Abstract

Mechanized tunneling is an established construction technology in particular in urban environments, which involves interactions between the tunnel boring machine (TBM), the ring-wise installed linings, the tail void grouting and the surrounding soil including the groundwater as well as existing buildings. Evidently, 3D computational simulations of the machine driven tunnel construction process requires time consuming preparation and excessive computing resources. In this work, a parallelization strategy is applied to perform large scale simulations of the advancement process in mechanized tunneling. To support the generation of realistic simulation models, the Building Information Modeling (BIM) concept is employed. In this approach, different simulation components in terms of geometrical representation, material modeling and process modeling are stored in the database, and can be selected on demand, depending on the objective of the analysis. In the highest modeling level, all components involved in the TBM advancement in urban environments, namely the soil, the tail-void grouting, the segmented lining, and the existing buildings are considered. The parallelization strategy employs the domain decomposition technique to decompose the large model into sub-domains. The contact between the TBM and the soil is resolved by using the surface-to-surface, so called mortar contact formulation. The spatial-temporal system is discretized using an LBB-compatible mixed formulation for space discretization and the generalized-α method for temporal discretization, leading to a linear system of block structure. This system is then solved iteratively by Krylov subspace method and a block preconditioner to speed-up the convergence. Selected numerical benchmarks are presented to showcase the effectiveness of the employed parallelization strategy for mechanized tunneling using a BIM concept.

Keywords: Mechanized tunneling, Parallel contact, Block-preconditioning, BIM.

1 Introduction

For the process oriented computational modeling of mechanized tunneling in soft soils, the different stages of the construction process (excavation, advancement of the tunnel boring machine (TBM) and installation of the linings) as well as the various interactions between the tunneling process and the environment needs to be considered. This includes the multiphase characteristic of the soil, modeled in the framework of poromechanics, the consideration of contact between the TBM and the soil, the ring-wise installation of the tunnel lining and the modeling of the pressurized tail void grouting. In the framework of the Collaborative Research Center SFB837 at Ruhr University Bochum, a process-oriented finite element model, which accounts for all relevant components and interactions has been developed (see, e.g. [1, 2, 3, 4, 5] for details). Considering the scale of the tail void gap in the range of $\approx$ 30- 50 cm and the scale of the complete analysis section in the range of several hundred meters, a sufficiently refined problem specific spatial resolution of the project section requires a large number of degrees of freedom. Furthermore, the staged construction process and the groundwater flow in the fully saturated ground requires also a fine temporal resolution. In addition to the strongly nonlinear character of the problem, which needs to be solved iteratively, these aspects put high demands on the spatio-temporal discretization and the computational methods used for the numerical simulation. In the usual practice, the geometrical model and the finite element mesh of the TBM and various components of the simulation model are constructed beforehand, based on design data, to provide as the input for the finite element solver. In case, that the design needs to be modified during the planning phase of a tunneling project, the re-generation of the discretized model is extremely time-consuming for such complex computational models. This leads to the demand for an efficient and user-friendly tool to automate the geometrical modeling process and manage construction data for large scale simulation.

The Building Information Modeling concept (BIM) is a vital tool for decision making process during the life cycle of infrastructure projects. Particularly, the information of complex, integrated, multi-disciplinary systems such as mechanized tunneling process can be well managed using BIM [6, 7]. In current engineering practice, the proof of design is often carried out by employing numerical simulation, which is usually generated based on design documents and reports. However, even if the underlying information needed for numerical analysis is stored in a BIM, the translation from information to computational model is still dominated by manual-editing process. Such an approach therefore incurs significant work carried out by experts, and is furthermore susceptible to human error. Hence, it is evident that an automated link between information management (in the form of a BIM) and numerical analysis is necessary. Such a link will enable the continuous, error-free exchange of information between BIM and numerical simulation from the design stages of a project over the construction phase to operation. Recently, examples for this integrated approach are presented in [8, 5, 9], where data obtained from a Tunnel Information Model (TIM) [10] is used for the automated generation of a numerical model for a real-world tunneling project, the Wehrhahn metro line in Düsseldorf (WHL).

With the geometrical data contained in BIM at hand, the large model resulting from automated modeling of typical tunnel projects is discretized, decomposed and solved by parallel techniques. These parallel techniques are presented partially in [2, 11]. Additional techniques, such as new type of block preconditioners and compatible decomposition of contact domains, are discussed briefly in Section 4. For an overview of the governing equations and discretization, the reader is referred to Section 3, and Section 5 for an overview of software used for the simulation. The paper is concluded by selected numerical benchmarks showcasing the recent simulation results obtained for a synthetic tunnel problem and the Wehrhahnline tunnel project in Dusseldorf, Germany.

2 Geometrical modeling using BIM-based approach

In this section, two different approaches of numerical modeling based on the BIM technology are presented. The first one is based on multi-level modeling concept developed within the state-of-the-art design environment, and the second one adopts the existing BIM database as the basis for automatic modeler. Both approaches address the ease and modularity of the generation process. The overview of each approach will be presented in subsequent sections.

2.1 Multi-level modeling with different Level of Details (LODs)

One of the challenges in the process of optimization of a project design is to preserve the consistency between the design alternatives and analysis among different sub-models and throughout different phases. This can be efficiently achieved by using an integrated design-analysis framework, where the numerical simulations are automatically generated based on the geometry and semantics stored in the BIM as illustrated in Figure 1. In addition, this also helps to minimize the time needed for model generation and computation.

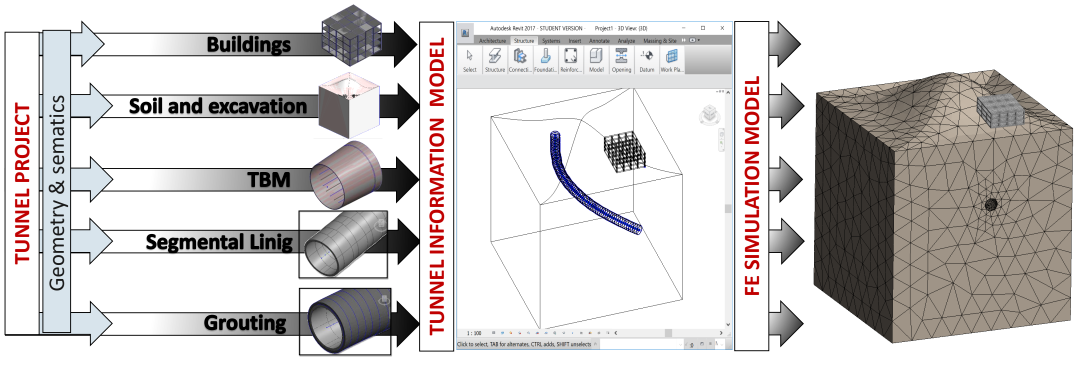

Figure 1: Generation of numerical simulation based on design in BIM framework

A characteristic of major tunneling projects is that they require analysis and design on vastly varying scales, from the kilometer scale (for the track alignment) to the centimeter scale (e.g. for detailed design of connection points). To carry out those tasks with high consistency when changing scales and modeling the components on different Levels of Details (LoDs), a sound foundation for handling such multi-level representations is proposed in [12].

Parametric representations for each of the system components (soil with excavation, tunnel lining with grouting, tunnel boring machine (TBM) and existing buildings) are developed on different LoDs. Figure 2a exemplifies the multi-level approach for two system components (building and tunnel lining).

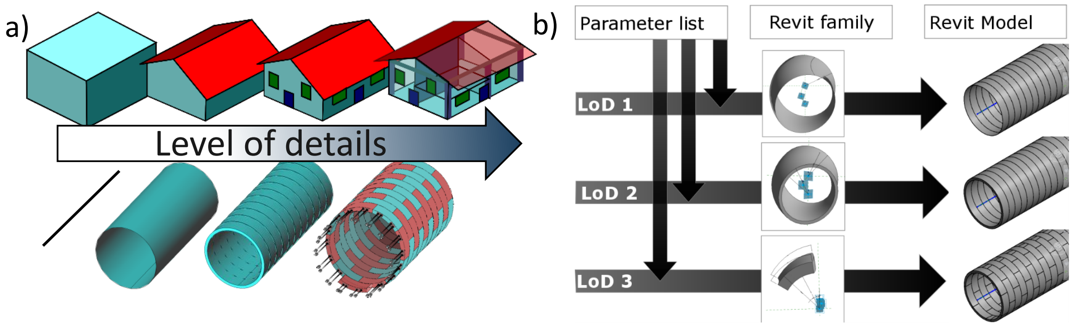

Figure 2: Multi-level representation of geometry: a) different LoDs for the representation of buildings and lining structure; b) parameter consistency between different LoDs for individual components

In order to keep consistency between different LoDs, the parametric dependency between classes representing different LoDs of the same component is defined as shown in Figure 2b. For each LoD of a specific component, a class with its attributes (parameters) and constraints is implemented, allowing to define specific attributes on individual model instances (objects), which represents the actual objects of a BIM model. The full set of parameters defining a component is needed for the definition on the highest LoD, while only a subset of the parameter list is used on lower LoDs.

2.2 Tunnel Information Model (TIM)-based modeling

The other approach to generate the model effectively using automatic modeler is to employ a Tunnel Information Model (TIM), which is a specialization of BIM for modeling tunneling scenarios. In this modeling scheme, the TIM automatic modeler is capable of performing the following tasks:

Acquisition of geometry and geology The topology of both the geological layers and the tunnel, in addition to the geometrical boundaries (simulation domain) must be defined. This information is imported from the TIM to the modeler as volumetric data with a Non-uniform rational basis spline (NURBS) in ACIS format. ACIS is a geometric modeling kernel used to enable robust and 3D modeling capabilities of Computer-aided design (CAD) software. The tunnel geometry can be either imported using the geometrical data for each individual ring, or as simply a tunnel alignment and a ring thickness and diameter. Initially the modeler stores these geometries separately.

Acquisition of input parameters Process parameters (e.g. face support pressure and grouting pressure) and material parameters (i.e. for soil, grouting, and lining stiffness) are required to generate the simulation model. This information is available in the TIM.

Generation and execution of the numerical simulation After reading all required data, the modeler combines the different geometries into one simulation and assigns the boundary conditions and the material parameters to their corresponding elements and generates the finite element mesh automatically. A simulation script is generated and executed.

Transmission of the simulation results to the TIM The database of the TIM will finally store all results in a structured way for several purposes, e.g. 3D visualization, validation and interaction with other sub-models of TIM.

To create a model using the TIM-based approach, engineers use a web-based service to communicate with the modeler. The results are then able to be accessed via online transferring or offline copying. This approach has the advantages that the computer resources to generate the model is centralized. In addition, the user can work and access the automatic tools across geographical locations. For an overview of the TIM-based modeling approach, the user is referred to Figure 3

3 Governing equations and finite element discretization

In this section, the governing equations describing the underlying physics of the particular problem at hand is briefly presented. The emphasis is given to additional terms representing the interaction between the buildings, the TBM and the soils. The strong form of the governing equation for soil is well described in [13], hence only the weak, discretized form of the governing equations is discussed. Following that, the variation of the total energy of the system is characterized by

$$\delta W = \delta W^s + \delta W^w + \delta W^c + \delta W^t \tag{1}$$

The reader is referred to [14, 15, 13] for the details description of the spatial discretization of system equilibrium δW^s, the fluid-mass balance equation δW^w and [16] for a short overview of the discretization for the virtual work δW^c of the contact force.

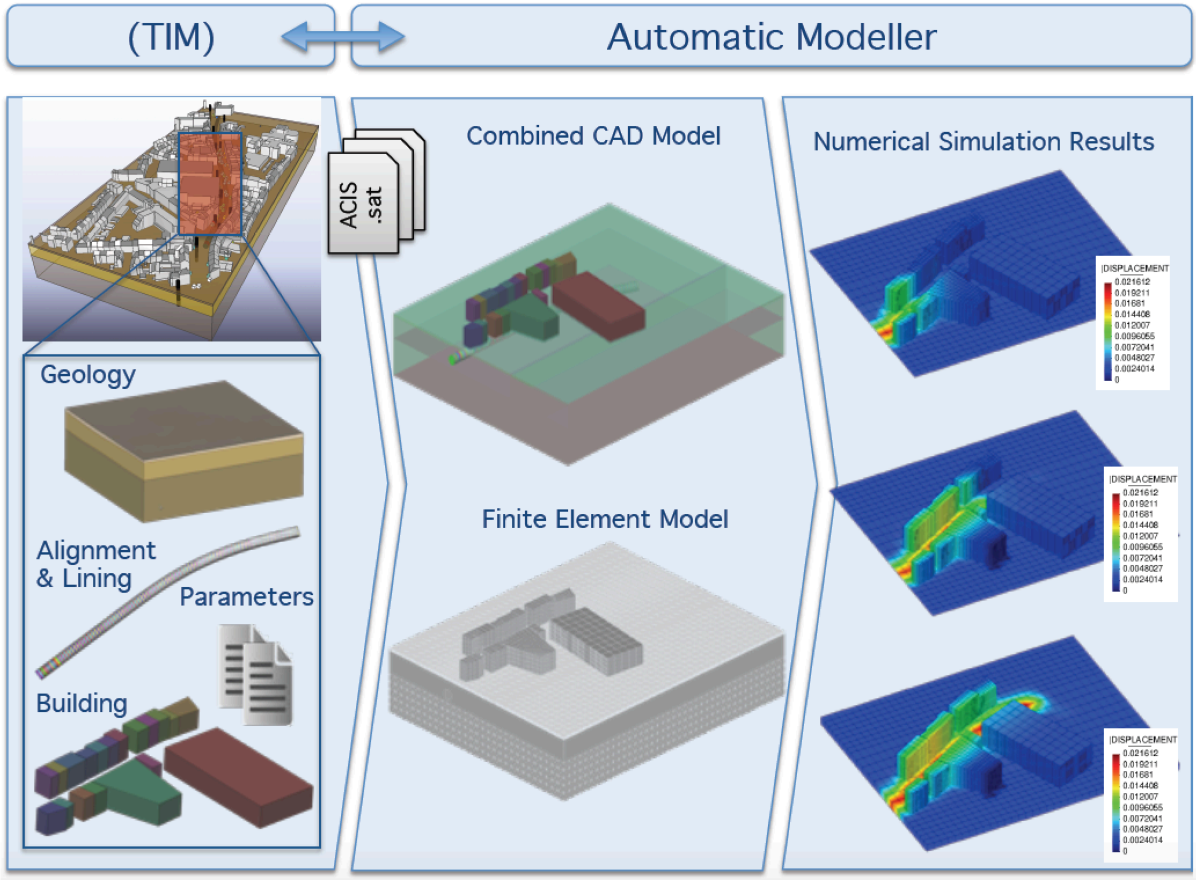

Figure 3: Scheme of the Tunnel Information Model (TIM) interacting with the automatic modeler

In analyses at higher level of detail, contact conditions are considered along the interface between the moving TBM and the surrounding soil. The interaction between the buildings and the soil is represented by mechanical tying of nonconforming domains. Using the penalty method, the soil-building interaction is considered by the contribution δW^t to the total virtual work of the system

$$\delta W^t = \int_{\Gamma^t} \rho_t (\delta \mathbf{u}^{(1)} - \delta \mathbf{u}^{(2)}) \cdot (\mathbf{u}^{(1)} - \mathbf{u}^{(2)}) dV \tag{2}$$

After spatial discretization by means of finite elements, the expression (1) constitutes the nonlinear weak form in terms of two sets of primal nodal variables, the displacement degree of freedom (d.o.f) and water pressure d.o.f. After linearization, one obtains straightforwardly an equation system in the format

$$\begin{bmatrix} \Delta u^s \\ \Delta p^w \end{bmatrix}_{n+1} = \begin{bmatrix} \mathbf{A}_{ss} & \mathbf{A}_{sw} \\ \mathbf{A}_{ws} & \mathbf{A}_{ww} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{R}^s_{ext} - \mathbf{R}^s_{int} \\ \mathbf{R}^w_{ext} - \mathbf{R}^w_{int} \end{bmatrix}, \tag{3}$$

where R^s_{ext} - R^s_{int} (R^w_{ext} - R^w_{int}) denote the residuals between the external and internal force vectors related to the mechanical equilibrium (index s) and the mass balance of the groundwater (index w). It is noted that, to make the discretization in Equation (1) LBB-compatible, the water pressure degree of freedom (d.o.f) is approximated using a functional space of lower order than of displacement d.o.f. In the numerical examples presented in Section 6, tetrahedral finite elements with 10 nodes for displacements and 4 nodes for the pressure is employed.

4 Computation and parallelization strategy

Application of a domain decomposition strategy for tunnel advancement simulations must account for the fact, that the contact surface, i.e. the interface between the TBM and the soil, is continuously changing to adapt with the new position of the TBM. The contact surfaces are identified from a contact search algorithm based on bounding volume hierarchy (BVH). In typical scenarios, when the TBM passes through the mesh boundary between computation domains, the contact search algorithm may fail since some of the master segments do not belong to the current working process. To cope with this issue, the master segments are distributed to all domains as ghost segments (see Figure 4 for an illustration of the ghost master segments distributed to the domain containing the TBM). It imposes additional memory overhead to store those ghost segments in the memory. However, this is negligible compared to the number of elements and boundary conditions of the complete system. On the rest of the domain, the system is decomposed by using the multilevel k-way partitioning algorithm from Metis package [17].

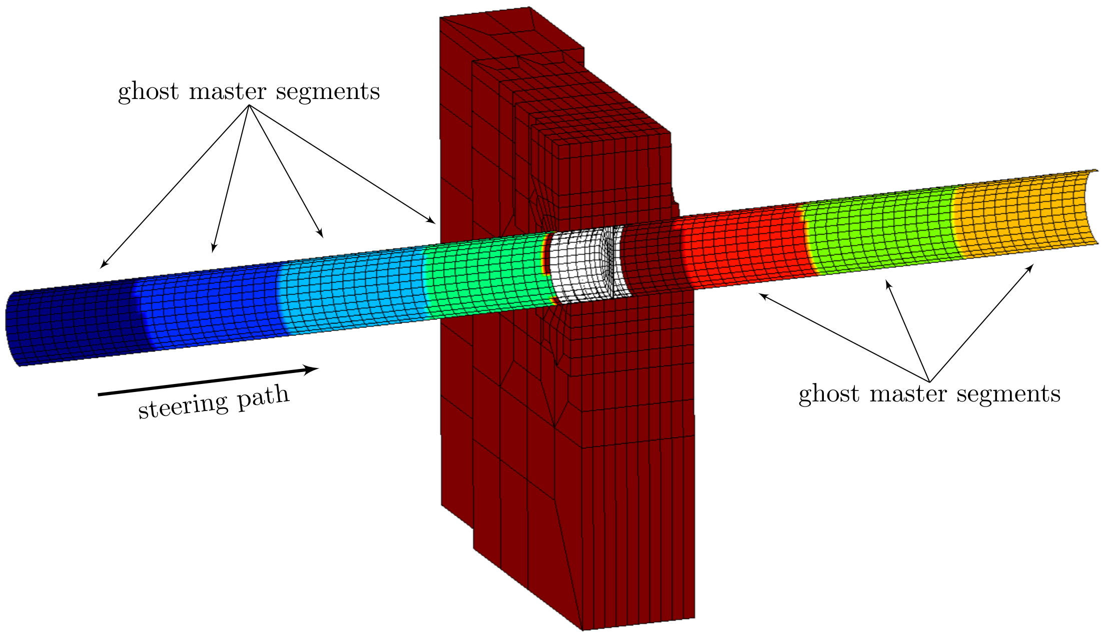

Figure 4: Ghost master segments in TBM partition

To solve the block system Equation (3) effectively using an iterative solver, the block preconditioner of the form

$$P^{-1} = \begin{bmatrix} I & 0 \\ 0 & -S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{ws} & I \end{bmatrix} \begin{bmatrix} A_{ss}^{-1} & 0 \\ 0 & I \end{bmatrix} \tag{4}$$

is employed. The Schur preconditioner S is approximated by $S = A_{ww} - A_{ws} diag(A_{ss}^{-1}) A_{sw}$ to retain the sparsity. It is computed more efficiently as compared to the dense counterpart. The quality of this block preconditioner depends substantially on the quality of the sub-preconditioners, which are used to invert the sub-block A^{-1} and S^{-1}. In the numerical example, the multigrid preconditioner [18] is chosen as sub-preconditioners. The setting for the multigrid preconditioner is chosen by manual calibrating and comparing the number of required iterations per each solution step.

5 Overview of modeling and computational software

The geometrical modeling framework for BIM-based numerical analysis used in this paper is SATBIM, an integrated platform for information modeling, structural analysis and visualization of the mechanized tunneling process for design support [12]. Within this platform, the industry-standard Autodesk tools Revit and Dynamo [19] are employed for the design of the tunnel structure and the surrounding infrastructure with consideration of different LoDs for the representation of all system components (soil, lining, TBM, infrastructure). For each tunnel component and on each LoD, a so-called "Revit family" for the corresponding component is created, preserving the geometric consistency between the levels as described in Figure 2b.

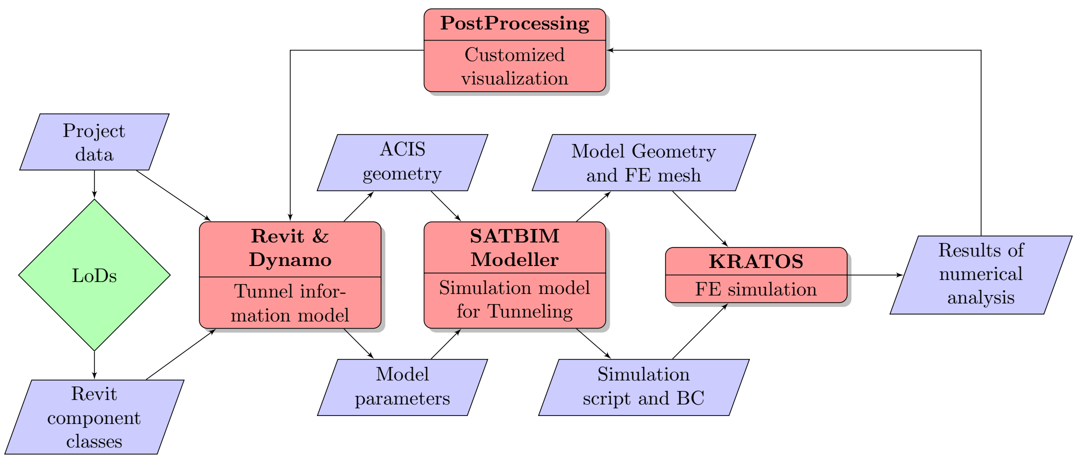

Figure 5: Data flow in SATBIM.

Based on a multi-level parametric TIM, numerical models for each component on different LoDs are developed, considering proper geometric as well as material representation, interfaces and the representation of the construction process. A fully automatic `SatBimModeller` for arbitrary tunnel alignments provides a high degree of automation for the generation, the setup and the execution of the simulation model, connecting the multi-level information model with the computational software. Finally, the geometrical modeling software allows for incorporation of simulation output back into Revit for further analysis of the soil-structure interaction effects induced by tunneling. The outline of the structure of SATBIM and the exchange of data is shown in Figure 5.

To compute and assemble the system in Equation (1) in parallel, a set of computational code is written on top of KRATOS [20], a multiphysics framework for finite element simulation. The structure of the code is illustrated in Figure 6. The central part

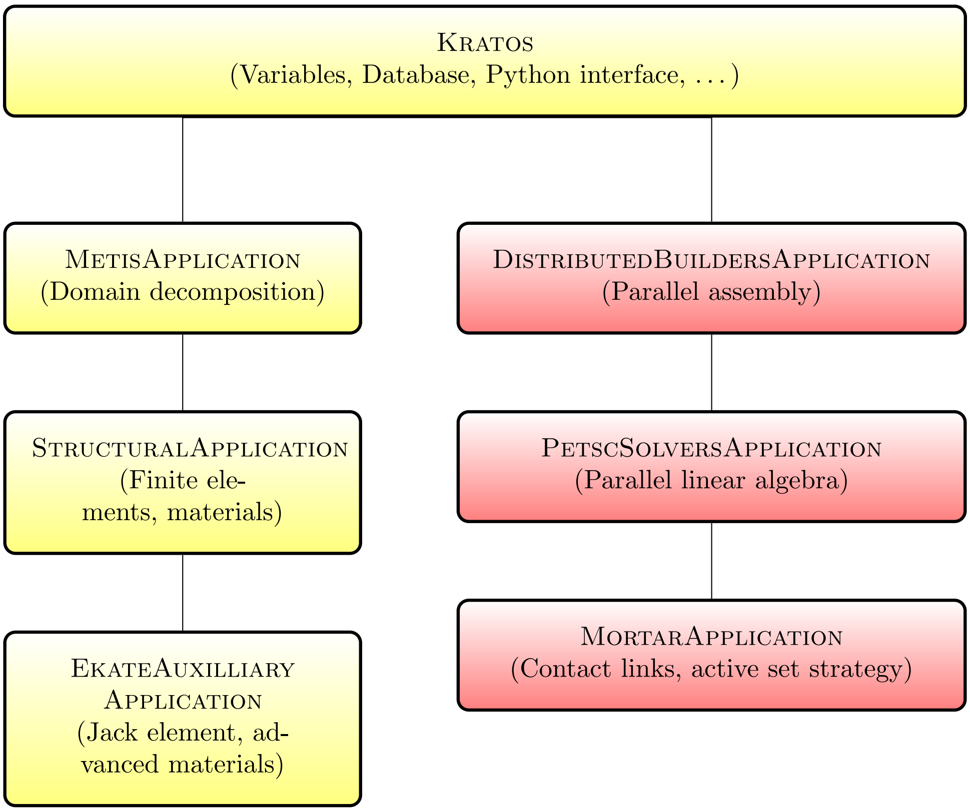

Figure 6: Simulation code structure, left: existing KRATOS components, right: extension code for parallel simulation

of the code is the KRATOS extension `DistributedBuildersApplication`, which manages the parallel data flow, and provides the interface for connection with various parallel solvers. The subroutines handling the contact constraint computation is implemented in `MortarApplication`. Finally, the `PetscSolversApplication` provides the interface to solvers and preconditioner in `PETSc` package [21, 22].

6 Numerical examples

In the conceptual phase, a designer often only needs approximate estimations for many different scenarios, for example tunnel track alternatives. To ensure a seamless work-flow, the computation time should be minimized. If preliminary analysis indicates the potential for hazards, a more detailed evaluation of the model is required. Therefore, in the following examples, models accounting for different levels of complexity are investigated. In numerical examples 1 and 2, the SATBIM framework described in Section 5 is used for generation of the large tunnel section combining components on different LODs, as shown in Figure 7. Based on this Revit model, a simulation model is generated using `SatBimModeller`. In both examples, the highest LoD for the representation of the overlaying structure and a medium LoD for the representation of the soil is selected.

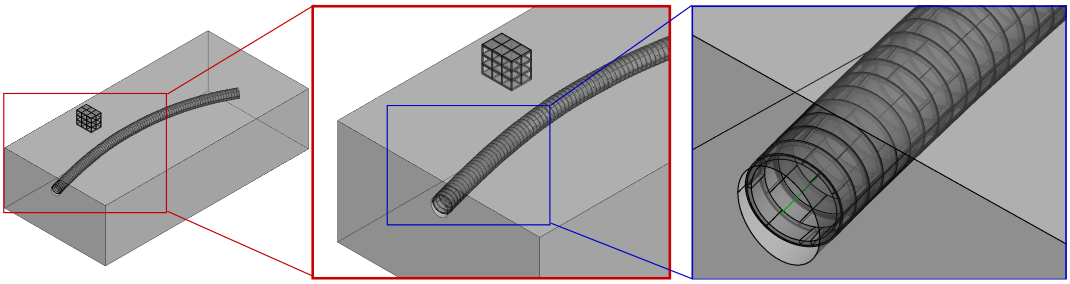

Figure 7: TIM for a more than 200 m long tunnel section used for generation of a large-scale simulation

6.1 Numerical example 1

In the first example, a simple representation of a machine driven tunnel construction is modeled with the so-called volume loss method. In this method, the support by the tunnel shell is modeled without explicit consideration of the lining structure. Instead, the confinement is described with the volume loss coefficient $V_l = (V_0 - V_{def})/V_0 \cdot 100\%$. In the implementation of the volume loss method, after the de-confinement due to soil excavation, the deformed area of the tunnel is continuously calculated in each computation cycle, and deformations of the excavation boundaries are fixed when the volume loss value of the tunnel boundary is reached [23].

The material parameters for soil used in the examples 1 and 2 are listed in Table 1. Considering sand-like soil conditions, the Mohr-Coulomb model is used as the constitutive relation for the effective stress and strain in the soil. In this work, associative plasticity and a linear hardening law are assumed.

The computation mesh is characterized by 148,650 tetrahedral elements for the ground. This results in 211,272 nodes and 635,356 d.o.fs during the tunnel advance-

Parameter	Value
Constitutive law	Mohr-Coulomb
Young modulus	100 MPa
Poisson ratio	0.25
Density	1732 kg/m^3
Porosity	0.4
Cohesion	200 kPa
Hardening modulus	5.83 MPa
Friction angle	30
Dilatancy angle	30

Table 1: Material parameters for example 1 & 2

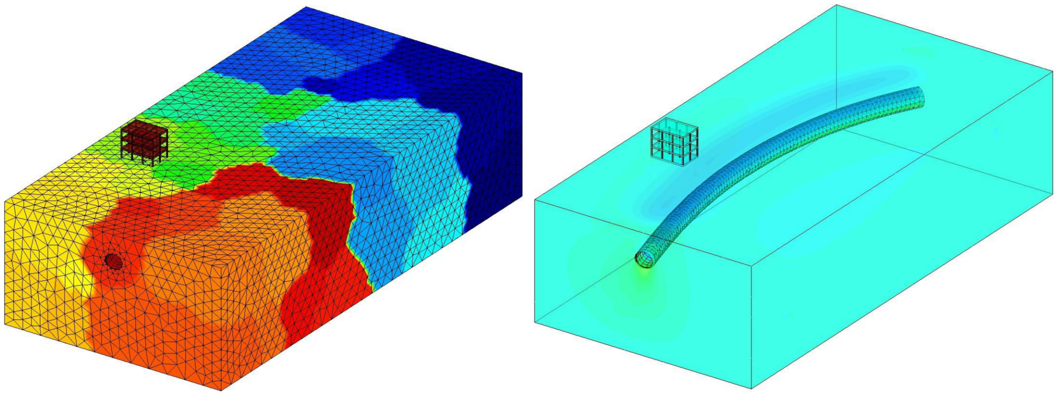

Figure 8: Left: decomposition of computing domain (32 sub-domains), Right: ground settlement at final computing step ($t = 79.4$ hours).

ment. The decomposition of the computing domain is shown Figure 8, left. The resulting ground settlements at the final computing step are presented in Figure 8.

Type of simulation	Average solution time	Average number of iterations
PARDISO (16 threads)	126.48	1
GMRES (4 processes)	28.21	34.2
GMRES (8 processes)	14.82	34.6
GMRES (16 processes)	10.26	34.7
GMRES (32 processes)	10.09	34.8

Table 2: Example 1: Comparison of the speed-up of the parallel solver with the direct solver PARDISO.

To understand the efficiency of a parallel solver, one needs to compare its performance with the direct solver counterpart. For this medium-sized problem, the direct solver (here we employ the PARDISO solver) is adequate to solve the resulting linear system by using 8 processes on a single computer with 256 GB of memory. A series of parallel simulations with 8, 16 and 32 processes is run to compare the average solving time per time step. The results are shown in Table 2. The results for the computation

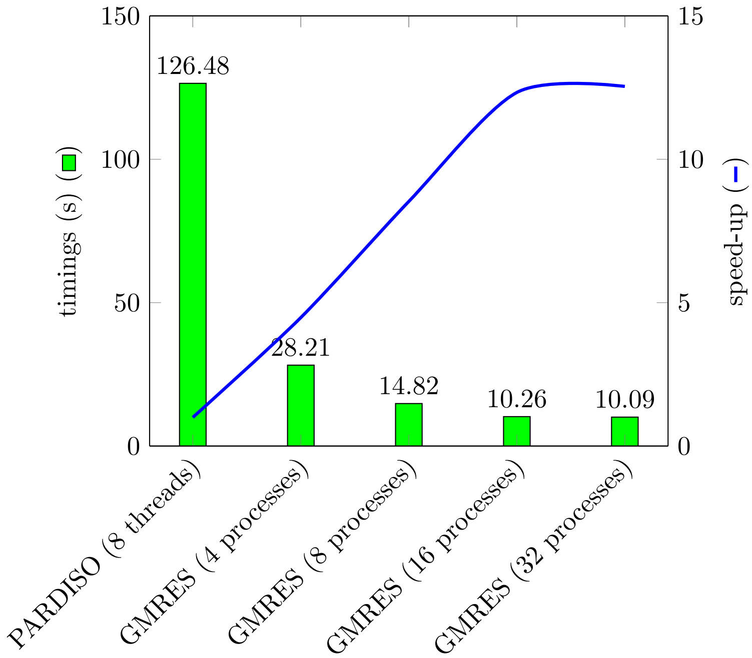

Figure 9: Example 1: Speed-up of the parallel and direct solver.

time in Table 2 and Figure 9 show that the parallel solver outperforms direct solver even when using the same number of physical computing cores. The speed-up starts to degrade at 16 processes and saturates at 32 processes. Neglecting the communication overhead, since the simulations are run with in-core MPI, it is clear from this example that starting from 16 processes, the time to compute the preconditioner begins to dominate and no additional speed-up can be generated.

6.2 Numerical example 2

In the second example, a higher level of definition (LoD 2) for the representation of the tunnel lining structure and the TBM is selected. This model accounts for the shield as a deformable body moving through the soil and interacting with the ground through surface-to-surface contact. The tunnel advance is modeled by means of deactivation of soil elements and installation of the tunnel lining and grouting elements. The annular gap between the segmented lining tube and the excavation boundary is refilled with grout, modeled as a fully saturated two-phase material, considering the time-dependent evolution of stiffness and permeability. Tunneling-induced deformations are controlled by applying the face support pressure and the grouting pressure at the tunnel face and in the steering gap, respectively.

The same tunnel section using the same material parameters as before is analyzed. In contrast to the previous example, the TBM and the annular grouting are enabled,

Type of simulation	Average solution time	Average number of iterations
GMRES (4 processes)	268.05	166.1
GMRES (8 processes)	141.85	166.8
GMRES (16 processes)	69.61	165.7
GMRES (32 processes)	59.23	167.1

Table 3: Example 2: Speed-up of parallel solver relative to the lowest number of computing processes ($n = 4$) using 4 to 32 processors.

resulting in an increase of the number of elements in the mesh. Furthermore, a finer spatial discretization of the soil is employed, which now contains 242,840 tetrahedra elements and 349,070 nodes. This amounts to a maximum of 1,030,652 d.o.fs during the advancement process. The penalty parameter used to enforce the contact constraint between the TBM and the soil is chosen as 1 MPa. The contact surfaces properties are considered frictionless. The penalty parameter used to connect the building to the ground is chosen as 100.0 GPa. The results of the computation, in terms of speed-up and number of average iterations of the parallel solver, are presented in Table 3 and Figure 10.

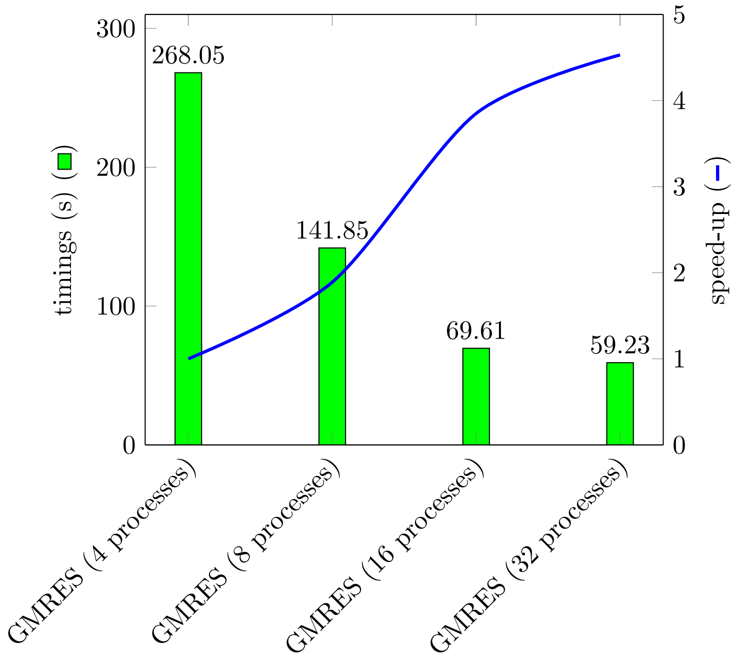

Figure 10: Example 2: Speed-up of parallel solver using 4 to 32 processors.

In this example, the multithreaded direct solver is not used, mainly due to the fact that the factorization matrix does not fit in the system memory. Nevertheless, the simulation is performed on four processes first. Subsequently, the speed-up from the computations using a larger number of processes is measured relative to the simulation using four processes. The results in Figure 10 show a similar characteristic as the results in Figure 9. Again, the speed-up starts to saturate after 16 computing processes.

6.3 Numerical examples 3

In terms of model complexity, the numerical examples 1 and 2 are only considering one building at the surface. To fully analyze the interaction between the tunnel advancement process and existing built environment in urban tunneling, a more realistic scenario must be considered, in which all relevant surface structures must be incorporated into the simulation model. Therefore, in this numerical example, a refined simulation model based upon a section of the Werhahn-Line tunnel project in Düsseldorf , Germany, which contains the complete building infrastructure along the alignment of the tunnel. This model is generated with the aid of the TIM-based service presented in Section 2.2. A snapshot of the model and the ground settlement at different time steps are illustrated in Figure 11. The buildings are considered in terms of substitute block-type structures with an equivalent stiffness. The stiffness of the individual buildings is calibrated and computed based on empirical models and measurement data [24].

A finite element mesh of 745,718 nodes and 409,368 tetrahedra elements is used to discretize the model. This results in a maximum of total 1,659,879 d.o.fs during the advancement process. The soil properties in this particular tunnel project possesses high permeability. Therefore, drained conditions may be assumed in this case, and the explicit consideration of the groundwater is not necessary. In this case, the model only contains displacement degrees of freedom. The water pressure and excess pore water pressure in the soil will be computed in the post-simulation phase. In regards to the level of detail of the advancement process, a lower LoD is used here. In this example, the TBM, and therefore, the contact conditions between the shield machine and the soil are not considered in this particular example.

Type of simulation	Average solution time	Average number of iterations
GMRES (8 processes)	711.73	354.4
GMRES (16 processes)	413.08	348
GMRES (32 processes)	256.39	355.9
GMRES (64 processes)	235.22	350.6

Table 4: Example 3: Performance of iterative solver in terms of computation time and number of iterations of the iterative solver.

Table 4 contains the average computing time for analyses using 8, 16, 32 and 64 processes. It is remarkable that the incorporation of the city model substantially slows down the computation of the tunnel-ground interaction. According to Table 4, the average time for solution of the computational simulation using 64 processes is 235.22s, which is slower than typical tunnel simulations, although the multiphase model for the soil and contact between the TBM and the soil is not enabled. Most probably, the enforcement of the deformation compatibility between the soil and the buildings using the penalty method is one of the reasons as it strongly affects the condition of the tangent matrices. However, the incorporation of multiple buildings and its effect on the

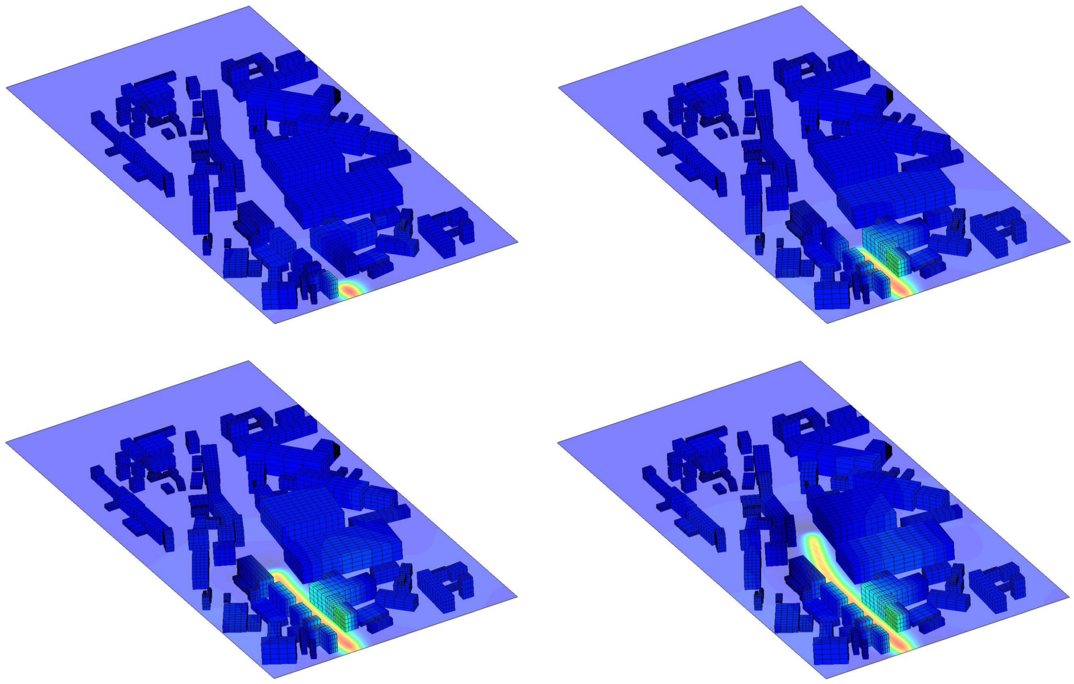

Figure 11: Example 3: Snapshots of the ground settlement at four different advancement stages (t=16, 61, 106, 151 hours).

performance of iterative solver deserves further investigation in the future. It should be noted, that the block preconditioning procedure with multigrid sub-preconditioners performs stable in this case. The number of iterations of the iterative solver does not differ much when more computing processes are used. The results in Table 4 and the speed-up are visualized in Figure 11. The maximum speed-up obtained with 64 computing processes compared with 8 computing processes is 3.03.

7 Conclusions

In this paper, recent results from large scale computational simulations of shield driven tunnel advancement in urban areas using iterative solvers have been presented. The simulation model has been generated by a BIM-based approach, allowing for the systematic consideration of different levels of detail. Being compatible with commercial CAD software, this approach is more intuitive to be used by engineers, and yet offers flexibility and modularity in model generation and realization. This model is tightly coupled with the parallel simulation tool to enable large scale simulation on demand, and to shorten the preparation time from model creation to parallelized computational analysis.

The results for the speed-up for large tunnel advancement simulations, in which the groundwater and the contact between the moving tunnel boring machine and the soil is considered, is satisfactory up to 16 processes for small cases (numerical examples

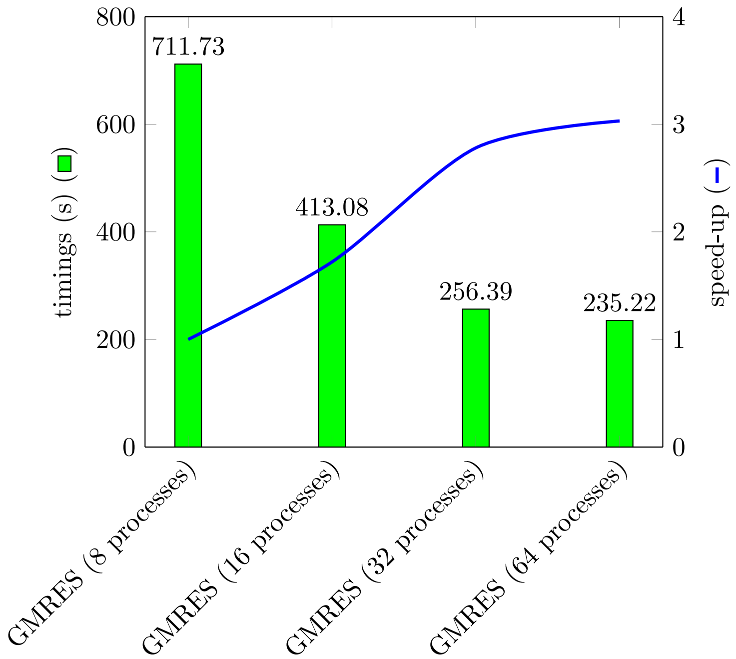

Figure 12: Example 3: Speed-up of the parallel iterative solver.

1 & 2) and up to 32 processes for the larger case, characterized by 1,6 Mio degrees of freedom (numerical example 3). Beyond that, the parallel simulation yields no more speed-up.

In the last example, a relatively complex model of the existing buildings has been integrated in the tunneling simulation. Most likely, the mesh tying technique, used to connect the buildings to the ground degrades the quality of the resulting preconditioning matrix. However, the effect of connecting large number of buildings to the simulation model deserves further investigation. Nevertheless, it was observed, that the multigrid sub-preconditioner yields a robust and predictable performance in terms of average number of required iterations over different number of computing processes.

In the future, the modeling and computational scheme outlined in this paper will be applied to simulation of further reference tunnel projects, with reallistic surrounding infrastructures. The results will help to understand the soil-structure interaction during TBM excavation, and to provide meaningful prediction for optimization of construction process parameters.

Acknowledgments

This presented work was conducted in the framework of the Collaborative Research Project SFB 837 ”Interaction Modeling in Mechanized Tunneling”, financed by the German Research Foundation (DFG). The authors would like to thank the DFG for the support of this project.

SATBIM project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 702874. The third author would like to gratefully acknowledge this support.

References

[1] H.G. Bui, J. Stascheit, G. Meschke, "A Parallel Block Preconditioner for Coupled Simulations of Partially Saturated Soils in Finite Element Analyses", in B. Topping, P. Iványi (Editors), *The Third International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering*. Civil-Comp, 2013.

[2] H.G. Bui, G. Meschke, "Study on Performance of Parallel Solvers for Coupled Simulations of Partially Saturated Soils in Tunnel Engineering Application", in *The Fourth International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering (PARENG 2015)*, 2015.

[3] F. Nagel, J. Stascheit, G. Meschke, "Process-oriented numerical simulation of shield tunneling in soft soils", *Geomechanics and Tunnelling*, 3(3): 268–282, 2010.

[4] A. Alsahly, J. Stascheit, G. Meschke, "Advanced finite element modeling of excavation and advancement processes in mechanized tunneling", *Advances in Engineering Software*, 100: 198 – 214, 2016.

[5] J. Ninic, S. Freitag, G. Meschke, "A hybrid finite element and surrogate modelling approach for simulation and monitoring supported TBM steering", *Tunnelling and Underground Space Technology*, 63: 12–28, 2017.

[6] A. Borrmann, M. Flurl, J.R. Jubierre, R.P. Mundani, E. Rank, "Synchronous collaborative tunnel design based on consistency-preserving multi-scale models", *Advanced Engineering Informatics*, 28(4): 499 – 517, 2014.

[7] M. König, J. Teizer, F. Schley, K. Kessoudis, *Taschenbuch für den Tunnelbau 2017*, Chapter Building Information Modeling (BIM) im maschinellen Tunnelbau, pages 199–232, Number 978-3-433-03168-1. Wilhelm Ernst & Sohn, 2016.

[8] G. Meschke, S. Freitag, A. Alsahly, J. Ninić, S. Schindler, C. Koch, "Numerische Simulation maschineller Tunnelvortriebe in innerstädtischen Gebieten im Rahmen eines Tunnelinformationsmodells", *Bauingenieur*, 89(11): 457–466, 2014.

[9] A. Alsahly, V.E. Gall, A. Marwan, J. Nini, G. Meschke, A. Vonthron, M. Knig, "From Building Information Modeling to Real time Simulation in Mechanized tunneling", in *Proceedings of the World Tunneling Congress*, 2016.

[10] J. Amann, A. Borrmann, F. Hegemann, J. Jubierre, M. Flurl, C. Koch, M. König, "A Refined Product Model for Shield Tunnels based on a Generalized Approach for Alignment Representation", in *Proc. of the 1st ICCBEI 2013*, 2013.

[11] J.A. White, R.I. Borja, “Block-preconditioned Newton–Krylov solvers for fully coupled flow and geomechanics”, *Computational Geosciences*, pages 1–13, 2011.

[12] J. Ninić, C. Koch, “Parametric multi-level tunnel modelling for design support and numerical analysis”, in *4th International Conference on Computational Methods in Tunneling and Subsurface Engineering (EURO:TUN 2017)*. Innsbruck University, Austria, 2017.

[13] F. Nagel, G. Meschke, “An elasto-plastic three phase model for partially saturated soil for the finite element simulation of compressed air support in tunnelling”, *International Journal for Numerical and Analytical Methods in Geomechanics*, 34: 605–625, 2010, doi:10.1002/nag.828.

[14] T. Kasper, *Finite Elemente Simulation maschineller Tunnelvortriebe in wassergesättigtem Lockergestein*, PhD thesis, Lehrstuhl für Statik und Dynamik, Ruhr-Universität Bochum, 2005.

[15] F. Nagel, *Numerical modelling of partially saturated soil and simulation of shield supported tunnel advance*, PhD thesis, Ruhr University Bochum, 2009.

[16] H.G. Bui, G. Meschke, “A parallelization strategy for hydro-mechanically coupled simulations in mechanized tunneling”, in *EURO:TUN 2017*, 2017.

[17] G. Karypis, V. Kumar, “A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs”, *SIAM J. Sci. Comput.*, 20: 359–392, Dec. 1998, ISSN 1064-8275.

[18] R.D. Falgout, U.M. Yang, “hypre: a Library of High Performance Preconditioners”, in *Preconditioners, Lecture Notes in Computer Science*, pages 632–641, 2002.

[19] Autodesk, *Autodesk Revit*, Autodesk, Inc, 2017, URL `http://www.autodesk.co.uk/products/revit-family/`.

[20] P. Dadvand, R. Rossi, E. Oñate, “An Object-oriented Environment for Developing Finite Element Codes for Multi-disciplinary Applications”, *Archives of Computational Methods in Engineering*, 17: 253–297, 2010.

[21] S. Balay, W.D. Gropp, L.C. McInnes, B.F. Smith, “Efficient Management of Parallelism in Object Oriented Numerical Software Libraries”, in E. Arge, A.M. Bruaset, H.P. Langtangen (Editors), *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

[22] S. Balay, S. Abhyankar, M.F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, K. Rupp, B.F. Smith, S. Zampini, H. Zhang, H. Zhang, “PETSc Users Manual”, Technical report, Argonne National Laboratory, 2016.

[23] N.A. Do, D. Dias, P. Oreste, I. Djeran-Maigre, “2D Tunnel Numerical Investigation: The Influence of the Simplified Excavation Method on Tunnel Behaviour”, *Geotechnical and Geological Engineering*, 32(1): 43–58, 2014.

[24] S. Schindler, P. Mark, “Evaluation of Building Stiffness in the Risk-assessment of Structures Affected by Settlements.”, in *Proc. 3rd Int. Conf. on Comp. Meth. in Tunneling and Subsurface Engineering - EURO:TUN 2013, RUB*, pages 477–486. Bochum, Germany, 2013.