

A NOTE ON MASSIVELY PARALLEL IMPLEMENTATION OF FETI FOR THE SOLUTION OF CONTACT PROBLEMS

David HORAK^{1,2}, Vaclav HAPLA², Jakub KRUIK², Radim SOJKA²,
Martin CERMAK², Jiri TOMCALA², Marek PECHA^{1,2}, Zdenek DOSTAL^{1,2}

¹Department of Applied Mathematics, Faculty of Electrical Engineering and Computer Science, VSB–Technical University of Ostrava, 17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

²IT4Innovations National Supercomputing Center, VSB–Technical University of Ostrava, 17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

david.horak@vsb.cz, vaclav.hapla@vsb.cz, jakub.kruzik@vsb.cz, radim.sojka@vsb.cz,
martin.cermak@vsb.cz, jiri.tomcala@vsb.cz, marek.pecha@vsb.cz, zdenek.dostal@vsb.cz

DOI: 10.15598/aeec.v15i2.2321

Abstract. *The paper deals with a solution of large multibody contact problems using massively parallel computers and domain decomposition methods. These methods can solve the problems discretized by billions of nodal variables at the cost nearly proportional to the number of variables using up to thousands cores before the communication costs start to dominate the computational costs. The paper describes the ingredients essential for efficient massively parallel implementation that increases the parallel scalability beyond the limit mentioned above. The improvements were enhanced into a new software package PERMON which is based on PETSc. The performance of the algorithm is demonstrated on the solution of an academic benchmark discretized by nearly billion of nodal variables.*

Keywords

Coarse problem, contact problems, MPRGP, PERMON, quadratic programming, SMALBE, TFETI.

1. Introduction

We are interested in the solution of large multibody contact problems such as those arising in the design of electromagnetic brakes [3]. A powerful tool for the solution of these problems are the Finite Element Tearing and Interconnecting (FETI) methods that were originally introduced by Farhat and Roux [2] for the solution of large linear systems arising from the discretization of elliptic partial differential equations. The key

ingredient of FETI is the decomposition of the spatial domain into non-overlapping subdomains that are glued by Lagrange multipliers. After elimination of the primal variables, the original problem is reduced to a small, relatively well conditioned equality constrained Quadratic Programming (QP) problem, so it can be solved by a suitable iterative method.

A unique feature of FETI is its numerical scalability, i.e., its capability to solve the discretized problems at the cost nearly proportional to the number of variables. Moreover, FETI also enjoys the parallel scalability – if each subdomain is assigned to a core, then the time of the solution can be reduced nearly proportionally to the number of available cores until the communication costs start to dominate the cost of the solution.

If the FETI procedure is applied to the contact problems, the duality not only reduces the dimension of the original problem, but it also turns all inequality constraints into bound constraints. Though the resulting QP has not only the equality constraints, but also the non-negativity constraints, the resulting problem can still be solved with the optimal complexity [1].

Let us mention that the optimality results mentioned above are rather qualitative; the estimates contain unknown constants, so it is not clear whether we can observe the scalability in the solution of realistic problems. The point of this paper is to present some implementation details that improve the performance of the algorithm and extend the domain of their parallel scalability. The improvements were implemented into a novel software package based on PETSc that is called PERMON toolbox. PERMON uses both our in-house software and efficient open source libraries. The core solver layer consists of two modules: PermonFLOP

for FETI and PermonQP for QP. The numerical and parallel scalability is demonstrated on the solution of an academic benchmark discretized by nearly a billion variables.

2. TFETI Method

Our development is based on a variant of FETI that we call Total-FETI (TFETI) [9]. TFETI uses Lagrange multipliers to enforce the Dirichlet boundary conditions so that all subdomains are floating and the kernels of the associated subdomain stiffness matrices are known a priori.

The original discretized primal problem has the following form:

$$\min \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} - \mathbf{f}^T \mathbf{u} \text{ s.t. } \mathbf{B}_I \mathbf{u} \leq 0 \text{ and } \mathbf{B}_E \mathbf{u} = 0. \quad (1)$$

The primal global distributed objects denote the stiffness matrix, its kernel, equality and inequality constraint matrices and the load vector

$$\begin{aligned} \mathbf{K} &= \text{diag}(\mathbf{K}^1, \dots, \mathbf{K}^{N_s}), \\ \mathbf{R} &= \text{diag}(\mathbf{R}^1, \dots, \mathbf{R}^{N_s}), \\ \mathbf{B} &= [\mathbf{B}_E^T, \mathbf{B}_I^T]^T = [\mathbf{B}^1, \dots, \mathbf{B}^{N_s}], \\ \mathbf{f} &= [(\mathbf{f}^1)^T, \dots, (\mathbf{f}^{N_s})^T]^T, \end{aligned}$$

where diag means a block-diagonal matrix consisting of the diagonal blocks in the parentheses, and \mathbf{K}^s , \mathbf{R}^s , \mathbf{B}^s and \mathbf{f}^s denote the local objects associated with s -th subdomain.

The primal problem Eq. (1) is transformed into the dual one,

$$\begin{aligned} \min \frac{1}{2} \hat{\boldsymbol{\lambda}}^T (\mathbf{P} \mathbf{F} \mathbf{P} + \rho \mathbf{Q}) \hat{\boldsymbol{\lambda}} - \hat{\boldsymbol{\lambda}}^T \mathbf{P} (\mathbf{d} - \mathbf{F} \tilde{\boldsymbol{\lambda}}) \\ \text{s.t. } \hat{\boldsymbol{\lambda}}_I \geq -\tilde{\boldsymbol{\lambda}}_I \text{ and } \mathbf{G} \hat{\boldsymbol{\lambda}} = \mathbf{o}. \end{aligned} \quad (2)$$

The corresponding dual objects are

$$\mathbf{F} = \mathbf{B} \mathbf{K}^\dagger \mathbf{B}^T, \mathbf{G} = \mathbf{R}^T \mathbf{B}^T, \mathbf{d} = \mathbf{B} \mathbf{K}^\dagger \mathbf{f}, \mathbf{e} = \mathbf{R}^T \mathbf{f},$$

where \mathbf{K}^\dagger denotes a left generalized inverse of \mathbf{K} , i.e., a matrix satisfying $\mathbf{K} \mathbf{K}^\dagger \mathbf{K} = \mathbf{K}$, $\tilde{\boldsymbol{\lambda}}$ satisfies $\mathbf{G} \tilde{\boldsymbol{\lambda}} = \mathbf{e}$, while the solution is looked for in the form $\boldsymbol{\lambda} = \hat{\boldsymbol{\lambda}} + \tilde{\boldsymbol{\lambda}}$, and $\rho > 0$ is the regularization parameter. The favorable spectral properties of the Hessian are given by orthogonal projectors $\mathbf{P} = \mathbf{I} - \mathbf{Q}$, $\text{Im } \mathbf{P} = \text{Ker } \mathbf{G}$, and $\mathbf{Q} = \mathbf{G}^T (\mathbf{G} \mathbf{G}^T)^{-1} \mathbf{G}$, $\text{Im } \mathbf{Q} = \text{Im } \mathbf{G}^T$. Evaluating $(\mathbf{G} \mathbf{G}^T)^{-1}$ is called the Coarse Problem (CP).

3. QP Problems

QP problems resulting from the application of the TFETI method to variational inequalities can be solved

by the MPRGP (Modified Proportioning with Reduced Gradient Projections) algorithm and SMALBE (SemiMonotonic Augmented Lagrangian algorithm for Bound and Equality constraints) algorithm, both developed by Dostal [1]. These algorithms have the rate of convergence given by the bounds on the spectrum of the Hessian matrix. This operator contains three projector applications that can be implemented using two CP solutions. In combination with TFETI, these algorithms were proved to enjoy both numerical and parallel scalability.

The rate of convergence does not depend on penalization parameter ρ , but it is bounded by the constant given by the ratio $\frac{H}{h}$:

$$\kappa(\mathbf{P} \mathbf{F} \mathbf{P} \mid \text{Im } \mathbf{P}) \leq C \frac{H}{h}, \quad (3)$$

with h and H denoting the discretization and decomposition parameter, respectively [2]. Efficiency of TFETI method could be further improved by application of preconditioning in face.

4. Massively Parallel Implementation

Parallelization of FETI/TFETI can be implemented using SPMD technique. The distribution of primal matrices is quite straightforward as every subblock reflects a subdomain. The matrices \mathbf{K} and \mathbf{R} have a favourable block-diagonal layout and can be implemented using a block-diagonal matrix composite type where blocks are ordinary sequential matrices and every core holds an array of blocks associated with its subdomains.

While the dual problem is solved iteratively, the auxiliary problems related to the application of an unassembled system matrix are solved directly.

The first auxiliary problem is the stiffness matrix pseudoinverse application. It is parallelizable without any data transfers because of a favourable block-diagonal structure so that core factorizes the regularized local blocks (subdomain stiffness matrices).

The cost of the factorization can be reduced by increasing the number of subdomains. On the other hand, this increases the null space dimension and the number of Lagrange multipliers on the subdomains interfaces (dual dimension). It can hardly be solved sequentially on the master core for large scale problems and becomes a bottleneck, see Fig. 1.

The second one is the CP solution appearing in the application of the projectors. However, this problem does not possess such a favourable structure suitable

for parallel processing and some communication is necessary.

An effect of a choice of the LU direct solver on the Cray XE6 machine HECToR (PETSc, MUMPS, SuperLU) on performance of the TFETI massively parallel implementation in PERMON library can be found in [11].

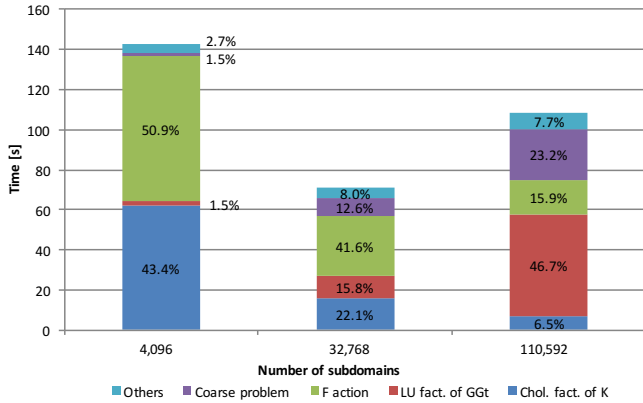


Fig. 1: Increase in number of subdomains leads to a decrease in size of the local stiffness matrices and improvement of bounds at the cost of the CP matrix size increase - times needed by individual actions.

5. K^\dagger Action

In the case of the pseudoinverse, each core regularizes a subdomain stiffness matrix using fixing nodes and factorizes it in the preprocessing phase. The application of K^\dagger then consists of purely local backward and forward substitutions in each CG iteration. A nice time reduction can be reached by the decomposition of the domain into more subdomains, see [10]. For this purpose it was necessary to implement special matrix formats enabling handling an array of diagonal blocks associated with more subdomains on one computational core, see Fig. 2 and Fig. 3.

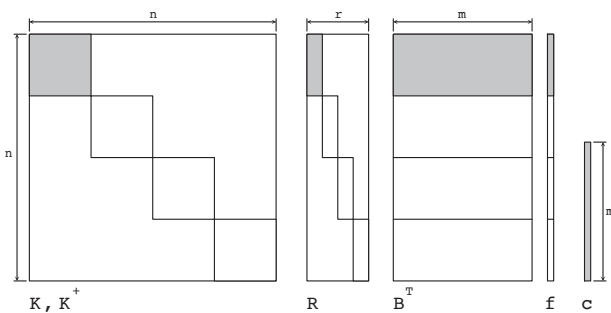


Fig. 2: Data distribution: one subdomain per one core.

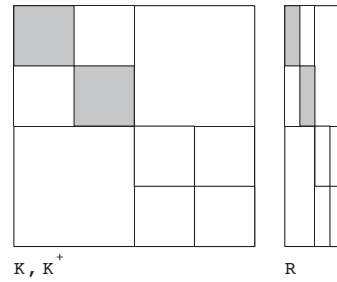


Fig. 3: Data distribution: more subdomains per one core.

6. $(GG^T)^{-1}$ Action

Firstly, it is necessary to assemble the natural coarse space matrix and CP matrix in parallel, see Fig. 4 and Fig. 5. The action time and level of communication depend primarily on the implementation of the CP solution $GG^T x = y$, which can be hardly solved sequentially on the master core for large scale problems. We have suggested and compared several strategies of CP solution:

1. directly using LU factorization,
2. iteratively using PCG,
3. applying explicit inverse of GG^T ,
4. orthonormalization performed by iterative classical version of Gram-Schmidt process, if matrix G^T has orthonormal columns then CP is eliminated because $(GG^T)^{-1} = I$.

The second strategy does not seem suitable because it harms robustness of the FETI method. The third strategy, i.e., explicit inverse assembling + dense matrix-vector products could be effective depending on the number of subdomains and the expected number of CP actions, for results with direct explicit inverse computation see [12]. The fourth orthonormalization strategy with the classical Gram-Schmidt method fails

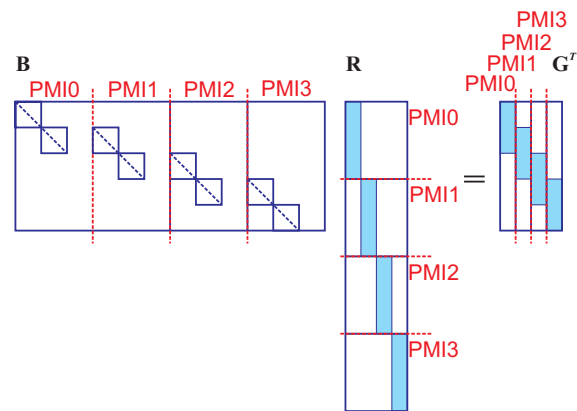


Fig. 4: Parallel computation of GG^T .

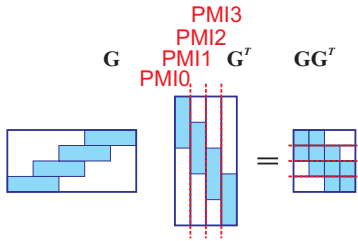


Fig. 5: Parallel computation of \mathbf{GG}^T .

when the nullspace is large enough (thousands) because the roundoff errors become an issue. The modified or iterative Gram-Schmidt methods have better numerical properties but worse scalability.

Let us describe in more details the first strategy using parallel direct solvers (MUMPS [13], SuperLU_DIST [14]) for factorization + forward/backward substitutions. The CP dimension is not large enough to justify the fully parallel approach; in this case, communication takes over computation. The significant efficiency improvement can be achieved by means of the partial parallelization of this CP solution. The approach splits cores of the global "world" communicator into groups of cores called subcommunicators in MPI; the number of these subcommunicators is N_r (number of cores doing redundant work), i.e., the number of cores in each subcommunicator is equal to N_c/N_r , with N_c denoting total number of cores. This CP matrix is factorized in parallel in subcommunicators. The application of $(\mathbf{GG}^T)^{-1}$ then consists of backward and forward substitutions which are not local and not negligible amount of the communication is needed in each CG iteration. Parallel approach has a big advantage consisting in the reduction of memory requirements for the CP solution; there are practically no memory limits as more and more cores can be engaged into the subcommunicators.

However, solving large CPs gets very complicated for tens or hundreds of thousands of subdomains, even if the best techniques are employed. This motivates our research in projector-less TFETI method for contact problems avoiding the CP solution at all, see [15].

7. PERMON Toolbox

The QP algorithms and FETI methods are implemented in our software package PERMON based on PETSc. PERMON (Parallel, Efficient, Robust, Modular, Object-oriented, Numerical) [4] and [5] is a collection of software libraries, uniquely combining QP and DDM. There are two core modules in PERMON: PermonQP and PermonFLLOP. They are built on the top of PETSc [6] and [7], using mainly its linear algebra

part. They extend PETSc with new specific functionality, algorithms for large scale sparse QP problems and DDM of the FETI type. The same coding style is used so that the users familiar with PETSc can utilize them with minimal effort. Among the main applications are contact problems of mechanics.

PermonQP provides a base for solving the QP problems. It includes data structures, transforms, algorithms, and supporting functions for QP. PermonQP is available for free under the FreeBSD open source license.

PermonFLLOP (FETI Light Layer on Top of PETSc) is an extension of PermonQP that adds support for DDM of the FETI type. PermonFLLOP is currently under preparation for publishing. The "gluing" signed Boolean matrix \mathbf{B}_g is constructed based on $l2g$ as described in [8]. The subdomain nullspace matrix \mathbf{R}^s is assembled using the rigid body modes assembled from the mesh nodal coordinates [9].

PermonFLLOP passes the global primal data to PermonQP and calls a specific series of QP transforms provided by PermonQP, resulting in the bound and equality constrained QP which is then solved with the QPSSolve function. Note that PermonFLLOP allows more than one subdomain per core, i.e. an array of \mathbf{K}^s and \mathbf{f}^s is passed per subdomain. The achieved numerical, weak parallel, strong parallel and memory scalability are reported in [10].

8. Numerical Experiments

The numerical experiments with PERMON's SMALBE and MPRGP were run on the Salomon supercomputer [16] at IT4Innovations. Salomon consists of 1008 compute nodes, totaling 24192 compute cores with 129 TB RAM and giving over 2 PFLOPS theoretical peak performance. Each node is a powerful x86-64 computer, equipped with 24 cores, at least 128 GB RAM. Nodes are interconnected by 7D Enhanced hypercube Infiniband network and equipped with Intel Xeon E5-2680v3 processors. The Salomon cluster consists of 576 nodes without accelerators and 432 nodes equipped with Intel Xeon Phi MIC accelerators.

As a model of 3D linear elasticity contact problem, we considered an elastic cube with the bottom face fixed, the top one loaded with a given vertical surface force directed downwards, and the right one in contact with a rigid obstacle, see Fig. 6. Young modulus is $E = 2 \cdot 10^5$ [MPa] and Poisson ratio $\mu = 0.33$.

The action of the stiffness matrix pseudoinverse \mathbf{K}^\dagger was implemented using the LU factorization of the regularized \mathbf{K} using the MUMPS library and

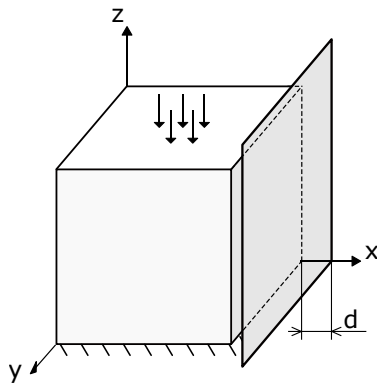


Fig. 6: 3D elastic cube contact problem.

SuperLU_DIST was used for the CP solution in MPI subcommunicators. Each subdomain was assigned to one core. The computations were carried out with decompositions into 512, 1024, 2048, 4096, 8192 and 10648 subdomains. The stopping criterion was defined by the relative precision of the projected gradient and the feasibility error equal to 10^{-4} and compared with the Euclidean norm of the dual linear term. The results of computations are depicted in Fig. 7.

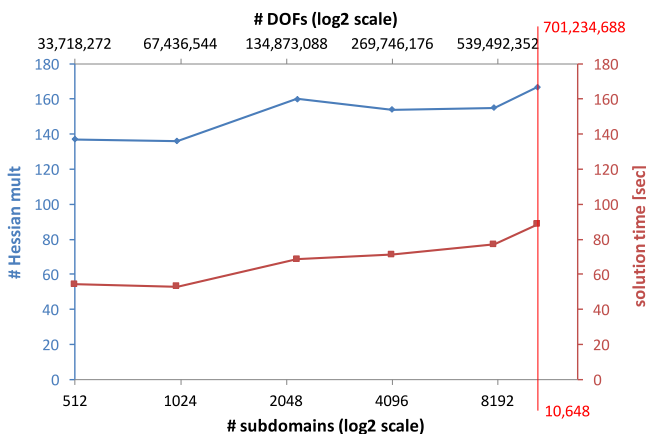


Fig. 7: Cube contact linear elasticity problem - numerical and weak parallel scalability highlight on Salomon.

9. Conclusion

We demonstrated the numerical scalability and weak scalability of the FETI based algorithm for contact problems using Salomon up to 702 millions of unknowns and 10648 subdomains (cores). The results document the efficiency of the implemented matrix formats, parallel direct solvers, and other ingredients (as e.g. adaptive expansion steplength in MPRGP). The results were obtained by means of TFETI + SMALBE + MPRGP implemented in PERMON toolbox.

Acknowledgment

The authors acknowledge the support of the Czech Science Foundation (GACR) project no. 15-18274S. The work was also supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project “IT4Innovations excellence in science - LQ1602” and from the Large Infrastructures for Research, Experimental Development and Innovations project “IT4Innovations National Supercomputing Center“ LM2015070”, and by the internal student grant competition project SP2017/169 “PERMON toolbox development III”.

References

- [1] DOSTAL, Z., T. KOZUBEK, M. SADOWSKA and V. VONDRAK. *Optimal Quadratic Programming Algorithms, with Applications to Variational Inequalities*. New York: Springer, 2009. ISBN 978-0-387-84806-8. DOI: 10.1007/b138610.
- [2] FARHAT, C., J. MANDEL and F-X. ROUX. Optimal convergence properties of the FETI domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*. 1994, vol. 115, iss. 3–4, pp. 365–385. ISSN 0045-7825. DOI: 10.1016/0045-7825(94)90068-X.
- [3] FLEMISH, B., Y. MADAY, F. RAPETTI and B. WOHLMUTH. Scalar and vector potentials’ coupling on nonmatching grids for the simulation of an electromagnetic brake. *The international journal for computation and mathematics in electrical and electronic engineering COMPEL*. 2005, vol. 24, iss. 3, pp. 1061–1070. ISSN 0332-1649. DOI: 10.1108/03321640510571246.
- [4] PERMON team. Parallel, Efficient, Robust, Modular, Object-oriented, Numerical. *PERMON* [online]. 2017. Available at: <http://permon.it4i.cz>.
- [5] HAPLA, V., D. HORAK, L. POSPISIL, M. CERMAK, A. VASATOVA and R. SOJKA. Solving Contact Mechanics Problems with PERMON. In: *High Performance Computing in Science and Engineering (HPCSE)*. Solan: Springer, 2015, pp. 101–115. ISBN 978-3-319-40361-8. DOI: 10.1007/978-3-319-40361-8_7.
- [6] Portable, Extensible Toolkit for Scientific Computation. *PETSc* [online]. 2017. Available at: <http://www.mcs.anl.gov/petsc/>.
- [7] PETSc Users Manual. *PETSc* [online]. 2016. Available at: <http://www.mcs.anl.gov/petsc/>.

- [8] VASATOVA, A., M. CERMAK and V. HAPLA. Parallel implementation of the FETI DDM constraint matrix on top of PETSc for the Permon-FLLOP package. In: *11th International Conference Parallel Processing and Applied Mathematics (PPAM)*. Krakow: Springer, 2016, pp. 150–159. ISBN 978-331932148-6. DOI: 10.1007/978-3-319-32149-3_15.
- [9] DOSTAL, Z., D. HORAK and R. KUCERA. Total FETI – an easier implementable variant of the FETI method for numerical solution of elliptic PDE. *Communications in Numerical Methods in Engineering*. 2006, vol. 22, iss. 12, pp. 1155–1162. ISSN 1069-8299. DOI: 10.1002/cnm.881.
- [10] SOJKA, R., D. HORAK, V. HAPLA and M. CERMAK. The impact of enabling multiple subdomains per MPI process in TFETI domain decomposition method. *Applied Mathematics and Computation*. Submitted in 2016. ISSN 0096-3003.
- [11] HAPLA, V., D. HORAK and M. MERTA. Use of Direct Solvers in TFETI Massively Parallel Implementation. In: *11th International Conference Applied Parallel and Scientific Computing (PARA)*. Helsinki: Springer, 2012, pp. 192–205. ISBN 978-364236802-8. DOI: 10.1007/978-3-642-36803-5_14.
- [12] VASATOVA, A., J. TOMCALA, R. SOJKA, M. PECHA, J. KRUIK, D. HORAK, V. HAPLA and M. CERMAK. Parallel strategies for solving the FETI coarse problem in the PERMON toolbox. In: *Programs and Algorithms for Numerical Mathematics 18*. Janov nad Nisou: Institute of Mathematics Czech Academy of Sciences, 2016, pp. 154–163. ISBN 978-80-85823-67-7. DOI: 10.21136/pam.2016.18.
- [13] MUMPS: A Parallel sparse direct Solver. *MUMPS* [online]. 2017. Available at: <http://graal.ens-lyon.fr/MUMPS/>.
- [14] LI, X. S., J. DEMMEL, J. GILBERT, L. GRIGORI, P. SAO, M. SHAO and I. YAMAZAKI. SuperLU. *SuperLU* [online]. 2014. Available at: <http://crd.lbl.gov/~xiaoye/SuperLU/>.
- [15] HORAK, D., Z. DOSTAL, V. HAPLA, J. KRUIK, R. SOJKA and M. CERMAK. Projector-less TFETI method for contact problems implemented in PERMON toolbox. In: *PARENG conference*. Submitted in 2017.
- [16] Hardware Overview. *SALOMON* [online]. 2017. Available at: <https://docs.it4i.cz/salomon/hardware-overview/>.

About Authors

David HORAK was born in Bohumin, Czech Republic. He has M.Sc. and Ph.D. degrees in Computer Science and Applied Mathematics and Associate Professor degree in Applied Mathematics at VSB–Technical University of Ostrava (VSB–TUO), 2007 Babuska price for Ph.D. thesis in Computer Sciences, since 2017 Associate Professor at Department of Applied Mathematics and since 2012 Researcher at IT4Innovations National Supercomputing Center, both at VSB–TUO. His professional interest covers development and implementation of the scalable quadratic programming solvers using domain decomposition methods of FETI type, integral-discrete transforms. He is co-author of PERMON library.

Vaclav HAPLA was born in Havirov, Czech Republic. He has been a researcher at IT4Innovations National Supercomputing Center since 2011. He is the main developer of the open source PERMON libraries based on top of PETSc. His research is focused on massively parallel implementation of scalable Domain Decomposition Methods (DDM) and Quadratic Programming (QP) algorithms, and applications of QP such as contact problems or machine learning via Support Vector Machines. He is the author of the PRACE video tutorials about PETSc (available on YouTube), and has lead full-day PETSc tutorials at PRACE Spring School 2012 in Krakow, Poland, at PRACE Autumn School 2016 in Linz, Austria. He was awarded two national prizes in the field of computational sciences: 1st prize in Joseph Fourier Prize 2014, and 3-4th prize in Babuska Prize 2016.

Jakub KRUIK was born in Slany, Czech Republic. He received his B.Sc. in Computational Mathematics at VSB–Technical University of Ostrava in 2016. Recipient of 2016 prof. Babuska price for student thesis in Computer Sciences. He studies Computational Sciences and works as a Research Assistant at IT4Innovations National Supercomputing Center. His research interests include quadratic programming, FETI and Krylov subspace methods.

Radim SOJKA was born in Frydek-Mistek, Czech Republic. He received his M.Sc. from Computational Mathematics in 2015 at VSB–Technical University of Ostrava. Since 2015, he is Research Assistant and Ph.D. student of study programme Computational science at IT4Innovations National Supercomputing Center, VSB–TUO. His research interests include optimization, e.g. quadratic programming and integer programming for Vehicle Routing Problem and parallel implementation of optimization methods.

Martin CERMAK was born in Hranice, Czech

Republic. He received his M.Sc. from Computational Mathematics in 2008. He finished Ph.D. in 2012 from Computational Mathematics at VSB–Technical University of Ostrava. Since 2012, he is Researcher at IT4Innovations National Supercomputing Center, VSB–TUO. His research interests include scalable algorithms for solving elastoplastic problems, material nonlinearities, contact problems, quadratic programming, non-overlapping domain decomposition and parallel implementation of these methods.

Marek PECHA was born in Zilina, Slovak republic. He has M.Sc. degree in Computer Science and Applied Mathematics at VSB–Technical University of Ostrava, Czech Republic (2016 Faculty of Electrical Engineering and Computer Science dean’s award for diploma thesis). Since 2016 until now Ph.D. student at Department of Applied Mathematics and since 2014 until now Research assistant at IT4Innovations National Supercomputing Center, both at VSB–TUO. His research of interest covers numerical analysis, machine learning, computer graphics, material sciences, and massively parallel programming. He is co-author of PermonSVM library for linear support vector machines problems, and also co-author of massively parallel version of PRAGTIC software for fatigue analysis.

Jiri TOMCALA was born in Frydek-Mistek, Czech Republic. Since 1998 till 2016 he worked as System Administrator at middle-sized building company, then he received M.Sc. degree in Computer Science and Applied Mathematics at VSB–Technical University of Ostrava and since 2016 until now he works as Researcher at IT4Innovations National Supercomputing Center, VSB–TUO. His professional interest covers C/C++ programming, development and implementation of the scalable solvers using domain decomposition methods of FETI type. He is co-author of PERMON library for the massively parallel solution of quadratic programming problems.

Zdenek DOSTAL was born in Olomouc, Czech Republic. He received his M.Sc. from Computer Science in 1970. He finished Ph.D. in 1979 in Mathematical Institute of the Czechoslovak Academy of Science, Prague, Associate Professor since 1992 from Charles University, Prague, Full Professor at VSB–Technical University of Ostrava since 1994, and DSc. from the Academy of Science of Czech Republic, Prague 2005. His research interests include numerical linear algebra, optimization, and computational mechanics.