Margarita Kovtun

(Department of Applied Mathematics,

St.Petersburg Polytechnical University)

Modeling of contact interaction with friction

Contents

1 Introduction

This paper is devoted to the development of fast algorithm for solving the contact problems. Contact problems often arise in different engineering applications. While solving these problems it is necessary to take into account the possibility of contact between body and an obstacle. The main difficulty is that the area of possible contact is unknown in advance. As a rule, iterative process is used to solve such kind of problems. However, it leads to high computational and time costs. In addition, it may happen that iterative process will not converge. At the same time one may require to perform the fast computations. Algorithm presented in this paper allows doing this in certain important cases.

2 General algorithm description

Let us consider a deformable body fixed in some points (see Fig. 1). Let us call a zone on the body surface where contact with obstacle may occur as a contact zone. We choose a system of computational nodes $CN = (cn_1, cn_2, \cdots, cn_n)$ in the contact zone (see. Fig. 1). Computational nodes are the points where computations of body's displacement are performed.

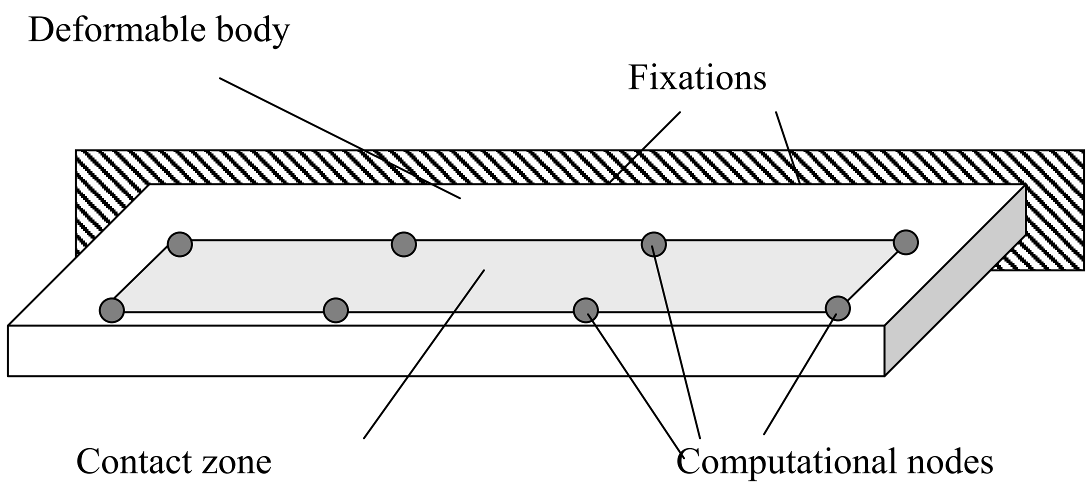


Fig. 1. An example of contact zone and system of computational nodes

Let $P^T = (p_{1x}, p_{1y}, p_{1z}, p_{2x}, p_{2y}, p_{2z}, \cdots, p_{nx}, p_{ny}, p_{nz})$ be a vector of loads, applied in computational nodes, where p_{ix} is a load, applied in *i*-th node in *X* direction, p_{iy} is a load, applied in *i*-th node in *Y* direction, etc. The action of loads causes displacements of computational nodes $U^T = (u_{1x}, u_{1y}, u_{1z}, u_{2x}, u_{2y}, u_{2z}, \cdots, u_{nx}, u_{ny}, u_{nz})$ (u_{ix} is a displacement of node cn_i under the action of *P* in *X* direction). During this stage the obstacle is not considered.

Forces *P* and displacements U are connected by the formula:

$$P = K \cdot U, \tag{1}$$

where *K* is rigidity matrix of mechanical system.

We take note that denoted matrix *K* doesn't correspond to the rigidity matrix of some finite element model. In present case *K* is computed in a following way:

A force equal to 1 Newton is applied in the first computational node in *X* direction. The values of displacements of computational nodes under applied load are calculated in all coordinate directions: *X, Y* и *Z* (for example, it can be done with the help of FEM complexes such as ANSYS, NASTRAN, etc.). Obtained vector of displacements becomes a first row of flexibility matrix *R*, inverse to matrix *K*:

$$U = R \cdot P \tag{2}$$

Then the unit load in *Y* direction is applied in first computational node and again the values of displacements of computational nodes are calculated in all directions. Thus we obtain the second row of flexibility matrix R. By analogy we compute the third row of matrix R applying the unit load in *Z* direction.

We fill in the flexibility matrix R by repeating the procedure described above for all computational nodes. Hereby flexibility matrix has $3n \times 3n$ dimensions: where *n* is the number of computational nodes.

To obtain the rigidity matrix we are to inverse the flexibility matrix $K = R^{-1}$.

Rigidity matrix corresponds to the body's ability to react on external load. It is defined by the geometry of the body, material properties, boundary conditions (fixations) and also by the computational nodes' positions $CN = (cn_1, cn_2, \cdots, cn_n)$. Consequently, variation of these parameters requires recalculation of rigidity matrix.

Introducing the rigidity matrix we replace the body itself by the system of computational nodes, where forces and displacements are related by (1). We are only interested in the displacements in the contact zone. Thereby computational nodes are chosen only in this area.

Our aim is to find the displacements of computational nodes, when the deformable body is subject to external forces, taking into consideration the presence of an obstacle, which constrains body's displacement and interacts with it via the friction.

Usually such type of problems is solved by finite element method so that the conditions of non-penetration are followed. Developed algorithm is based on a different approach, which uses the minimization of quadratic functional under linear constraints. Due to this approach high computational speed is achieved.

Let us write down the expression for potential energy of a system:

$$W(u_{1x}, u_{1y}, u_{1z}, u_{2x}, u_{2y}, u_{2z}, \ldots u_{nx}, u_{ny}, u_{nz}) = \frac{1}{2} U^T \cdot K \cdot U - P^T \cdot U, \tag{3}$$

where K – rigidity matrix, U – vector of displacements, P – vector of loads.

Fundamental mechanical statement says that the system is in equilibrium if its potential energy is minimal.

Presence of an obstacle eliminates body’s movement in a following way:

$$AU \le \Delta, \tag{4}$$

here A – matrix which defines the direction, Δ – vector of initial distances between computational nodes and an obstacle in certain direction. More detailed description of matrix A and vector Δ is given below.

Moreover, if it necessary to take into account the friction between body and obstacle, then we have to impose the constraints not only on displacements but also on forces. According to Coulomb-Mohr law, tangential and normal forces (F_τ and F_n correspondingly) acting in contact zone are related by the formula:

$$|F_\tau| \le |\mu F_n|, \tag{5}$$

where μ – friction coefficient.

Thus, the problem of computation of displacements in contact zone turns into the problem of quadratic functional minimization (3) under constraints on displacements (4) and forces (5). To provide fast computations means to provide the linearity of constraints.

Now let’s consider how described algorithm may be applied to certain problems.

3 Applications

3.1 Computation of bodies' displacements connected by springs taking into account the friction

Let us consider a system of N bodies. Each body has the mass m. Bodies are connected by springs of stiffness c (see Fig. 2). The bodies are assumed to be material points. The end of outside left spring is fixed in the wall. Initially all the springs were not stretched then the horizontal force F was applied to outside right body that caused the displacement in the system. Our objective is to calculate bodies' final disposition.

Each body is subject to the friction force F_{fr}, which doesn't exceed μmg (g is the acceleration of gravity).

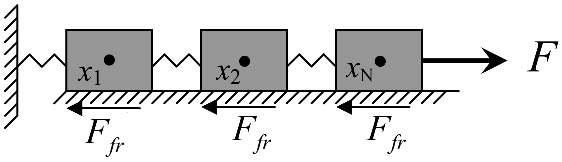


Fig. 2. Considered system

We refer one computational node to each body. As a result we obtain a system of N computational nodes. We consider only horizontal displacements so the problem is one-dimensional.

Rigidity matrix which connects horizontal forces and displacements has a form:

$$K = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{pmatrix}_{N \times N}$$

The energy of deformation is written like that:

$$W(x_1, x_2, \ldots x_N) = \frac{c}{2} \cdot X^T K X = \frac{c}{2} x_N^2 + c \sum_{i=1}^{N-1} x_i^2 - c \sum_{i=1}^{N} x_i x_{i-1} , \quad (6)$$

where $X = (x_1, x_2, \ldots, x_N)^T$ is the vector of horizontal displacements of computational nodes.

Constraints (5) take on form

$$-\mu mg \le c(u_{i+1} - u_i) - c(u_i - u_{i-1}) \le \mu mg, \quad i = 1, \ldots, N-1; \tag{7}$$

$$-\mu mg \le F - c(u_N - u_{N-1}) \le \mu mg\,. \tag{8}$$

Inequalities (7) provide the fulfillment of Coulomb-Mohr law in «inner» computational nodes ($i = 1, \ldots, N-1$), and inequality (8) – in the last computational node (outside right in Fig. 2).

Thus it is necessary to find such a vector of displacements *X*, which provides minimum to functional (6) under constraints (7) and (8).

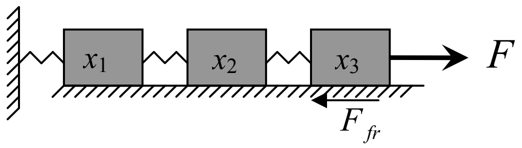
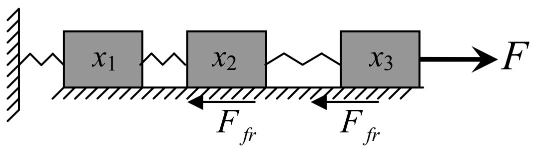
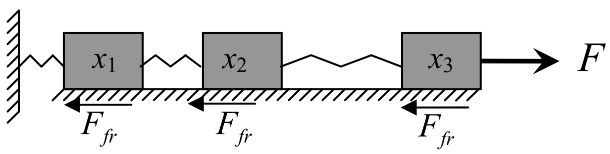
Let's find numerical solution for the system of three bodies (*N*=3). Input data are as follows: spring stiffness $c = 1\ \frac{\text{N}}{\text{m}}$, friction coefficient $\mu = 0.2$, body's mass $m = 2$ kg. Then friction force doesn't exceed value of 4 N.

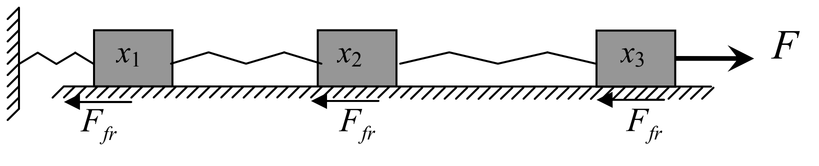
Stated problem has analytical solution:

$$x_i = x_{i-1} + \frac{\max\{F - (N+1-i)F_{fr}, 0\}}{c}, \quad i = \overline{1, N} \tag{9}$$

Table 1 presents the results of numerical solution using the described algorithm. We should note that numerical solution is the same as analytical one.

Table 1 Numerical solution for the problem

№	F (N)	x_1 (m)	x_2 (m)	x_3 (m)	Scheme of bodies' displacements
1	3	0	0	0	
2	6	0	0	2	
3	9	0	1	6	

4	14	2	8	18	

3.2 Rode bending over the obstacle

Let us consider the system of elastic rode and absolutely rigid obstacle (see Fig. 3). Rode has following characteristics: length 40 mm, Young's Modulus E=10^{12} Pa, Poisson coefficient ν=0.3, circle cross-section with 0.2 mm^2 square. Rode is fixed in the right point (Fig. 3). All displacements and rotations in this point are forbidden. The gap between rode and obstacle changes linearly from 1 to 5 mm.

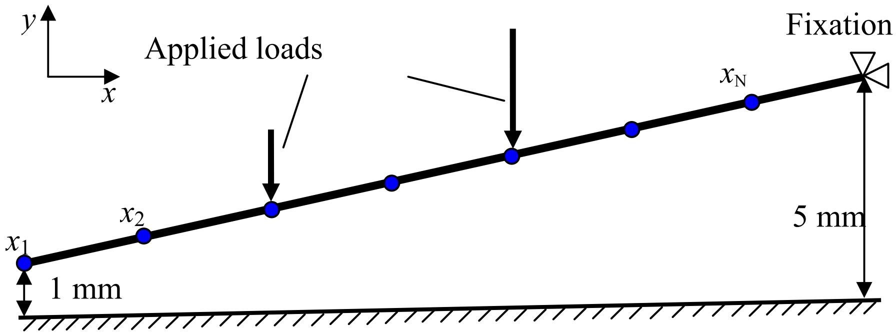


Fig. 3. Geometry of rode and obstacle

The rode is divided by eight computational nodes (N=8). Every computational node can move in horizontal and vertical direction (along x and y axes). Then the vector of displacements has a form $U = (u_{1x}, u_{2x}, \ldots, u_{Nx}, u_{1y}, u_{2y}, \ldots, u_{Ny})^T$. Computational nodes are subject to a load $P = (p_{1x}, p_{2x}, \ldots, p_{Nx}, p_{1y}, p_{2y}, \ldots, p_{Ny})^T$ (here p_{ix} and p_{iy} are horizontal and vertical components of load, applied to i-th computational node).

Rigidity matrix K of the rode is computed with the help of ANSYS Mechanical using the procedure described above.

As it can be seen in Fig. 3, the obstacle constrains rode's movement in vertical direction. Consequently, inequalities (4) take form:

$$A_y U \leq \Delta_y. \tag{10}$$

Here A_y is the matrix of projection operator on vertical direction, Δ_y is the vector of initial distances between computational nodes and obstacle.

In this case we don't take into consideration the friction between rode and obstacle. That's why constraints (5) are not taken into account.

Thus, the problem is to find the displacement vector in computational nodes U, which provides minimum to functional (3) under constraints (10).

To verify obtained results ANSYS Mechanical is used which allows solving the contact problem by finite element method.

Two numerical experiments were held corresponding to different configurations of external loads. Figures 4 and 5 show two schemes of loadings and computed deformations.

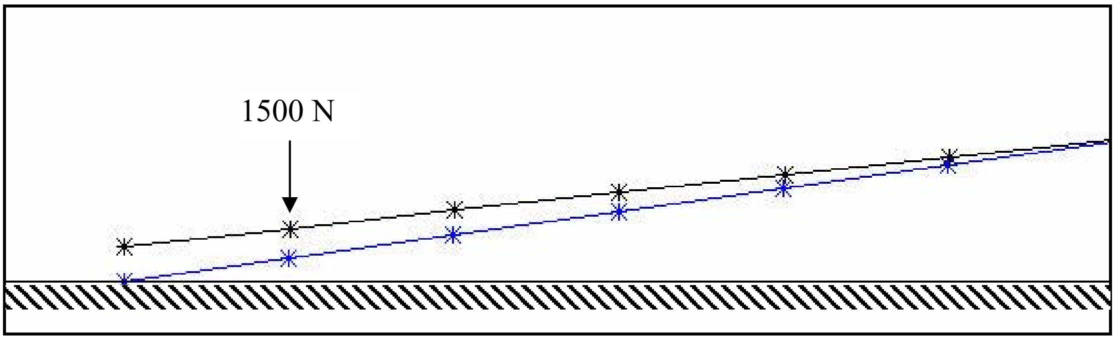


Fig. 4. Rode deformation under one force

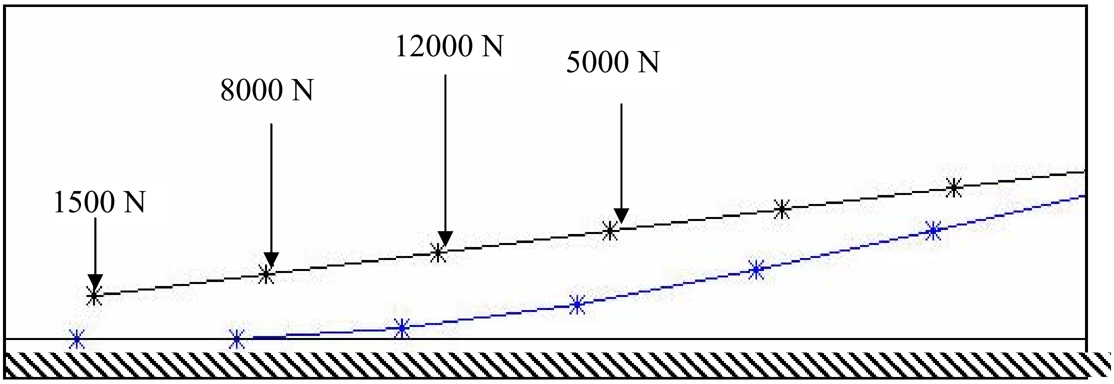


Fig. 5. Rode deformation under several forces

Relative deviation of calculation of displacement (in i-th node) is computed according to formulas:

$$(\varepsilon_X)_i = \left| \frac{(U_X^{ANSYS})_i - (U_X)_i}{(U_X^{ANSYS})_i} \right| \cdot 100\%, \quad (\varepsilon_Y)_i = \left| \frac{(U_Y^{ANSYS})_i - (U_Y)_i}{(U_Y^{ANSYS})_i} \right| \cdot 100\%.$$

Table 2 contains relative deviations in each computational node for both variants of loading. As it can be seen from a table, maximal relative error doesn't exceed 0.3%.

Table 2 Relative deviations for both cases of loading

	Variant of loading			
	One force (Fig. 4)		Several forces (Fig. 5)	
Node number	$(\varepsilon_X)_i$, %	$(\varepsilon_Y)_i$, %	$(\varepsilon_X)_i$, %	$(\varepsilon_Y)_i$, %
1	0.253	0.000	0.008	0.000
2	0.034	0.003	0.032	0.082
3	0.012	0.013	0.045	0.081
4	0.060	0.014	0.041	0.054
5	0.183	0.010	0.079	0.043
6	0.138	0.042	0.069	0.095
7	0.143	0.020	0.216	0.080

4 Conclusion

Algorithm presented in this paper is based on the minimization of potential energy of the system. It allows solving some particular important contact problems fast and accurately. Efficiency of proposed approach is confirmed by two basic test examples. In both cases numerical solution is compared to analytical one or to the solution obtained in ANSYS.

Described algorithm may be applied in geomechanics while calculating the relative displacement of soil layers. Another sphere of application is modeling of riveting process between different parts (usually in aircrafts).