Programs and Algorithms of Numerical Mathematics 18
J. Chleboun, P. Kůs, P. Přikryl, K. Segeth, J. Šístek, T. Vejchodský (Eds.)
Institute of Mathematics CAS, Prague 2017

IMPLEMENTATION OF FULL LINEARIZATION IN SEMISMOOTH NEWTON METHOD FOR 2D CONTACT PROBLEM

Ladislav Foltyn, Oldřich Vlach

Department of Applied Mathematics, FEECS
VŠB – Technical University of Ostrava
17. listopadu 15, 708 33 Ostrava, Czech Republic
ladislav.foltyn@vsb.cz, oldrich.vlach2@vsb.cz

Abstract: To solve the contact problems by using a semismooth Newton method, we shall linearize stiffness and mass matrices as well as contact conditions. The latter are prescribed by means of mortar formulation. In this paper we describe implementation details.

Keywords: semismooth Newton method, contact problem, active set strategy, mortar formulation

MSC: 74M15, 65N30

1. Introduction

After a finite element discretization, a mathematical model of the contact problem is a problem of quadratic programming with an equality and an inequality constraints in a special form, see for example [1]. If a system matrix isn't positive definite, we cannot use this approach. Positive definiteness of the system matrix may be impaired in a case of material nonlinearity and computing in increments. One of the possible solutions is the solution of a nonlinear equation system instead of a minimization problem. The inequality constraints can be also written as the equality constraints unfortunately compensated by the price of a nonsmooth function appearance in the formulation. This reformulated problem can be solved by the semismooth Newton method. Therefore, it is necessary to linearize the stiffness matrix and the contact matrices which is in detail described in [3]. The mortar contact topic is described in [4]. Our goal was to implement solution for the linearized contact problem using the semismooth Newton method in the framework of the MatSol library.

DOI: 10.21136/panm.2016.04

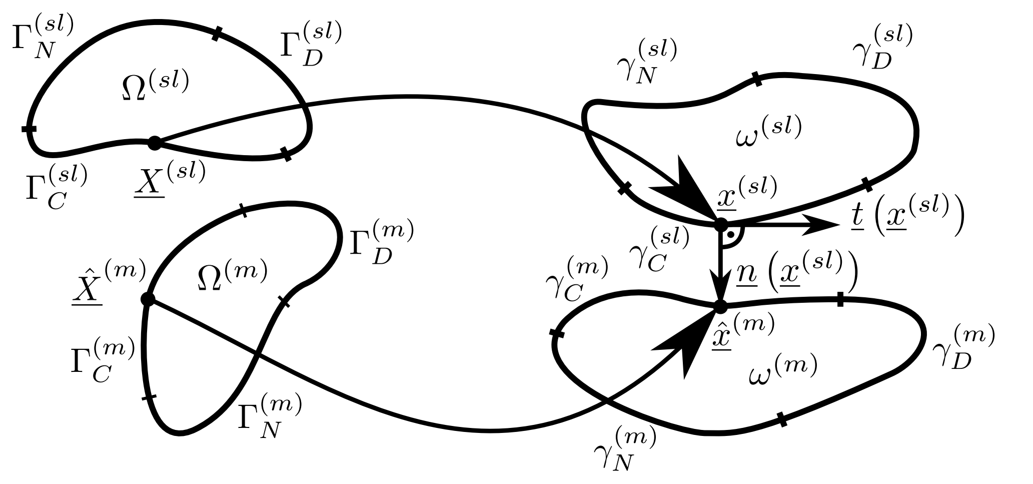

Figure 1: Reference and actual configuration.

2. Formulation of a contact problem

Consider a 2D contact problem with definite deformations of two elastic bodies (see Figure 1). Both bodies are represented as open sets $\Omega^{(sl)} \subset \mathbb{R}^2$ (a slave), $\Omega^{(m)} \subset \mathbb{R}^2$ (a master). Their boundary $\partial\Omega^{(sl)}, \partial\Omega^{(m)}$ can be divided into the following parts:

- the part with prescribed Neumann condition $\Gamma_N^{(i)}$,
- the part with prescribed Dirichlet condition $\Gamma_D^{(i)}$,
- and the part with contact boundary $\Gamma_C^{(i)}$,

where $i \in \{sl,\, m\}$. We will assume that all boundary parts $\Gamma_N^{(i)}$, $\Gamma_D^{(i)}$ and $\Gamma_C^{(i)}$ are mutually disjoint. We also distinguish two types of configuration in the contact problem, an actual configuration ($\omega^{(i)}$, $\gamma_D^{(i)}$, $\gamma_N^{(i)}$, $\gamma_C^{(i)}$, $\underline{x}^{(i)}$) and a reference configuration ($\Omega^{(i)}$, $\Gamma_D^{(i)}$, $\Gamma_N^{(i)}$, $\Gamma_C^{(i)}$, $\underline{X}^{(i)}$). Actual configuration of both bodies is described by a displacement vector

$$\underline{u}^{(i)} = \underline{X}^{(i)} - \underline{x}^{(i)}. \tag{1}$$

On contact boundary a gap function is introduced to define gap between the slave and the master body

$$g(\underline{X}^{(sl)}) = -\underline{n}\left(\underline{x}^{(sl)}(\underline{X}^{(sl)})\right) \cdot \left[\underline{x}^{(sl)}(\underline{X}^{(sl)}) - \hat{\underline{x}}^{(m)}(\hat{\underline{X}}^{(m)})\right], \tag{2}$$

where $\underline{n} = \underline{n}^{(sl)}$ is a normal vector of the slave surface $\gamma_C^{(sl)}$ in the actual configuration, $\hat{\underline{x}}^{(m)}$ is a projection of the slave node $\underline{x}^{(sl)}$ to the master surface $\gamma_C^{(m)}$ in the direction of the normal vector and $\underline{X}$ is a corresponding point to $\underline{x}$ in the reference configuration. Using the gap function we are able to find a corresponding node on the master surface in the reference configuration to the slave node in the same configuration.

After a discretization process (for details see [5]), we can reformulate the contact problem with the KKT conditions to an algebraic form

$$\begin{aligned} \underline{\underline{\mathrm{K}}}\,\underline{\mathrm{d}} + \underline{\underline{\mathrm{D}}}^\top \underline{\mathrm{z}} - \underline{\underline{\mathrm{M}}}^\top \underline{\mathrm{z}} - \underline{\mathrm{f}} &= \underline{0} \\ \tilde{g}_j &\geq 0 \;\; \text{(nonnegative gap between bodies)} \\ (z_n)_j &\geq 0 \\ (z_n)_j\, \tilde{g}_j &= 0 \;\text{(the complementarity condition)} \\ (z_t)_j &= 0 \;\text{(no friction)}, \end{aligned} \tag{3}$$

where $\underline{\underline{\mathrm{K}}}$ is a stiffness matrix, $\underline{\underline{\mathrm{D}}}$ and $\underline{\underline{\mathrm{M}}}$ are mortar contact matrices, $\underline{\mathrm{d}}$ is a displacement vector, $\underline{\mathrm{z}}$ is a vector of multiplicators and $\underline{\mathrm{f}}$ is a vector of volume and boundary forces.

If we want to calculate mortar matrices, we have to divide the contact elements to smaller parts called the contact segments (see Figure 2). It is necessary for the numerical integration of the master shape functions because one segment is connected with exactly one master and one slave element and the shape function formulas have nonchanging prescriptions. A segmentation process is based on the node projection from the slave (the master) surface to the other surface along the normal of the slave surface.

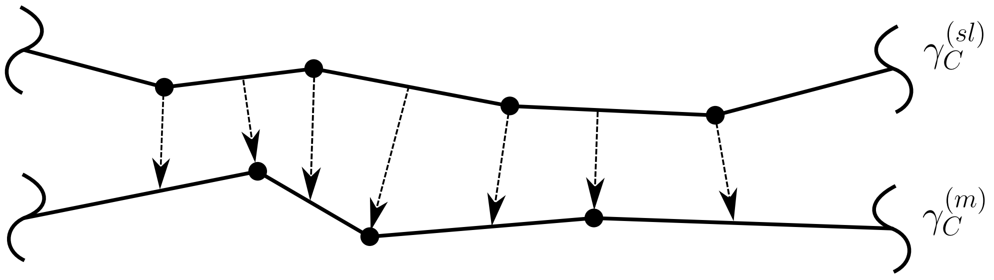

Figure 2: Surface segmentation.

3. Nonsmooth formulation for discrete problem

The semismooth Newton method solves

$$\mathbf{F}^{\circ}(\underline{\mathrm{x}}) = \underline{0}\ , \tag{4}$$

where $\mathbf{F}^{\circ}$ is a nonsmooth function, so it is necessary to transform all inequalities in (3) to equality. Moreover, the semismooth Newton method uses iterative prescription which is similar to continuous one

$$\mathbf{F}^{\circ}(\mathbf{x}^k)\Delta\mathbf{x}^k = -\mathbf{F}(\mathbf{x}^k), \quad \mathbf{x}^{k+1} = \mathbf{x}^k + \Delta\mathbf{x}^k, \tag{5}$$

but there is the nonsmooth function $\mathbf{F}^\circ$ and we reach only a superlinear convergence. For details see [2].

With the use of the active set strategy idea, it is possible to reformulate inequalities in (3) to equalities but there is no apriori information which nodes of the slave contact surface belong to an active set $\mathcal{A}$ or to an inactive set $\mathcal{I}$. It is known only if the solution is available. Luckily, the inequalities can be formulated as finding a zero level of the nonsmooth function. Therefore we introduce a so-called complementarity function C_j

$$C_j(\underline{z}_j, \underline{d}) = (\underline{z}_n)_j - \max\left(0, (\underline{z}_n)_j - c_n \tilde{g}_j\right) \quad \forall j \in \mathcal{S}, \tag{6}$$

where $\mathcal{S}$ denotes a set of all contact nodes of the slave surface. The zero level of the complementarity function C_j is equal with the KKT conditions

$$\left.\begin{array}{r} \tilde{g}_j \geq 0 \\ (z_n)_j \geq 0 \quad \forall j \in \mathcal{S} \\ (z_n)_j \tilde{g}_j = 0 \end{array}\right\} \Leftrightarrow C_j(\underline{z}_j, \underline{d}) = 0 \quad \forall j \in \mathcal{S} \ . \tag{7}$$

The function C_j is continuous but nonsmooth, we cannot determine a derivative at $(z_n)_j - c_n \tilde{g}_j = 0$. With use of the complementarity function C_j, we are able to reformulate (3) to

$$\begin{aligned} \underline{r} = \underline{\underline{K}}\,\underline{d} + \underline{\underline{D}}^\top \underline{z} - \underline{\underline{M}}^\top \underline{z} - \underline{f} = \underline{0} \ , \\ C_j\left(\underline{z}_j, \underline{d}\right) = 0 \quad \forall j \in \mathcal{S} \ , \\ \left(z_t\right)_j = 0 \quad \forall j \in \mathcal{S} \ . \end{aligned} \tag{8}$$

If we want to be able to linearize the function C_j, we have to define a generalized derivative of the $\max(a, x)$ function

$$f(x) = \max(a, x) \longrightarrow \Delta f(x) = \begin{cases} 0, & \text{for } x \leq a \\ 1, & \text{for } x > a \end{cases} . \tag{9}$$

Sets which are described bellow are used to determine which nodes of the slave contact surface belong to the active $\mathcal{A}_k$ or to the inactive $\mathcal{I}_k$ set in each step of the algorithm

$$\mathcal{I}_k = \left\{ j \in \mathcal{S} |\ \left(\underline{n}_j^k \cdot \underline{z}_j^k - c_n \tilde{g}_j^k \right) \leq 0 \right\} \ , \tag{10}$$

$$\mathcal{A}_k = \left\{ j \in \mathcal{S} |\ \left(\underline{n}_j^k \cdot \underline{z}_j^k - c_n \tilde{g}_j^k \right) > 0 \right\} \ . \tag{11}$$

Using these sets also allows us to use a block matrix notation which you can see below.

Complete linearization of the problem yields the system of linear equations

$$\begin{bmatrix} \underline{\underline{K}}_{\mathcal{N}\mathcal{N}} & \underline{\underline{K}}_{\mathcal{N}\mathcal{M}} & \underline{\underline{K}}_{\mathcal{N}\mathcal{I}} & \underline{\underline{K}}_{\mathcal{N}\mathcal{A}} & \underline{\underline{0}} & \underline{\underline{0}} \\ \underline{\underline{K}}_{\mathcal{M}\mathcal{N}} & \underline{\underline{\tilde{K}}}_{\mathcal{M}\mathcal{M}} & \underline{\underline{\tilde{K}}}_{\mathcal{M}\mathcal{I}} & \underline{\underline{\tilde{K}}}_{\mathcal{M}\mathcal{A}} & -\underline{\underline{M}}_{\mathcal{I}}^{\top} & -\underline{\underline{M}}_{\mathcal{A}}^{\top} \\ \underline{\underline{K}}_{\mathcal{I}\mathcal{N}} & \underline{\underline{\tilde{K}}}_{\mathcal{I}\mathcal{M}} & \underline{\underline{\tilde{K}}}_{\mathcal{I}\mathcal{I}} & \underline{\underline{\tilde{K}}}_{\mathcal{I}\mathcal{A}} & \underline{\underline{D}}_{\mathcal{I}} & \underline{\underline{0}} \\ \underline{\underline{K}}_{\mathcal{A}\mathcal{N}} & \underline{\underline{\tilde{K}}}_{\mathcal{A}\mathcal{M}} & \underline{\underline{\tilde{K}}}_{\mathcal{A}\mathcal{I}} & \underline{\underline{\tilde{K}}}_{\mathcal{A}\mathcal{A}} & \underline{\underline{0}} & \underline{\underline{D}}_{\mathcal{A}} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{I}}_{\mathcal{I}} & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{\tilde{M}}}_{\mathcal{A}} & \underline{\underline{\tilde{S}}}_{\mathcal{A}\mathcal{I}} & \underline{\underline{\tilde{S}}}_{\mathcal{A}\mathcal{A}} & \underline{\underline{0}} & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{\tilde{F}}}_{\mathcal{A}\mathcal{I}} & \underline{\underline{\tilde{F}}}_{\mathcal{A}\mathcal{A}} & \underline{\underline{0}} & \underline{\underline{T}}_{\mathcal{A}} \end{bmatrix} \begin{bmatrix} \Delta\underline{d}_{\mathcal{N}} \\ \Delta\underline{d}_{\mathcal{M}} \\ \Delta\underline{d}_{\mathcal{I}} \\ \Delta\underline{d}_{\mathcal{A}} \\ \underline{z}_{\mathcal{I}} \\ \underline{z}_{\mathcal{A}} \end{bmatrix} = - \begin{bmatrix} \underline{r}_{\mathcal{N}} \\ \underline{r}_{\mathcal{M}} \\ \underline{r}_{\mathcal{I}} \\ \underline{r}_{\mathcal{A}} \\ \underline{0} \\ \underline{\tilde{g}}_{\mathcal{A}} \\ \underline{0} \end{bmatrix} , \qquad (12)$$

where elements with a tilde above are affected by a linearization process, $\mathcal{N}$ denotes all nodes which aren't on the contact surfaces, $\mathcal{M}$ denotes all nodes of the master contact surface, $\mathcal{A}$ denotes nodes of the slave contact surface which are actually in contact (active set) and $\mathcal{I}$ denotes all nodes of the slave contact surface which aren't actually in contact (inactive set).

We can also eliminate all multiplicators $\underline{z}$ from the system above by using

$$\underline{z}_{\mathcal{I}} = \underline{0}\,, \qquad (13)$$

$$\underline{z}_{\mathcal{A}} = \underline{\underline{D}}_{\mathcal{A}}^{-1}\left(-\underline{\underline{K}}_{\mathcal{A}\mathcal{N}}\,\Delta\underline{d}_{\mathcal{N}} - \underline{\underline{K}}_{\mathcal{A}\mathcal{M}}\,\Delta\underline{d}_{\mathcal{M}} - \underline{\underline{K}}_{\mathcal{A}\mathcal{I}}\,\Delta\underline{d}_{\mathcal{I}} - \underline{\underline{K}}_{\mathcal{A}\mathcal{A}}\,\Delta\underline{d}_{\mathcal{A}} - \underline{r}_{\mathcal{A}}\right) . \qquad (14)$$

4. Algorithm

The algorithm for solving the contact problem, which was formulated above, can be written in this way

1. In the step $k = 0$, set initial value of the vector $\begin{bmatrix} \Delta\underline{d}^0 \\ \underline{z}^0 \end{bmatrix}$.
2. Determine $\mathcal{A}_0$ and $\mathcal{I}_0$, where $\mathcal{A}_0 \cup \mathcal{I}_0 = \mathcal{S}$ and $\mathcal{A}_0 \cap \mathcal{I}_0 = \emptyset$.
3. Find primal-dual couple $\left(\Delta\underline{d}^k,\, \underline{z}^{k+1}\right)$ by solving the system of linear equations (12) (or the system with eliminated multiplicators).
4. Update $\underline{d}^{k+1} = \underline{d}^k + \Delta\underline{d}^k$.
5. Determine $\mathcal{A}_{k+1}$ and $\mathcal{I}_{k+1}$
$$\mathcal{I}_{k+1} = \left\{ j \in \mathcal{S} | \left(\underline{n}_j^{k+1} \cdot \underline{z}_j^{k+1} - c_n \tilde{g}_j^{k+1} \right) \leq 0 \right\} \;,$$
$$\mathcal{A}_{k+1} = \left\{ j \in \mathcal{S} | \left(\underline{n}_j^{k+1} \cdot \underline{z}_j^{k+1} - c_n \tilde{g}_j^{k+1} \right) > 0 \right\} \;.$$
6. If $\mathcal{A}_{k+1} = \mathcal{A}_k$, $\mathcal{I}_{k+1} = \mathcal{I}_k$ and $\|\underline{r}_{tot}\| \leq \varepsilon_r$, then stop, else increment $k = k + 1$ and continue from the 3rd step.

ε_r represents accuracy of our calculation and a vector $\underline{r}_{tot}$ contains vector of a residual force $\underline{r}$ and the residual contact constraints.

5. Numerical experiment

The semismooth Newton method was tested on a static problem, in which the slave body was divided into 10×10 elements and the master body into 20×10 elements (see Figure 3). On the upper halves of sides of the slave boundary a Dirichlet condition was defined. The Dirichlet condition was also defined on the bottom of the master boundary. On the other parts of the boundaries, except the contact boundary, a Neumann condition was prescribed. We choose the calculation accuracy equal to $\varepsilon_r = 10^{-9}$.

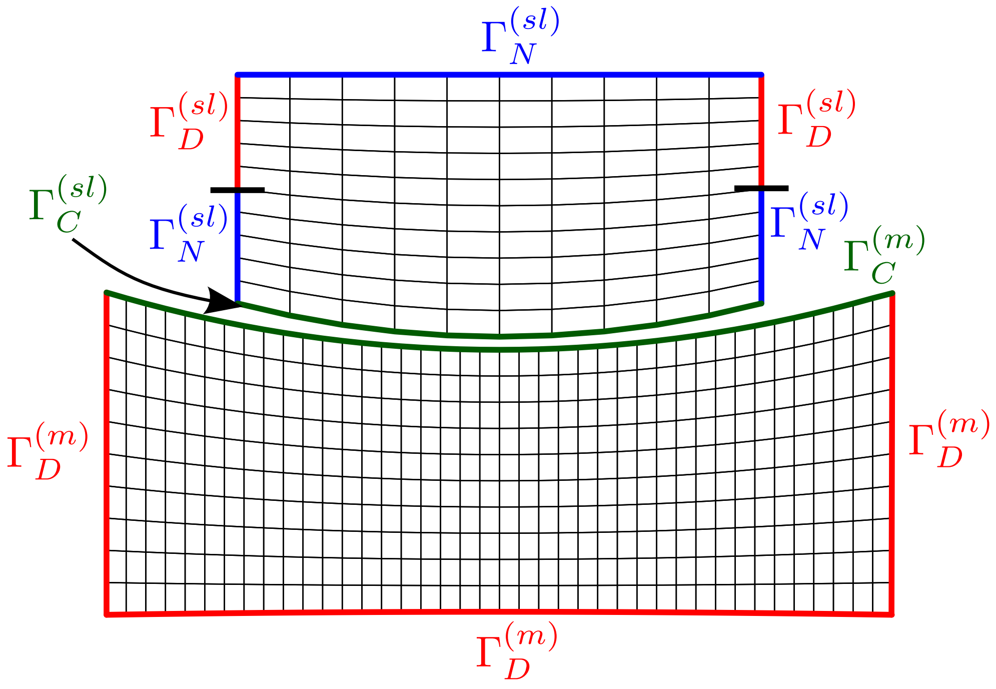

Figure 3: Static problem.

We compared both modifications of the semismooth Newton method (the first modification, denoted SSNM – Alg. 1, uses the system which contains multiplicators $\underline{z}$ and the second modification, denoted SSNM – Alg. 2, uses the system without multiplicators $\underline{z}$) with a fixed point problem. In each step of the algorithms the $\|\underline{r}_{tot}\|$ value was used to stop the algorithm. The individual $\|\underline{r}_{tot}\|$ values are shown in Table 1 listed below .

Individual matrices assembling was implemented in a C++ language with the use of the mex interface for ability to employ this code in MATLAB. Both of the mentioned algorithms (semismooth Newton method modification and fixed point algorithm) were implemented in MATLAB and were added to existing MatSol library which was developed at the Department of Applied Mathematics of the VŠB – Technical University of Ostrava.

k	SSNM – Alg. 1	SSNM – Alg. 2	fixed point
1	$6.13 \cdot 10^{1}$	$6.13 \cdot 10^{1}$	$6.00 \cdot 10^{4}$
2	$6.53 \cdot 10^{-1}$	$6.53 \cdot 10^{-1}$	$3.30 \cdot 10^{0}$
3	$5.62 \cdot 10^{-4}$	$5.62 \cdot 10^{-4}$	$2.97 \cdot 10^{0}$
4	$3.17 \cdot 10^{-7}$	$3.17 \cdot 10^{-7}$	$3.30 \cdot 10^{-3}$
5	$8.08 \cdot 10^{-10}$	$7.85 \cdot 10^{-10}$	$3.13 \cdot 10^{-4}$
6			$7.38 \cdot 10^{-6}$
7			$4.75 \cdot 10^{-7}$
8			$1.92 \cdot 10^{-8}$
9			$1.01 \cdot 10^{-9}$
10			$4.54 \cdot 10^{-11}$

Table 1: $\|\underline{\mathrm{r}}_{tot}\|$ values.

Acknowledgements

This work was supported by grant SGS . SP2015/100, VŠB – Technical University of Ostrava, Czech Republic.

References

[1] Dostál, Z., Kozubek, T., Brzobohatý, T., Markopoulos, A., and Vlach, O.: Scalable TFETI with optional preconditioning by conjugate projector for transient frictionless contact problems of elasticity. Comput. Methods Appl. Mech. Engrg. **247–248** (2012), 37–50.

[2] Motyčková, K. and Kučera, R.: Semi-smooth Newton method for solving 2D contact problems with Tresca and Coulomb friction. Advances Electr. Electron. Engrg. **11** (2013), 218–226.

[3] Popp, A., Gee, M. W., and Wall, W.A.: A finite deformation mortar contact formulation using a primal-dual active set strategy. Internat. J. Numer. Methods Engrg. **79** (2009), 1354–1391.

[4] Wohlmuth, B. I.: Variationally consistent discretization schemes and numerical algorithms for contact problems. Acta Numer. **20** (2011), 569–734.

[5] Wriggers, P.: *Computational contact mechanics*. J. Wiley, Hoboken, NJ, 2002.