps the normal reaction to itself (or the friction threshold) such that

$$h(r_{\text{N}}) = r_{\text{N}}. \quad (88) \quad \{\text{eq:Haslinger-g}\}$$

Using this artifact, we can define a new problem from Problem FC such that

$$\left\{ \begin{array}{l} \theta = h(r_{\text{N}}) \\ u = Wr + q \\ -u_{\text{N}} \in N_{\mathbb{R}_+^{n_c}}(r_{\text{N}}) \\ -u_{\text{T}} \in N_{D(\mu, \theta)}(r_{\text{T}}). \end{array} \right. \quad (89) \quad \{\text{eq:Haslinger-1}\}$$

Since  $W$  is a symmetric positive semi-definite matrix, the last three lines are equivalent to a convex optimization problem over the product of semi-cylinders  $C(\mu, \theta)$ , that is

$$\left\{ \begin{array}{l} \theta = h(r_{\text{N}}) \\ \min \frac{1}{2} r^{\top} W r + r^{\top} q \\ \text{s.t. } r \in C(\mu, \theta) \end{array} \right. \quad (90) \quad \{\text{eq:Haslinger-2}\}$$

The method of successive approximation has been extensively used for proving existence and uniqueness of solutions to the discrete frictional contact problems. We refer to (Haslinger et al, 1996) which summarizes the seminal work of the Czech school (Nečas et al, 1980; Haslinger, 1983, 1984). We will see in the sequel that this approach also provides us with very efficient numerical solvers in Section 7.2.

The convex SOCP

approach is in the same vein as the previous one, with the difference that a SOCQP sub-problem is identified. To this aim, we augment the problem by introduction an auxiliary variable  $s$ , the image of  $g(u)$  introduced in (25). We then obtain

$$\begin{cases} s = g(u) \\ \hat{u} = Wr + q + s \\ K^* \ni \hat{u} \perp r \in K. \end{cases} \quad (91) \quad \{\text{eq:ACLM-2}\}$$

Since  $W$  is a positive semi-definite matrix, a new convex optimization sub-problem can be defined

$$\{\text{eq:ACLM-3}\} \quad \begin{cases} s = g(u) \\ \begin{cases} \min \frac{1}{2} r^\top W r + r^\top (q + s) \\ \text{s.t. } r \in K. \end{cases} \end{cases} \quad (92)$$

This formulation introduced in (Cadoux, 2009) and developed in (Acary and Cadoux, 2013; Acary et al, 2011) has been used to give an existence criteria to the discrete frictional contact problems. Furthermore, this existence criteria can be numerically checked by solving a linear program of second-order cone (SOCLP).

## 4 Numerical methods for VIs

### 4.1 Fixed point and projection methods for VI

Starting from the VI formulations (35) or more precisely an associated nonsmooth equation through the natural map,

$$\{\text{eq:skew-natural-NE-bis}\} \quad F_R^{\text{nat}}(z) = z - P_{X,R}(z - R^{-1}F(z)). \quad (93)$$

The basic idea of the algorithm is to perform fixed point iterations on the mapping

$$\{\text{eq:skew-fixed-point}\} \quad z \mapsto P_{X,R}(z - R^{-1}F(z)), \quad (94)$$

yielding to Algorithm 1 with the specific choice of  $R = \rho_k^{-1}I$ . The choice of the updating rule of  $\rho_k$  is detailed in Section 4.2.

For the formulation (38), the following iterations are performed

$$\{\text{eq:FP-vi-II}\} \quad r_{k+1} \leftarrow P_{K,R}(r_k - R^{-1}(Wr_k + q + g(Wr_k + q))). \quad (95)$$

In the sequel when a parameter  $\rho$  is specified, it is assumed that  $R = \rho^{-1}I$ .

The convergence of such methods are generally shown for strongly monotone VI. In our case, this assumption is not satisfied, but we will see in the sequel that such methods can converge in practice.

*Remark 3.* Algorithm 1 with the iteration rule (95) and a fixed value of  $\rho_k$  has been originally proposed in (De Saxcé and Feng, 1991, 1998). The algorithm is called Uzawa's algorithm by reference to the algorithm due to Uzawa in computing the



**Algorithm 1** Fixed point iterations for the VI (35)

---

**Require:**  $F, X$  Data of VI (35)  
**Require:**  $z_0$  initial values  
**Require:**  $\text{tol} > 0$  a tolerance value and  $\text{iter}_{\max} > 0$  the max number of iterations  
**Require:**  $\rho_0$  initial value for  $\rho$   
**Ensure:**  $z$  solution of VI (35)

```

  k ← 0
  while error > tol and k < itermax do
    Update the value of  $\rho_k$ 
     $z_{k+1} \leftarrow P_X(z_k - \rho_k F(z_k))$ 
    Evaluate error.
    k ← k + 1
  end while
  z ← zk

```

---

optimal values of convex program by primal-dual techniques (Glowinski et al, 1976; Fortin and Glowinski, 1983). Note that the algorithm in (Simo and Laursen, 1992) is similar to the fixed point algorithm with projection though based on augmented Lagrangian concept (see Remark 2).

## Extragradient methods

The extragradient method (Korpelevich, 1976) is also a well-known method for VI which improves the previous projection method. It can be described as

$$\begin{aligned}
 \bar{z}_k &\leftarrow P_X(z_k - \rho F(z_k)) \\
 z_{k+1} &\leftarrow P_X(z_k - \rho F(\bar{z}_k))
 \end{aligned} \tag{96} \quad \{\text{eq:vi-ge5}\}$$

and formally defined in Algorithm 2. The convergence of this method is guaran-

**Algorithm 2** Extragradient method for the VI (35)

---

**Require:**  $F, X$  Data of VI (35)  
**Require:**  $z_0$  initial values  
**Require:**  $\text{tol} > 0$  a tolerance value and  $\text{iter}_{\max} > 0$  the max number of iterations  
**Ensure:**  $z$  solution of VI (35)

```

  k ← 0
  while error > tol and k < itermax do
    Update the value of  $\rho_k$ 
     $\bar{z}_k \leftarrow P_X(z_k - \rho_k F(z_k))$ 
     $z_{k+1} \leftarrow P_X(z_k - \rho_k F(\bar{z}_k))$ 
    Evaluate error.
    k ← k + 1
  end while
  z ← zk

```

---

teed under the following assumptions: there exists a solution and the function  $F$  is Lipschitz-continuous and pseudo-monotone.

## 4.2 Self-adaptive step-size rules

A key ingredient in this efficiency and the convergence of the numerical methods for VI presented above is the choice of the sequence  $\{\rho_k\}$ . A sensible work has been done in the literature mainly motivated by some convergence proofs under specific assumption. Besides the relaxation of the assumption for the convergence, we are interesting in improving the numerical efficiency and robustness. We present in this section, the most popular approach for choosing the sequence  $\{\rho_k\}$ .

In (Khobotov, 1987), a method is proposed to improve the extragradient method of Korpelevich (1976) by adapting  $\rho_k$  in the following way. The goal is the find  $\rho_k$  that satisfies

$$\{eq:khobotov1\} \quad 0 < \rho_k \leq \min \left\{ \bar{\rho}, L \frac{\|z_k - \bar{z}_k\|}{\|F(z_k) - F(\bar{z}_k)\|} \right\} \text{ with } L \in (0, 1) \quad (97)$$

where  $\bar{\rho}$  is the maximum value of  $\rho_k$  which is chosen in the light of the specific problem. The objective is to find a coefficient that is bounded by the local Lipschitz constant. The standard way to do that is to use an Armijo-type procedure by successively trying value of  $\rho_k = \bar{\rho} v^m$  with  $m \in \mathbb{N}$  and  $v \in (0, 1)$ , with a typical value of  $2/3$ . In the original article of (Khobotov, 1987), there is no procedure to size  $\bar{\rho}$  or to update it. In (He and Liao, 2002) and in the context of prediction-correction, the authors propose to use the rule  $\rho_k = \rho_{k-1} v^m$  and if the criteria (97) is largely satisfied for  $\rho_k$ , the value is increased. In (Han and Lo, 2002), a similar procedure is used for the extragradient method by adding an increasing step of  $\rho_k$ , which is done after the correction as in (He and Liao, 2002). The criteria (97) is verified by computing the ratio

$$\{eq:Khobotov-r\} \quad r_k \leftarrow \frac{\rho_k \|F(z_k) - F(\bar{z}_k)\|}{\|z_k - \bar{z}_k\|}. \quad (98)$$

In (Solodov and Tseng, 1996), similar Armijo-like technique is used, and the ratio  $r_k$  is computed as follows:

$$\{eq:SolodovTseng-r\} \quad r_k \leftarrow \frac{\rho_k (z_k - \bar{z}_k)^\top (F(z_k) - F(\bar{z}_k))}{\|z_k - \bar{z}_k\|^2}. \quad (99)$$

The approach is summarized in Algorithm 3. The parameter  $L$  typically chosen around 0.9 is a safety coefficient in the evaluation of  $\rho_k$ . The parameter  $L_{\min}$  that triggers an increase of  $\rho_k$  is chosen around 0.3. In Algorithm 3, a Boolean option is added to the standard Armijo approach. The approximation  $\bar{z}_k$  is updated within the self-adaptive loop. This trick is not justified by any theoretical argument and in most of the articles this operation is not performed but in practice (see Section

9.1) it appears to improve the convergence speed. The update of the Armijo rule  $\rho_k \leftarrow \nu \rho_k$  can also be replaced by  $\rho_k \leftarrow \nu \rho_k \min \{1, 1/r_k\}$  but it appears that this trick does improve the self-adaptive procedure. Other more evolved step-lengths strategies can be found in (Wang et al, 2010) that have been tried in this study.

---

#### REDACTION NOTE O.H. 4.1

*The last paragraph is hard to follow. Maybe we could put some informations in the algorithm description, like the typical values.*

---



---

#### Algorithm 3 Updating rule for $\rho_k$

---

**Require:**  $F, X$

**Require:** Search and safety parameters.  $L \in (0, 1), 0 < L_{\min} < L, \nu \in (0, 1)$

**Require:** Initial values  $z_k \in X, \rho_{k-1} > 0$

$\rho_k \leftarrow \rho_{k-1}$

$\bar{z}_k \leftarrow P_X(z_k - \rho_k F(z_k))$

Evaluate  $r_k$  with (98) (or (99))

**while**  $r_k > L$  **do**

$\rho_k \leftarrow \nu \rho_k$

$\bar{z}_k \leftarrow P_X(z_k - \rho_k F(z_k))$

    Evaluate  $r_k$  with (98) (or (99))

**end while**

Perform the correction step of extragradient or prediction–correction method.

**if**  $r_k < L_{\min}$  **then**

$\rho_k = \frac{1}{\nu} \rho_k$

**end if**

---

### 4.3 Nomenclature

A nomenclature for the algorithms based on the VI formulation is given in Table 1.

Name	Algo.	Additional informations
FP-DS	1	iteration rule (95) and fixed $\rho$
FP-VI-UPK	1 and 3	iteration rule (95) and updating rule (98)
FP-VI-UPTS	1 and 3	iteration rule (95) and updating rule (99)
EG-VI-UPK	2 and 3	iteration rule (96) and updating rule (98)
EG-VI-UPTS	2 and 3	iteration rule (96) and updating rule (99)

Table 1: Naming convention for the algorithms based on VI formulations.

## 5 Newton based methods

### 5.1 Principle of the nonsmooth Newton methods

In Section 3.3, several formulations of the frictional contact problem by means of nonsmooth equations have been presented. These nonsmooth equations call for the use of nonsmooth Newton's methods. Remember that the standard Newton method is to solve

$$\{eq: NSN1\} \quad G(z) = 0 \quad (100)$$

by performing the following Newton iteration

$$\{eq: NSN2\} \quad z_{k+1} = z_k - J^{-1}(z_k)G(z_k). \quad (101)$$

If the mapping  $G$  is smooth enough, the matrix  $J$  is the Jacobian matrix of  $G$  with respect to  $z$ , that is  $J(z) = \nabla_z^\top G(z)$ . When the  $G$  is nonsmooth but locally Lipschitz continuous, the Jacobian matrix is replaced by an element of the generalized Jacobian at  $z$  denoted by  $\Phi(z) \in \partial G(z)$ . Let us recall the definition of the generalized Jacobian. By Rademacher's Theorem, if  $G$  is locally Lipschitz continuous, then  $G$  is almost everywhere differentiable and let us define the set  $D_G$  by

$$\{eq: NSN4\} \quad D_G := \{z \mid G \text{ is differentiable at } z\}. \quad (102)$$

The generalized Jacobian of  $G$  at  $z$  can be defined by

$$\{eq: NSN5\} \quad \partial G(z) = \text{conv} \partial_B G(z), \quad (103)$$

with

$$\{eq: NSN6\} \quad \partial_B G(z) = \left\{ \lim_{\bar{z} \rightarrow z, \bar{z} \in D_G} \nabla G(\bar{z}) \right\}. \quad (104)$$

If  $\Phi(z)$  is nonsingular, then an iteration of the nonsmooth Newton method is given by

$$\{eq: NSN3\} \quad z_{k+1} = z_k - \Phi^{-1}(z_k)(G(z_k)). \quad (105)$$

The resulting nonsmooth Newton method is detailed in Algorithm 4.

The convergence of nonsmooth Newton methods is based on the assumption of semi-smoothness of the nonsmooth function in (100). For this reason they are often called semi-smooth Newton methods (see (Facchinei and Pang, 2003, Section 7.5) and references therein).

**Algorithm 4** Nonsmooth Newton method for (100)

---

**Require:**  $G$  data of Problem (100)  
**Require:**  $z_0$  initial values  
**Require:**  $\text{tol} > 0$  a tolerance value and  $\text{iter}_{\max} > 0$  the max number of iterations  
**Ensure:**  $z$  solution of Problem (100)  
 $k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\max}$  **do**  
    compute (select)  $\Phi(z_k) \in \partial G(z_k)$   
     $z_{k+1} \leftarrow z_k - \Phi^{-1}(z_k)(G(z_k))$   
    Evaluate error.  
     $k \leftarrow k + 1$   
**end while**  
 $z \leftarrow z_k$

---

**5.2 Application to the discrete frictional contact problem**

Newton method based on the Jean–Moreau and Alart–Curnier functions

Let us consider now the Alart–Curnier function  $F_{\text{ac}}(u, r)$  in (61) or the Jean–Moreau function  $F_{\text{mj}}(u, r)$  (59) for Problem FC. Algorithm 4 is applied with

$$\{\text{eq:phiphi-ac}\} \quad \Phi(r) \in \partial F_{\text{ac}}(r) \quad \text{or} \quad \Phi(r) \in \partial F_{\text{mj}}(r). \quad (106)$$

The details of a possible computation of  $\Phi$  can be found in Appendix 13.2.

Newton method based on SOCC-function

Let us consider now the Fischer–Burmeister function  $F_{\text{FB}}(u, r)$  in (77) for Problem FC. Algorithm 4 is applied **with**

$$\Phi(r) \in \partial F_{\text{FB}}(r). \quad (107) \quad \{\text{eq:phiphi-soccp}\}$$

Nonsmooth newton based on the natural map

Let us consider the natural map  $F_{\text{vi}}^{\text{nat}}$  in (49) that enables to write Problem FC **as** a nonsmooth equation. Algorithm 4 is applied **with**

$$\Phi(r) \in \partial F_{\text{vi}}^{\text{nat}}(r). \quad (108) \quad \{\text{eq:phiphi}\}$$

The details of a possible computation of  $\Phi$  can be found in Appendix 13.1. Similar computations can also be found in (Joli and Feng, 2008) where a Newton method based on the formulation (49) is used contact by contact in a Gauss–Seidel loop.

### Lipschitz continuity properties

For the mapping  $F_{vi}^{\text{nat}}, F_{ac}, F_{FB}, F_{mj}, F_{xsw}$  that are mainly generated by a composition of the Lipschitz functions  $P_X, \min, \max$  and  $\|\cdot\|$ ; the local Lipschitz properties can be proven without difficulties. For the mapping  $F_{FB}$ , the proof of Lipschitz continuity of  $\phi_{FB}$  can be found in (Sun and Sun, 2005) and references therein. This ensures the consistency of the definition of the generalized Jacobians.

### 5.3 Convergence and robustness issues.

The local convergence of the nonsmooth Newton methods is based on the semi-smoothness of the mapping  $G$  and the fact that all elements of the generalized Jacobian at the solution point  $z^*$ ,  $\Phi(z^*) \in \partial G(z^*)$  are non singular (see (Qi and Sun, 1993) and Chapter 1 of (Qi et al, 2018) for a survey of mathematical results). For our application, the semi-smoothness of the mapping  $F_{ac}, F_{mj}$ , or  $F_{hsw}$  is proven in several papers (Christensen and Pang, 1998; Hüeber et al, 2008). The strong semi-smoothness of  $\phi_{FB}$  can be found in (Sun and Sun, 2005).

On the contrary, the regularity of all elements of the generalized Jacobians is not necessarily ensured. The first reason is the possible rank deficiency of the matrix  $W$ , which is usual in rigid body applications as discussed in Section 2.3. Even if we consider a full rank matrix  $W$ , as in the standard one contact case for instance, the invertibility of all the elements of the generalized Jacobian at the solution point is not straightforward. For the mapping  $F_{ac}, F_{mj}$ , some results are given in (Alart, 1993, 1995; Jourdan et al, 1998). Some of the results depends on the value of the coefficient of friction and the exact penalty parameters  $\rho, \rho_N, \rho_T$  parameters. For the mapping  $F_{hsw}$ , some other results can be found in (Hüeber et al, 2008).

In the numerical practice and even if  $W$  is full-rank, it may happen that the elements of the generalized Jacobians are not regular or very badly conditioned even when we are far from the solution. This fact is reported in (Alart, 1993, 1995; Jourdan et al, 1998; Hüeber et al, 2008; Koziara and Bićanić, 2008). Some divergence of the Newton algorithm can be encountered. A few work has been done to understand this problem. Among them, we cite (Hüeber et al, 2008) where some modifications of the elements of the generalized Jacobian are performed far from the solution to keep the Newton iteration matrix regular and well conditioned when the function  $F_{hsw}$  is chosen. This very interesting work opens new directions of research for the other mappings. In (Koziara and Bićanić, 2008), some other heuristics are developed to try avoid divergence of the Newton loop. In the two next sections, we present two complementary ways to partly solve this problem by choosing consistently the parameters  $\rho, \rho_N, \rho_T$  and by applying some line-searches techniques to globalize the convergence.

### 5.4 Estimation of $\rho, \rho_N, \rho_T$ parameters

One of the key parameters in the efficiency of the nonsmooth Newton methods are the choice of the parameter  $\rho$  in the parameterized natural map in (48) and the parameters  $\rho_N$  and  $\rho_T$  in the Jean–Moreau and Alart–Curnier functions (59) and (61). The default choice is to set these parameters equal to 1 but the numerical practice shows that the convergence of the nonsmooth solvers is drastically deteriorated, especially if the norm or the conditioning of the matrix  $W$  is far from this unit value. There is no theoretical rules to size this parameters, but some heuristics may be found in the literature for a single contact problem that we expose in the sequel.

Inverse of a norm of  $W$

A first simple choice is to consider the inverse of a norm of the matrix  $W$ . With this heuristics, we set the  $\rho$  parameter before the Newton loop as follows:

$$\rho = \frac{1}{\|W\|}, \quad \rho_N = \rho_T = \frac{1}{\|W\|}. \quad (109) \quad \{\text{eq:rho-1}\}$$

This choice is mainly based on a guess of the inverse of the local Lipschitz constant of the operator  $Wr + q$ . In the case of the natural map, it amounts to neglecting the nonlinear contribution of  $g$ . Naturally, this choice depends on the choice of the matrix norm. If we assume that the matrix is a symmetric definite positive matrix, a possibility is to choose a 2-norm based on the spectral radius  $\|W\|_2 = \rho(W) = \lambda_{\max}(W)$  of the matrix:

$$\rho = \frac{1}{\lambda_{\max}(W)}, \quad \rho_N = \rho_T = \frac{1}{\lambda_{\max}(W)}. \quad (110) \quad \{\text{eq:rho-2}\}$$

Estimation based on the splitting  $W_{NN}$  and  $W_{TT}$

A second possible choice for the map (59) and (61) is to use the fact that the problem is split with respect to the normal and the tangent directions. In that case, we compute a value of  $\rho_N$  that is based on the eigenvalues of  $W_{NN}$  and a value of  $\rho_T$  based on the eigenvalue of  $W_{TT}$ . For a single contact, we set

$$\rho_N = \frac{1}{W_{NN}}, \quad \rho_T = \frac{1}{\lambda_{\max}(W_{TT})} \quad (111) \quad \{\text{eq:rho-3}\}$$

A third option it also take into account the conditioning of the matrix  $W_{TT}$  by choosing

$$\rho_N = \frac{1}{W_{NN}}, \quad \rho_T = \frac{\lambda_{\min}(W_{NN})}{\lambda_{\max}^2(W_{TT})} \quad (112) \quad \{\text{eq:rho-4}\}$$

### Adaptive estimation of the parameters

In (Koziara and Bićanić, 2008), an adaptive way of updating  $\rho$  is proposed that has been implemented for our experiments.

### Default choices

By default, we use the rule (112) for the mapping (59) and (61) and the rule (110) for the natural. When other rules are chosen in the comparison, they are specified.

## 5.5 *Damped Newton and line-search procedures*

These line-searches algorithms follow the presentation done in chapter 3 of (Bonnan et al, 2003), where one may find notably all the mathematical explanations of why they terminate. However, despite the mathematical proofs, in practice, it is recommended that some emergency tests should be added during extrapolation and interpolation phases to avoid infinite loops.

The choice of the values for the parameters  $m_1, m_2$  for the Goldstein-Price line-search and the parameter  $m_1$  alone for the Armijo line-search is also discussed and it is advised to choose  $m_1 < \frac{1}{2}$  and  $m_2 > \frac{1}{2}$ .

### Merit functions

#### Goldstein–Price (GP) line search

#### Armijo line–search

#### Non-monotone line–search

see (Koziara and Bićanić, 2008) based on (Ferris and Lucidi, 1994; Grippo et al, 1986)



**Algorithm 5** Goldstein–Price (GP) line search

---

**Require:**  $x$ , the starting point of the line-search.  
**Require:**  $d$ , the direction of search.  
**Require:**  $t$ , an initial stepsize-value.  
**Require:**  $t \rightarrow q(t)$ , for  $t \geq 0$ , with  $q \in C^1$  bounded from below and  $q'(0) < 0$ , a merit function representing  $f(x+td)$   
**Require:**  $m_1, m_2$ , parameters with  $0 < m_1 < m_2 < 1$   
**Require:**  $a$ , with  $a > 1$ , parameter for extrapolation  
**Ensure:** a finite line-search

```

 $t_L \leftarrow 0$ 
 $t_R \leftarrow 0$ 
 $\Delta \leftarrow \frac{q(t)-q(0)}{t}$ 
while  $m_2 q'(0) > \Delta$  or  $\Delta > m_1 q'(0)$  do
  if  $m_1 q'(0) < \Delta$  then
     $t_R \leftarrow t$ 
  end if
  if  $\Delta < m_2 q'(0)$  then
     $t_L \leftarrow t$ 
  end if
  if  $t_R = 0$  then
     $t \leftarrow at$ 
  else
     $t \leftarrow \frac{t_L + t_R}{2}$ 
  end if
   $\Delta \leftarrow \frac{q(t)-q(0)}{t}$ 
end while

```

---

**Algorithm 6** Armijo(A) line search

---

**Require:**  $x$ , the starting point of the line-search.  
**Require:**  $d$ , the direction of search.  
**Require:**  $t$ , an initial stepsize-value.  
**Require:**  $t \rightarrow q(t)$ , for  $t \geq 0$ , with  $q \in C^1$  bounded from below and  $q'(0) < 0$ , a merit function representing  $f(x+td)$   
**Require:**  $m_1$ , a parameter with  $0 < m_1 < 1$   
**Require:**  $a$ , with  $a > 1$ , parameter for extrapolation  
**Ensure:** a finite line-search

```

while  $m_1 q'(0) < \frac{q(t)-q(0)}{t}$  do
  if  $t_R = 0$  then
     $t \leftarrow at$ 
  else
     $t_R \leftarrow t$ 
     $t \leftarrow \frac{t_R}{2}$ 
  end if
end while

```

---

## 5.6 Nomenclature

A nomenclature for the algorithms based on the nonsmooth Newton methods is given listed in Table 2.

Name	Algo.	Additional informations
NSN-NM	4	Natural map formulation (49)
NSN-AC	4	Alart–Curnier formulation (61)
NSN-JM	4	Jean–Moreau formulation (59)
NSN-FB	4	Fischer–Burmeister formulation (77)
NSN-NM-GP	4 and 5	Natural map formulation (49) and the Goldstein–Price (GP) line search
NSN-AC-GP	4 and 5	Alart–Curnier formulation (61) and the Goldstein–Price (GP) line search
NSN-JM-GP	4 and 5	Jean–Moreau formulation (59) and the Goldstein–Price (GP) line search
NSN-FB-GP	4 and 5	Fischer–Burmeister formulation (77) and the Goldstein–Price (GP) line search
NSN-NM-A	4 and 6	Natural map formulation (49) and the Armijo(A) line search
NSN-AC-A	4 and 6	Alart–Curnier formulation (61) and the Armijo(A) line search
NSN-JM-A	4 and 6	Jean–Moreau formulation (59) and the Armijo(A) line search
NSN-FB-A	4 and 6	Fischer–Burmeister formulation (77) and the Armijo(A) line search
NSN-AC-HYBRID	4	Alart–Curnier formulation (61) with a pre computation of the initial guess with 100 iterations of EG-VI-UPK algorithm the

Table 2: Naming convention for the algorithms based on nonsmooth Newton (NSN) method

## 6 Splitting techniques and proximal point algorithm

Splitting techniques are standard techniques to solve  $\text{VI}(F, X)$  when the function  $F$  is affine that is  $F(z) = Mz + q$  and the set  $X$  can be decomposed in a Cartesian product of independent smaller sets  $X = \prod_i X_i$ . Usually, a block splitting of the matrix  $M$  is performed and a Projected Successive Over Relaxation (PSOR) method is used to solve the VI. Since the cone  $K$  is a product of second-order cones in  $\mathbb{R}^3$ , a natural way to split the problem is to form sub-problems by using single contact as a building block. The sub-problems can be solved by any method for the VI that have been presented in the previous sections. In the same way, the proximal point algorithm can also be used which amounts to solving the original  $\text{VI}(F, X)$  by solving a sequence of  $\text{VI}(F_{c, x_k}, X)$  problems such that  $F_{c, x_k}(z) = z - x_k + cF(z)$ ,  $c > 0$  and  $\lim_{k \rightarrow +\infty} \|x_k - z\| = 0$ .

### 6.1 Splitting techniques

The particular structure of the cone  $K$  as a product of second-order cone in  $\mathbb{R}^3$  calls for a splitting of the problem contact by contact. For Problem FC, the relation

$$\{eq:delassus-bis\} \quad u = Wr + q \quad (113)$$

is splitted along each contact as follows

$$u^\alpha = W^{\alpha\alpha} r^\alpha + \sum_{\beta \neq \alpha} W^{\alpha\beta} r^\beta + q^\alpha, \text{ for all } \alpha \in 1 \dots n_c, \quad (114) \quad \{eq:delassus-ter\}$$

where the matrices  $\alpha$  and  $\beta$  are used to label the variable for each contact. The matrices  $W^{\alpha\beta}$  with  $\alpha \in 1, \dots, n_c$  and  $\beta \in 1, \dots, n_c$  are easily identified from (113). From (114), a projected Gauss–Seidel (PGS) method is obtained by using the following update rule at the  $k$ -th iterate:

$$u_{k+1}^\alpha = W^{\alpha\alpha} r_{k+1}^\alpha + \sum_{\beta < \alpha} W^{\alpha\beta} r_{k+1}^\beta + \sum_{\beta > \alpha} W^{\alpha\beta} r_k^\beta + q^\alpha, \text{ for all } \alpha \in 1 \dots n_c. \quad (115) \quad \{eq:pgs-1\}$$

A Projected Successive Over Relaxation (PSOR) scheme is derived by introducing a relaxation parameter  $\omega > 0$  such that

$$u_{k+1}^\alpha = \frac{1}{\omega} W^{\alpha\alpha} r_{k+1}^\alpha - \frac{1}{\omega} W^{\alpha\alpha} r_k^\alpha + \sum_{\beta < \alpha} W^{\alpha\beta} r_{k+1}^\beta + \sum_{\beta \geq \alpha} W^{\alpha\beta} r_k^\beta + q^\alpha, \text{ for all } \alpha \in 1 \dots n_c. \quad (116) \quad \{eq:psor-1\}$$

At the  $k$ -th iteration, the following problem is solved for each contact  $\alpha$ :

$$\begin{cases} u_{k+1}^\alpha = \bar{W}^{\alpha\alpha} r_{k+1}^\alpha + \bar{q}_{k+1}^\alpha, \\ \hat{u}_{k+1}^\alpha = u_{k+1}^\alpha + g(u_{k+1}^\alpha), \\ K^{\alpha,*} \ni \hat{u}_{k+1}^\alpha \perp r_{k+1}^\alpha \in K^\alpha, \end{cases} \quad (117) \quad \{eq:psor-3\}$$

where

$$\begin{cases} \bar{W}^{\alpha\alpha} = \frac{1}{\omega} W^{\alpha\alpha} \\ \bar{q}_{k+1}^\alpha = -\frac{1}{\omega} W^{\alpha\alpha} r_k^\alpha + \sum_{\beta < \alpha} W^{\alpha\beta} r_{k+1}^\beta + \sum_{\beta \geq \alpha} W^{\alpha\beta} r_k^\beta + q^\alpha \end{cases}, \text{ for all } \alpha \in 1 \dots n_c. \quad (118) \quad \{eq:psor-2\}$$

The problem (117) has exactly the same structure as Problem FC, but is of lower size since it is only for one contact. It is solved by a *local solver*, which can be any of the algorithms presented in this article or even an analytical method (enumerating all the possible cases as in (Bonneton and Daviet, 2011)).

The PSOR algorithm is summarized in Algorithm 7 and the NSGS can be recovered by setting  $\omega = 1$ .

Applications methods in frictional contact date back to the work of (Mitsopoulou and Doudoumis, 1988, 1987) for two-dimensional friction. In (Jourdan et al, 1998),

**Algorithm 7** PSOR algorithm for Problem FC

---

**Require:**  $W, q, \mu$   
**Require:**  $r_0$  initial values  
**Require:**  $\text{tol} > 0, \text{tol}_{\text{local}}$  tolerance values and  $\text{iter}_{\text{max}} > 0, \text{iter}_{\text{local max}} > 0$  the max number of local iterations  
**Require:**  $\omega$  a relaxation parameter  
**Ensure:**  $r, u$  solution of Problem FC  
**while** error  $> \text{tol}$  and  $k < \text{iter}_{\text{max}}$  **do**  
  **for**  $\alpha = 1 \dots n_c$  **do**  
     $\bar{W}_{k+1}^{\alpha\alpha} \leftarrow \frac{1}{\omega} W^{\alpha\alpha}$   
     $\bar{q}_{k+1}^{\alpha} \leftarrow -\frac{1}{\omega} W^{\alpha\alpha} r_k^{\alpha} + \sum_{\beta < \alpha} W^{\alpha\beta} r_{k+1}^{\beta} + \sum_{\beta \geq \alpha} W^{\alpha\beta} r_k^{\beta} + q^{\alpha}$   
    Solve the single contact problem  $\text{FC}(\bar{W}^{\alpha\alpha}, \bar{q}_{k+1}^{\alpha}, \mu)$  at accuracy  $\text{tol}_{\text{local}}$  with a maximum of iteration  $\text{iter}_{\text{local max}} > 0$   
  **end for**  
  Evaluate error.  
   $k \leftarrow k + 1$   
**end while**  
 $r \leftarrow r_k$   
 $u \leftarrow u_k$

---

this method is developed in the Gauss-Seidel configuration ( $\omega = 1$ ) with a local Newton solver based on the Alart–Curnier formulation. If the local solver is only one iteration of the VI solver based on projection, we get a standard splitting technique for VI. In Table 3, the methods based on PSOR used in the comparison are summarized.

## 6.2 Proximal points techniques

---

### REDACTION NOTE O.H. 6.1

*The following paragraph is a work in progress.*

---

Proximal points methods, or more generally augmented Lagrangian methods, find their roots in the study of variational inequalities. They rely on rather theoretical tools (monotone operators and resolvents), introduced to study VI. However, they are also useful to devise numerical methods as we shall see. The basic idea is to design an iterative algorithm to find a solution to the inclusion

$$0 \in F(x) + N_X(x) \quad \text{also} \quad 0 \in T(x), \quad (119)$$

where  $T$  is a maximal monotone operator (see Definition 1). Remember that the first inclusion is a VI.

---

**REDACTION NOTE V.A. 6.1**

*Small introduction on proximal functions of Moreau, algo of Martinet and result of Rockafellar (see (Chen and Teboulle, 1993))*

Without entering into theoretical aspects, the key idea of proximal points algorithms is to replace the original  $\text{VI}(F, X)$  by a sequence of  $\text{VI}(F_{\rho, x_k}, X)$  problems such that

$$F_{\rho, x}(z) = z - x_k + \alpha F(z), \quad \rho > 0 \quad (120) \quad \{\text{eq:prox-algo-1}\}$$

with the property that  $\lim_{k \rightarrow +\infty} \|x_k - z\| = 0$ . This is usually performed by defining a sequence  $x_k$  such that

$$x_{k+1} = (1 - \omega)x_k + \omega z_{k+1} \quad (121) \quad \{\text{eq:prox-algo-2}\}$$

where  $\omega$  is a relaxation parameter and  $z_{k+1}$  the solution of  $\text{VI}(F_{\rho, x_k}, X)$ . The algorithm is described in algorithm 8

---

**Algorithm 8** Proximal point algorithm for the VI (35)

---

**Require:**  $F, X$  Data of VI (35)  
**Require:**  $\omega$  relaxation parameter  
**Require:**  $\alpha$  proximal point parameter  
**Require:**  $x_0$  initial values  
**Require:**  $\text{tol} > 0, \text{tol}_{\text{in}}$  tolerance values and  $\text{iter}_{\text{max}} > 0$  the max number of iterations  
**Ensure:**  $z$  solution of VI (35)  
 $k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\text{max}}$  **do**  
    Solve  $\text{VI}(F_{\alpha, x_k}, X)$  for  $z_{k+1}$  at accuracy  $\text{tol}_{\text{int}}$   
     $x_{k+1} \leftarrow (1 - \omega)x_k + \omega z_{k+1}$   
    Evaluate error.  
     $k \leftarrow k + 1$   
**end while**  
 $z \leftarrow z_{k+1}$

---

For solving the sub-problem  $\text{VI}(F_{\alpha, x_k}, X)$ , any of the previous presented algorithms can be used. In this sense, the proximal point algorithm defined a general family of algorithm. The main interest of the proximal point algorithm is the regularization it introduces in the definition (120). For instance, let consider an affine VI, that is defined with  $F(z) = Mz + q$ . The function is the sub-problem is given by

$$F_{\alpha, x}(z) = (I + \alpha M)z - x_k + \rho q, \quad \rho > 0. \quad (122) \quad \{\text{eq:prox-algo-4}\}$$

We can easily that for sufficiently small  $\rho$ , we get a monotone affine VI and even better a strongly monotone VI. For Problem FC with  $F_{\text{vi}}$ , the proximal point algorithm yields

$$F_{\text{vi}, \rho, x}(r) = (I + \alpha W)r - x_k + \alpha(q + g(Wr + q)), \quad \rho > 0. \quad (123) \quad \{\text{eq:prox-algo-5}\}$$

and it should not be difficult to prove that with a small  $\rho$  parameter that we get a monotone VI. Naturally, there is a price to pay, smaller the parameter  $\rho$  is, easier the VI problem is, but the resulting solution of  $z_{k+1}$  is far from the solution.

The use of proximal point algorithm can be very interesting when the solving of the problem suffers for the lack of regularity of the operator. For instance, the Newton methods are in trouble when the Jacobian is not invertible. Thanks to the proximal point algorithm, we can retrieve invertible Jacobian.

---

#### REDACTION NOTE V.A. 6.2

- *Prove it !*
- *Is there an analogy with choosing a small  $r$  ?*
- *Techniques of (Han, 2008) for sizing alpha*

---

In (He and Liao, 2002), the extragradient method is reinterpreted as a prediction-correction method applied to the proximal point algorithm. Indeed, the proximal point algorithm amounts to solving the following VI for a given  $z_k$

$$\{\text{eq:PPA-VI1}\} \quad ((z - z_k) + \alpha_k F(z))^\top (y - z) \geq 0, \text{ for all } y \in X \text{ and } \alpha_k > 0 \quad (124)$$

or equivalently

$$\{\text{eq:PPA-VI2}\} \quad z = P_X(z - [z - z_k + \alpha_k F(z)]) = P_X(z_k - \alpha_k F(z)). \quad (125)$$

Searching for a solution  $z_{k+1}$  of (125) yields

$$\{\text{eq:PPA-VI3}\} \quad z_{k+1} = P_X(z_k - \alpha_k F(z_{k+1})). \quad (126)$$

A prediction–correction method based may be written as

$$\{\text{eq:PPA-VI4}\} \quad \begin{aligned} \bar{z}_k &\leftarrow P_X(z_k - \rho_k F(z_k)) \\ z_{k+1} &\leftarrow P_X(z_k - \alpha_k F(\bar{z}_k)) \end{aligned} \quad (127)$$

where the scalars  $\alpha_k$  and  $\rho_k$  have to be updated at each iteration.

---

#### REDACTION NOTE O.H. 6.2

*I think that the analogy with the proximal point algorithm is premature. We only introduce it in Section 6.2.*

*I think we are presenting result with the extragradient. Maybe add a teaser at the end of the paragraph?*

*VA OK we can discuss this point. It is difficult to do a (European) perfect linear presentation.*

---

### 6.3 Control of the tolerance of internal solvers $\text{tol}_{\text{int}}$ and $\text{tol}_{\text{local}}$ in the splitting and proximal approaches

In Algorithms 8 and 7, an internal tolerance is used to control the accuracy of the internal solver. It is generally not useful to solve the internal problem at the accuracy of the global one. In the comparison study, we set the internal tolerance  $\text{tol}_{\text{int}}$  to a fraction of the error  $\text{error}/10.0$  for the Algorithm 8. For Algorithm 7, the local tolerance  $\text{tol}_{\text{local}}$  is set by default to a very low value of  $10^{-14}$ . When it is specified, an adaptive local tolerance is set to  $\text{error}/10.0$ .

### 6.4 Nomenclature

A nomenclature for the algorithms based on the projection/splitting approach is given in Table 3.

Algorithm	Description
NSGS-AC	Algorithm 7 with $\omega = 1$ with the local solver NSN-AC with tolerance $\text{tol}_{\text{local}}$
NSGS-JM	Algorithm 7 with $\omega = 1$ with the local solver NSN-JM with tolerance $\text{tol}_{\text{local}}$
NSGS-AC-GP	Algorithm 7 with $\omega = 1$ with the local solver NSN-AC-GP with tolerance $\text{tol}_{\text{local}}$
NSGS-JM-GP	Algorithm 7 with $\omega = 1$ with the local solver NSN-JM=GP with tolerance $\text{tol}_{\text{local}}$
NSGS-FP-DS-One	Algorithm 7 with $\omega = 1$ with one iteration of FP-DS for the local solver
NSGS-FP-VI-UPK	Algorithm 7 with $\omega = 1$ with FP-VI-UPK for the local solver with tolerance $\text{tol}_{\text{local}}$
NSGS-EXACT	Algorithm 7 with $\omega = 1$ with the exact local solver.
PSOR-AC	Algorithm 7 with the local solver NSN-AC with tolerance $\text{tol}_{\text{local}}$
PPA-NSN-AC	Algorithm 8 with the NSN-AC solver as internal solver.
PPA-NSGS-AC	Algorithm 8 with the NSGS-AC solver as internal solver.

Table 3: Naming convention for the algorithms based on splitting and proximal algorithms

## 7 Optimization based methods

### 7.1 Alternating optimization problem

The Panagiotopoulos approach described in Section 3.4 generates a family of solvers by choosing a particular solver for the normal contact problem (86) and the tangential contact problem (87). This method may be viewed as a two-block

Gauss-Seidel method as it has been pointed out by (Tzaferopoulos, 1993). More precisely, the following choices may be made for the normal and tangent problems.

The normal contact problem

$$\{\text{eq:AO-3-bis}\} \quad \begin{cases} \min \frac{1}{2} r_N^\top W_{NN} r_N + r_N^\top \tilde{q}_N \\ \text{s.t. } r_N \geq 0 \end{cases} \quad \text{with } \tilde{q}_N = q_N + W_{NT} r_{T,k}, \quad (128)$$

is a convex quadratic program with simple bound constraints. In the literature, a bunch of solvers has been developed to solve such problems. Among others, we might cite the active set strategy solvers (Fletcher, 1987; Nocedal and Wright, 1999) that are mainly dedicated to small-scale systems and the projected gradient (Calamai and More, 1987) and projected conjugate gradient methods (Moré and Toraldo, 1989; Moré and Toraldo, 1991) that are more dedicated to large-scale systems. Note that there exists also a bunch of methods in the literature that improves the methods of (Moré and Toraldo, 1991) for large-scale systems. For the reader interested in deeper details, we refer to the book of (Dostál, 2016). (see especially (Dostál, 2016), Section 8.7 for a review of the different approaches). It is clear that we might also use semi-smooth Newton methods or interior point methods but our experience has shown that such methods are not efficient when  $\ker(W_{NN}) \neq \{0\}$ . The optimality conditions of this quadratic reduced to a linear complementarity problem with a semi-definite matrix. In that case, it is also possible to solve the problem with PSOR techniques with line-searches. In order to keep the article in a reasonable size, we decided in this work to use the projected gauss-seidel (PGS) algorithm and the projected gradient algorithm of (Calamai and More, 1987) to solve the normal problem described. The projected gradient algorithm solved the following QP for a convex set  $C$

$$\{\text{eq:ConvexQP}\} \quad \begin{cases} \min q(r) := \frac{1}{2} r^\top W r + r^\top \tilde{b} \\ \text{s.t. } r \in C \end{cases} \quad (129)$$

with the algorithm described in Algorithm 9.

The tangential problem,

$$\{\text{eq:AO-4-bis}\} \quad \begin{cases} \min \frac{1}{2} r_T^\top W_{TT} r_T + r_T^\top \tilde{q}_T \\ \text{s.t. } r_T \in D(\mu, \tilde{r}_N) \end{cases} \quad \text{with } \tilde{q}_T = q_T + W_{TN} r_{N,k+1}, \quad (130)$$

is also a convex program but with a more complex structure since the constraints are quadratic one. It exists also some dedicated methods to solve this specific problem as the method in (Dostál and Kozubek, 2012) (an extension of (Moré and Toraldo, 1991)) which is an algorithm for solving QP over convex constraints (disk constraint for instance). Earlier application of projected gradient and projected gradient techniques for the frictionless problem can also be found in (Barbosa et al, 1997) with comparison with PSOR techniques.

In the article, we will use a) a reformulation of the optimality conditions of this problem as a variational inequality and we apply the fixed point algorithm and the



**Algorithm 9** Projected gradient algorithm for QP 129

---

**Require:**  $W, b$  that defines  $q(r)$   
**Require:**  $C$  a convex set  
**Require:**  $r_0$  initial values  
**Require:**  $\text{tol} > 0$  a tolerance value and  $\text{iter}_{\max} > 0$  the max number of iterations  
**Require:**  $\rho_0 > 0, l, \sigma \in (0, 1)$   
**Require:**  $i_{\text{lsmax}}$  maximum number of line-search iterations  
**Ensure:**  $r$  solution of Problem 129  
 $r_k \leftarrow r_0 ; \theta_0 \leftarrow q(r_0) ; k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\max}$  **do**  
    *Armijo like-search procedure*  
     $i_{\text{ls}} \leftarrow 0$   
    **while**  $\text{criterion} > 0$  and  $i_{\text{ls}} < i_{\text{lsmax}}$  **do**  
         $\rho \leftarrow \rho_0 l^{i_{\text{ls}}}$   
         $r \leftarrow P_C(r_k - \rho(Mr_k + b))$   
         $\theta \leftarrow q(r)$   
         $\text{criterion} \leftarrow \theta - \theta_k - \sigma(Mr_k + b)^\top (r - r_k)$   
         $i_{\text{ls}} \leftarrow i_{\text{ls}} + 1$   
    **end while**  
     $r_k \leftarrow r ; \theta_k \leftarrow q(r);$   
    evaluate error.  
**end while**

---

extra gradient algorithm of Section 4 or b) an adaptation of one of the splitting techniques detailed in Section 6. The algorithm is described in Algorithm 10.

In Table 4, we detailed the algorithms we use in the present study.

**Algorithm 10** Panagiotopoulos decomposition algorithm for Problem FC

---

**Require:**  $W, q, \mu$   
**Require:**  $r_0$  initial values  
**Require:**  $\text{tol} > 0, \text{tol}_{\text{int}}$  tolerance values and  $\text{iter}_{\max} > 0$  the max number of iterations  
**Ensure:**  $r, u$  solution of Problem FC  
 $r_k \leftarrow r_0 ; k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\max}$  **do**  
     $\tilde{q}_N \leftarrow q_N + W_{NT} r_{T,k}$   
    solve (128) for  $r_{N,k+1}$  at accuracy  $\text{tol}_{\text{int}}$   
     $\tilde{q}_T \leftarrow q_T + W_{TN} r_{N,k+1}$   
    solve (130) for  $r_{T,k+1}$  at accuracy  $\text{tol}_{\text{int}}$   
     $k \leftarrow k + 1$   
    evaluate error.  
**end while**  
 $r \leftarrow r_k$   
 $u \leftarrow Wr + q$

---

## 7.2 Successive approximation method

The methods of successive approximation is a natural tool for the numerical realization of Problem FC. It is based on the Tresca approximation of the Coulomb cone as it is described in Section 3.4 and the work of the celebrated Czech school which summarizes the seminal work of the Czech school (Nečas et al, 1980; Haslinger, 1983, 1984; Haslinger et al, 1996). Each iterative step is represented by an auxiliary contact problem with given friction described by quadratic program over a cylinder (90), that we recall there:

$$\left\{ \begin{array}{l} \theta = h(r_N) \\ \min \frac{1}{2} r^\top W r + r^\top q \\ \text{s.t. } r \in C(\mu, \theta). \end{array} \right. \quad (131)$$

The radius of the cylinder is then updated in an iterative procedure. The algorithm is described in Algorithm 11

---

**Algorithm 11** Tresca approximation algorithm for Problem FC

---

**Require:**  $W, q, \mu$   
**Require:**  $r_0$  initial values  
**Require:**  $\text{tol} > 0, \text{tol}_{\text{int}}$  tolerance values and  $\text{iter}_{\text{max}} > 0$  the max number of iterations  
**Ensure:**  $r, u$  solution of Problem FC  
 $r_k \leftarrow r_0 ; k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\text{max}}$  **do**  
     $\theta \leftarrow h(r_{N,k})$   
    solve (131) for  $r_{N,k+1}, r_{T,k+1}$  at accuracy  $\text{tol}_{\text{int}}$   
     $k \leftarrow k + 1$   
    evaluate error.  
**end while**  
 $r \leftarrow r_k$   
 $u \leftarrow W r + q$

---

In the literature, the successive approximation technique has been used in the bidimensional case in (Haslinger et al, 2002) & (Dostál et al, 2002) with improved and dedicated QP solvers over box-constraints. Two strategies are implemented : a) the classical Tresca iteration (called FPMI) and b) the Panagiotopoulos decomposition plus a Fixed point (called FPMII). They use a specific QP solver for box constraint (Dostál, 1997) that is an improvement of Moré–Toraldo method (Moré and Toraldo, 1991). This technique has been directly extended in the three-dimensional case with a faceting of the cone in (Haslinger et al, 2004). In the latter case, the problem is still a box constrained QP since it contains only polyhedral constraints. In (Haslinger et al, 2012) the authors propose a successive approximation technique in 3D with the special solver of (Kučera, 2007, 2008) which is itself an extension to disk constraints of the Polyak method (conjugate gradient with active set on the

bounds constraint) and its improvements (Dostál, 1997; Dostál and Schöberl, 2005). Other improvements of the method may be found in (Dostál and Kučera, 2010) with a last improvement of the method in (Dostál and Kozubek, 2012). All this work is summarized and details in (Dostál, 2016)

### 7.3 ACLM approach

In the convex SOCCP approach described in Section 3.4, we have to solve for a given value the following problem

$$\begin{cases} \min \frac{1}{2} r^\top W r + r^\top (q + s) \\ \text{s.t. } r \in K. \end{cases} \quad (132) \quad \{\text{eq:ACLM-4}\}$$

which is again a convex quadratic program over second-order cone. The approach listed below could again be used to solve this problem. In this work, we solve it by three different ways: a) an adaptation of one of the splitting techniques detailed in Section 6, b) using the projected gradient algorithm dedicated to convex QP described in 9 or c) the fixed point algorithm and the extra gradient algorithm of Section 4. The algorithm is described in Algorithm 12 and we detailed the algorithms we use in the present study in Table 4.

---

#### Algorithm 12 ACLM approximation algorithm for Problem FC

---

**Require:**  $W, q, \mu$   
**Require:**  $r_0$  initial values  
**Require:**  $\text{tol} > 0, \text{tol}_{\text{int}}$  tolerance values and  $\text{iter}_{\text{max}} > 0$  the max number of iterations  
**Ensure:**  $r, u$  solution of Problem FC  
 $u_0 \leftarrow W r_0 + q$  ;  $k \leftarrow 0$   
**while**  $\text{error} > \text{tol}$  and  $k < \text{iter}_{\text{max}}$  **do**  
     $s \leftarrow g(u_k)$   
    solve (132) for  $r_{k+1}$  at accuracy  $\text{tol}_{\text{int}}$   
     $u_{k+1} \leftarrow W r_{k+1} + q$   
     $k \leftarrow k + 1$   
    evaluate error.  
**end while**  
 $r \leftarrow r_k$   
 $u \leftarrow u_k$

---

A nomenclature for the algorithms based on the optimisation approach is given in Table 4.

#### 7.4 Convex relaxation and the SOCCP approach

Finally, we propose to compare the optimization based algorithm to a complete convex relaxation of the problem by solving the convex SOCCP (132) with  $s = 0$ . This procedure is very similar to the approach in (Tasora and Anitescu, 2009; Anitescu and Tasora, 2010; Tasora and Anitescu, 2011) where the convex problem is only solved.

#### 7.5 Control of the tolerance of internal solvers $\text{tol}_{\text{int}}$ in optimization approach

In Algorithms 10, 11 and 12, an internal tolerance is used to control the accuracy of the internal solver. It is generally not useful to solve the internal problem at the accuracy of the global one. In the comparison study, we set the internal tolerance  $\text{tol}_{\text{int}}$  to  $\text{error}/10.0$ .

Name	Algo.	Additional informations
PANA-PGS-FP-VI-UPK	10	The normal problem is solved by a PGS algorithm and the tangent problem is solved with the FP-VI-UPK algorithm
PANA-PGS-EG-VI-UPK	10	The normal problem is solved by a PGS algorithm and the tangent problem is solved with the EG-VI-UPK algorithm
PANA-PGS-CONVEXQP-PG	10 & 9	The normal problem is solved by a PGS algorithm and the tangent problem is solved with Algorithm 9
PANA-CONVEXQP-PG	10&9	Both normal and tangent problems are solved with Algorithm 9
TRESCA-NSGS-FP-VI-UPK	11	The problem 131 is solved with the FP-VI-UPK algorithm
TRESCA-FP-VI-UPK	11 & 1	The problem 131 is solved with the FP-VI-UPK algorithm
TRESCA-EG-VI-UPK	11 & 2	The problem 131 is solved with the EG-VI-UPK algorithm
TRESCA-CONVEXQP-PG	11 & 9	The problem 131 is solved with Algorithm 9
ACLM-NSGS-FP-VI-UPK	12	The problem 132 is solved with the NSGS-FP-VI-UPK algorithm
ACLM-FP-VI-UPK	12& 1	The problem 132 is solved with the FP-VI-UPK algorithm
ACLM-EG-VI-UPK & 2	12	The problem 132 is solved with the EG-VI-UPK algorithm
ACLM-CONVEXQP-PG	12	The problem 132 is solved with the Algorithm 9

Table 4: Naming convention for optimization based algorithms

## 8 Comparison framework

In this section, we present our comparison framework. Especially, we specify how we measure performance and how the performance profile are built.

### 8.1 Measuring errors

A key parameter in the measurement of performance of the solver is the definition of errors. The absolute error is given by the norm of the natural map. A relative error is computed with respect to the norm of the vector  $q$ . More precisely, the error is given by

$$\text{error} = \frac{\|F_{\text{vi}}^{\text{nat}}(r)\|}{\|q\|}. \quad (133) \quad \{\text{eq:error-1}\}$$

This error is used as the final error measurement for all solvers.

For some iterative solvers such as VI-FP, VI-EG, NSGS and PSOR, the computation of the error at each iteration penalizes the performance of the solver since it amounts to computing a matrix-vector product that costs more than one iteration. This is the reason why a cheaper error measurement is used inside the main loop in Algorithms 1, 2 and 7. This cheaper error measurement is based on

$$\text{error}_{\text{cheap}} = \frac{\|r_{k+1} - r_k\|}{\|r_k\|}. \quad (134) \quad \{\text{eq:error-2}\}$$

The tolerance of solver is then self-adapted in the loop to meet the required tolerance based on the error given by (133).

### 8.2 Performance profiles

The concept of performance profiles was introduced in (Dolan and Moré, 2002) for bench-marking optimization solvers. It enables to comparison of several solvers on a large set of problems. For a set  $P$  of  $n_p$  problems, and a set  $S$  of  $n_s$  solvers, we define a performance criterion for a solver  $s$  and a problem  $p$  by

$$t_{p,s} = \text{computing time required for solver } s \text{ to solve problem } p, \quad (135) \quad \{\text{eq:measure}\}$$

and a performance ratio over all the solvers

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s}, s \in S\}} \geq 1. \quad (136)$$

For  $\tau \geq 1$ , we define a distribution function  $\rho_s$  for the performance ratio for a solver  $s$  as

$$\rho_s(\tau) = \frac{1}{n_p} \text{size}\{p \in P, r_{p,s} \leq \tau\} \leq 1. \quad (137)$$

This distribution computes the number of problems  $p$  that are solved with a performance ratio below a given threshold  $\tau$ . In other words,  $\rho_s(\tau)$  represents the probability that the solver  $s$  has a performance ratio not larger than a factor  $\tau$  of the best solver. It is worth noting that  $\rho_s(1)$  represents the probability that the solver  $s$  beats the other solvers, and  $\rho_s(\infty)$  characterizes the robustness of the method. The higher  $\rho_s$  is, the better the method is. The term *performance profile* will be used in the sequel for this graph of the functions  $\rho_s(\tau), \tau \geq 1$ .

In (135), the computational time is used to measure performance. Other criterion can be used. The number of floating point operations (flops) is a better measure of performance since it is independent of the computer. Unfortunately, it is usually difficult to measure automatically it in a robust over various platform. This is the reason why we prefer use the computational time. In our experiments, we decided to fix the required accuracy with the tolerance of each solver. Another performance criteria could be also used. For instance, we can fix a given computational time and measure to obtained error of the solver. This a way to measure the ability of solver to give a solution in a given time that may be interesting for real-time applications. Another way to measure performance may also to divide the computational time by the number of contacts in order to judge of the ability of the solver to be scalable. For the sake of conciseness, this has not been done in this article.

### 8.3 Benchmarks presentation

To perform the comparison of the solvers on a fair basis, we use a large set of problems that comes from various applications. This collection is FCLib (Frictional Contact libraries) which is an open collection of problems in a hdf5 format described in (Acary et al, 2014) and more information can be found at <https://frictionalcontactlibrary.github.io>. The whole collection of problems can be found at <https://github.com/FrictionalContactLibrary/fclib-library> and we used the version v1.0 for the comparisons that contains 2368 problems.

The test sets are illustrated in Figure 3 and some details on their contents are given in Table 5. All the problems has been generated thanks to the software codes LMGC90 and Siconos. In Table 5, the number of degrees of freedom  $n$  corresponds to the degrees of freedom of the systems before its condensation to local variables. In other words, the number of rows of the matrix  $M$  and  $H$  in (5). The contact density  $c$  is the ratio of the number of contact unknowns over the number of degrees of freedom:

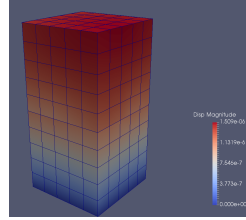
$$c = \frac{3n_c}{n} = \frac{m}{n}. \quad (138)$$

The coefficient  $c$  corresponds also to the ratio between the number of rows of  $H$  over its number of columns. If this number is larger than 1, the matrix  $H$  can not be full row rank and then the matrix  $W$  is also rank deficient. In the case that  $m > n$ , we can

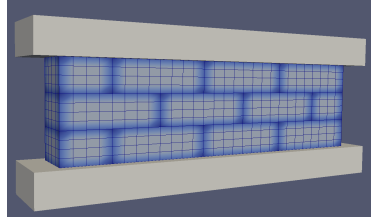
{eq:fclib-1}

observe in Table 5 that this number  $c$  is a good approximation of the rank ratio of the matrix  $W$  in our applications. The estimation of the rank of matrix  $W$  shows that is is very close to the number of degrees of freedom of the system when  $c > 1$ . For  $c \gg 1$ , the contact density is really high and the system suffers from hyperstaticity as we discussed in Section 2.3. In Table 5, we also give an estimation of the conditioning of the matrix  $W$ . When is was possible from a computational point of view, we perform a singular value decomposition (SVD) of the matrix  $W$  to estimate the spectral radius and then the conditioning by cutting the small eigenvalues. This process has two drawbacks. Firstly, the computation of the SVD decomposition can be really expensive for large dense matrix. Secondly, the value of the condition number of the matrix is very sensitive to the threshold for cutting off the small eigenvalues renders. This is we also use LSMR (Fong and Saunders, 2011) algorithm to give a better approximation of the condition number of rank deficient matrix.

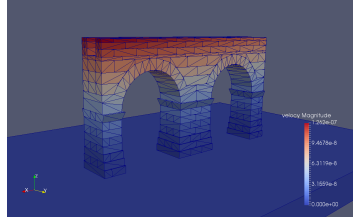
The four first tests in 5 are examples that involve flexible elastic bodies meshed by element methods. The matrix  $W$  in that case is full rank. We will call this set of examples the flexible test set in the sequel.



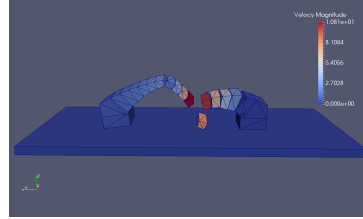
(a) Cubes\_H8



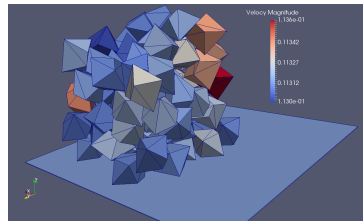
(b) LowWall\_FEM



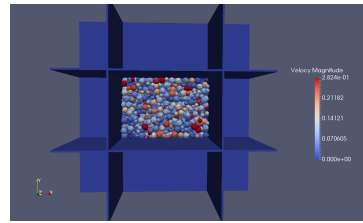
(c) Aqueduct\_PR



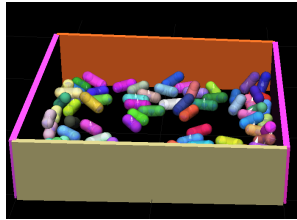
(d) Bridge\_PR



(e) 100\_PR\_Peribox



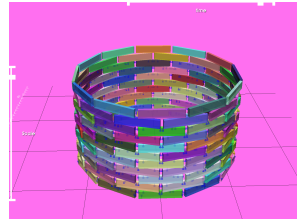
(f) 945\_SP\_Box\_PL



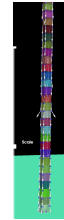
(g) Capsules



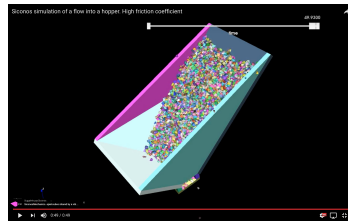
(h) Chain



(i) KaplasTower



(j) BoxesStack



(k) Chute\_1000, Chute\_4000, Chute\_local\_problems

Fig. 3: Illustrations of the FClib test problems



Test set	code	friction coefficient $\mu$	# of problems	# of d.o.f.	# of contacts	contact density $c$	rank ratio(W)	cond(W)	cond(W) LSMR
Cubes_H8_2	LMGC90	0.3	15	162	[3:5]	[0.02:0.09]	1	$[2.2 \cdot 10^1, 1.3 \cdot 10^3]$	$[8.1 \cdot 10^5, 1.5 \cdot 10^6]$
Cubes_H8_5	LMGC90	0.3	50	1296	[17:36]	[0.02:0.09]	1	$[3.3 \cdot 10^4, 7.2 \cdot 10^4]$	$[1.3 \cdot 10^6, 3.1 \cdot 10^6]$
Cubes_H8_20	LMGC90	0.3	50	55566	[361:388]	[0.019:0.021]	1	$[2.4 \cdot 10^5, 2.5 \cdot 10^5]$	$[1.3 \cdot 10^6, 5.2 \cdot 10^6]$
LowWall_FEM	LMGC90	0.83	50	{7212}	[624:688]	[0.28:0.29]	1	–	$[9.3 \cdot 10^2, 5.0 \cdot 10^5]$
Aqueduct_PR	LMGC90	0.8	10	{1932}	[4337:4811]	[6.81:7.47]	[6.80:7.46]	$[4.7 \cdot 10^7, 3.4 \cdot 10^8]$	$[6.7 \cdot 10^1, 1.5 \cdot 10^2]$
Bridge_PR	LMGC90	0.9	50	{138}	[70:108]	[1.5:2.3]	[2.27,2.45]	$[8.3 \cdot 10^4, 1.1 \cdot 10^5]$	$[1.9 \cdot 10^3, 2.6 \cdot 10^4]$
100_PR_Peribox	LMGC90	0.8	106	{606}	[14:578]	[0.2:3]	[1.76:3.215]	$[4.3 \cdot 10^2, 1.0 \cdot 10^6]$	$[6.3 \cdot 10^5, 3.5 \cdot 10^6]$
945_SP_Box_PL	LMGC90	0.8	60	{5700}	[2322:5037]	[1.22:2.65]	[1.0, 2.66]	$[2.2 \cdot 10^4, 4.4 \cdot 10^5]$	$[2.9 \cdot 10^1, 9.2 \cdot 10^2]$
Capsules	Siconos	0.7	249	{300}	[1:200]	[1.2:1.5]	[1.08:1.55]	$[1.2 \cdot 10^6, 7.5 \cdot 10^9]$	–
Chain	Siconos	0.3	242	{60}	[8:28]	[0.5:1.3]	[1.05:1.6]	$[7.4 \cdot 10^4, 4.0 \cdot 10^9]$	$[1.5 \cdot 10^1, 4.7 \cdot 10^5]$
KaplasTower	Siconos	0.7	201	[72:792]	[48:933]	[3.0:3.6]	[2.0:3.53]	[67 : 2174]	[8 : 67]
BoxesStack	Siconos	0.7	255	[6:300]	[1:200]	[1.86:2.00]	[1.875,2.0]	$[3.8 \cdot 10^4, 2.5 \cdot 10^7]$	$[9.0, 5.4 \cdot 10^3]$
Chute_1000	Siconos	1.0	156	[276:5508]	[74:5056]	[0.69:2.95]	[1.0:2.95]	$[2.1 \cdot 10^1, 1.9 \cdot 10^3]$	
Chute_4000	Siconos	1.0	40	[17280:20034]	[15965:19795]	[2.51:3.06]	–	–	$[5.5 \cdot 10^1, 9.0 \cdot 10^3]$
Chute_local_problems	Siconos	1.0	834	3	1	1	1	[1.04:4.66]	$[2.6, 2.6 \cdot 10^1]$

Table 5: Description of the test sets of FCLib library (v1.0)

---

**REDACTION NOTE V.A. 8.1***Can we say more on the description of the examples ?*

---

**8.4 Software & implementation details**

All the solvers that are used in this article are implemented in standard C in the component of Siconos called Siconos/Numerics. The aim of Siconos is to provide a common platform for the modeling, simulation, analysis and control of general nonsmooth dynamical systems. More information on the software is available at <http://siconos.gforge.inria.fr> and the software can be downloaded at <https://github.com/siconos/siconos>. The algebraic manipulations are based on BLAS/LAPACK and we use SuiteSparse for the implementation of sparse matrices. The algorithm VI-FP, VI-EG, NSGS and PSOR uses the sparse block structure of the matrix  $W$ . The NSN solvers relies on a standard sparse implementation. The simulations are performed on the University of Grenoble-Alpes cluster CIMENT<sup>3</sup>.

---

**REDACTION NOTE V.A. 8.2***something more ?*

---

**8.5 Simulation campaign**

In Sections 9 and 10, we report the results for the simulation campaign. Since the large amount of data and the numerous of performance profiles for each test sets that are generated, we do not report any profiles when a list of solvers fails or is not able to meet the required tolerance within the prescribed time defined by the timeout values. Nevertheless, the reader may have access to the complete list of performance profiles at [https://github.com/siconos/faf/blob/master/TeX/Full-test/full-test\\_current.pdf](https://github.com/siconos/faf/blob/master/TeX/Full-test/full-test_current.pdf).

---

**REDACTION NOTE V.A. 8.3***a permanent address on hal would be better*

---

---

<sup>3</sup> <https://ciment-grid.ujf-grenoble.fr/>

Test set	precision	timeout (s)	mean performance of the fastest solver $\mu\{\min\{t_{p,s}, s \in S\}\}$	std. deviation performance of the fastest solver $\sigma(\min\{t_{p,s}, s \in S\})$	mean performance of the fastest solver by contact $\mu\{\min\{t_{p,s}/n_{c,p}, s \in S\}\}$	std. deviation performance of the fastest solver by contact $\sigma(\min\{t_{p,s}/n_{c,p}, s \in S\})$	# of unsolved problems
Cubes_H8_*	$10^{-08}$	100	1.73	2.13	$4.83^{-03}$	$5.78^{-03}$	0
Cubes_H8_* II	$10^{-04}$	100	0.92	1.06	$2.66^{-03}$	$2.83^{-03}$	0
LowWall_FEM	$10^{-04}$	400	14.8	2.85	$2.16^{-02}$	$4.54^{-03}$	0
Aqueduct_PR	$10^{-04}$	200	5.80	6.36	$4.90^{-04}$	$3.03^{-04}$	0
Bridge_PR	$10^{-08}$	100	11.0	13.6	$1.05^{-01}$	$1.33^{-01}$	0
Bridge_PR II	$10^{-04}$	100	0.048	0.038	$1.30^{-03}$	$1.42^{-03}$	0
100_PR_Peribox	$10^{-04}$	100	0.064	0.062	$1.56^{-04}$	$1.22^{-04}$	0
945_SP_Box_PL	$10^{-04}$	100	3.20	1.71	$6.45^{-04}$	$3.36^{-04}$	0
Capsules	$10^{-08}$	50	$1.46 \cdot 10^{-02}$	$1.74 \cdot 10^{-02}$	$5.67^{-05}$	$6.26^{-05}$	0
Chain	$10^{-08}$	50	$6.19 \cdot 10^{-04}$	$3.68 \cdot 10^{-04}$	$3.15 \cdot 10^{-05}$	$1.46 \cdot 10^{-05}$	0
KaplasTower	$10^{-08}$	200	$1.27 \cdot 10^{-01}$	$3.75 \cdot 10^{-01}$	$1.84 \cdot 10^{-04}$	$4.57 \cdot 10^{-04}$	0
KaplasTower II	$10^{-04}$	100	$2.84 \cdot 10^{-02}$	$1.51 \cdot 10^{-01}$	$3.39 \cdot 10^{-05}$	$1.84 \cdot 10^{-04}$	0
BoxesStack	$10^{-08}$	100	$3.42 \cdot 10^{-02}$	$8.87 \cdot 10^{-02}$	$3.24 \cdot 10^{-04}$	$9.77 \cdot 10^{-04}$	0
Chute_1000	$10^{-04}$	200	2.62	3.06	$6.76^{-04}$	$6.58^{-04}$	0
Chute_4000	$10^{-04}$	200	10.52	7.88	$5.71^{-04}$	$4.07^{-04}$	0
Chute_local_problems	$10^{-08}$	10	$1.80 \cdot 10^{-04}$	$1.57 \cdot 10^{-05}$	$1.80 \cdot 10^{-04}$	$1.57 \cdot 10^{-05}$	0

Table 6: Parameters of the simulation campaign

## 9 Comparison of methods by family

In this section, we perform a comparison of the solvers by family in order to study the influence of the various possible parameters on the performance of the solvers.

### 9.1 Numerical methods for VI: FP-DS, FP-VI- $\star$ and FP-EG- $\star$

In Figure 4, we compare the different numerical solvers for VI described in Section 4. Only the test sets for which the solvers are reached the precision before the prescribed time (timeout) are presented. For that reason, the results for the test sets LowWall\_FEM, Cubes\_H8, Bridge\_PR, AqueducPR, 945\_SP\_Box\_PL, BoxesStack, Chute\_4000 and Chute\_1000 are not depicted. The solvers are quite slow to converge in practice for large problems although they are robust except for FP-DS. The main conclusions are as follows:

1. The solver FP-DS suffers from robustness problems and a lot of divergence has been observed.
2. The solvers FP-VI- $\star$  and FP-EG- $\star$  are really robust but slow. They are able to solve all the problems but they require a lot of time and we didn't observe divergence issues. Comparing to FP-DS, the self-adaptive rule for size the parameter  $\rho_k$  is of utmost importance for robustness.
3. Except for the test set KaplasTower II, the FP-EG- $\star$  performs better than FP-VI- $\star$ .
4. The difference between the adaptive strategy for sizing  $\rho_k$ , UPK and UPTS is very small in all the examples and the choice of the update rule is not fundamentally important.

### 9.2 Splitting based algorithms: NSGS- $\star$ and PSOR- $\star$

Again the results on the test sets Bridge\_PR, Chute\_local\_problems, BoxesStack are not reported since the NSGS- $\star$  are not able to meet the required tolerance for a sufficient percentage of the problems mainly due to a slow convergence rate. We can see that this problem also appears in the Capsules test set.

Effect and influence of the local solver in NSGS- $\star$  algorithms

In Figure 4, we report the performance profiles of the NSGS- $\star$  for the different local solvers. The main conclusions are as follows:

1. When the timeout value is sufficiently large, we observe the NSGS- $\star$  solvers are quite robust. In other words, the convergence is slow but robust. Indeed, we are

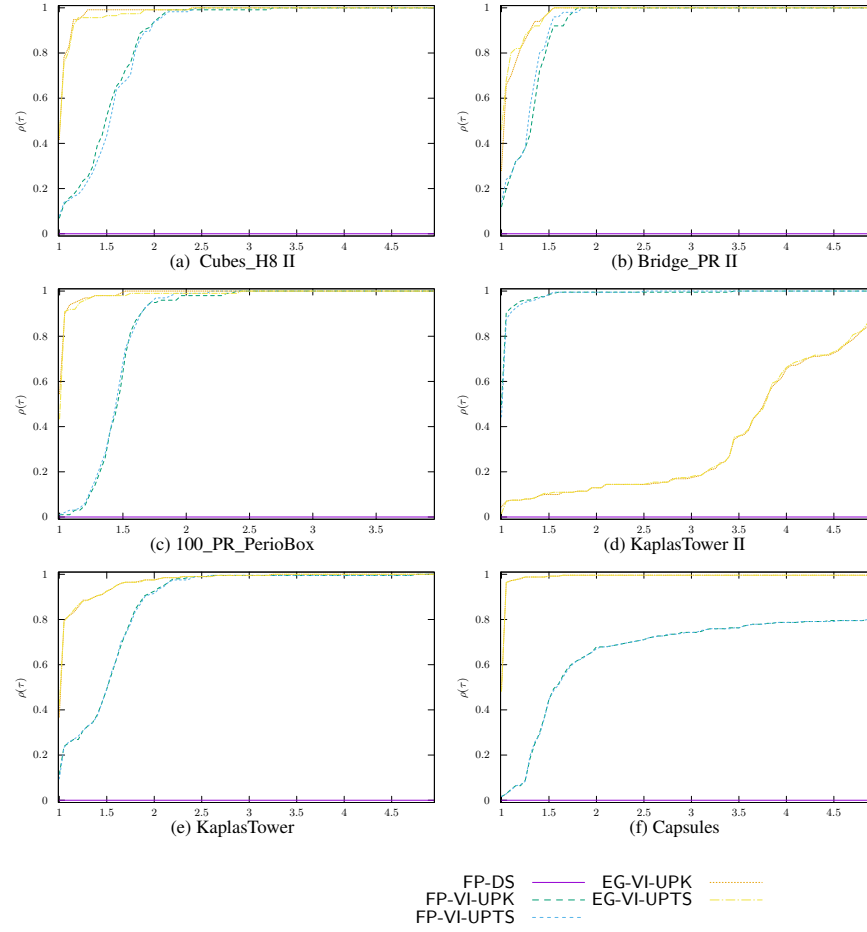
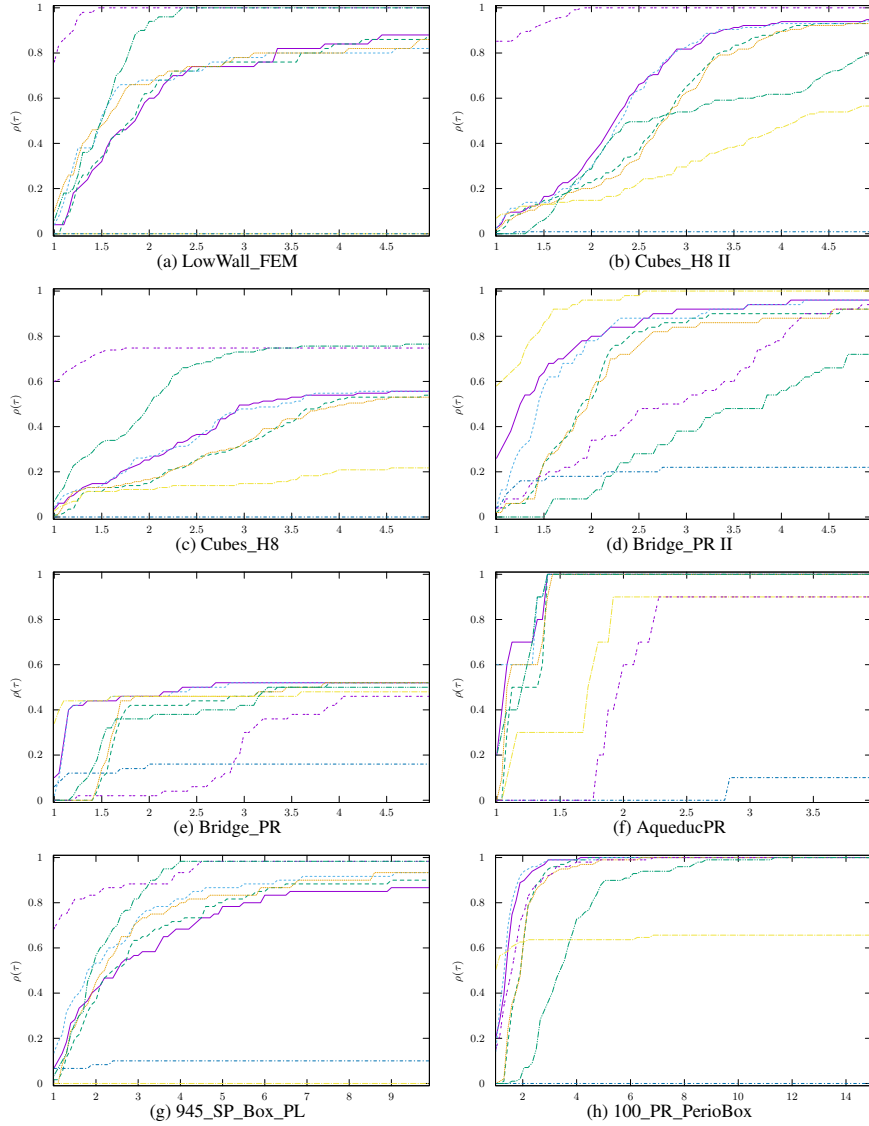


Fig. 4: Comparison of numerical method for VI FP-DS, FP-VI-★ and FP-EG-★

able to find to a local solver for each test sets that is able to give a solution at the required accuracy. Nevertheless, there is no universal efficient local solver that outperforms the other ones and the global efficiency depends on the local solver in various ways according to the test sets.

2. Except for the test sets KaplasTower II and BoxesStack, the solver NSGS-EXACT behaves poorly. This is mainly due to the fact that the solver is not robust to find a solution for one contact when we are far from the solutions for all the contacts. This behavior was already reported in (Daviet et al, 2011) where



another solver based a nonsmooth Newton technique was used when the exact solution was not satisfactory.

3. The solver NSGS-FP-DS-One is most efficient on the test sets Bridge\_PR II, KaplasTower II, Chain and BoxesStack. These tests sets seem to be among the easiest one the solve (see Table 6). Nevertheless, this local solver seems slow or suffers from robustness issues for other test sets.

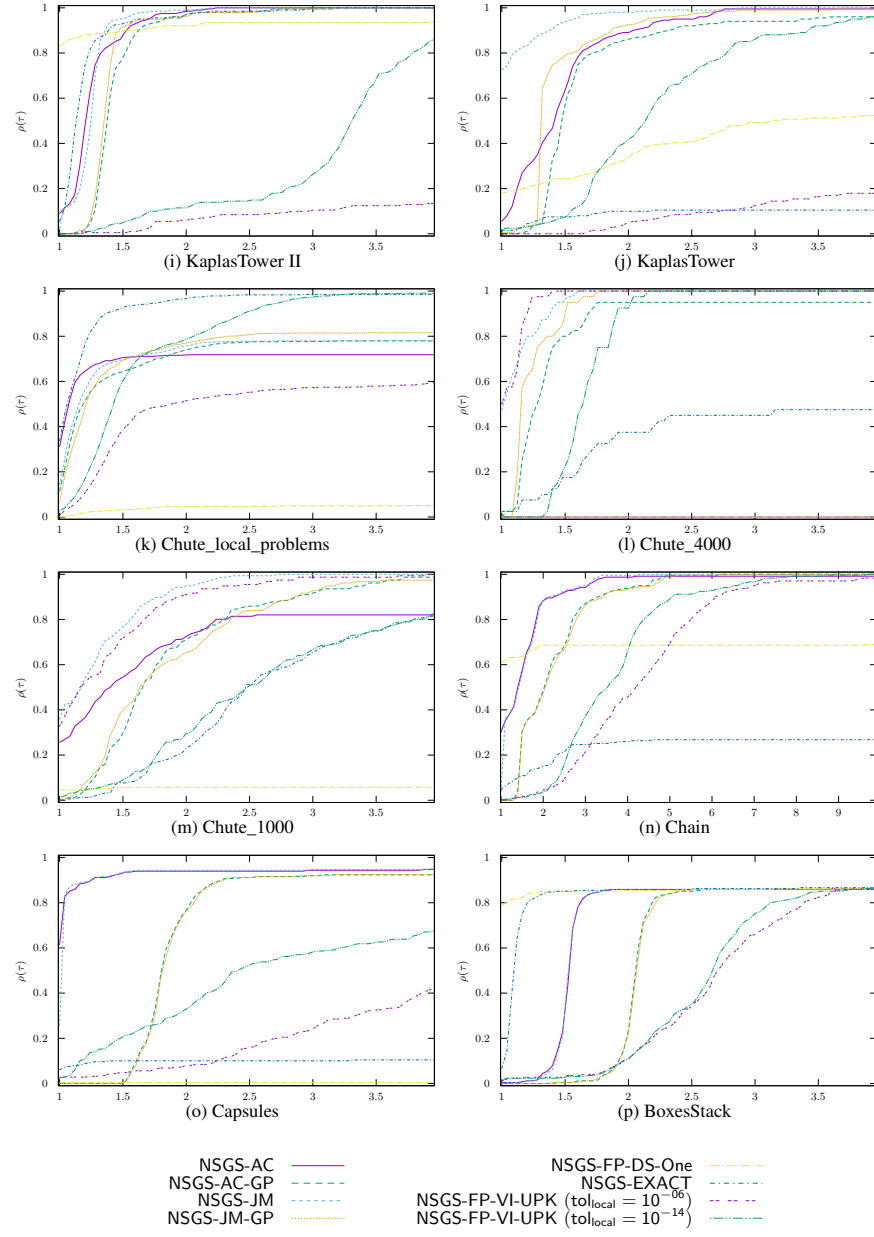


Fig. 4: Influence of the local solver in NSGS-★ algorithms.

4. On the flexible test sets and 945\_SP\_Box\_PL, the best solver is NSGS-FP-VI-UPK for a relatively low required tolerance ( $\text{tol}_{\text{local}} = 10^{-06}$ ). For these test sets, an approximate solution of the single contact problems seems sufficient to ensure an efficient convergence towards the solution without entailing robustness.
5. On the test sets 100\_PR\_PerioBox, KaplasTower, Chain, Capsules, the solver NSGS-NSN- $\star$  are the best solvers and behave very well on Bridge\_PR II. It seems that when a tight accuracy is required, the solvers NSGS-NSN- $\star$  are useful and helps with a tight local tolerance to speed-up the convergence.
6. For the Chute\_1000, Chute\_4000 test sets, we observe large differences between the local formulations of the nonsmooth equations for the Newton solvers (NSGS-NSN-AC or NSGS-NSN-JM). The solver NSGS-NSN-JMx is the best solver and really better than NSGS-NSN-AC although their theoretical formulation are very close. These two test sets are characterized by difficult local problems where the matrix  $W$  is asymmetric.
7. Comment on the line searches method

#### Influence of the tolerance of the local solver $\text{tol}_{\text{local}}$ in NSGS-FP-VI-UPK algorithms

In this paragraph, the tolerance of the local solver is varied and its effect on the global convergence of the solver is reported. In Figure 5, we report the performance profiles of NSGS-FP-VI-UPK algorithms for the  $\text{tol}_{\text{local}}$  in the range  $[10^{-04}, 10^{-16}]$ . We also report the efficiency of the adaptive strategy for setting the value of the local tolerance (see Section 6.3). The main observations are

1. For the test sets that are quickly solved (see Table 6), such as Capsules a tight tolerance on the local solver  $10^{-16}$  improves the efficiency of the NSGS-FP-VI-UPK solver. Similar results are obtained for BoxesStack Chain, Chute\_local\_problems, KaplasTower, KaplasTower II and are not depicted.
2. For the other problems that are longer to solve, that is, when we expect more iterations of the NSGS-FP-VI-UPK solver, the adaptive rule or tight local tolerances are better.

From these results, it is quite difficult to guess in advance the internal dynamics of the solver. Note that the range of  $\tau$  that we used in the graph is quite small so the difference in performance between the solvers is not crucial.

#### Influence of the tolerance of the local solver $\text{tol}_{\text{local}}$ in NSGS-AC-GP algorithms.

In Figure 6, we report the performance profile of NSGS-AC-GP algorithms for the  $\text{tol}_{\text{local}}$  in the range  $[10^{-04}, 10^{-16}]$ . We also test the adaptive strategy value for the local tolerance. Except for the test set Chute\_local\_problems, the main observation is that the local tolerance does not change fundamentally the convergence of the solver. For the test set Chute\_local\_problems, there is no internal dynamics of the



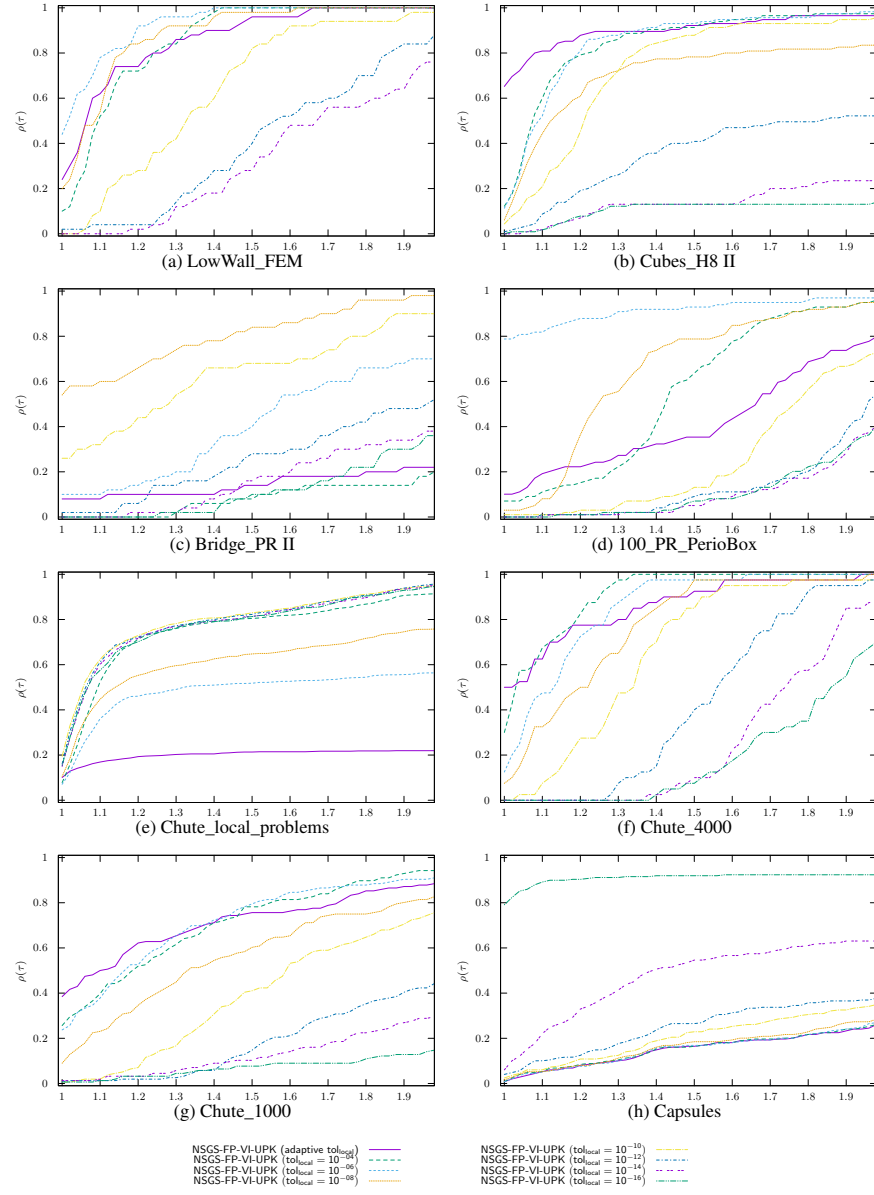


Fig. 5: Influence of the tolerance of the local solver  $\text{tol}_{\text{local}}$  in NSGS-FP-VI-UPK algorithms.

main loop of the NSGS since there is only one contact. It is therefore reasonable to see that adaptive performs better than the other.

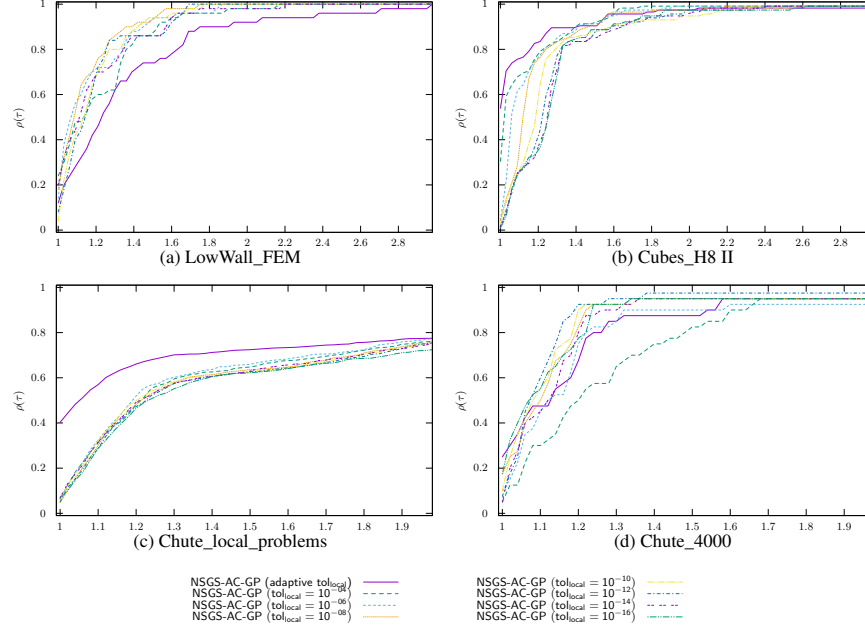


Fig. 6: Influence of the tolerance of the local solver  $\text{tol}_{\text{local}}$  in NSGS-FP-NSN-AC-GP algorithms.

#### Influence of the choice of the parameters $\rho_N, \rho_T$ in the local solver of the NSGS-AC algorithms

In Figure 7, we evaluate the influence of the choice of the parameters  $\rho_N, \rho_T$  on the convergence of the solver. The main conclusion are:

1. For the test sets 945\_SP\_Box\_PL, 100\_PR\_PerioBox, KaplasTower II, Kaplas-Tower, Chute\_local\_problems, Chute\_4000, Chute\_1000, Capsules, a fixed value of  $\rho_N = \rho_T = 1$  has a dramatic effect on the convergence of the algorithm. The scaling of  $\rho$  is of utmost importance for the efficiency and the robustness of the solver. Note that the rule (112) that takes into account the condition number of the local matrix  $W$  deteriorates the performance for Chute\_local\_problems, Chute\_4000, Chute\_1000. In these problems, the local matrix is asymmetric due to large Coriolis effects.

2. For the other tests, the choice of  $\rho_N, \rho_T$  does not really change the results such as LowWall\_FEM, mainly due the fact that the order of magnitude of the chosen  $\rho$  with the rules (110), (111) or (112) is around 1. Cubes\_H8 II, Cubes\_H8, Bridge\_PR II, Bridge\_PR, 100\_PR\_PerioBox, Chain, BoxesStack and Aque-ducPR are not displayed since the results are similar.

One of the conclusion of this study is as follows, the rules (110), (111) improves a lot some simulations without increasing the computational cost for the others. Therefore, it is strongly advised to use it.

#### Influence of the contacts order in NSGS algorithms

In this section, we study the influence of the contact order within the loop of the NSGS-AC-GP solver. We reproduce in Figure 8 the result of the solvers with the original contact list of the problem (NSGS-AC-GP) and with two other ways of iterating over the contacts. The solver NSGS-AC-GP Shuffled corresponds to a single randomization of the list of contacts at the beginning of the algorithm. In the solver NSGS-AC-GP Fully shuffled, the list is shuffled at each iteration.

1. The solver NSGS-AC-GP Fully shuffled performs really better on the flexible test sets
2. For the rigid test sets, we reproduce here only the test set 100\_PR\_PerioBox because the other test sets behave identically. The NSGS-AC-GP Fully shuffled has a really bad influence on the convergence of the solver. It seems that it modify the internal dynamics of the solver in a way the rate of convergence is really decreased.

#### Comparison of PSOR algorithm with respect to the relaxation parameter $\omega$

In Figure 9, the relaxation parameter  $\omega$  is varied ranging in  $[0.5, 1.8]$ . Two conclusions can be drawn:

1. For the flexible tests, the efficiency of the solver is really improved as we decreased the value of  $\omega$ . Moreover, this is done without destroying the robustness of the solver.
2. For the rigid tests, the effect of the relaxation is not so clear. For values of  $\omega$  greater than 1.0, the efficiency is improved but the robustness deteriorates and we observe the contrary for the  $\omega$  less than 1.0. Note in particular that for the test sets Chute\_1000, Chute\_4000 the convergence is completely destroyed for  $\omega = 1.8$ .

To conclude, it is difficult to advice to use PSOR algorithm with  $\omega \neq 1$ . If it accelerates drastically the rate of convergence of the algorithm for some problems it deteriorates the convergence for others. Further studies would be needed to design self-adaptive schemes for the choice of  $\omega$ .

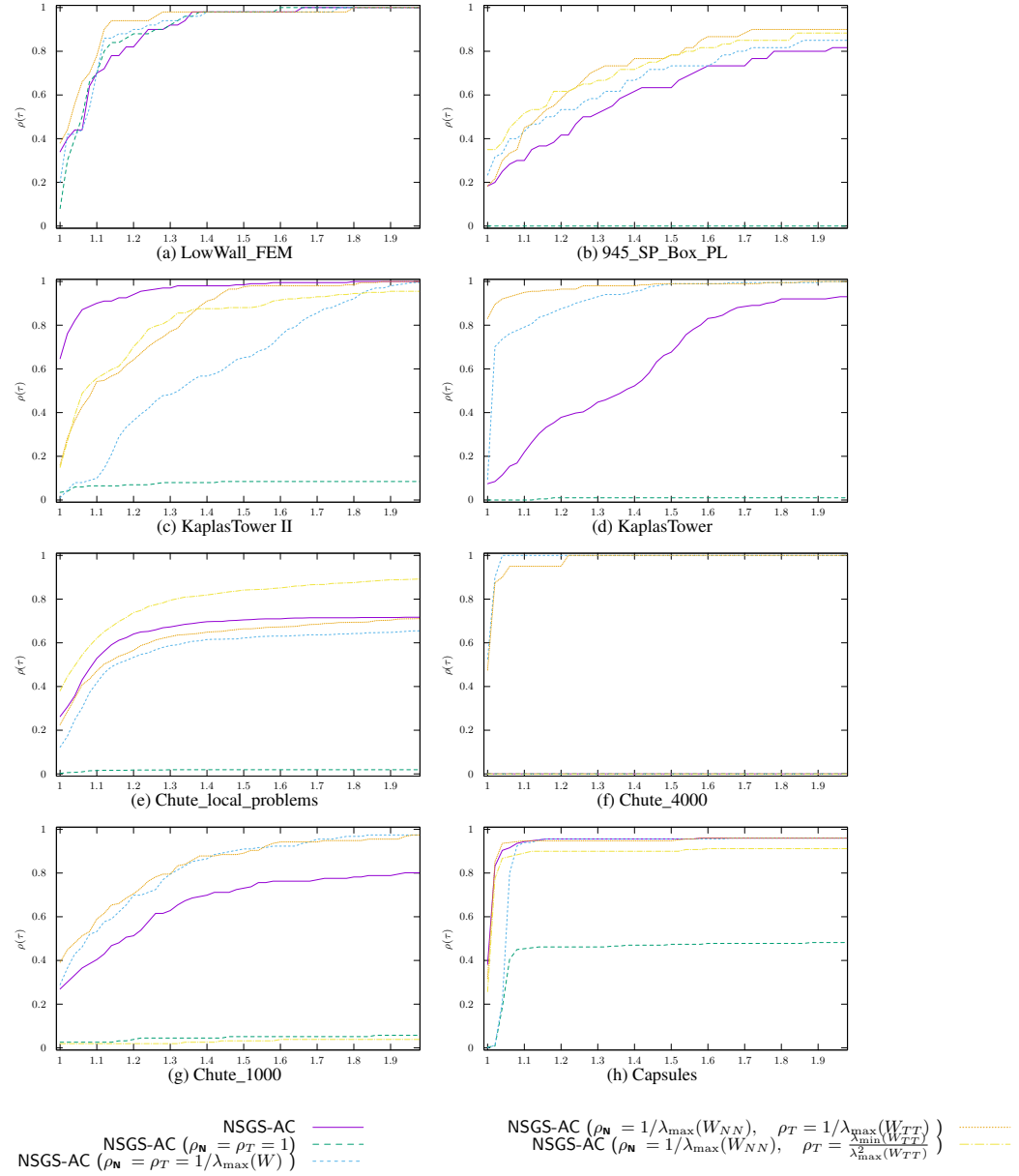


Fig. 7: Influence of the choice of the parameters  $\rho_N, \rho_T$  in the local solver of the NSGS-AC algorithms

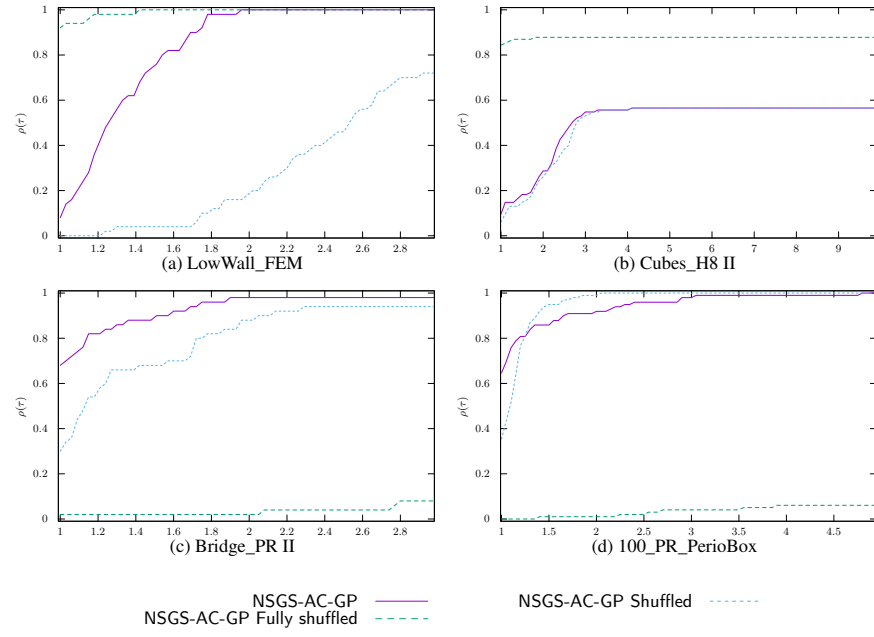
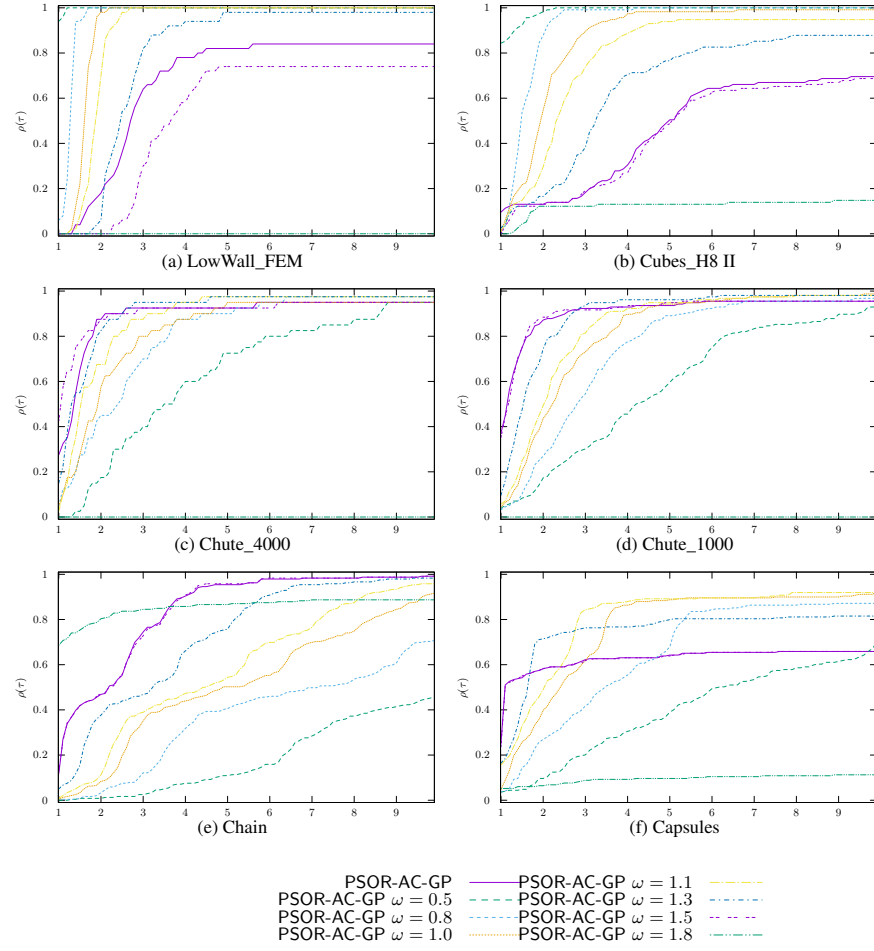


Fig. 8: Influence of the contacts order in NSGS algorithms.

Fig. 9: Effect of relation coefficient  $\omega$  in PSOR-AC-GP algorithm.

### 9.3 Comparison of NSN- $\star$ algorithms

In this section, the nonsmooth Newton methods are compared and we depict the performance profiles in Figure 10. The main conclusion are as follows:

1. For the flexible tests, most of the Newton methods succeed to solve the problems within the prescribed time. The solver NSN-AC-HYBRID appears to be the best solver. The effect of computing an initial condition of the solver with a robust method such as EG-VI-UPK improves the convergence. In practice, we observe the precomputation allows one to determine roughly the set of closed and sliding contacts and it helps a lot the convergence of the Newton solvers. The solvers without line-searches perform also better than those with a line-search procedure which seems to slow down the convergence without improving the robustness. For the different formulations, the NSNS-AC and NSN-JM give equivalent results and are better than the NSN-NM solver which is in turns better than the NSN-FB solver. Note that the Goldstein-Price line search is usually better than the Armijo despite the fact the merit function is not necessarily smooth. Finally, we note that NSN-FB and NSN-FB-A are really the slowest solver on these flexible examples.
2. For the rigid test sets with an high value of the rank ratio or the contact density  $c$  (see Table 5), the Newton methods fail to converge and a lot of divergence has been noted in practice. This is the case for the test sets Bridge\_PR II, Bridge\_PR, AqueducPR, 945\_SP\_Box\_PL, 100\_PR\_PerioBox.
3. For the rigid test sets with a low value of the rank ratio or the contact density  $c$  less than 1 such as Chute\_1000 and Chain, we observe that the Newton methods are able to solve some problems. We note also that in the Chain test set, the use of a fixed value of  $\rho$  is penalizing a lot the convergence of the solver. Contrary to flexible test sets, the use of a line-search procedure helps to get a better robustness of the solver. This is particularly true for NSN-NM-GP.
4. Chute\_local\_problems
5. Finally, for the test sets KaplasTower and Capsules, the NSN-FB-GP is able to solve more than 80% in a very efficient way. Some further studies would be needed to understand why this specific solver performs really better than the others.

As general conclusion, the success of the NSN- $\star$  algorithms relates on the rank of the matrix  $W$  and then on the contact density value  $c$ . For full rank matrix  $W$ , the solvers are robust and efficient. For values of  $c$  not larger than 1, the methods are able to find a solution with a tight accuracy. For larger values of  $c$  and larger rank ratio, the nonsmooth Newton methods are not robust.

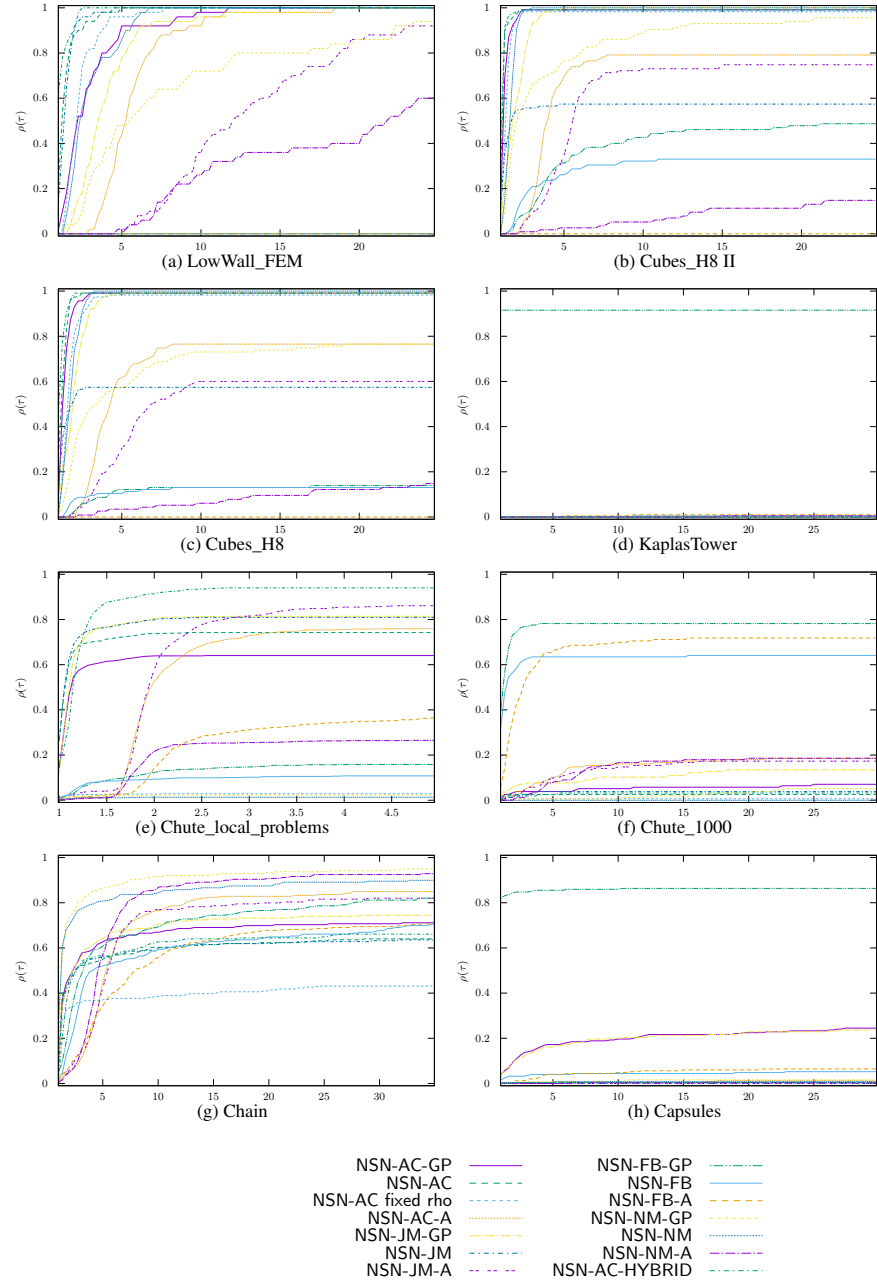


Fig. 10: Comparison of NSN-★ algorithms.



#### 9.4 Comparison of the proximal point algorithm PPA-NSN- $\star$ algorithms

In Figure 11, we compare various the proximal point algorithm with various internal solvers based on nonsmooth Newton methods NSN- $\star$ . The main observations are:

1. For the flexible test sets (see for an illustration the test set LowWall\_FEM), for which the nonsmooth Newton solvers work pretty well, the use of a proximal point algorithm has no interest since it slows down the convergence of the algorithm by performing a first iteration with a large value of the regularization parameter  $\alpha$ .
2. For the test sets KaplasTower, Chute\_1000, Chain, Capsules and BoxesStack, the proximal point approach improves the efficiency of the NSN-AC-GP solver and often it improves its reliability (see for comparison Figure 10). Clearly, the regularization introduced in the proximal point algorithm changes the rank of the matrix  $W$  and it has a strong effect on the convergence of the Newton methods.
3. The efficiency of the proximal point algorithm depends on the internal solver. As is illustrated on the Capsules test, the improvement brought by the proximal point algorithm does not necessarily follows the success of the Newton solver when it is used alone.
4. The strategy for updating the regularization parameter  $\alpha$  plays also an important role. Quite surprisingly, for the Bridge\_PR test set, the adaptive rule that does not take into account the current error is really efficient and allows to get a robust and efficient solver with respect to the others.

In Figure 12, we compare the NSGS-AC solver when it is used directly or inside the proximal point algorithm. On most of the test sets such as KaplasTower, A direct application of the NSGS-AC solver is already efficient and its embedding into a proximal point algorithm does not bring any improvements. Nevertheless, we can see on Figure 12 that the proximal point algorithm improves the robustness and the efficiency for the test sets 945\_SP\_Box\_PL, Chute\_4000, Chute\_1000 and Capsules. For the test set Chute\_4000, the result is quite impressive.

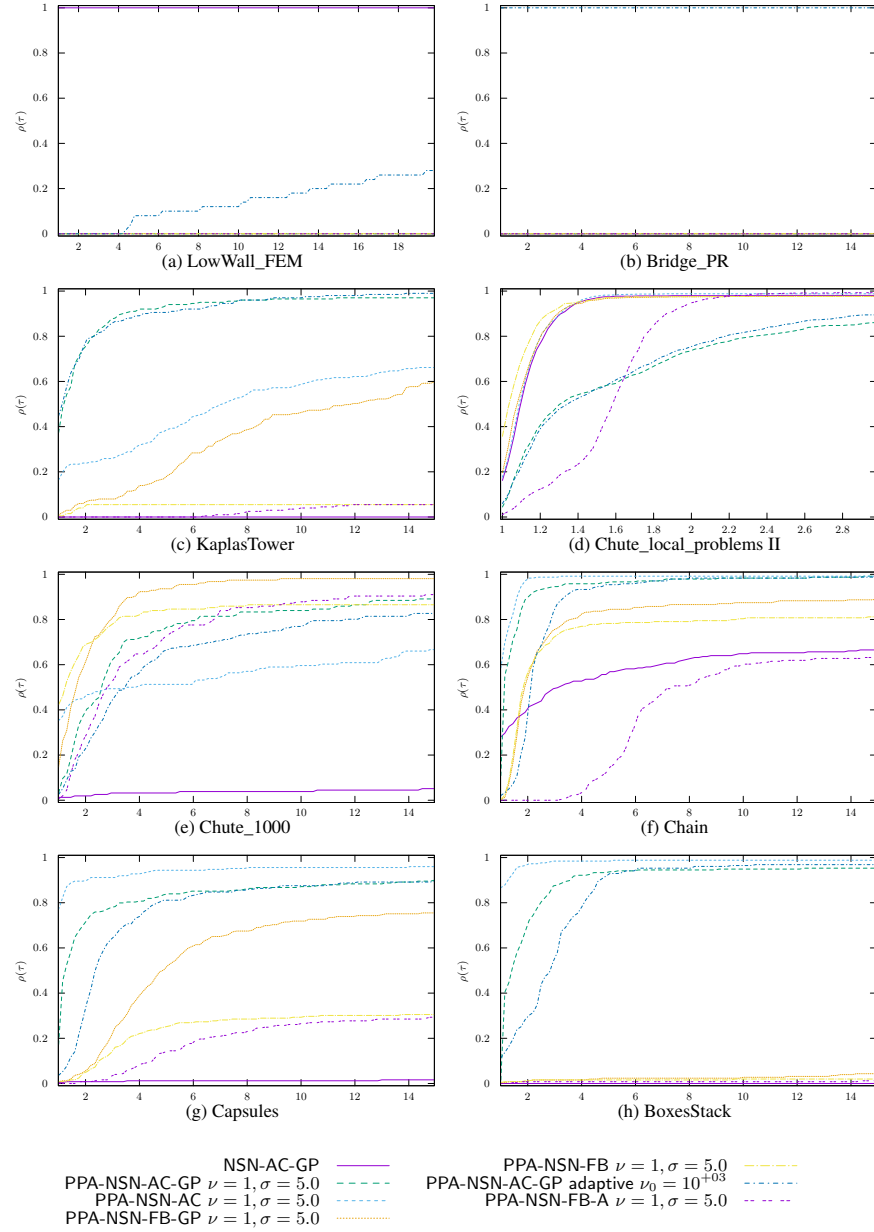
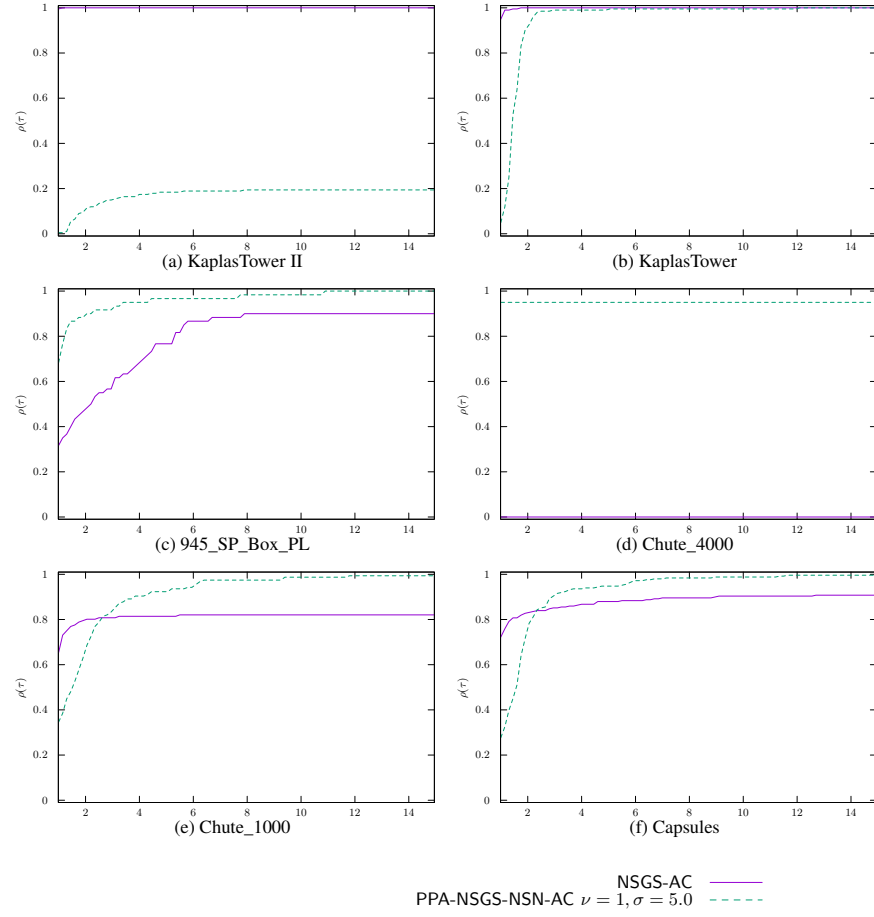
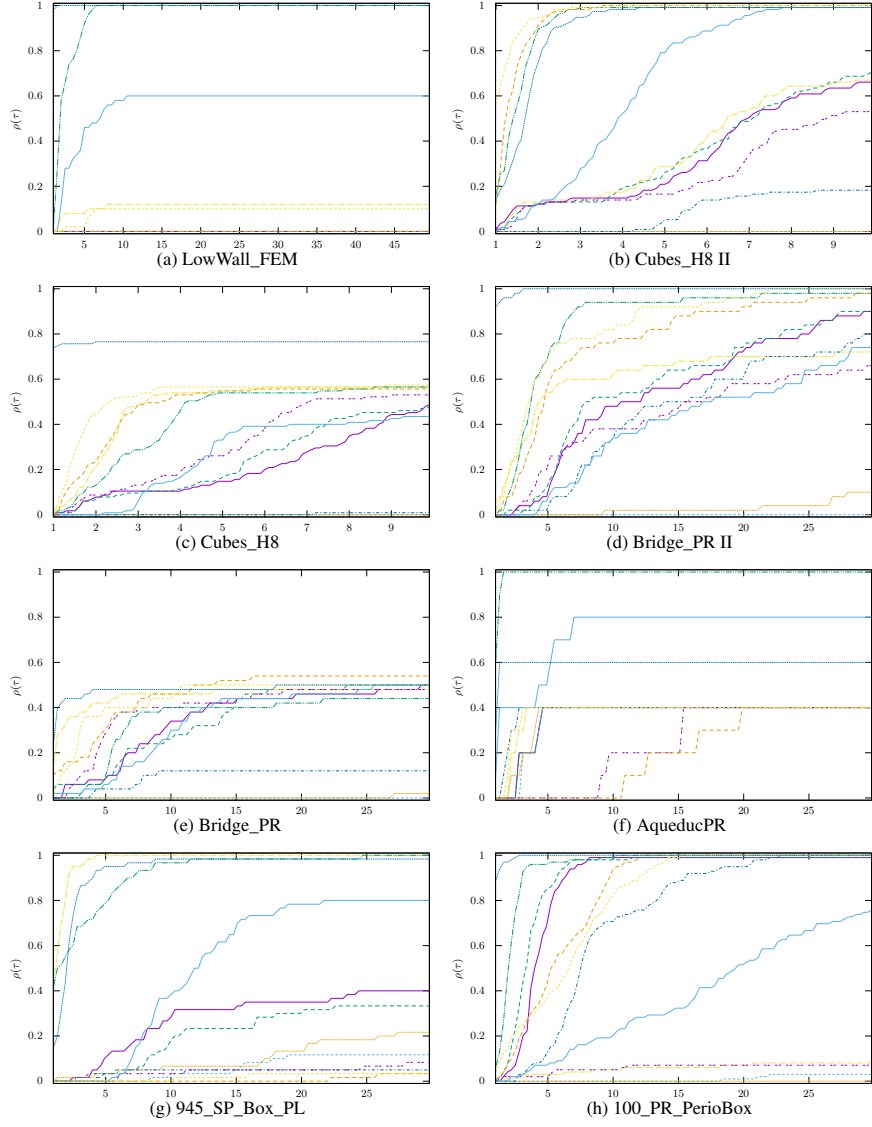


Fig. 11: Comparison of internal solvers in PPA-NSN-★ algorithms.

Fig. 12: Comparison of internal solvers in PPA-NSGS- $\star$  algorithms.

### 9.5 Comparison of optimization-based algorithms



In Figure 12, we compare the algorithms based on the optimization approach presented in Section 7.

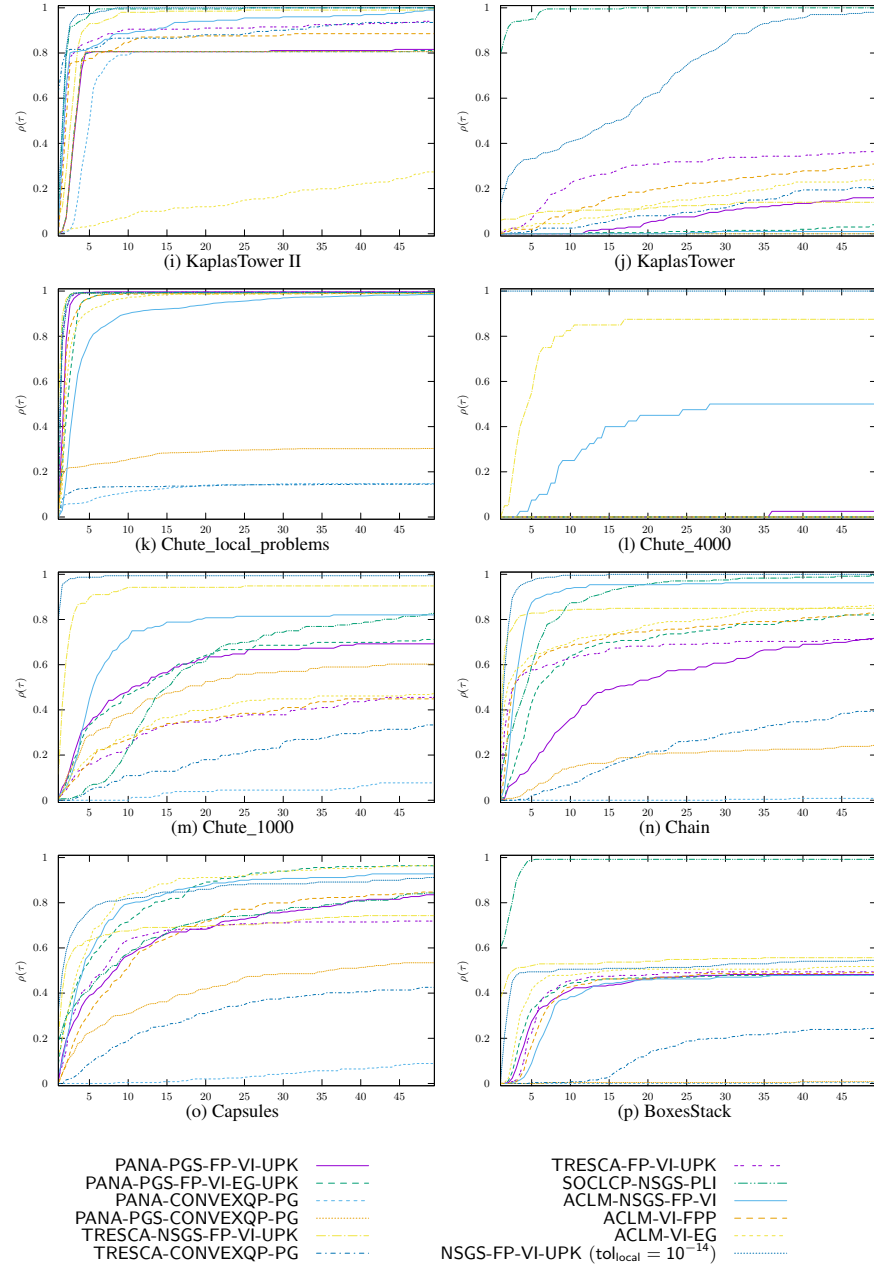
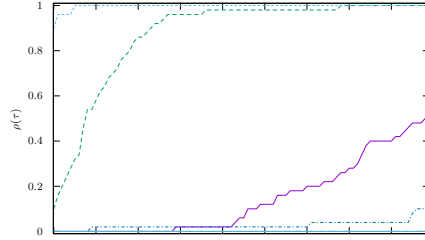
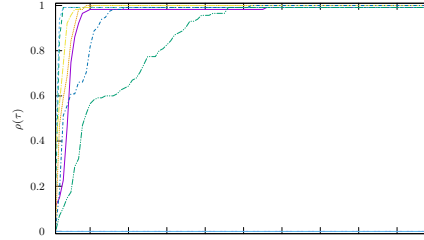


Fig. 12: Comparison of the optimization based solvers

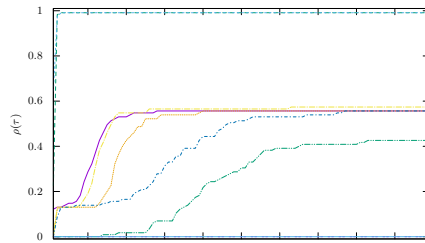
## 10 Comparison of different families of solvers.



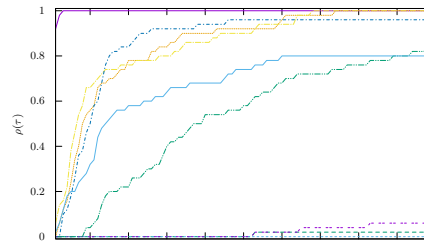
(a) LowWall\_FEM



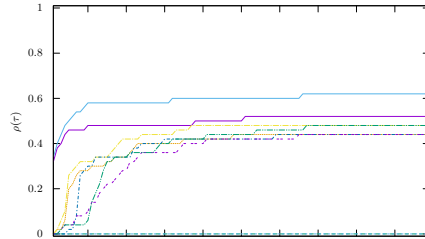
(b) Cubes\_H8 II



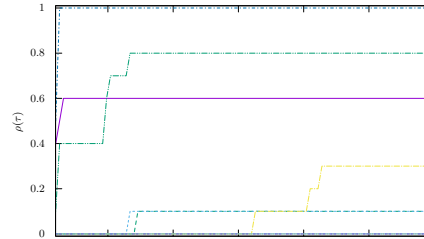
(c) Cubes\_H8



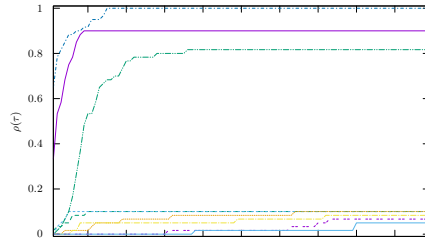
(d) Bridge\_PR II



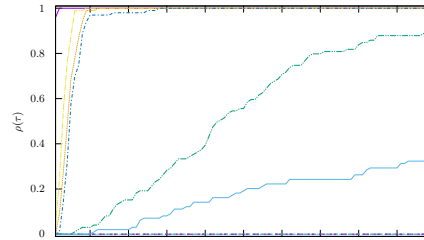
(e) Bridge\_PR



(f) AqueducPR



(g) 945\_SP\_Box\_PL



(h) 100\_PR\_PerioBox

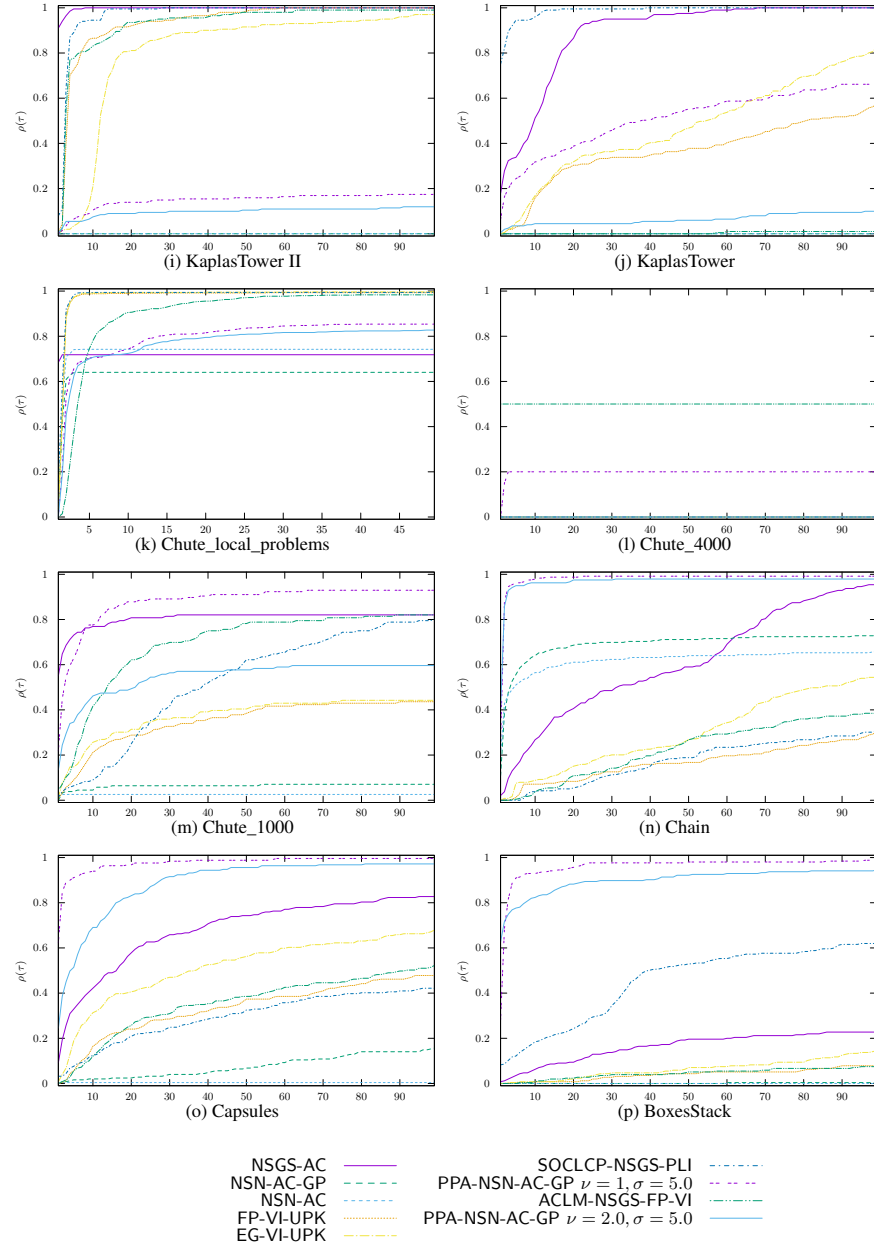


Fig. 12: Comparison of the solvers between families

## 11 General conclusions

- NSGS-★
  1. robust
  2. slow
  3. local solvers must preserve robustness.
- NSN-★
  1. Choice of  $\rho$  and robustness convergence. more studies
- PPA-★
  1. some huge improvements in efficiency and robustness on some tests. Need further studies.

**Acknowledgements** If you want to include acknowledgments of assistance and the like at the end of an individual chapter please use the `acknowledgement` environment – it will automatically render Springer’s preferred layout.

## Appendix



## 12 Basics in Convex Analysis

**Definition 1 ((Rockafellar and Wets, 1997)).** Let  $X \subseteq \mathbb{R}^n$ . A multivalued (or point-to-set) mapping  $T: X \rightrightarrows X$  is said to be (strictly) monotone if there exists  $c(>) \geq 0$  such that for all  $\hat{x}, \tilde{x} \in X$

$$(\hat{v} - \tilde{v})^\top (\hat{x} - \tilde{x}) \geq c \|\hat{x} - \tilde{x}\| \quad \text{with } \hat{v} \in T(\hat{x}), \tilde{v} \in T(\tilde{x}). \quad (139)$$

Moreover  $T$  is said to be maximal when it is not possible to add a pair  $(x, v)$  to the graph of  $T$  without destroying the monotonicity.

The Euclidean projector  $P_X$  onto a closed convex set  $X$ : for a vector  $x \in \mathbb{R}^n$ , the projected vector  $z = P_X(x)$  is the unique solution of the convex quadratic program

$$\begin{cases} \min \frac{1}{2} (y - x)^\top (y - x), \\ \text{s.t. } y \in X. \end{cases} \quad (140) \quad \{\text{eq:opt-proj}\}$$

$$y = P_K(x) \iff \min_{y \in K} \frac{1}{2} (y - x)^\top (y - x) \quad (141)$$

$$\iff -(y - x) \in N_K(y) \quad (142)$$

$$\iff (x - y)^\top (y - z) \geq 0, \forall z \in K \quad (143) \quad \{\text{eq:tutu}\}$$

$$-F(x) \in N_K(x) \iff -\rho F(x)^\top (y - x) \geq 0, \forall y \in K \quad (144)$$

$$\iff (x - (x - \rho F(x)))^\top (y - x) \geq 0, \forall y \in K \quad (145)$$

$$\iff x = P_K(x - \rho F(x)) \text{ thanks to (143)} \quad (146)$$

$$\partial \|z\|_2 = \begin{cases} \frac{z}{\|z\|} \\ \{x, \|x\| = 1\} \end{cases} \quad (147) \quad \{\text{eq:sub-norm}\}$$

### 12.1 Euclidean projection on the disk in $\mathbb{R}^2$ .

Let  $D = \{x \in \mathbb{R}^2, \|x\| \leq 1\}$ . We have

$$P_D(z) = \begin{cases} z & \text{if } z \in D \\ \frac{z}{\|z\|} & \text{if } z \notin D \end{cases} \quad (148) \quad \{\text{eq:projD}\}$$

and

$$\partial P_D(z) = \begin{cases} I & \text{if } z \in D \setminus \partial D \\ I + (s-1)zz^\top, s \in [0, 1] & \text{if } z \in \partial D \\ \frac{I}{\|z\|} - \frac{zz^\top}{\|z\|^3} & \text{if } z \notin D \end{cases} \quad (149) \quad \{\text{eq:sub-projD}\}$$

## 12.2 Euclidean projection on the second order cone of $\mathbb{R}^3$ .

Let  $K = \{x = [x_N x_T]^T \in \mathbb{R}^3, x_N \in \mathbb{R}, \|x_T\| \leq \mu x_N\}$ . We have

$$\{\text{eq:proj}\} \quad P_K(z) = \begin{cases} z & \text{if } z \in K \\ 0 & \text{if } -z \in K^* \\ \frac{1}{1+\mu^2}(z_N + \mu\|z_T\|) \begin{bmatrix} 1 \\ \mu \frac{z_T}{\|z_T\|} \end{bmatrix} & \text{if } z \notin K \text{ and } -z \notin K^* \end{cases} \quad (150)$$

Direct computation of an element of the subdifferential

The computation of the subdifferential are given as follows

- if  $z \in K \setminus \partial K$ ,  $\partial_z P_K(z) = I$ ,
- if  $-z \in K^* \setminus \partial K^*$ ,  $\partial_z P_K(z) = 0$ ,
- if  $z \notin K$  and  $-z \notin K^*$  and,  $\partial_z P_K(z) = 0$ , we get

$$\{\text{eq:sub-proj_n}\} \quad \partial_{z_N} P_K(z) = \frac{1}{1+\mu^2} \begin{bmatrix} 1 \\ \mu z_T \end{bmatrix} \quad (151)$$

and

$$\{\text{eq:sub-proj_t1}\} \quad \partial_{z_T} [P_K(z)]_N = \frac{\mu}{1+\mu^2} \frac{z_T}{\|z_T\|} \quad (152)$$

$$\{\text{eq:sub-proj_2}\} \quad \partial_{z_T} [P_K(z)]_T = \frac{\mu}{(1+\mu^2)} \left[ \mu \frac{z_T}{\|z_T\|} \frac{z_T^\top}{\|z_T\|} + (z_N + \mu\|z_T\|) \left( \frac{I_2}{\|z_T\|} - \frac{z_T z_T^\top}{\|z_T\|^3} \right) \right] \quad (153)$$

that is

$$\{\text{eq:sub-proj_2_bis}\} \quad \partial_{z_T} [P_K(z)]_T = \frac{\mu}{(1+\mu^2)\|z_T\|} \left[ (z_N + \mu\|z_T\|) I_2 + z_N \frac{z_T z_T^\top}{\|z_T\|^2} \right] \quad (154)$$

---

### REDACTION NOTE V.A. 12.1

to be checked carefully

---

## Computation of the subdifferential using the spectral decomposition

In (Hayashi et al, 2005), the computation of the Clarke subdifferential of the projection operator is also done by inspecting the different cases using the spectral decomposition

$$\partial P_K(x) = \begin{cases} I & (\lambda_1 > 0, \lambda_2 > 0) \\ \frac{\lambda_2}{\lambda_1 + \lambda_2} I + Z & (\lambda_1 < 0, \lambda_2 > 0) \\ 0 & (\lambda_1 < 0, \lambda_2 < 0) \\ \text{co}\{I, I + Z\} & (\lambda_1 = 0, \lambda_2 > 0) \\ \text{co}\{0, Z\} & (\lambda_1 < 0, \lambda_2 = 0) \\ \text{co}\{0 \cup I \cup S\} & (\lambda_1 = 0, \lambda_2 = 0) \end{cases} \quad (155) \quad \{\text{eq:Jordan-projection-Jacob}\}$$

where

$$\begin{aligned} Z &= \frac{1}{2} \begin{bmatrix} -y_N & y_T^\top \\ y_T & -y_N y_T y_T^\top \end{bmatrix}, \\ S &= \left\{ \frac{1}{2}(1 + \beta)I + \frac{1}{2} \begin{bmatrix} -\beta & w^\top \\ w & -\beta w w^\top \end{bmatrix} \mid -1 \leq \beta \leq 1, \|w\| = 1 \right\} \end{aligned} \quad (156) \quad \{\text{eq:Jordan-projection-Jacob}\}$$

with  $y = x/\|x_T\|$ . A simple verification shows that the previous computation is an element of the subdifferential.

---

**REDACTION NOTE V.A. 12.2**

*to be checked carefully*

---

### 13 Computation of Generalized Jacobians for Nonsmooth Newton methods

#### 13.1 Computation of components of subgradient of $F_{vi}^{\text{nat}}$

Let us introduce the following notation for an element of the sub-differential

$$\Phi(u, r) = \begin{bmatrix} \rho I & -\rho W \\ \Phi_{ru}(u, r) & \Phi_{rr}(u, r) \end{bmatrix} \in \partial F_{vi}^{\text{nat}}(u, r) \quad (157) \quad \{\text{eq:Phi-natural-typeII}\}$$

where  $\Phi_{xy}(u, r) \in \partial_x [F_{vi}^{\text{nat}}]_y(u, r)$ . Since  $\Phi_{uu}(u, r) = I$ , a reduction of the system is performed in practise and Algorithm 4 is applied or  $z = r$  with

$$\begin{cases} G(z) = [F_{vi}^{\text{nat}}]_r(Wr + q, r) \\ \Phi(z) = \Phi_{rr}(r, Wr + q) + \Phi_{ru}(r, Wr + q)W \end{cases} \quad (158) \quad \{\text{eq:phiphi_ter}\}$$

**REDACTION NOTE V.A. 13.1**

*to be checked carefully. compare with (Hayashi et al, 2005)*

Let us introduce the following notation for an element of the sub-differential with a obvious simplification

{eq:Phi-natural-typeI}

$$\Phi(v, r) = \begin{bmatrix} \rho M & -\rho H & \\ -\rho H^\top & \rho I & 0 \\ 0 & \Phi_{ru}(v, u, r) & \Phi_{rr}(v, u, r) \end{bmatrix} \in \partial F_{vi}^{\text{nat}}(u, r) \quad (159)$$

where  $\Phi_{xy}(v, u, r) \in \partial_x[F_{vi-1}^{\text{nat}}]_y(v, u, r)$ . A possible computation of  $\Phi_{ru}(v, u, r)$  and  $\Phi_{rr}(v, u, r)$  is directly given by (161) and (160). In this case, the variable  $u$  can be also substituted.

For one contact, a possible computation of the remaining parts in  $\Phi(u, r)$  is given by

{eq:Phi-natural-typeII-rr}

$$\Phi_{ru}(u, r) = \begin{cases} 0 & \text{if } r - \rho(u + g(u)) \in K \\ I - \partial_r[P_K(r - \rho(u + g(u)))] & \text{if } r - \rho(u + g(u)) \notin K \end{cases} \quad (160)$$

$$\Phi_{ru}(u, r) = \begin{cases} \rho \left( I + \begin{bmatrix} 0 & 0 & 0 \\ \frac{u_T}{\|u_T\|} & 0 & 0 \end{bmatrix} \right) & \text{if } \begin{cases} r - \rho(u + g(u)) \in K \\ u_T \neq 0 \end{cases} \\ \rho \left( I + \begin{bmatrix} 0 & 0 & 0 \\ s & 0 & 0 \end{bmatrix} \right), s \in \mathbb{R}^2, \|s\| = 1 & \text{if } \begin{cases} r - \rho(u + g(u)) \in K \\ u_T = 0 \end{cases} \\ I + \rho \left( I + \begin{bmatrix} 0 & 0 & 0 \\ \frac{u_T}{\|u_T\|} & 0 & 0 \end{bmatrix} \right) \partial_u[P_K(r - \rho(u + g(u)))] & \text{if } r - \rho(u + g(u)) \notin K \end{cases} \quad (161)$$

{eq:Phi-natural-typeII-ru}

The computation of an element of  $\partial P_K$  is given in Appendix 12.

### 13.2 Alart–Curnier function and its variants

For one contact, a possible computation of the remaining parts in  $\Phi(u, r)$  is given by

{eq:Phi-AC-II-rnun}

$$\Phi_{r_N u_N}(u, r) = \begin{cases} \rho_N & \text{if } r_N - \rho_N u_N > 0 \\ 0 & \text{otherwise} \end{cases} \quad (162)$$

{eq:Phi-AC-II-rnrn}

$$\Phi_{r_N r_N}(u, r) = \begin{cases} 0 & \text{if } r_N - \rho_N u_N > 0 \\ 1 & \text{otherwise} \end{cases} \quad (163)$$

$$\Phi_{r_T u_N}(u, r) = \begin{cases} 0 & \text{if } \|r_T - \rho_T u_T\| \leq \mu \max(0, r_N - \rho_N u_N) \\ 0 & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N \leq 0 \end{cases} \\ \mu \rho_N \frac{r_T - \rho_T u_T}{\|r_T - \rho_T u_T\|} & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N > 0 \end{cases} \end{cases} \quad (164) \quad \{\text{eq:Phi-AC-II-rtun}\}$$

$$\Phi_{r_T u_T}(u, r) = \begin{cases} \rho_T & \text{if } \|r_T - \rho_T u_T\| \leq \mu \max(0, r_N - \rho_N u_N) \\ \mu \rho_T (r_N - \rho_N u_N)_+ \Gamma(r_T - \rho_T u_T) & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N > 0 \end{cases} \end{cases} \quad (165) \quad \{\text{eq:Phi-AC-II-rtut}\}$$

$$\Phi_{r_T r_N}(u, r) = \begin{cases} 0 & \text{if } \|r_T - \rho_T u_T\| \leq \mu \max(0, r_N - \rho_N u_N) \\ 0 & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N \leq 0 \end{cases} \\ -\mu \frac{r_T - \rho_T u_T}{\|r_T - \rho_T u_T\|} & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N > 0 \end{cases} \end{cases} \quad (166) \quad \{\text{eq:Phi-AC-II-rtrn}\}$$

$$\Phi_{r_T r_T}(u, r) = \begin{cases} 0 & \text{if } \|r_T - \rho_T u_T\| \leq \mu \max(0, r_N - \rho_N u_N) \\ I_2 - \mu (r_N - \rho_N u_N)_+ \Gamma(r_T - \rho_T u_T) & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu \max(0, r_N - \rho_N u_N) \\ r_N - \rho_N u_N > 0 \end{cases} \end{cases} \quad (167) \quad \{\text{eq:Phi-AC-II-rtrt}\}$$

with the function  $\Gamma(\cdot)$  defined by

$$\Gamma(x) = \frac{I_{2 \times 2}}{\|x\|} - \frac{xx^\top}{\|x\|^3} \quad (168) \quad \{\text{eq:AC-L12}\}$$

If the variant (58) is chosen, the computation of  $\Phi_{r_\bullet}$  simplify in

$$\Phi_{r_T u_N}(u, r) = 0 \quad (169) \quad \{\text{eq:Phi-CKPS-II-rtun}\}$$

$$\Phi_{r_T u_T}(u, r) = \begin{cases} \rho_T & \text{if } \|r_T - \rho_T u_T\| \leq \mu r_N \\ -\mu \rho_T r_{n,+} \Gamma(r_T - \rho_T u_T) & \text{if } \|r_T - \rho_T u_T\| > \mu r_N \end{cases} \quad (170) \quad \{\text{eq:Phi-CKPS-II-rtut}\}$$

$$\Phi_{r_T r_N}(u, r) = \begin{cases} 0 & \text{if } \|r_T - \rho_T u_T\| \leq \mu r_N \\ 0 & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu r_N \\ r_N \leq 0 \end{cases} \\ -\mu \frac{r_T - \rho_T u_T}{\|r_T - \rho_T u_T\|} & \text{if } \begin{cases} \|r_T - \rho_T u_T\| > \mu r_N \\ r_N > 0 \end{cases} \end{cases} \quad (171) \quad \{\text{eq:Phi-CKPS-II-rtrn}\}$$

$$\Phi_{r_T r_T}(u, r) = \begin{cases} 0 & \text{if } \|r_T - \rho_T u_T\| \leq \mu r_N \\ I_2 - \mu(r_N)_+ \Gamma(r_T - \rho_T u_T) & \text{if } \|r_T - \rho_T u_T\| > \mu r_N \end{cases} \quad (172) \quad \{\text{eq:Phi-CKPS-II-rtrt}\}$$

---

## REDACTION NOTE V.A. 13.2

\* Is there a difference with the computation of Florent in his thesis?  
\*

---

## 13.3 Fischer-Burmeister function

### References

- Acary V, Brogliato B (2008) Numerical methods for nonsmooth dynamical systems. Applications in mechanics and electronics. Lecture Notes in Applied and Computational Mechanics 35. Berlin: Springer. xxi, 525 p. 5, 10
- Acary V, Cadoux F (2013) Recent Advances in Contact Mechanics, Stavroulakis, Georgios E. (Ed.), Lecture Notes in Applied and Computational Mechanics, vol 56, Springer Verlag, chap Applications of an existence result for the Coulomb friction problem 5, 12, 24
- Acary V, Cadoux F, Lemaréchal C, Malick J (2011) A formulation of the linear discrete coulomb friction problem via convex optimization. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik 91(2):155–175, DOI 10.1002/zamm.201000073, URL <http://dx.doi.org/10.1002/zamm.201000073> 4, 5, 10, 12, 24
- Acary V, Brémond M, Koziara T, Pérignon F (2014) FCLIB: a collection of discrete 3D Frictional Contact problems. Technical Report RT-0444, INRIA, URL <https://hal.inria.fr/hal-00945820> 46
- Acary V, Brémond M, Huber O, Pérignon F (2015) An introduction to Siconos. Tech. Rep. TR-0340, second version, INRIA, <http://hal.inria.fr/inria-00162911/en/> 7
- Al-Fahed A, Stavroulakis G, Panagiotopoulos P (1991) Hard and soft fingered robot grippers. the linear complementarity approach. Zeitschrift für Angewandte Mathematik und Mechanik 71:257–265 5
- Alart P (1993) Injectivity and surjectivity criteria for certain mappings of  $\mathbb{R}^n$  into itself; application to contact mechanics. (Critères d’injectivité et de surjectivité pour certaines applications de  $\mathbb{R}^n$  dans lui-même; application à la mécanique du contact.). RAIRO, Modélisation Math Anal Numér 27(2):203–222 30
- Alart P (1995) Méthode de newton généralisée en mécanique du contact. Journal de Mathématiques Pures et Appliquées 30

- Alart P, Curnier A (1991) A mixed formulation for frictional contact problems prone to Newton like solution method. *Computer Methods in Applied Mechanics and Engineering* 92(3):353–375 17, 18
- Anitescu M, Potra F (1997) Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics, Transactions of ASME* 14:231–247 4, 5
- Anitescu M, Tasora A (2010) An iterative approach for cone complementarity problems for nonsmooth dynamics. *Comput Optim Appl* 47(2):207–235, DOI 10.1007/s10589-008-9223-4, URL <http://dx.doi.org/10.1007/s10589-008-9223-4> 5, 44
- Barbosa H, Feijóo R (1985) A numerical algorithm for signorini problem with Coulomb friction. In: (Del Piero and Maceri, 1985) 23
- Barbosa H, Raupp F, Borges C (1997) Numerical experiments with algorithms for bound constrained quadratic programming in Mechanics. *Computers & Structures* 64(1–4):579–594 40
- Bonnans J, Gilbert J, Lemaréchal C, Sagastizábal C (2003) *Numerical Optimization: Theoretical and Practical Aspects*. Springer-Verlag 32
- Bonnefon O, Daviet G (2011) Quartic formulation of Coulomb 3D frictional contact. Technical Report RT-0400, INRIA, URL <https://hal.inria.fr/inria-00553859> 35
- Cadoux F (2009) *Analyse convexe et optimisation pour la dynamique non-régulière*. PhD thesis, Université Joseph Fourier, Grenoble I 24
- Calamai P, More J (1987) Projected gradient methods for linearly constrained problems. *Mathematical Programming* 39(1):93–116, DOI <http://dx.doi.org/10.1007/BF02592073> 40
- Chabrand P, Dubois F, Raous M (1998) Various numerical methods for solving unilateral contact problems with friction. *Mathematical and Computer Modelling* 28(4–8):97 – 108, DOI [http://dx.doi.org/10.1016/S0895-7177\(98\)00111-3](http://dx.doi.org/10.1016/S0895-7177(98)00111-3), URL <http://www.sciencedirect.com/science/article/pii/S0895717798001113>, recent *Advances in Contact Mechanics* 6
- Chaudhary A, Bathe K (1986) A solution method for static and dynamic analysis of three-dimensional contact problems with friction. *Computers & Structures* 24(6):855–873 2
- Chen G, Teboulle M (1993) Convergence analysis of a proximal-like minimization algorithm using bregman functions. *SIAM Journal on Optimization* 3(3):538–543, DOI 10.1137/0803026, URL <http://dx.doi.org/10.1137/0803026>, <http://dx.doi.org/10.1137/0803026> 37
- Christensen P, Pang J (1998) Frictional contact algorithms based on semismooth newton methods. In: Qi MFL (ed) *Reformulation - Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, Kluwer Academic Publishers, Dordrecht, pp 81–116 6, 30
- Christensen P, Klarbring A, Pang J, Stromberg N (1998) Formulation and comparison of algorithms for frictional contact problems. *International Journal for Numerical Methods in Engineering* 42:145–172 6, 17

- Curnier A, Alart P (1988) A generalized Newton method for contact problems with friction. *Journal de Mécanique Théorique et Appliquée* supplément no 1 to 7:67–82 17, 18
- Daviet G, Bertails-Descoubes F, Boissieux L (2011) A Hybrid Iterative Solver for Robustly Capturing Coulomb Friction in Hair Dynamics. *ACM Transactions on Graphics* 30(6):139:1–139:12, DOI 10.1145/2070781.2024173, URL <https://hal.inria.fr/hal-00647497> 53
- De Saxcé G (1992) Une généralisation de l'inégalité de Fenchel et ses applications aux lois constitutives. *Comptes Rendus de l'Académie des Sciences t 314,série II*:125–129 10, 21
- De Saxcé G, Feng ZQ (1991) New inequality and functional for contact with friction : The implicit standard material approach. *Mech Struct & Mach* 19(3):301–325 10, 24
- De Saxcé G, Feng ZQ (1998) The bipotential method: A constructive approach to design the complete contact law with friction and improved numerical algorithms. *Mathematical and Computer Modelling* 28(4):225–245 24
- Del Piero G, Maceri F (eds) (1983) *Unilateral Problems in Structural Analysis, CISM courses and lectures, vol 288*, Springer, Ravello, Italy 2
- Del Piero G, Maceri F (eds) (1985) *Unilateral Problems in Structural Analysis – II., CISM courses and lectures, vol 304*, Springer, Prescudin, Italy 2, 80
- Dolan E, Moré J (2002) Benchmarking optimization software with performance profiles. *Mathematical Programming* 91(2):201–213 45
- Dostál Z (1997) Box constrained quadratic programming with proportioning and projections. *SIAM J Optim* 7(3):871–887, DOI 10.1137/S1052623494266250, URL <http://dx.doi.org/10.1137/S1052623494266250> 42, 43
- Dostál Z (2016) *Scalable Algorithms for Contact Problems*. Springer New York, New York, NY, DOI 10.1007/978-1-4939-6834-3\_8, URL [https://doi.org/10.1007/978-1-4939-6834-3\\_8](https://doi.org/10.1007/978-1-4939-6834-3_8) 40, 43
- Dostál Z, Kozubek T (2012) An optimal algorithm and superrelaxation for minimization of a quadratic function subject to separable convex constraints with applications. *Math Program* 135(1-2, Ser. A):195–220, DOI 10.1007/s10107-011-0454-2, URL <http://dx.doi.org/10.1007/s10107-011-0454-2> 40, 43
- Dostál Z, Kučera R (2010) An optimal algorithm for minimization of quadratic functions with bounded spectrum subject to separable convex inequality and linear equality constraints. *SIAM Journal on Optimization* 20(6):2913–2938, DOI 10.1137/090751414, URL <http://dx.doi.org/10.1137/090751414>, <http://dx.doi.org/10.1137/090751414> 43
- Dostál Z, Schöberl J (2005) Minimizing quadratic functions subject to bound constraints with the rate of convergence and finite termination. *Comput Optim Appl* 30(1):23–43, DOI 10.1007/s10589-005-4557-7, URL <http://dx.doi.org/10.1007/s10589-005-4557-7> 43
- Dostál Z, Haslinger J, Kučera R (2002) Implementation of the fixed point method in contact problems with Coulomb friction based on a dual splitting type technique.



- J Comput Appl Math 140(1-2):245–256, DOI 10.1016/S0377-0427(01)00405-8, URL [http://dx.doi.org/10.1016/S0377-0427\(01\)00405-8](http://dx.doi.org/10.1016/S0377-0427(01)00405-8) 42
- Facchinei F, Pang JS (2003) Finite-dimensional Variational Inequalities and Complementarity Problems, Springer Series in Operations Research, vol I & II. Springer Verlag NY. Inc. 3, 13, 14, 15, 28
- Ferris MC, Lucidi S (1994) Nonmonotone stabilization methods for nonlinear equations. *Journal of Optimization Theory and Applications* 81(1):53–71, DOI 10.1007/BF02190313, URL <https://doi.org/10.1007/BF02190313> 32
- Fletcher R (1987) *Practical Methods of Optimization*. Chichester: John Wiley & Sons, Inc. 40
- Fong DCL, Saunders M (2011) Lsmr: An iterative algorithm for sparse least-squares problems. *SIAM Journal on Scientific Computing* 33(5):2950–2971, DOI 10.1137/10079687X, URL <https://doi.org/10.1137/10079687X>, <https://doi.org/10.1137/10079687X> 47
- Fortin M, Glowinski R (1983) *Augmented Lagrangian methods, Studies in Mathematics and its Applications*, vol 15. North-Holland Publishing Co., Amsterdam, applications to the numerical solution of boundary value problems, Translated from the French by B. Hunt and D. C. Spicer 25
- Fukushima M, Luo Z, Tseng P (2001) Smoothing functions for second-order-cone complementarity problems. *SIAM Journal on Optimization* 12(2):436–460, DOI 10.1137/S1052623400380365, URL <http://link.aip.org/link/?SJE/12/436/1> 19, 20
- Glowinski R, JL L, Trémolières R (1976) *Approximations des Inéquations Variationnelles*. Dunod, Paris 25
- Grippo L, Lampariello F, Lucidi S (1986) A nonmonotone line search technique for newton’s method. *SIAM Journal on Numerical Analysis* 23(4):707–716, DOI 10.1137/0723046, URL <https://doi.org/10.1137/0723046>, <https://doi.org/10.1137/0723046> 32
- Han D (2008) A generalized proximal-point-based prediction correction method for variational inequality problems. *Journal of Computational and Applied Mathematics* 221(1):183 – 193, DOI <http://dx.doi.org/10.1016/j.cam.2007.10.063>, URL <http://www.sciencedirect.com/science/article/pii/S037704270700578X> 38
- Han D, Lo HK (2002) Two new self-adaptive projection methods for variational inequality problems. *Computers & Mathematics with Applications* 43(12):1529 – 1537, DOI [http://dx.doi.org/10.1016/S0898-1221\(02\)00116-5](http://dx.doi.org/10.1016/S0898-1221(02)00116-5), URL <http://www.sciencedirect.com/science/article/pii/S0898122102001165> 26
- Harker P, Pang JS (1990) Finite-dimensional variational inequality and complementarity problems: a survey of theory, algorithms and applications. *Mathematical Programming* 48:160–220 13
- Haslinger J (1983) Approximation of the signorini problem with friction, obeying the coulomb law. *Mathematical Methods in the Applied Sciences* 5:422–437 2, 23, 42

- Haslinger J (1984) Least square method for solving contact problems with friction obeying coulomb's law. *Applications of mathematics* 29(3):212–224, URL <http://dml.cz/dmlcz/104086> 2, 23, 42
- Haslinger J, Panagiotopoulos PD (1984) The reciprocal variational approach to the signorini problem with friction. approximation results. *Proceedings of the Royal Society of Edinburgh: Section A Mathematics* 98:365–383, DOI 10.1017/S0308210500013536, URL [http://journals.cambridge.org/article\\_S0308210500013536](http://journals.cambridge.org/article_S0308210500013536) 2, 23
- Haslinger J, Hlaváček I, Nečas J (1996) Numerical methods for unilateral problems in solid mechanics. In: Ciarlet P, Lions J (eds) *Handbook of Numerical Analysis*, North-Holland, Amsterdam, 1996, vol IV, Part 2, pp 313–485 23, 42
- Haslinger J, Dostál Z, Kučera R (2002) On a splitting type algorithm for the numerical realization of contact problems with Coulomb friction. *Comput Methods Appl Mech Engrg* 191(21-22):2261–2281, DOI 10.1016/S0045-7825(01)00378-4, URL [http://dx.doi.org/10.1016/S0045-7825\(01\)00378-4](http://dx.doi.org/10.1016/S0045-7825(01)00378-4) 42
- Haslinger J, Kučera R, D Z (2004) An algorithm for the numerical realization of 3D contact problems with Coulomb friction. In: *Proceedings of the 10th International Congress on Computational and Applied Mathematics (ICCAM-2002)*, vol 164/165, pp 387–408, DOI 10.1016/j.cam.2003.06.002, URL <http://dx.doi.org/10.1016/j.cam.2003.06.002> 5, 42
- Haslinger J, Kučera R, Vlach O, Baniotopoulos C (2012) Approximation and numerical realization of 3d quasistatic contact problems with coulomb friction. *Mathematics and Computers in Simulation* 82(10):1936 – 1951, DOI <http://dx.doi.org/10.1016/j.matcom.2011.01.004>, URL <http://www.sciencedirect.com/science/article/pii/S0378475411000310>, the Fourth IMACS Conference : "Mathematical Modelling and Computational Methods in Applied Sciences and Engineering" Devoted to Owe Axelsson in occasion of his 75th birthday 42
- Hayashi S, Yamashita N, Fukushima M (2005) A combined smoothing and regularization method for monotone second-order cone complementarity problems. *SIAM J on Optimization* 15(2):593–615, DOI <http://dx.doi.org/10.1137/S1052623403421516> 20, 76, 77
- He B, Liao L (2002) Improvements of some projection methods for monotone nonlinear variational inequalities. *Journal of Optimization Theory and Applications* 112(1):111–128, DOI 10.1023/A:1013096613105, URL <http://dx.doi.org/10.1023/A:1013096613105> 26, 38
- Hestenes M (1969) Multiplier and gradient methods. *Journal of Optimization Theory and Applications* 4:303–320 18
- Heyn T (2013) On the modeling, simulation, and visualization of many-body dynamics problems with friction and contact. PhD thesis, University of Wisconsin–Madison 6
- Heyn T, Anitescu M, Tasora A, Negrut D (2013) Using Krylov subspace and spectral methods for solving complementarity problems in many-body contact dynamics simulation. *Internat J Numer Methods Engrg* 95(7):541–561, DOI 10.1002/nme.4513, URL <http://dx.doi.org/10.1002/nme.4513> 5, 6

- Hüeber S, Stadler G, Wohlmuth BI (2008) A primal-dual active set algorithm for three-dimensional contact problems with coulomb friction. *SIAM J Sci Comput* 30(2):572–596, DOI 10.1137/060671061, URL <http://dx.doi.org/10.1137/060671061> 18, 19, 30
- Jean M, Moreau J (1987) Dynamics in the presence of unilateral contacts and dry friction: a numerical approach. In: Del Pietro G, Maceri F (eds) *Unilateral problems in structural analysis. II*, CISM 304, Springer Verlag, pp 151–196 2, 17
- Joli P, Feng ZQ (2008) Uzawa and newton algorithms to solve frictional contact problems within the bi-potential framework. *International Journal for Numerical Methods in Engineering* 73(3):317–330, DOI 10.1002/nme.2073, URL <http://dx.doi.org/10.1002/nme.2073> 29
- Jourdan F, Alart P, Jean M (1998) A Gauss Seidel like algorithm to solve frictional contact problems. *Computer Methods in Applied Mechanics and Engineering* 155(1):31–47 30, 35
- Katona MG (1983) A simple contact–friction interface element with applications to buried culverts. *International Journal for Numerical and Analytical Methods in Geomechanics* 7(3):371–384, DOI 10.1002/nag.1610070308, URL <http://dx.doi.org/10.1002/nag.1610070308> 2
- Khobotov E (1987) Modification of the extra-gradient method for solving variational inequalities and certain optimization problems. *{USSR} Computational Mathematics and Mathematical Physics* 27(5):120 – 127, DOI [http://dx.doi.org/10.1016/0041-5553\(87\)90058-9](http://dx.doi.org/10.1016/0041-5553(87)90058-9), URL <http://www.sciencedirect.com/science/article/pii/0041555387900589> 26
- Kikuchi N, Oden JT (1988) Contact problems in elasticity: a study of variational inequalities and finite element methods, *SIAM Studies in Applied Mathematics*, vol 8. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, DOI 10.1137/1.9781611970845, URL <http://dx.doi.org/10.1137/1.9781611970845> 6
- Klarbring A (1986) A mathematical programming approach to three-dimensional contact problem with friction. *Compt Methods Appl Math Engrg* 58:175–200 5
- Klarbring A, Pang JS (1998) Existence of solutions to discrete semicoercive frictional contact problems. *SIAM Journal on Optimization* 8(2):414–442 4, 12
- Kleinert J, Simeon B, Obermayr M (2014) An inexact interior point method for the large-scale simulation of granular material. *Computer Methods in Applied Mechanics and Engineering* 278(0):567 – 598, DOI <http://dx.doi.org/10.1016/j.cma.2014.06.009>, URL <http://www.sciencedirect.com/science/article/pii/S0045782514001959> 6
- Korpelevich G (1976) The extragradient method for finding saddle points and other problems. *Matecon* 12(747–756) 25, 26
- Koziara T, Bićanić N (2008) Semismooth newton method for frictional contact between pseudo-rigid bodies. *Computer Methods in Applied Mechanics and Engineering* 197(33–40):2763 – 2777, DOI <http://dx.doi.org/10.1016/j.cma.2008.01.006>, URL <http://www.sciencedirect.com/science/article/pii/S0045782508000194> 18, 19, 30, 32

- Krabbenhoft K, Lyamin A, Huang J, da Silva MV (2012) Granular contact dynamics using mathematical programming methods. *Computers and Geotechnics* 43:165 – 176, DOI <http://dx.doi.org/10.1016/j.compgeo.2012.02.006>, URL <http://www.sciencedirect.com/science/article/pii/S0266352X12000262> 5, 6
- Kučera R (2007) Minimizing quadratic functions with separable quadratic constraints. *Optim Methods Softw* 22(3):453–467, DOI 10.1080/10556780600609246, URL <http://dx.doi.org/10.1080/10556780600609246> 42
- Kučera R (2008) Convergence rate of an optimization algorithm for minimizing quadratic functions with separable convex constraints. *SIAM J Optim* 19(2):846–862, DOI 10.1137/060670456, URL <http://dx.doi.org/10.1137/060670456> 42
- Laursen T (2003) *Computational Contact and Impact Mechanics – Fundamentals of Modeling Interfacial Phenomena in Nonlinear Finite Element Analysis*. Springer Verlag, 1st ed. 2002. Corr. 2nd printing, 5, 11
- Leung A, Guoqing C, Wanji C (1998) Smoothing Newton method for solving two– and three–dimensional frictional contact problems. *International Journal for Numerical Methods in Engineering* 41:1001–1027 18
- Mijar A, Arora J (2000a) Review of formulations for elastostatic frictional contact problems. *Structural and Multidisciplinary Optimization* 20(3):167–189, DOI 10.1007/s001580050147, URL <http://dx.doi.org/10.1007/s001580050147> 6
- Mijar A, Arora J (2000b) Study of variational inequality and equality formulations for elastostatic frictional contact problems. *Archives of Computational Methods in Engineering* 7(4):387–449, DOI 10.1007/BF02736213, URL <http://dx.doi.org/10.1007/BF02736213> 6
- Mijar A, Arora J (2004a) An augmented lagrangian optimization method for contact analysis problems, 1: formulation and algorithm. *Structural and Multidisciplinary Optimization* 28(2-3):99–112, DOI 10.1007/s00158-004-0423-y, URL <http://dx.doi.org/10.1007/s00158-004-0423-y> 6
- Mijar A, Arora J (2004b) An augmented lagrangian optimization method for contact analysis problems, 2: numerical evaluation. *Structural and Multidisciplinary Optimization* 28(2-3):113–126, DOI 10.1007/s00158-004-0424-x, URL <http://dx.doi.org/10.1007/s00158-004-0424-x> 6
- Mitsopoulou E, Doudoumis I (1987) A contribution to the analysis of unilateral contact problems with friction. *Solid Mechanics Archives* 12(3):165–186 35
- Mitsopoulou E, Doudoumis I (1988) On the solution of the unilateral contact frictional problem for general static loading conditions. *Computers & Structures* 30(5):1111–1126 2, 35
- Miyamura T, Kanno Y, Ohsaki M (2010) Combined interior-point method and semismooth newton method for frictionless contact problems. *International Journal for Numerical Methods in Engineering* 81(6):701–727, DOI 10.1002/nme.2707, URL <http://dx.doi.org/10.1002/nme.2707> 6

- Morales JL, Nocedal J, Smelyanskiy M (2008) An algorithm for the fast solution of symmetric linear complementarity problems. *Numerische Mathematik* 111(2):251–266, DOI 10.1007/s00211-008-0183-5, URL <http://dx.doi.org/10.1007/s00211-008-0183-5> 6
- Moré J, Toraldo G (1991) On the solution of large quadratic convex programming problems with bound constraints. *SIAM J Optimization* 1(1):93–113 40, 42
- Moré JJ, Toraldo G (1989) Algorithms for bound constrained quadratic programming problems. *Numerische Mathematik* 55(4):377–400, DOI 10.1007/BF01396045, URL <https://doi.org/10.1007/BF01396045> 40
- Moreau J (1988) Unilateral contact and dry friction in finite freedom dynamics. In: Moreau J, PD P (eds) *Nonsmooth Mechanics and Applications*, no. 302 in CISM, Courses and lectures, CISM 302, Springer Verlag, Wien- New York, pp 1–82, formulation mathématiques tire du livre *Contacts mechanics* 8
- Mylapilli H, Jain A (2017) Complementarity techniques for minimal coordinates contact dynamics. *ASME Journal of Computational and Nonlinear Dynamics* 12(2) 6
- Nečas J, Jarušek J, Haslinger J (1980) On the solution of the variational inequality to the Signorini problem with small friction. *Bollettino UMI* 5(17-B):796–811 2, 23, 42
- Nocedal J, Wright S (1999) *Numerical Optimization*. Springer Verlag 40
- Panagiotopoulos P (1975) A nonlinear programming approach to the unilateral contact-, and friction-boundary value problem in the theory of elasticity. *Ingenieur-Archiv* 44(6):421–432, DOI 10.1007/BF00534623, URL <http://dx.doi.org/10.1007/BF00534623> 2, 23
- Pang J, Trinkle J (1996) Complementarity formulations and existence of solutions of dynamic multi-rigid-body contact problems with Coulomb friction. *Mathematical Programming* 73:199–226 4, 5
- Park J, Kwak B (1994) Three dimensional frictional contact analysis using the homotopy method. *Journal of Applied Mechanics, Transactions of ASME* 61:703–709 18
- Qi L, Sun J (1993) A nonsmooth version of Newton’s method. *Mathematical Programming* 58:353–367 30
- Qi L, Sun D, Ulbrich M (eds) (2018) *Semismooth and Smoothing Newton Methods*. Springer Verlag 30
- Raous M, Chabrand P, Lebon F (1988) Numerical methods for frictional contact problems and applications. *J Méc Théor Appl* 7(1):111–18 6
- Rockafellar R (1974) Augmented Lagrange multiplier functions and duality in non-convex programming. *SIAM Journal on Control* 12:268–285 18
- Rockafellar R (1993) Lagrange multipliers and optimality. *SIAM Review* 35(2):183–238 18
- Rockafellar R, Wets R (1997) *Variational Analysis*, vol 317. Springer Verlag, New York 74
- Saxcé GD, Feng ZQ (1998) The bipotential method: A constructive approach to design the complete contact law with friction and improved numerical algorithms. *Mathematical and Computer Modelling* 28(4–8):225

- 245, DOI [http://dx.doi.org/10.1016/S0895-7177\(98\)00119-8](http://dx.doi.org/10.1016/S0895-7177(98)00119-8), URL <http://www.sciencedirect.com/science/article/pii/S0895717798001198>, recent *Advances in Contact Mechanics* 10
- Sibony M (1970) Méthodes itératives pour les équations et inéquations aux dérivées partielles non linéaires de type monotone. *Calcolo* 7:65–183 16
- Simo J, Laursen T (1992) An augmented Lagrangian treatment of contact problems involving friction. *Computers & Structures* 42(1):97–116 18, 25
- Solodov M, Tseng P (1996) Modified projection-type methods for monotone variational inequalities. *SIAM Journal on Control and Optimization* 34(5):1814–1830, URL [citeseer.ist.psu.edu/article/solodov95modified.html](http://citeseer.ist.psu.edu/article/solodov95modified.html) 26
- Stadler G (2004) Semismooth newton and augmented lagrangian methods for a simplified friction problem. *SIAM Journal on Optimization* 15(1):39–62, DOI 10.1137/S1052623403420833, URL <https://doi.org/10.1137/S1052623403420833>, <https://doi.org/10.1137/S1052623403420833> 18
- Stewart D, Trinkle J (1996) An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and Coulomb friction. *International Journal for Numerical Methods in Engineering* 39(15), reference tirée du site WILEY 4, 5
- Sun D, Sun J (2005) Strong semismoothness of the fischer-burmeister sdc and soc complementarity functions. *Mathematical Programming* 103(3):575–581, DOI 10.1007/s10107-005-0577-4, URL <https://doi.org/10.1007/s10107-005-0577-4> 30
- Tasora A, Anitescu M (2009) A fast NCP solver for large rigid-body problems with contacts, friction, and joints. In: *Multibody dynamics, Comput. Methods Appl. Sci.*, vol 12, Springer, Berlin, pp 45–55 5, 44
- Tasora A, Anitescu M (2011) A matrix-free cone complementarity approach for solving large-scale, nonsmooth, rigid body dynamics. *Comput Methods Appl Mech Engrg* 200(5-8):439–453, DOI 10.1016/j.cma.2010.06.030, URL <http://dx.doi.org/10.1016/j.cma.2010.06.030> 5, 44
- Tasora A, Anitescu M (2013) A complementarity-based rolling friction model for rigid contacts. *Meccanica* 48(7):1643–1659, DOI 10.1007/s11012-013-9694-y, URL <http://dx.doi.org/10.1007/s11012-013-9694-y> 5
- Temizer I, Abdalla M, Gürdal Z (2014) An interior point method for isogeometric contact. *Computer Methods in Applied Mechanics and Engineering* 276(0):589 – 611, DOI <http://dx.doi.org/10.1016/j.cma.2014.03.018>, URL <http://www.sciencedirect.com/science/article/pii/S0045782514001042> 6
- Tzaferopoulos M (1993) On an efficient new numerical method for the frictional contact problem of structures with convex energy density. *Computers & Structures* 48(1):87–106 23, 40
- Wang X, He B, Liao LZ (2010) Steplengths in the extragradient type methods. *Journal of Computational and Applied Mathematics* 233(11):2925 – 2939, DOI <http://dx.doi.org/10.1016/j.cam.2009.11.037>,

URL <http://www.sciencedirect.com/science/article/pii/S0377042709007845> 27

Wriggers P (2006) Computational Contact Mechanics, 2nd edn. Springer Verlag, originally published by John Wiley & Sons Ltd., 2002 5, 11

Xuewen L, Soh AK, Wanji C (2000) A new nonsmooth model for three-dimensional frictional contact problems. Computational Mechanics 26:528–535 18