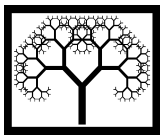
Paper 24



©Civil-Comp Press, 2013
Proceedings of the Third International Conference on
Parallel, Distributed, Grid and Cloud Computing for Engineering,
B.H.V. Topping and P. Iványi, (Editors),
Civil-Comp Press, Stirlingshire, Scotland

A Parallel Block Preconditioner for Coupled Simulations of Partially Saturated Soils in Finite Element Analyses

G. Bui, J. Stascheit and G. Meschke
Institute for Structural Mechanics, Ruhr University Bochum
Bochum, Germany

Abstract

Coupled simulations of partially saturated soils using finite element analysis lead to systems of equations that are difficult to solve for iterative solvers. In the framework of a simulation model for shield tunnelling that considers various components of the tunnelling process - the ground, the lining, the grouting, hydraulic jacks and supports, the ground and the grouting mortar are formulated as two- and three-phase continua, consisting of solid particles, water and air. The coupled formulation of the underlying discretised differential equations leads to a block-structured system of equations. An iterative solver is employed to solve the system within an implicit Newton-Raphson solution strategy. In this contribution, we examine a block-preconditioning technique to accelerate and stabilise iterative solvers in large scale simulations and that has been specifically designed to address the various challenges of the problem, mainly ill-conditioning and non-symmetry of the coefficient matrix. The key result is to realise the effectiveness of the technique in a parallel regime which ensures scalability and efficiency. Furthermore, the mesh-independent convergence properties and the insensitivity to model parameters of the implemented preconditioner are shown.

Keywords: Coupled problems, FEM, shield tunnelling, mixed formulation, block-preconditioning, preconditioned Krylov solver.

1 Introduction

Shield tunnelling is characterised by a complex construction process and often involves intricate interactions between the ground, the supporting measures, the tail void and the tunnel boring machine (TBM). The tunnels are often constructed in severe geological conditions such as high ground water pressures and soft soils. Therefore, it is impacted by water ingress and soil settlements in front of cutting face and along

and behind the shield skin. Settlements are also caused by consolidation of soil in a long term. Moreover, in the case of fixing and maintenance of the cutting wheel using temporary compressed air support, air may flow into the heading face and lead to a partially saturated zone in the soil ahead of the face. All of the above yields reactions in the ground, characterised by a stress redistribution in the soil skeleton, changing water pressures and pore saturation as well as ground deformations. To mitigate failure, a range of support measures is kept along with the tunnelling process: the heading face is supported by a pressurised support medium, the soil is in frictional contact with the shield skin and the annular gap is filled with grouting mortar at the time the machine is thrusted forward to ensure the stability of the surrounding soil, to minimise settlements and to prevent water ingress into the tunnel.

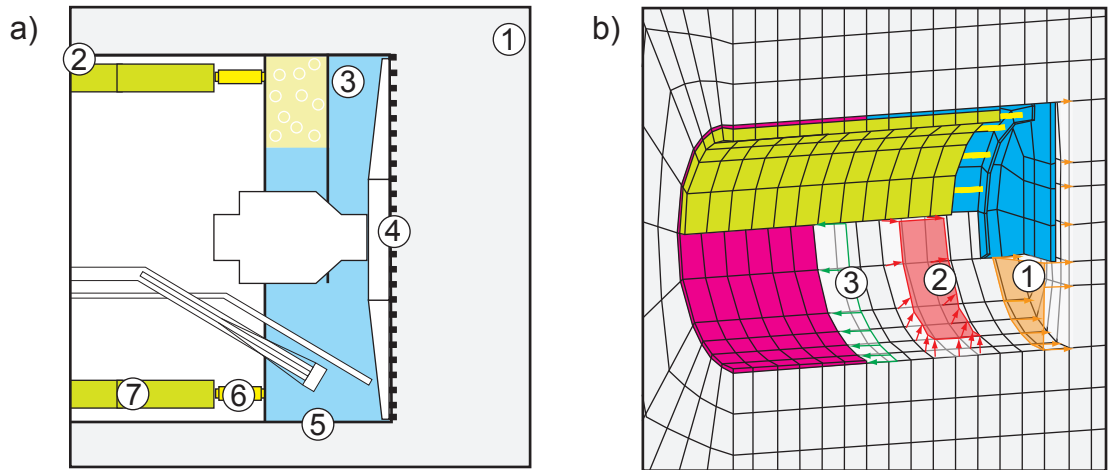


Figure 1: Components of typical mechanised tunnelling process and spatial discretisation of tunnelling simulation

Fig.1 presents typical components of a tunnelling process. In Fig.1a, (1) represents the ground model, (2) the tail void grouting, (3) the diving wall, (4) the heading face support, (5) the shield tail, (6) the hydraulic jacks, and (7) the lining. In Fig.1b, the spatial discretisation of the tunnelling simulation model is shown. (1) marks the mesh for heading face support where a boundary condition of water pressure is applied, (2) are contact elements which represent the interaction between the tunnel lining and the soil, (3) are pressure conditions to simulate the pressurised grouting of the annular gap.

During maintenance phases, the support medium can be temporarily replaced by compressed air (compressed air intervention). To model this, it is compulsory to use a three-phase formulation to capture the complex interactions with sufficient accuracy. In the numerical simulation framework for tunnelling (**ekate**), the triphasic model for a time variant description of partially saturated soil [2] is used. The model is formulated within the framework of the theory of porous media (TPM), based upon phase balance equations and constitutive relations for the stress-strain behaviour of the soil

skeleton, the pressure-density relation of gaseous phase, the pressure-suction relation and the pressure-seepage flow relation. The balance equations are solved simultaneously instead of consecutive steps of solving the fluid flow problem and the momentum balance of soils separately. Therefore, it allows for a fully coupled triphasic soil modelling and enables a consistent formulation of two-phase (i.e. water phase and gaseous phase) flow problem.
In the coupled model - consisting of an elasto-plastic soil skeleton, pore water and pore air - the water is modelled as an incompressible phase and air as a compressible phase. Within the TPM framework, each fluid phase is associated with its own state of motion. The fluid flows are characterised by Darcy's law. The capillary effect of suction of water into the pore space is described by the Soil-Water characteristic curve (SWCC). The clay and sand model (CAS-model) [2] is implemented to account for partially saturated soil behaviour. A large deformation formulation is used in variational form to fully describe the soil deformation.
The coupled formulation leads to a block structure of the system stiffness matrix. To solve the resulting system of equations and to ensure scalability, an iterative solver employing the Krylov subspace method is used. Since direct solvers require much memory as the size of problem grows, an iterative method is apparently the choice of interest. Nevertheless, the main challenge of an iterative method is to design a good preconditioner by means of improving the condition of the system matrix and accelerating the convergence behaviour. In the coupled simulation of partially saturated soil, the preconditioner is designed in the way that the iterative solver is able to exhibit mesh-independent convergence [1]. Preconditioning a matrix means finding a good approximation of its inverse with reasonable computational effort. In this sense, the system matrix can be decomposed by a block LU factorisation and the problem of finding approximate inverse can be reduced to finding the approximate inverse of the block matrix of the solid part and its Schur complement.

2 Model of coupled partially saturated soil and the two-phase flow

The partially saturated soil consists of three phases ϕ^α: the solid phase ϕ^s, the liquid phase of pore water ϕ^w and the gaseous phase of pore air ϕ^a (see Fig.2a). Each phase is represented by its volume fraction n^α. The pore space of solid phase is assumed to be completely filled with fluid phase ϕ^β [β=w(ater),a(ir)] which are able to flow through its pore voids. The exact microstructure of these pore voids is not important and the interaction between fluids and pore skeleton is described by using the averaging principle.

$$n^s + n^w + n^a = 1 \tag{1}$$

Using the averaging principle, the density of each phase can be defined knowing its intrinsic density ϱ^α and it volume fraction:

$$\rho^{\alpha} = n^{\alpha}\varrho^{\alpha} \tag{2}$$

The pore volume is defined as volume fraction of fluid phases $n = n^w + n^a = 1 - n^s$. The saturation S^{β} of the pore space with the respective fluid phase ϕ^{β} is expressed as

$$S^{\beta} = \frac{n^{\beta}}{n} \tag{3}$$

Note that $S^a = 1 - S^w$. Following eq.(2), the overall density of the porous mixture can be expressed as function that depends on the pore volume and the water saturation

$$\rho = \Sigma_{\alpha} n^{\alpha}\varrho^{\alpha} = (1-n)\varrho^s + n(S^w\varrho^w + (1-S^w)\varrho^a) \tag{4}$$

In case of deformation of the mixture, the motion of a material point in each phase is a function of time and its location in reference configuration (Lagrangean description):

$$\mathbf{x}^{\alpha}(\mathbf{X}^{\alpha}, t) = \mathbf{X}^{\alpha} + \mathbf{u}^{\alpha}(\mathbf{X}^{\alpha}, t) \tag{5}$$

Moreover, the motion of the fluid phase ϕ^{β} with respect to the solid phase is denoted as diffusion velocity

$$\mathbf{v}^{\beta s} = \dot{\mathbf{x}}^{\beta} - \dot{\mathbf{x}}^{s} \tag{6}$$

Averaging eq.(6) leads to Darcy velocity

$$\tilde{\mathbf{v}}^{\beta s} = nS^{\beta}\mathbf{v}^{\beta s} \tag{7}$$

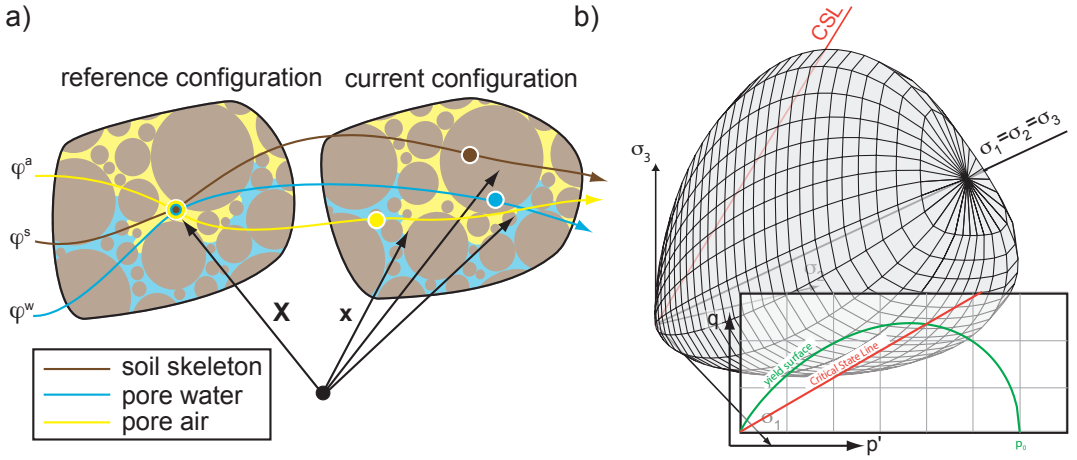


Figure 2: Ground model: a) The multiphase character of unsaturated soil, (b) Illustration of yield surface of Clay and Sand model in the principal stress space and in the deviatoric plane

2.1 Balance equations for the solid phase and two phase flows

To derive the equilibrium state of the mixture, it's important to assume that the solid phase and water phase is incompressible and the air phase is a compressible, ideal gas. This condition can be described as the derivative w.r.t time of intrinsic density of solid phase and water phase in its respective configuration is vanished. Moreover, taking into account the mass balance equation of multiphase mixture, the phase balance equation of partially saturated soil can be described as

$$n\frac{\partial^s S^w}{\partial t} + \text{div } \tilde{\mathbf{v}}^{ws} + S^w \text{div } \dot{\mathbf{u}}^s = 0 \tag{8}$$

$$\frac{nS^a}{\varrho^a}\frac{\partial^s \varrho^a}{\partial t} + n\frac{\partial^s S^a}{\partial t} + \frac{1}{\varrho^a}\text{grad } \varrho^a \cdot \tilde{\mathbf{v}}^{as} + \text{div } \tilde{\mathbf{v}}^{as} + S^a \text{div } \dot{\mathbf{u}}^s = 0 \tag{9}$$

Assuming quasi-static condition ($\ddot{\mathbf{u}} = 0$), The overall momentum balance of the mixture can be read

$$\text{div } \boldsymbol{\sigma} + \rho \mathbf{g} = 0 \tag{10}$$

With ρ is described in eq.(4) and the total stress $\boldsymbol{\sigma}$ is constituted by stresses in each phase: $\boldsymbol{\sigma} = (1-n)\boldsymbol{\sigma}^s - n(S^a p^a + S^w p^w)\mathbf{I}$

From eq.(8), (9) & (10), it is essential to choose the primary field variables as displacement $\mathbf{u}^s$ of the solid phase and hydrostatic pressure p^w, p^a of the water and air phase respectively.

2.2 Constitutive relation in solid phase

To describe the elasto-plastic behaviour of the soil skeleton, the elastic Hookean law and the CAS-model [2] are used. The CAS-model is a generalisation of the Cam-Clay model and is characterised by yield surface (see Fig.2b)

$$\mathcal{F} = \left(\frac{\sqrt{3}J}{M(\theta)p'}\right)^n + \frac{1}{\ln r}\ln\frac{p'}{p'_0} = 0 \tag{11}$$

The CAS-model employed in this formulation use the same hardening law as in Cam-Clay model to express the elastic-plastic behaviour while using a modified stress-dilatancy relation. The model also incorporates the capillary pressure into the yield rule and hardening law according to Barcelona basic model [7].

For two-phase flow within solid skeleton, the water saturation S^w depends on capillary pressure $p^c = p^a - p^w$. Owing to capillary effect, if the pore volume is small then the water can resist more capillary pressure and vice versa. The relation between water saturation and capillary pressure within the pore voids is found by experiment and is described by soil-water characteristic curve. In this work the formulation according to Van Genuchten [8] is used:

$$S^w(p^c) = S^w_{min} + (S^w_{max} - S^w_{min}) \left(1 + \left(\frac{p^c}{p^r_b}\right)^n\right)^{-m} \tag{12}$$

An example of SWCC is shown in Fig.3. The SWCC depends on the grain size, the pore size distribution and the compaction of the soil.

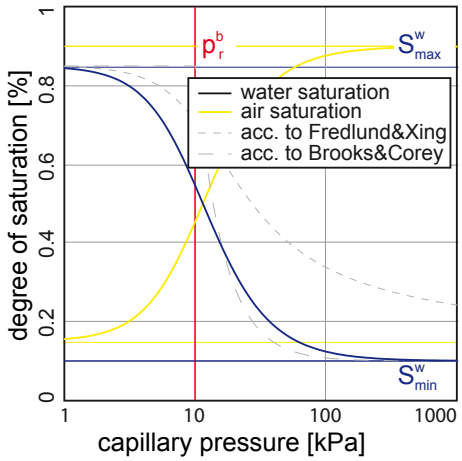


Figure 3: Computed relation between saturation and capillary pressure according to Van Genuchten [8]

3 Finite element formulation and discretisation procedure

The coupled equations for partially saturated soil and multiphase flow is space and time-dependent, hence space and time discretisation are required. Using an usual finite element discretisation procedure, the formulation in strong form is converted to variational form by multiplying the strong form equations with the trial functions and integrate over the whole domain. In this case, the trial functions are the variation of displacement and the variation of pressures. The integration is converted to bilinear form with the application of Green formula. From eq.(8) & (9), these results in

$$\begin{aligned} &\int_\Omega \delta p^w n \frac{\partial S^w}{\partial p^c}(\dot{p}^a - \dot{p}^w) dV + \int_{\Gamma^w_N} \delta p^w (\tilde{\mathbf{v}}^{ws} \cdot \mathbf{n}) dA \\ &- \int_\Omega \text{grad}\, \delta p^w \cdot \tilde{\mathbf{v}}^{ws} dV + \int_\Omega \delta p^w S^w \text{div}\, \dot{\mathbf{u}}^s dV = 0 \end{aligned} \tag{13}$$

And

$$\int_{\Omega} \delta p^a \frac{nS^a}{\varrho^a} \frac{\partial \varrho^a}{\partial p^a} \dot{p}^a dV + \int_{\Gamma_N^a} \delta p^a (\tilde{\mathbf{v}}^{as} \cdot \mathbf{n}) dA - \int_{\Omega} \text{grad}\, \delta p^a \cdot \tilde{\mathbf{v}}^{as} dV$$
$$+ \int_{\Omega} \delta p^a n \frac{\partial S^a}{\partial p^c} (\dot{p}^a - \dot{p}^w) dV + \int_{\Omega} \delta p^a S^a \text{div}\, \dot{\mathbf{u}}^s dV \tag{14}$$
$$+ \int_{\Omega} \delta p^a \frac{1}{\varrho^a} \frac{\partial \varrho^a}{\partial p^a} \text{grad}\, p^a \cdot \tilde{\mathbf{v}}^{as} dV = 0$$

Finally, the momentum balance in its variational form can be expressed as

$$\int_{\Gamma_N^s} \delta \mathbf{u}^s \cdot \boldsymbol{\sigma} \cdot \mathbf{n} d\Gamma - \int_{\Omega} \text{grad}\, \delta \mathbf{u}^s : \boldsymbol{\sigma} dV + \int_{\Omega} \delta \mathbf{u}^s \cdot \rho \mathbf{g} dV = 0 \tag{15}$$

Eq.(13), (14), (15) constitute the system of equations to be solved for phase balances of multiphase mixture. To discretise these equations of multiphase materials which can behave incompressibly, the mixed-FEM approach has been used in which quadratic Lagrangean shape functions are used for approximation of the displacement field and linear Lagrangean shape functions are used for approximation of the pressure field. By using this approach, Babuŝka-Brezzi stability constraint for multiphase materials is fulfilled.

In terms of time discretisation, the generalised Newmark-α method [9] has been used to ensure unconditional stability and second-order accuracy. This time integration scheme use a modified mid-point rule to approximate primary variables $(\bullet)_{n+1-\alpha} = \alpha_f (\bullet)_n + (1-\alpha_f)(\bullet)_{n+1}$ and its time derivatives $(\dot{\bullet})_{n+1-\alpha} = \alpha_f (\dot{\bullet})_n + (1-\alpha_f)(\dot{\bullet})_{n+1}$. The velocity and acceleration of primary variables at current time step is approximated by:

$$\begin{aligned}
(\dot{\bullet})_{n+1} &= \frac{\gamma}{\beta \Delta t} \left((\bullet)_{n+1} - (\bullet)_n \right) - \frac{\gamma - \beta}{\beta} (\dot{\bullet})_n - \frac{\gamma - 2\beta}{2\beta} \Delta t (\ddot{\bullet})_n \\
(\ddot{\bullet})_{n+1} &= \frac{1}{\beta \Delta t^2} \left((\bullet)_{n+1} - (\bullet)_n \right) - \frac{1}{\beta \Delta t} (\dot{\bullet})_n - \frac{1 - 2\beta}{2\beta} (\ddot{\bullet})_n
\end{aligned} \tag{16}$$

Within the Newton-Raphson iteration scheme, eq.(14), (15) and (13) is linearised and solved in each time step using the state update equations:

$$\begin{bmatrix} \Delta \mathbf{u}^s \\ \Delta p^w \\ \Delta p^a \end{bmatrix}_{n+1} = \left(\begin{bmatrix} \mathbf{K}_{mu} & \mathbf{K}_{mw} & \mathbf{K}_{ma} \\ \mathbf{K}_{wu} & \mathbf{K}_{ww} & \mathbf{K}_{wa} \\ \mathbf{K}_{au} & \mathbf{K}_{aw} & \mathbf{K}_{aa} \end{bmatrix} (1 - \alpha_f) + \begin{bmatrix} \mathbf{D}_{mu} & \mathbf{D}_{mw} & \mathbf{K}_{ma} \\ \mathbf{D}_{wu} & \mathbf{D}_{ww} & \mathbf{K}_{wa} \\ \mathbf{D}_{au} & \mathbf{D}_{aw} & \mathbf{K}_{aa} \end{bmatrix} (1 - \alpha_f) \frac{\gamma}{\beta \Delta t} \right)^{-1}_{n+1-\alpha}$$
$$\times \begin{bmatrix} \mathbf{R}^m_{ext} - \mathbf{R}^m_{int} \\ R^w_{ext} - R^w_{int} \\ R^a_{ext} - R^a_{int} \end{bmatrix} \tag{17}$$

4 Iterative solver

The state update equation (17) can be rearranged in block form as

$$\begin{bmatrix} A & B_1 \\ B_2 & C \end{bmatrix} \begin{bmatrix} \Delta U \\ \Delta P \end{bmatrix} = \begin{bmatrix} R_u \\ R_p \end{bmatrix} \tag{18}$$

In which $A = \mathbf{K}_{mu}$, $B_1 = [\mathbf{K}_{mw}\ \mathbf{K}_{ma}]$, $B_2 = [\mathbf{K}_{wu}\ \mathbf{K}_{au}]^T$ and $C = \begin{bmatrix} \mathbf{K}_{ww} & \mathbf{K}_{wa} \\ \mathbf{K}_{aw} & \mathbf{K}_{aa} \end{bmatrix}$ are matrices representing the solid phase, coupling between the solid and fluid and fluid phase respectively. For ease of representation, $\begin{bmatrix} A & B_1 \\ B_2 & C \end{bmatrix}$ is denoted as J and $\begin{bmatrix} R_u \\ R_p \end{bmatrix}$ as b.

In this contribution, the main goal is to solve (18) in the efficient and scalable manner. The direct method becomes memory-limited when the size of problem is large. Although there are direct solvers which can solve the large linear system of equations in scalable manner (i.e. frontal solvers), iterative solvers are the method of choice to solve the large system involved in large-scale simulations of shield tunnelling.

Eq.(18) is subject to ill-conditioning considering that when the mixture is in saturated state, the C block vanish, leaving the system of equations has the form of saddle-point structure [1]. From a physical point of view, this means that the pore fluid exerts an incompressibility condition on the deformation of solid when permeability goes to zero. Even though in this work, the mixture is considered as unsaturated, the difference in C block and A block is huge, leading to ill-condition and instability in the result of linear solver.

The iterative solver employing Krylov subspace method is the method of choice for many parallel applications. However, given the sensitivity of the Krylov subspace method to the conditioning properties of system matrix, preconditioning is required. In this work, the left preconditioning technique is chosen, i.e. to find a good approximate inverse of matrix J.

$$P^{-1}Jx = P^{-1}b \tag{19}$$

The computation of P is addressed to solve the coupled model in the sense that a mesh convergence property can be achieved. As opposed to the linear solver, which is standardised, the preconditioning method for a particular type of problem is not fixed since there are various methods to approximate the inverse of J. General preconditioning techniques (i.e. ILU, ILUT) perform well in most applications, however the convergence properties are unpredictable in the sense that the convergence rate depends on the problem size and the iterative method chosen. Observed numerical results for coupled problems show that the standard preconditioning techniques behave stably but the convergence varies among time steps and when the mesh is refined, the solver takes more iterations to converge. Despite the upper limit for the number of

iterations for standard Krylov subspace iterative method (i.e. BiCG) is n (i.e. number of degrees of freedom), this is unacceptable since n can be millions. Apparently, the structure of the coupled matrix has some effects on the efficiency of this kind of preconditioner. For example, if ILU is used, the inverse of the coupled matrix is computed from its partial LU decomposition. Therefore, the approximation may be good in the solid block but it is not in the fluid block.

4.1 Block preconditioner

Block preconditioning is the strategy of choice for preconditioning the coupled system matrix for the aforementioned reason. In this contribution, the block LU factorisation [1] is of particular interest:

$$J = \begin{bmatrix} A & B_1 \\ B_2 & C \end{bmatrix} = \begin{bmatrix} A & 0 \\ B_2 & S \end{bmatrix} \begin{bmatrix} I & A^{-1}B_1 \\ 0 & I \end{bmatrix} \tag{20}$$

Where $S = C - B_2 A^{-1} B_1$ is the Schur complement w.r.t the solid block A of the system. The preconditioner to be chosen is to approximate the inverse of the lower part of the block LU and has the form:

$$P^{-1} = L^{-1} = \begin{bmatrix} P_A^{-1} & 0 \\ -P_S^{-1} B_2 P_A^{-1} & P_S^{-1} \end{bmatrix} \tag{21}$$

P_A^{-1} and P_S^{-1} are sub-preconditioners to approximate the inverse of A and S since the exact inverse of those blocks are expensive to compute. The choice of them is various and can be customised. While preconditioning for the solid block is straightforward and can be chosen as ILU preconditoner or multigrid preconditioner (i.e. AMG), the preconditioner for the Schur complement is non-trivial to compute in the sense that the A^{-1} is dense while C is sparse. Therefore the Schur complement is replaced by its approximation

$$S_D = C - B_2 \text{diag}(A)^{-1} B_1 \tag{22}$$

Hence the preconditioner for S becomes $P_S^{-1} = P_{S_D}^{-1}$. This preconditioner is easier to compute, though its quality depends on the diagonal dominance of A.

In parallel computing, the domain partitioning method is frequently used to partition the primary domain to sub-domains in order to distribute the work to several processors. This contribution exploits the distributed computing model using a cluster with multi-core processing units per node to perform the parallel assembling of the coupled stiffness matrix and parallel solving of the coupled system of equations. The domain partitioning is used to separate domains with criteria to minimise the interface between adjacent domains in order to reduce communication time of transferring information between different domains. The use of domain partitioning leads to block

structure for each domain in the final assembled system of equations. In case the domain is decomposed to two sub-domains, the coupled stiffness matrix has the form of

$$J = \begin{bmatrix} \begin{bmatrix} A_1^1 & B_1^1 \\ B_2^1 & C^1 \end{bmatrix} & E_2^1 \\ E_1^2 & \begin{bmatrix} A_1^2 & B_1^2 \\ B_2^2 & C^2 \end{bmatrix} \end{bmatrix} \tag{23}$$

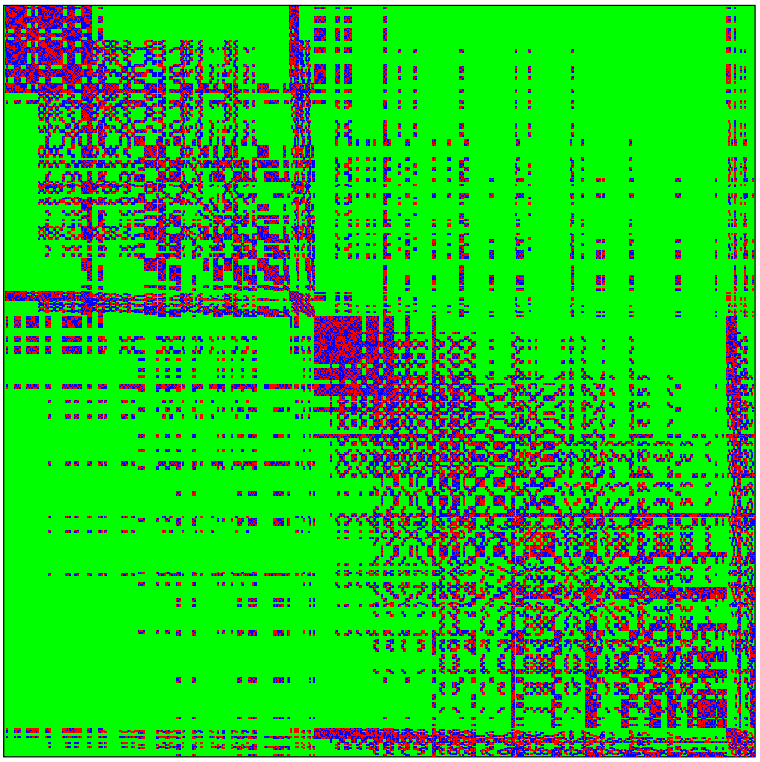


Figure 4: Coupled stiffness matrix for problem with 2 phases, computing on 2 processes

Fig.5 shows a typical stiffness matrix distributed on two processes. In this form, one can see that the matrix is composed of separated part for each process. Each process contains a block of the solid part and a block of the fluid part. The interface matrices are sparse and can be ignored for preconditioning purpose.

In Eq.(24), the upper index is the domain index, the terms E_2^1 and E_1^2 are related to d.o.fs belonging to the interface between two domains. If the sub-domains are completely separated, these matrices become zero. Therefore, the minimisation of domain interface is important to preserve a good block-structure. To design the preconditioner for partitioning matrix J, the interface matrices E_2^1 and E_1^2 is eliminated and J is replaced with $\tilde{J}$.

$$\tilde{J} = \begin{bmatrix} \begin{bmatrix} A_1^1 & B_1^1 \\ B_2^1 & C^1 \end{bmatrix} & 0 \\ 0 & \begin{bmatrix} A_1^2 & B_1^2 \\ B_2^2 & C^2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} J^1 & 0 \\ 0 & J^2 \end{bmatrix} \tag{24}$$

The preconditioning problem for $\tilde{J}$ becomes finding preconditioner for J^1 and J^2 and can be solved by the above approach. The preconditioner $\tilde{P}^{-1}$ for $\tilde{J}$ is computed as:

$$\tilde{P}^{-1} = \begin{bmatrix} (P^1)^{-1} & 0 \\ 0 & (P^2)^{-1} \end{bmatrix} \tag{25}$$

5 Numerical examples

As pointed out earlier, the block preconditioning strategy depends on the choice of sub-preconditioners to precondition the solid block and its Schur complement. In this contribution, the same preconditioning strategy is applied for both matrices. In the first step to experiment the performance of this preconditioner for a partially saturated soil problem, the ILU0 preconditioning technique is tested. The block preconditioner is combined and tested using robust iterative solvers which are biconjugate gradient stabilised method (BICGSTAB) and generalised minimum residual method (GMRES). The results obtained is then validated for accuracy with a solution obtained with a reliable direct solver (MKL Pardiso solver).

The Finite element framework employed in the analysis is KRATOS [10], developed at International Center for Numerical Methods in Engineering (CIMNE) at Barcelona. KRATOS is an object-oriented multidisciplinary finite element framework supporting a flexible data structure and is easy to extend through templates and class inheritances. KRATOS has a built-in solver and preconditioner interface, hence external solvers/preconditioners can be incorporated easily. **ekate** (Enhanced KRATOS for Tunnel Engineering) is an application developed within KRATOS to support tunnelling simulations. Among others, **ekate** features a finite element formulation for partially saturated soil problems and utilities to generate Python scripts for convenient and flexible simulation setups. **ekate** also extends KRATOS functionalities by introducing a contact algorithm which is an important component for a realistic simulation of tunnel advance.

As noted earlier, four solution strategies are tested in order to compare the performance of the proposed preconditioner. The two preconditioning strategies which are block preconditioning (BP) and ILU0 are coupled with BICGSTAB and GMRES iterative solvers. The simulations are run using shared memory parallelisation on a quad-core computer (AMD Phenom X4 2.2 MHz). OpenMP parallelisation is employed in the assembling phase to accelerate the assembly process. OpenMP is also used in matrix-vector operations in the iterative solvers to improve the performance.

5.1 Dewatering of a sand column

The testing problem is the model of column with height of one meter [4]. At the first step of simulation, water is put on the top surface of the column. This is achieved by setting a fix amount of pressure on this surface. In the subsequent step, the water pressure on the top surface is fixed. Under the influence of gravity, water starts to flow out slowly from the column. Due to capillary effects, some residual water still retains in the column. This behaviour is expressed in the figures below.

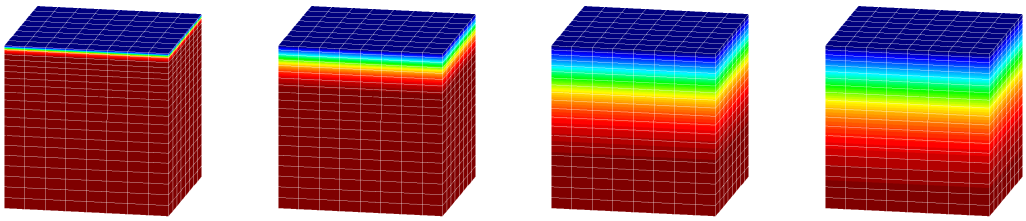


Figure 5: Dewatering of the sand column

The simulation parameters of this example are as follows:

Table 1: Simulation parameters for column dewatering problem

Parameter	Symbol	Column dewatering	Units
Young modulus	E	1e9	Pa
Poisson ratio	ν	0.3	
Gravity	g	-9.81	m/s^2
Soil density	ϱ^s	1500	kg/m^3
Water density	ϱ^w	1000	kg/m^3
Air density	ϱ^a	1.295	kg/m^3
Soil porosity	n^s	0.2	
Water permeability	k^w	4.4e-6	m^2
Air permeability	k^a	3.2e-7	m^2

Three mesh discretisations are used as input for the finite element simulation. The tested meshes are structured and are refined in the vicinity of top surface to account for rapid change of water pressures in this area.

Table 2: Mesh discretisations

Ref	Total unknowns	Displacement	Pressure
1	625	580	45
2	4516	4266	250
3	34090	32470	1620

In order to obtain representative results, the performance measures (i.e. number of iterations, solving time and building time) are averaged after each simulation. In

fact, depending on the characteristics of partially saturated soil and relative state of water and air, the stiffness matrix of the system may vary and affect the convergence bahaviour. The time step also states a dominant factor to deteriorate the number of iterations. Larger time steps lead to a softer matrix which is more ill-conditioned and hence needs more iterations to converge. Nevertheless, the avarage quantities are a valid indicator to verify the efficiency of the respective method.

Table 3: Computation results with OpenMP code

Strategy	Ref	Building time (s)	Average iteration	Average time (s)
BICGSTAB-BP	1	0.14	18.5	0.16
BICGSTAB-ILU0	1	0.14	27.0	0.18
GMRES-BP	1	0.14	25.0	0.15
GMRES-ILU0	1	0.14	30.5	0.15
BICGSTAB-BP	2	1.01	40.4	2.26
BICGSTAB-ILU0	2	1.00	35.5	1.95
GMRES-BP	2	1.00	46.8	1.71
GMRES-ILU0	2	1.01	45.9	1.58
BICGSTAB-BP	3	8.15	91.8	35.70
BICGSTAB-ILU0	3	8.03	71.1	26.67
GMRES-BP	3	8.09	100.7	22.66
GMRES-ILU0	3	8.11	86.7	19.28

Table 3 shows the results of the performance test for the four tested schemes w.r.t different mesh size. It can be seen that the block preconditioning strategy in fact does not perform better than ILU0 when fine meshes are used. It also performs worse if the same iterative method is used for both preconditioning strategies. with respect to mesh dependency, it is clear that more iterations are needed as the problem size grows. As mentioned earlier, the current block preconditioning technique uses ILU0 as the sub-preconditioners for the diagonal blocks. The same behaviour has been observed in [1] when block preconditioning combines with ILU did not exhibit mesh-independent convergence properties. This initiates the idea to apply a different preconditioning strategy for the sub-preconditioners (e.g. AMG). However, one can note that block preconditioning works better if it is used with BICGSTAB method.

6 Conclusion

In this work, a block-preconditioning technique for partially saturated soil simulation is studied. Although a mesh-independent convergence property is not shown in the current setup, it provides some valuable insight of the convergence properties for this particular problem. The current work shows that the ILU0 preconditioner is not a good candidate for sub-preconditioning of the block matrices. On the other hand, the key challenge is to implement the proposed method into a distributed memory environment which provides better scalability. This will be a subject of future research.

Acknowledgement

Financial support was provided by the German Science Foundation (DFG) in the framework of project C1 of the Collaborative Research Center SFB 837 “Interaction Modelling in Mechanised Tunnelling”. This support is gratefully acknowledged.

References

[1] J. White and R. Borja, “Block-preconditioned Newton-Krylov solvers for fully coupled flow and geomechanics”, Computational Geoscience, DOI 10.1007/s10596-011-9233-7, 4/2011.

[2] F. Nagel and 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.

[3] F. Nagel, J. Stascheit and G. Meschke, “Process-oriented numerical simulation of shield supported tunnelling in soft soils”, Geomechanics and Tunnelling, 3, 268-282, 2010.

[4] F. Nagel, J. Stascheit and G. Meschke, “A Numerical Simulation Model for Shield Tunnelling with Compressed Air Support”, Geomechanics and Tunnelling, 1, 222-228, 2008.

[5] G. Meschke, F. Nagel and J. Stascheit, “Numerical simulation of mechanized tunnelling as part of an integrated optimization platform for tunnelling design”, IACMAG, India, 2008.

[6] A.W. Bishop, “The principle of effective stress”, Teknisk Ukeblad 1956, 39(106):859-863.

[7] E.E. Alonso, A. Gens, A. Josa “Constitutive model for partially saturated soil”, Geotechnique 1990, 40(3):405-430.

[8] M.T. Van Genuchten, D.R. Nielsen “On describing and predicting the hydraulic properties of unsaturated soils”, Annales Geophysicae 1985, 3(5):615-628.

[9] J. Chung, G.M. Hulbert “Time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method”, Journal of Applied Mechanics 1993, 60(2):371-375.

[10] P. Dadvand, R. Rossi, E. Oñate “An Object-oriented Environment for Developing Finite Element Codes for Multi-disciplinary Applications”, Arch Comput Methods Eng 2010, 17:253-297.