Transactions of the VŠB – Technical University of Ostrava, Mechanical Series
No. 3, 2009, vol. LV
article No. 1724

Petr HORYL[*], Tomáš KOZUBEK[**], Alexandros MARKOPOULOS[***], Tomáš BRZOBOHATÝ[****]

DEMONSTRATION OF THE MATSOL LIBRARY DEVELOPED FOR THE EFFICIENT SOLUTION OF CONTACT PROBLEMS AND ITS COMPARISON WITH ANSYS

DEMONSTRACE KNIHOVNY MATSOL VYVINUTÉ PRO EFEKTIVNÍ ŘEŠENÍ KONTAKTNÍCH ÚLOH A JEJÍ SROVNÁNÍ S ANSYSEM

Abstract

The goal of this contribution is to present the MatSol library developed for the efficient solution of large problems in contact mechanics. The MatSol solvers are based on FETI domain decomposition methods which are well known by their parallel and numerical scalability. The performance is illustrated on a model contact problem of the cantilever cube over the obstacle. Finally, we compare the MatSol solution with the ANSYS one of a simplified contact problem with Coulomb friction of the yielding clamp connection.

Abstrakt

Cílem tohoto příspěvku je představení knihovny MatSol vytvořené za účelem efektivního řešení rozsáhlých úloh kontaktní mechaniky. Řešiče v MatSolu jsou založeny na FETI metodách rozložení oblastí, které jsou dobře známé svou paralelní a numerickou škálovatelností. Efektivnost knihovny je demonstrována na modelové úloze nosníku tvaru krychle nad tuhou překážkou. Na závěr srovnáváme řešení zjednodušené kontaktní úlohy s Coulombovským třením poddajného svěrného spoje spočtená MatSolem a ANSYSem.

1 INTRODUCTION

During last several years, our MatSol team [3] has been focused on development of scalable algorithms for contact problems in mechanics. These algorithms are based on FETI (Finite Element Tearing and Interconnecting) domain decomposition methods which are well known by their numerical and parallel scalability, i.e., we are able to solve resulting quadratic programming problems in $O(1)$ iterations and in $O(1)$ seconds independently on the problem size only by adding directly proportional number of processors [1]. All the algorithms were implemented into a new library that is developed in the Mathworks Matlab [2] environment which is equipped with many helpful functions for mathematics, plotting, debugging, quick testing and efficient implementation. We call this library MatSol (MATlab SOLvers) [3]. Nowadays, it is our primary testing and developing library. To parallelize the algorithms we use Matlab Distributed Computing Engine which allows running Matlab functions also on parallel computers. Hence, the MatSol has full functionality to solve efficiently large problems of mechanics.

[*] prof., Ing., CSc., Department of Mechanics, Faculty of Mechanical Engineering, VSB-TUO, 17. listopadu 15, Ostrava-Poruba, tel. (+420) 59 732 4351, e-mail: petr.horyl@vsb.cz

[**] doc., Ing., Ph.D., Department of Applied Mathematics, Faculty of Electrical Engineering and Computer Sciences, VSB-TUO, 17. listopadu 15, Ostrava-Poruba, tel. (+420) 59 732 3494, e-mail: tomas.kozubek@vsb.cz

[***] Ing., Department of Mechanics, Faculty of Mechanical Engineering, VSB-TUO, 17. listopadu 15, Ostrava-Poruba, tel. (+420) 59 732 5752, e-mail: alexandros.markopoulos@vsb.cz

[****] Ing., Department of Mechanics, Faculty of Mechanical Engineering, VSB-TUO, 17. listopadu 15, Ostrava-Poruba, tel. (+420) 59 732 5752, e-mail: tomas.brzobohaty@vsb.cz

2 STRUCTURE OF THE MATSOL LIBRARY

Todays structure of the MatSol library looks as follows. The solution process starts from the model which is stored in the model database. Models may be converted to the model database from standard commercial and non-commercial preprocessors like ANSA, ANSYS, COMSOL, PMD, etc. The list of preprocessing tools is not limited and any new one can be simply plugged into the MatSol library by creating a proper database convertor. Preprocessing part continues in dependence on the solved problem. User can solve deterministic or stochastic problems, static or transient analysis, optimization problems, problems in linear and non-linear elasticity and contact problems with various friction models. To discretize our problems we can choose between finite and boundary element methods. Furthermore, FETI or BETI based methods are used as the domain decomposition approaches. The solution process could be run either in sequential or parallel mode, but the algorithms are implemented in such a way that the code is the same for both modes. MatSol library includes also tools for postprocessing of the results and advanced tools for postplotting. The results of the problem may be converted through the model database to the modelling tools for further postprocessing.

The above described structure of the MatSol library allows overriding standard solvers in commercial and non-commercial finite element packages and substituting them by those implemented in MatSol. This gives a very useful alternative to users of commercial packages and a great tool for algorithm developers to test the new algorithms on the realistic problems.

3 DEMONSTRATION OF SCALABILITY OF THE MATSOL SOLVERS

Now we shall illustrate the performance of the MatSol solvers on a model example solved on the computational cluster HP BLc7000 with 9 nodes. Each node is equipped with 2 dual core AMD Opteron processors and 8GB RAM and interconnected by infiniband network. On this cluster we have installed 24 licences of Matlab distributed computing engine.

Let us consider a cantilever cube over the obstacle (see Fig. 1). The elastic body is represented by the steel cube. The body is fixed in all directions along the left face and loaded by traction along the top face. The bottom face of the body may touch the rigid plane obstacle. In Figure 2, we depict the deformed body together with the traces of decomposition.

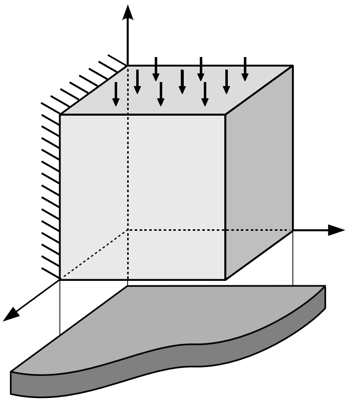

Fig. 1 Geometry of the problem.

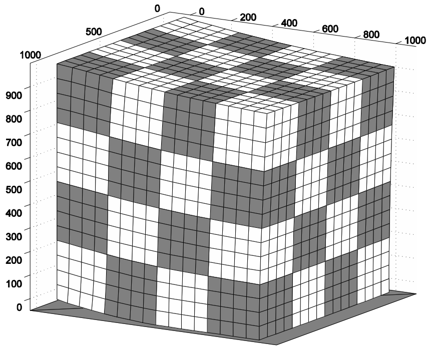

Fig. 2 Deformed body and decomposition.

The numerical scalability of our algorithm is illustrated in Figure 3. We can observe that the number of iterations with increasing problem size from thousands to millions of DOFs increases only moderately in agreement with the theory. Finally, the parallel scalability is depicted in Figure 4, where we fix the problem size (number of DOFs is equal to 30,000) and increase the number of partitions into subdomains accordingly to the number of used CPUs. The behavior agrees with the theoretical results.

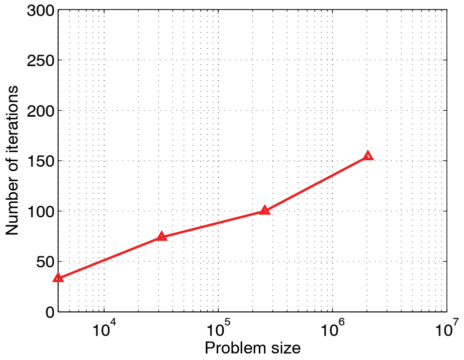

Fig. 3 Numerical scalability.

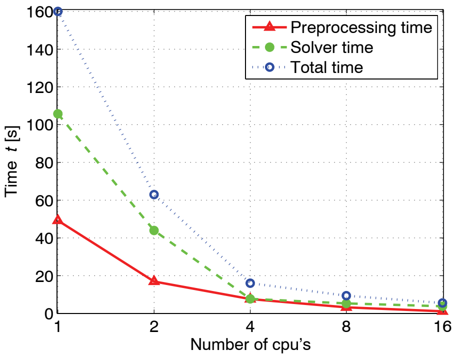

Fig. 4 Parallel scalability.

4 COMPARISON WITH ANSYS

For our experiments we take a yielding clamp connection (see Fig. 5) of steel arched supports. This type of the construction is very often used for supporting of mining seams. It is a typical multibody contact, where the yielding connection plays a role of the mechanical protection against destruction, it means against the total deformation of the supporting arches. For the purposes of this contribution we simplify the 3D model and we get our testing 2D example depicted in Figure 6, where we assume the symmetry with respect to the vertical axis, zero gaps between contacts and nodal loads as marked in the same figure.

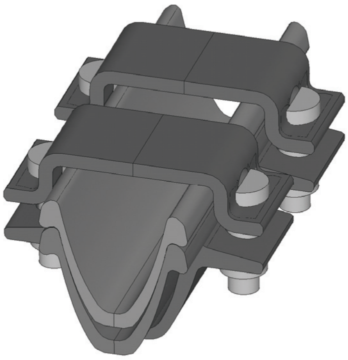

Fig. 5 3D model of the yielding clamp connection.

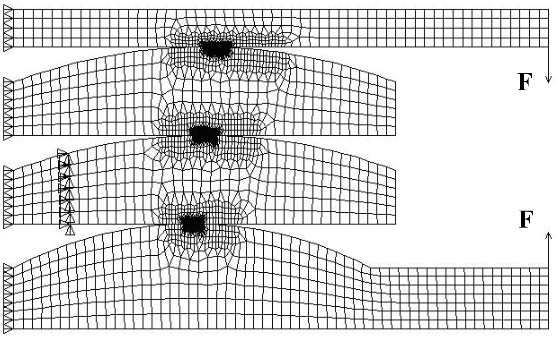

Fig. 6 2D simplified model for the numerical solution.

Next, we compare the results (maximum displacement, maximum contact pressure, maximum of the essential components of the stress tensor, and solution time) achieved by the MatSol library and ANSYS.

The simplified model is created in ANSYS and to discretize it we generate a mesh with 8,616 nodes, 17,232 DOFs and 9,036 elements (8,009 of them are elements PLANE42, 487 elements CONTA172 and 540 elements TARGE169). Only the third body from above is fixed in space at the interior nodes marked with two triangles per node in Figure 6. Other bodies of the structure are bonded only by imposed symmetry and contact conditions. Contact conditions are enforced by the augmented or pure Lagrangian method with the allowed penetration set to 0.01mm. The setting of the contact is a “standard” type and the Coulomb friction coefficient is $f = 0.1$. The behavior of the contact pressure is illustrated in Figure 7. The contact zones are numbered from above to down (from 1 to 3) and the corresponding maximum contact pressures are denoted by P_1, P_2, and P_3.

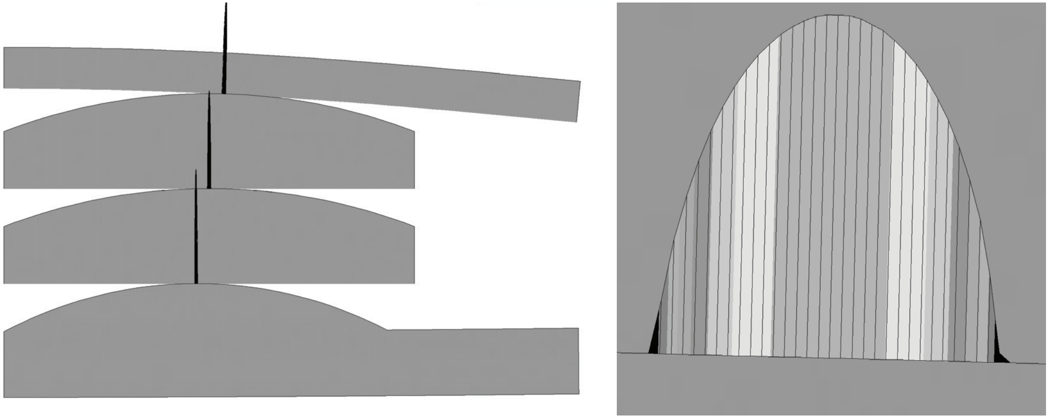

Fig. 7 Contact pressure, left – whole structure, right – detail of the contact pressure in zone 3.

In Figure 7, we see relatively narrow contact places in all three contact zones. However, if we zoom we will see - at the detail - the desirable continuity of the contact pressure. The correctness of the numerical solution of contact problems in ANSYS is given by the value of the normal penalty stiffness factor FKN. It can be interpreted as a spring stiffness in the contact interface. In the theory, this stiffness ought to be infinity but the computational process does not converge in this case. This value depends on the Young’s modulus of the bodies in contact and on the sizes of elements creating the contact boundary. The typical initial value of FKN is 1. If the process is converging, then the value is increasing. The process starts to converge for our model example after entering the instruction "close gap" for all contact pairs [4] which closes gaps between the contacts. The question is when the changing of FKN ought to be stopped. It showed to be useful to add an auxiliary convergence criterion $\delta_P = |P^{(i)} - P^{(i-1)}| / P^{(i)} < \varepsilon$, where $P^{(i)}$ is the maximum contact pressure in the *i-th* step for $FKN^{(i)}$ and ε is the relative tolerance (for example $\varepsilon = 0.001$). In our example, it is possible to change the value FKN from 1 to 300. But many other examples do not convergence for increasing FKN. The behavior of the maximum contact pressure P_3 in dependence on FKN is illustrated in Figure 8. The value of the contact pressure achieved by MatSol is depicted in dashed line.

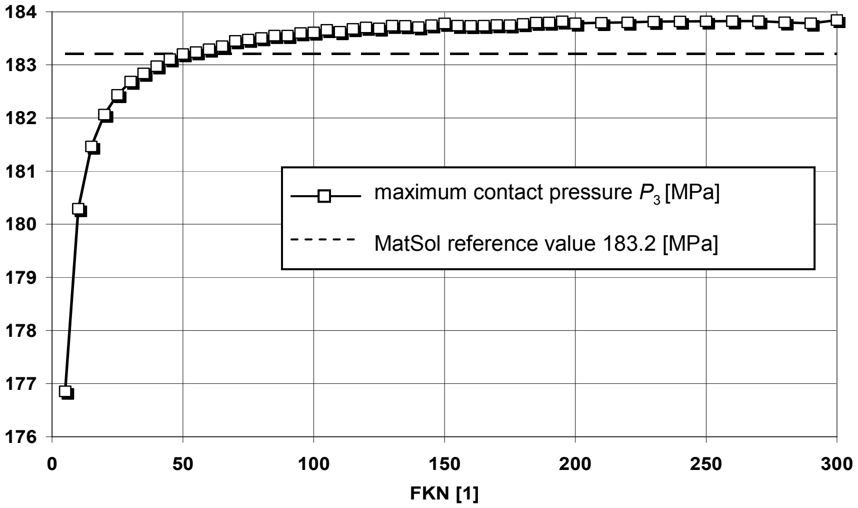

Fig. 8 Behavior of the maximum contact pressure P_3 in dependence on FKN.

It is clear that the values of the contact pressure start to stabilize for FKN>190. This is confirmed also by the behavior of the convergence criterion δ_p depicted in Figure 9. The computation can be stopped for FKN=190 which corresponds to the value P_3=183,8 MPa. The penetration is rapidly decreasing in accordance with increasing FKN. The penetration becomes negligible, circa 4E-5mm, for FKN=10 but the contact pressure P_3 is still far from the reference value. Figure 10 illustrates the influence of the maximum contact pressure P_3 on the tolerance TOL in the convergence criterion on the out of balance of nodal forces in nonlinear analysis (instruction CNVTOL,F,TOL). It is clear from this figure that TOL must be less than 0.002.

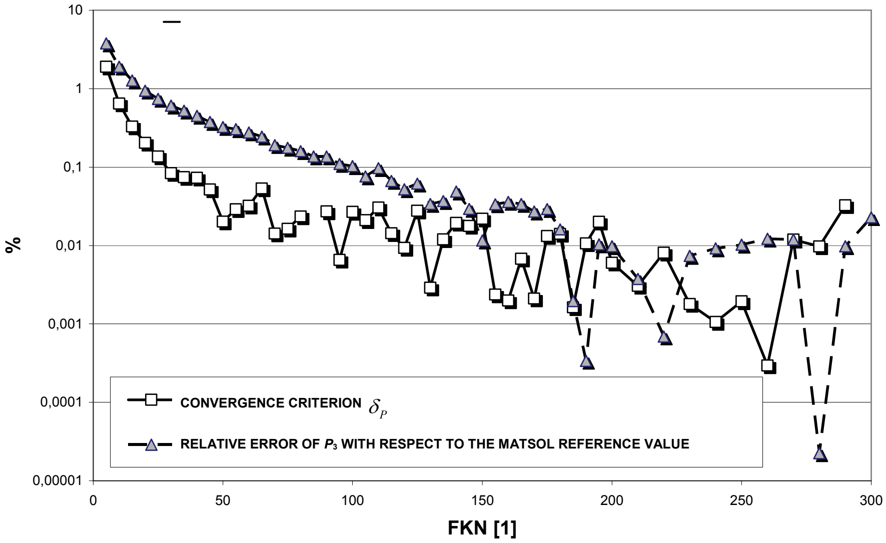

Fig. 9 Convergence criterion in dependence on FKN.

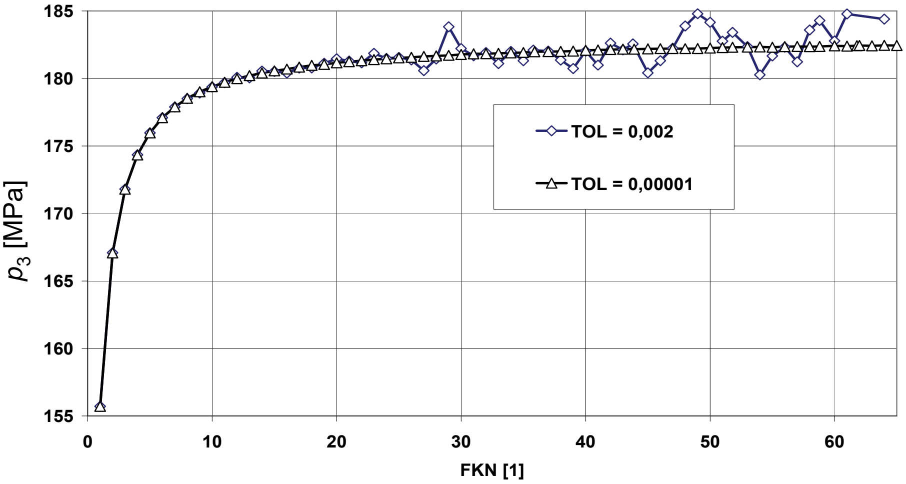

Fig. 10 Influence of the contact pressure P_3 on the parameter TOL in the convergence criterion on the out of balance of nodal forces in nonlinear analysis.

The results of the numerical solution achieved by ANSYS and MatSol are reported in Table 1. The differences in the values of the considered quantities are relatively small, e.g., the difference of the maximum contact pressure P_3 from the reference MatSol value is 0.4%. The greatest difference 1.15% is in the first principal stress. In some cases the computation in ANSYS may be accelerated using the pure Lagrangian method instead of the augmented Lagrangian one with the normal stiffness penalty factor [4] as you can see in Table 1.

Quantity		**MatSol**		**ANSYS**			
				Augmented Lagrangian*		**Pure Lagrangian**	
Umax	**[mm]**	3.468		3.468		3.471	
Sx	**[MPa]**	-646.6	517.8	-646.0	517.3	-645.8	517.7
Sy		-190.8	95.5	-190.6	95.5	-190.7	95.5
Txy		-48.0	50.3	-48.5	50.0	-48.7	50.1
S1		523.4		517.4		517.8	
S3		-647.4		-646.7		-646.3	
S_{HMH}		587.3		587.1		585.4	
P1		150.6		151.0		150.9	
P2		155.3		156.3		155.9	
P3		183.2		183.8		184.0	
CPU time	**[s]**	175.0		3,024.0		171.0	

* for values FKN = 190, CNVTOL,F,1E-6

Tab. 1 Summarizing results of the testing example.

5 CONCLUSIONS

We have presented the MatSol library developed for the efficient solution of large problems in contact mechanics. The implemented numerically and parallely scalable solvers are based on FETI domain decomposition methods. They enable to solve resulting quadratic programming problems in $O(1)$ iterations and in $O(1)$ seconds independently on the problem size only by adding directly proportional number of processors. The scalability was illustrated on a model contact problem of the cantilever cube over the obstacle. The behavior corresponds to the theoretical results.

Next, we compared the MatSol solution with the ANSYS one of a simplified contact problem with Coulomb friction of the yielding clamp connection. On the basis of our experiments with multibody contact problems, where some bodies are free in space (rigid body motions are admitted), we can say that some problems with zero gaps between the contacts converge without inserting any "artificial interventions" into the structure in ANSYS but generally we need them. As interventions we consider inserting the weak springs limiting the rigid body motions, inserting the artificial damping into nonlinear analysis, the use of the instruction STABILIZE and transformation of the static problem to the dynamic one. On the other hand, we do not need any of these interventions in MatSol because the implemented algorithms work directly with singular matrices.

ACKNOWLEDGMENTS

This research has been supported by the grants GA CR 101/08/0574 and the Ministry of Education of the Czech Republic No. MSM6198910027.

REFERENCES

[1] Z. Dostál, *Optimal Quadratic Programming Algorithms, with Applications to Variational Inequalities,* 1st edition, Springer US, NY 2009, SOIA 23.

[2] Matlab - http://www.mathworks.com.

[3] MatSol - http://www.am.vsb.cz/matsol.

[4] László I. *Interní zpráva SVS FEM*, SVS FEM s.r.o., Brno, 2009.