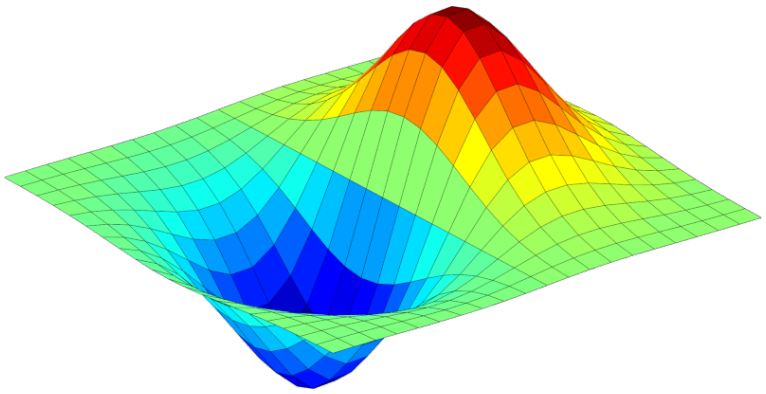
VŠB – Technical University of Ostrava
Institute of Geonics of the CAS, Ostrava

SNA'25

Seminar on Numerical Analysis

Modelling and Simulation
of Challenging Engineering Problems



Winter School

Methods of Numerical Mathematics and Modelling,
High-Performance Computing, Numerical Linear Algebra

Ostrava, January 27 – 31, 2025

Programme committee:

Marek Brandner	University of West Bohemia in Pilsen
David Horák	VŠB - Technical University of Ostrava & Institute of Geonics of the CAS, Ostrava
Jaroslav Kruis	Czech Technical University in Prague
Dalibor Lukáš	VŠB - Technical University of Ostrava
Miroslav Rozložník	Institute of Mathematics of the CAS, Prague
Stanislav Sysala	Institute of Geonics of the CAS, Ostrava
Petr Tichý	Charles University in Prague

Organising committee:

Hana Bílková	Institute of Mathematics of the CAS, Prague
Petra Frélichová	VŠB - Technical University of Ostrava
David Horák	VŠB - Technical University of Ostrava & Institute of Geonics of the CAS, Ostrava
Dalibor Lukáš	VŠB - Technical University of Ostrava
Eva Schäfferová	VŠB - Technical University of Ostrava
Jiří Starý	Institute of Geonics of the CAS, Ostrava
Stanislav Sysala	Institute of Geonics of the CAS, Ostrava

Institute of Geonics of the Czech Academy of Sciences
ISBN 978-80-86407-88-3

Preface

The Seminar on Numerical Analysis (SNA) started in 2003 and since then it has become an established event for the Czech scientific community working in the field of mathematical modelling, numerical analysis, and computer simulations. The SNA consists of a winter school, contributed research talks and poster presentations. The winter school comprises invited survey lectures delivered by renown specialists, possibly from foreign countries.

SNA'25 is held in Ostrava and organized by the Department of Applied Mathematics, Faculty of Electrical Engineering and Computer Science, VŠB - Technical University of Ostrava jointly with the Institute of Geonics of the Czech Academy of Sciences, Ostrava. The conference is under the auspices of the Union of Czech Mathematicians and Physicists and EU-MATHS-IN.CZ.

In memory of one of the conference founders and a long term main organizer of these events, Prof. Radim Blaheta, we newly establish **The Blaheta Lecture**. Its first speaker is Prof. Ulrich Rüde, FAU Erlangen-Nürnberg, providing the talk entitled *Solvers for Extreme Scale Computing*. Further, several other foreign guests, who cooperated with Radim Blaheta, accepted our invitation to participate at SNA'25. Namely, Prof. Massimiliano Ferronato (University of Padova), Prof. Akhtar Khan (Rochester Institute of Technology, on-line), Prof. Johannes Kraus (University of Duisburg-Essen), and Prof. Youseph Saad (University of Minnesota, on-line).

SNA'25 has 66 participants. It includes 25 short talks, 15 posters, and the following tutorial lectures within the Winter School:

- *M. Ferronato*: Coupled Simulation of Flow and Deformation in Fractured Porous Media: Discretizations, Solvers and Applications
- *M. Tůma*: Solving Large Sparse Linear Systems and Least Squares
- *P. Vodstrčil*: Selected Inequalities in the Analysis of Domain Decomposition Methods
- *J. Vybíral*: Mathematics of Neural Networks

This proceedings contains extended abstracts to several talks or posters. Short abstracts to all contributions, presentations of invited speakers, and the list of participants are available at the conference web pages `sna.vsb.cz`.

Last but not least we would like to thank all the participants for supporting the SNA conference series. We are honored that we can continue this traditional event that has been established by our teachers.

On behalf of the Programme and Organizing Committee of SNA'25,

David Horák, Dalibor Lukáš, Jiří Starý and Stanislav Sysala

Extended Abstracts

Contents

Bayesian inversion with neural network surrogates for TSX parameter estimation

S. Béreš[1,2], *M. Béreš*[1,2], *T. Luber*[1]

[1] Institute of Geonics, Czech Academy of Sciences, Ostrava
[2] Faculty of Electrical Engineering and Computer Science, VŠB - Technical University of Ostrava

1 Introduction

This contribution presents a Bayesian framework for solving a geotechnical inverse problem based on the Tunnel Sealing Experiment (TSX), conducted in an underground research laboratory in Canada, see [1].

We consider the standard Bayesian inversion scheme with an additive noise model. In this framework, we work with three random variables: U, Y, and Z, representing the unknown parameters, the observed data, and the observational noise, respectively. A mathematical model $G : \mathbb{R}^n \to \mathbb{R}^m$ is provided, and the objective is to determine the posterior distribution of U given a specific vector of noisy outputs $y \in \mathbb{R}^m$ (i.e., a realization of Y). For the additive noise model, Y is expressed as

$$Y = G(U) + Z. \tag{1}$$

The Bayesian approach also incorporates prior information about the input parameters, independent of the observed data. This prior distribution reflects preliminary knowledge about the input parameters obtained from prior experience.

The probability density function (pdf) of U is referred to as the prior pdf and is denoted by f_U. Similarly, f_Z, the pdf of Z, is called the noise pdf. For a given y, f_U, and f_Z, the Bayesian approach yields the conditional distribution of U given $Y = y$, known as the posterior distribution:

$$f_{U|Y}(u|y) = \frac{f_Z(y - G(u))\, f_U(u)}{\int_{\mathbb{R}^n} f_Z(y - G(u))\, f_U(v)\, \mathrm{d}v} \propto \underbrace{f_Z(y - G(u))}_{\text{data likelihood}} \underbrace{f_U(u)}_{\text{prior}}. \tag{2}$$

Here, the symbol $\propto$ indicates proportionality, meaning equality up to a multiplicative constant. The subsequent step is to generate samples from the posterior distribution using Markov Chain Monte Carlo (MCMC) methods, which constitutes the main objective of this contribution.

The remainder of the text is organized as follows: Section 2 defines the observation operator G for the TSX inverse problem. Section 3 details the MCMC sampling framework, which is enhanced by a dynamically refined deep neural network surrogate model. Section 4 provides visualizations of the posterior distribution.

2 Poroelasticity problem

The following Biot poroelasticity model with zero volume forces is considered:

$$-\mathrm{div}\left(C:\varepsilon\left(u\right)\right)+\left[\alpha_B\nabla p\right] = 0 \quad (3)$$

$$c_s\frac{\partial p}{\partial t}-\nabla\cdot\left(\frac{k}{\mu}\nabla p\right)+\left[\alpha_B\frac{\partial}{\partial t}\left(\mathrm{div}\left(u\right)\right)\right] = 0 \quad (4)$$

in Ω (see Fig. 1). Here, C is the elasticity tensor, u is the displacement vector, and $\varepsilon\left(u\right) = \frac{1}{2}\nabla u + \frac{1}{2}\left(\nabla u\right)^T$. Additionally, p denotes the pore pressure. For the isotropic linear elastic case, $C:\varepsilon\left(u\right) = 2G\varepsilon\left(u\right) + \lambda\mathrm{div}\left(u\right)I$, where $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$ and $G = \frac{E}{2(1+\nu)}$.

The material parameters are as follows: Biot-Willis constant α_B, storativity c_s, porosity k, dynamic viscosity μ, Young's modulus E, and Poisson's ratio ν.

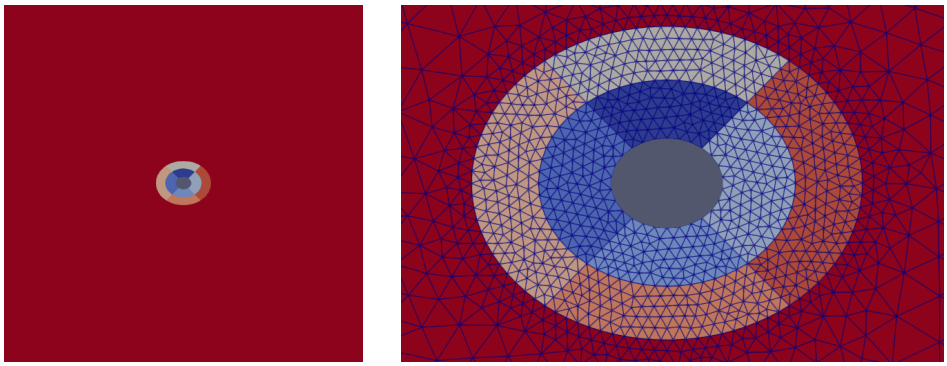


Figure 1: Whole domain Ω (left), cutout - subdomains around the tunnel (right)

The initial conditions are as follows: zero initial displacement $u_0 = 0$ m, initial pore pressure $p_0 = 3\cdot 10^6$ Pa, and an initial stress tensor characterized by nonzero values σ_x and σ_y and an angle α. Constant pressure of $3\cdot 10^6$ Pa and zero normal displacements are prescribed on the outer (square) boundary. The boundary conditions on the inner boundary (elliptic hole) simulate the tunnel drilling phase (17 days) followed by a relaxation period (18 to 365 days). During the drilling phase, the initial pore pressure and the initial stress, transformed into the normal direction, decrease linearly to zero. In the relaxation period, both the pore pressure and the normal stress remain at zero. The model was implemented using FEniCSx. For further details, see [2].

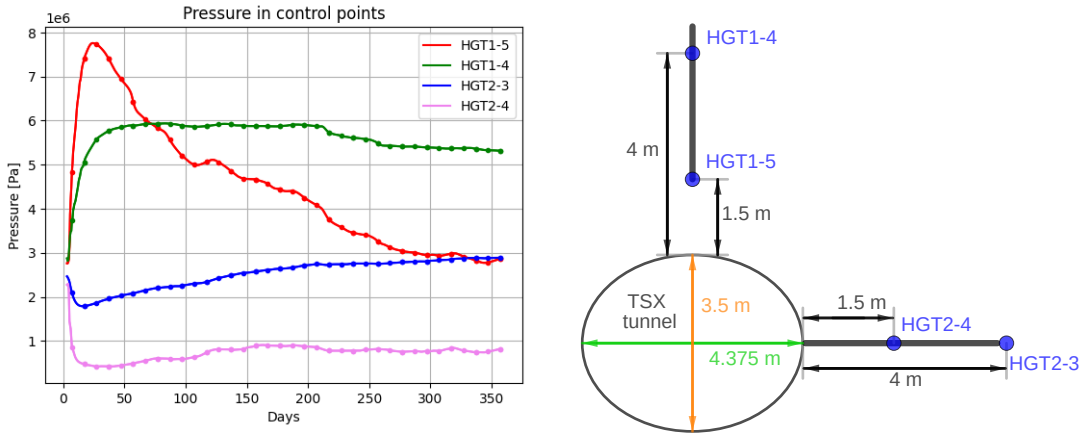


Figure 2: Observed data (left), position of control points (right)

The Bayesian inverse problem considered here is inspired by inverse problems solved deterministically by [1]. The available data consist of four time series representing pore pressure measurements at four control points over one year, obtained from TSX (see Fig. 2). Each time series is sampled at 36 uniformly distributed time points, represented as dots in Fig. 2.

The unknown parameters are the values of $\frac{k}{\mu}$, c_{pp}, E, and ν, which are assumed to be constant within each of the nine subdomains. Additional unknown parameters include the initial stresses σ_x and σ_y, as well as the angle α. This defines the observation operator $G : \mathbb{R}^{39} \to \mathbb{R}^{144}$.

3 MCMC sampling

The MCMC sampling framework is implemented in Python using the SurrDAMH library, which is available at `https://github.com/dom0015/surrDAMH`. It is based on the Delayed Acceptance Metropolis-Hastings (DAMH) algorithm, accelerated through the adaptive construction of a surrogate model (DAMH-SMU), see [3].

To further enhance efficiency, proposals are generated using a subchain approach, as described in [4], and it is also possible to adapt the proposal distribution following the method of [5]. The steps of the algorithm are summarized in Alg. 1, where the proposal distributions $q^{(k)}$ are Gaussian.

Algorithm 1 DAMH-SMU adaptive algorithm with subchains

Collect an initial set S of snapshots of G, i.e., pairs $\left(u^{(i)}, G\left(u^{(i)}\right)\right)$.
For $k = 0, 1, \ldots$

1. Construct surrogate model $\widetilde{G}^{(k)} : \mathbb{R}^n \to \mathbb{R}^m$ based on snapshots in S.
2. Generate proposal v using a subchain: Set $w^{(0)} = u^{(k)}$. For $m = 0, 1, \ldots, m_{\max} - 1$
   (a) Propose a sample z from $q^{(k)}\left(\cdot|w^{(m)}\right)$.
   (b) With probability $\widetilde{a}\left(w^{(m)}, z\right) = \min\left\{1, \frac{q^{(k)}\left(w^{(m)}|z\right) f_Z\left(y - \widetilde{G}^{(k)}(z)\right) f_U(z)}{q^{(k)}\left(z|w^{(m)}\right) f_Z\left(y - \widetilde{G}^{(k)}\left(w^{(m)}\right)\right) f_U\left(w^{(m)}\right)}\right\}$, accept z, i.e., set $w^{(m+1)} = z$. Otherwise, set $w^{(m+1)} = w^{(m)}$.
3. Set $v = w^{(m_{\max})}$.
4. With probability $a_{\widetilde{Q},\mu}\left(u^{(k)}, v\right) = \min\left\{1, \frac{f_Z\left(y - \widetilde{G}^{(k)}\left(u^{(k)}\right)\right) f_Z(y - G(v))}{f_Z\left(y - \widetilde{G}^{(k)}(v)\right) f_Z\left(y - G\left(u^{(k)}\right)\right)}\right\}$, accept v, i.e., set $u^{(k+1)} = v$. Otherwise, reject v, i.e., set $u^{(k+1)} = u^{(k)}$.
5. Add $(v, G(v))$ to S.
6. Construct updated proposal pdf $q^{(k+1)}$.

The algorithm assumes the use of non-intrusive surrogate models $\widetilde{G}^{(k)} \sim G$, which can be constructed from collected snapshots $\left(u^{(k)}, G\left(u^{(k)}\right)\right)$ and adaptively refined. In addition to standard approaches such as polynomial chaos approximation or radial basis function interpolation, surrogate models can also be constructed using machine learning techniques.

In this work, we use a multilayer perceptron (MLP) neural network regressor with a tanh activation function, the ADAM (Adaptive Moment Estimation) learning algorithm, and a mean square error loss function. The MLP is trained continuously during the sampling process. Whenever new data become available, the MLP is trained on them until a specified loss threshold is achieved; subsequently, the new data are appended to the entire training set. Training is skipped if the target loss is already met and no new data are added (until additional data arrive).

For the TSX problem, the layer sizes were set to $(39, 128, 128, 128, 144)$. However, the choice of MLP structure, along with other settings such as learning rates and loss thresholds, is generally highly problem-specific.

4 Bayesian inversion results

The posterior distribution is approximated by samples generated using Alg. 1. These samples can be visualized, for instance, in the form of one- and two-dimensional marginal histograms (see Fig. 4). The figure displays histograms for a subset of random parameters—the values of $\frac{k}{\mu}$ for each of the nine subdomains shown in Fig. 1. The prior distribution for these unknown parameters was chosen as $\mathrm{Log}\mathcal{N}(-42.78, 2)$, independently for each parameter.

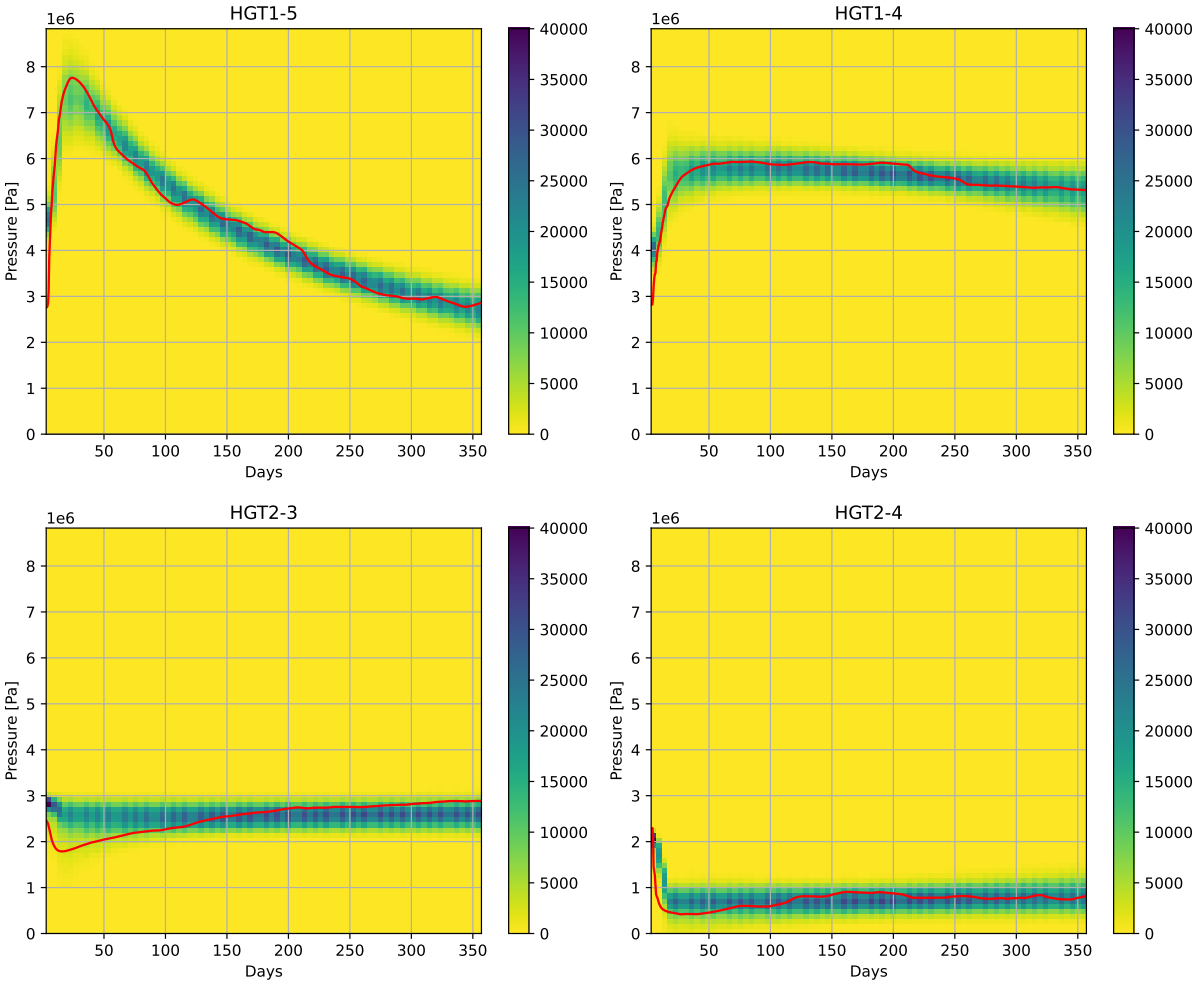


Figure 3: Posterior distribution reflected into observed data

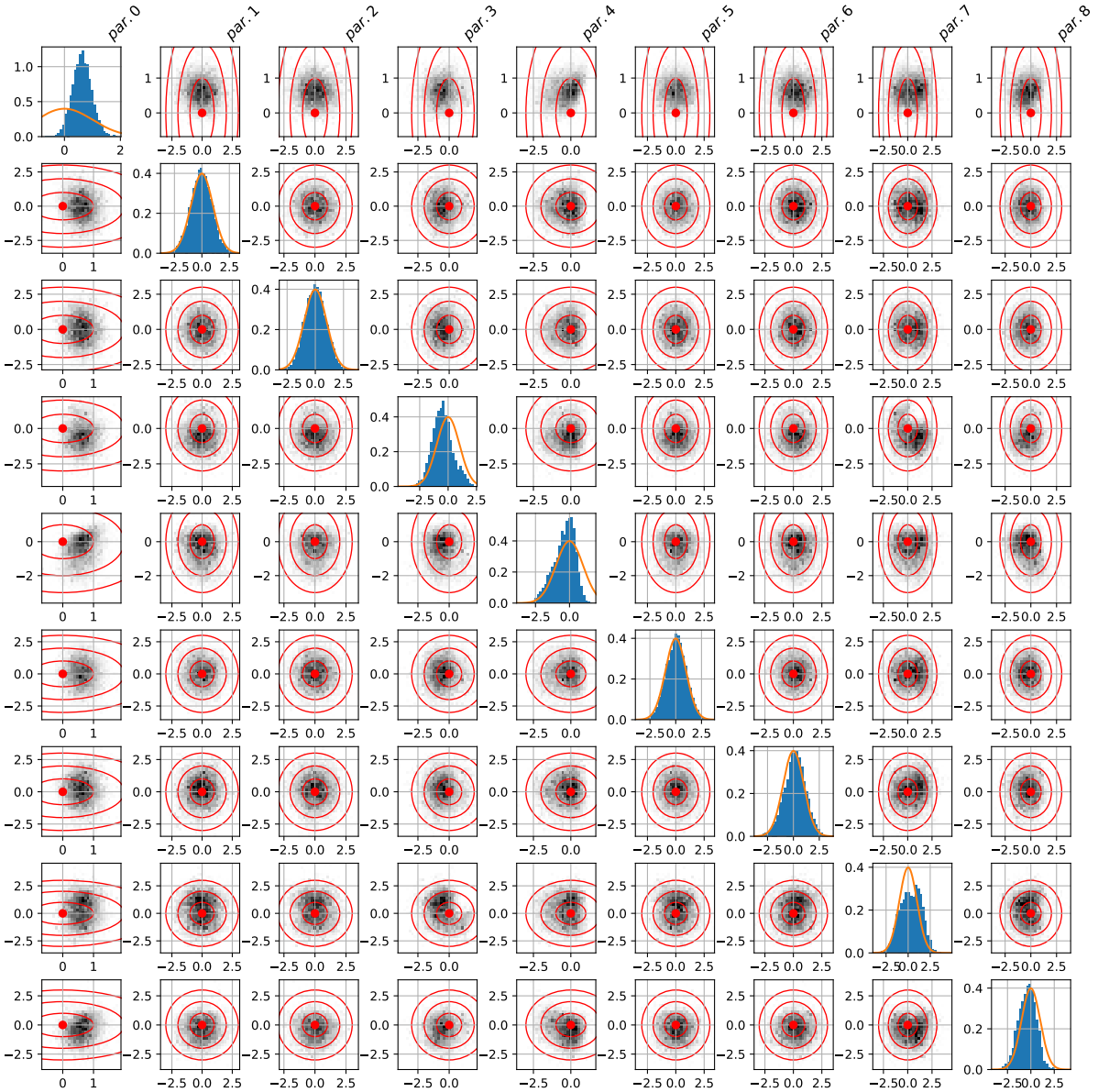


Figure 4: Histograms of posterior samples; parameters 0-8 corresponding to $\frac{k}{\mu}$ in all subdomains

Internally, the sampling framework employed a standard normal distribution $\mathcal{N}(0,1)$ for each independent component, with a transformation to the lognormal distribution applied prior to evaluating G for each sample. The orange curves in Fig. 4 represent the prior pdf (i.e., the pdf of $\mathcal{N}(0,1)$), while the red circles indicate 1, 2, and 3 standard deviations around the mean. Subdomain 0 contains control point HGT1-5, subdomain 3 contains HGT2-4, subdomain 4 contains HGT1-4, and subdomain 7 contains HGT3-4. For these parameters, we observe that the marginal posterior pdfs differ significantly from the prior pdfs.

Additionally, the posterior distribution can also be represented in the form of observed data—in this case, pore pressure timelines, see Fig. 3. These histograms allow us to assess the model's ability to reproduce the observations. Figure 5 also displays the best fit among all generated samples, determined in terms of data likelihood.

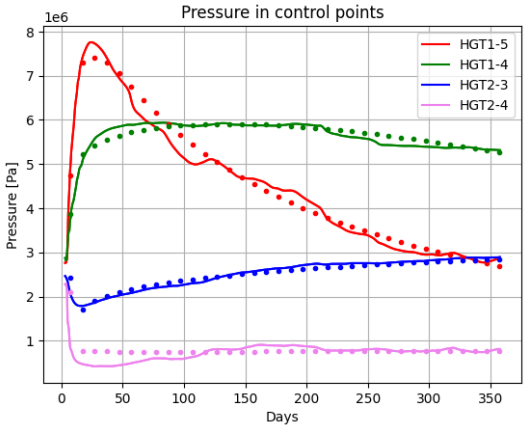


Figure 5: Best fit; dots represent $G(u_{\text{best}})$, where u_{best} is a best fitting sample

5 Conclusions

In this contribution, we demonstrated that our Bayesian inversion framework successfully integrates all available data into the model, achieving a level of consistency not attained in our previous results or in the deterministic approach presented by [1]. However, the current parameterization may be overly flexible, as evidenced by the histograms, which show minimal deviation from the prior distributions. This suggests that the model may include too many degrees of freedom for the inversion process.

Future work will focus on identifying parameters that have the most significant impact on the measurements. By reducing the parameter set while maintaining the same approximation properties, we aim to improve the efficiency and interpretability of the inversion framework.

Acknowledgement: This work was supported by the European Union under Grant Agreement no 101166718 (EURAD-2 project).

References

[1] J. Rutqvist, L. Börgesson, M. Chijimatsu et al.: *Modeling of damage, permeability changes and pressure responses during excavation of the TSX tunnel in granitic rock at URL, Canada.* Environmental Geology 57, 2009, pp. 1263–1274, https://doi.org/10.1007/s00254-008-1515-6

[2] T. Luber, S. Sysala: *Robust block diagonal preconditioners for poroelastic problems with strongly heterogeneous material.* Numerical Linear Algebra with Applications 31(3):e2546, 2024, https://doi.org/10.1002/nla.2546

[3] S. Bérešová: *Bayesian approach to the identification of parameters of differential equations.* PhD thesis, VŠB - Technical University of Ostrava, 2022.

[4] M.B. Lykkegaard, T.J. Dodwell, C. Fox, G. Mingas, R. Scheichl: *Multilevel Delayed Acceptance MCMC.* SIAM/ASA Journal on Uncertainty Quantification 11(1), 2023, pp.1–30, https://doi.org/10.1137/22M1476770

[5] H. Haario, E. Saksman, J. Tamminen: *An adaptive Metropolis algorithm.* Bernoulli 7(2), April 2001, pp. 223–242.

Shape optimization of water turbines

M. Brandner, J. Egermaier, H. Kopincová

Univeristy of West Bohemia, Pilsen

1 Introduction

The problem of determining the optimal shape of a domain or its parts (e.g., water turbine blades) is a constrained optimization problem. This requires defining an objective function (typically in integral form), specifying constraint conditions (fluid flow equations in this case), and selecting design parameters to describe the shape. Crucially, the optimization process demands the computation of the gradient, which is fundamental for any gradient-based optimization methods and includes the shape derivative. To compute the gradient, the continuous adjoint method is applied, that means we starting with the derivation of the adjoint problem, followed by its discretization. This methodology, will be demonstrated using a simplified two-dimensional laminar flow model. The chosen solver for both the state and adjoint problems is thoroughly discussed in [2]. Consider the following optimization problem:

$$\min_{\mathbf{u},p,\Omega} F(\mathbf{u},p,\Omega) \tag{1}$$

subject to incompressible steady-state Navierâ€“Stokes equations

$$R_i^u = -\frac{\partial \tau_{ij}}{\partial x_j} + u_j \frac{\partial u_i}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0, \quad i = 1,2 \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^2, \tag{2}$$

$$R^p = \frac{\partial u_j}{\partial x_j} = 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^2, \tag{3}$$

where u_i is a component of the velocity vector, $p := \frac{p}{\rho}$ is static pressure divided by constant density of the liquid and constant kinematic viscosity ν is considered in the stress tensor $\tau_{ij} = \nu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$. The Lipschitz domain boundary $\partial\Omega := \Gamma$ consists of several disjoint parts: inflow Γ_{in}, outflow Γ_{out}, periodic parts Γ_1, Γ_2 (it holds that $\Gamma_2 = T(\Gamma_1)$ is a translational copy of Γ_1 under a map T with the opposite normal vector with respect to Γ_1 in the corresponding points of both boundaries) and optimized part of the boundary Γ_{opt} with following boudary conditions:

$$\mathbf{u} = \mathbf{u}_{in}, \qquad \mathbf{x} \in \Gamma_{in}, \tag{4}$$

$$\mathbf{u} = \mathbf{0}, \qquad \mathbf{x} \in \Gamma_{opt}, \tag{5}$$

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1, \tag{6}$$

$$p(\mathbf{x}) = p(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1,$$

$$\tau_{ij}(\mathbf{x}) = \tau_{ij}(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1,$$

$$\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) n_j = 0, \quad i = 1,2 \qquad \mathbf{x} \in \Gamma_{out}, \tag{7}$$

$$p = p_{out}, \qquad \mathbf{x} \in \Gamma_{out},$$

where n_j is the jth component of the outward unit normal vector to the corresponding part of the boundary. $\mathbf{u}_{in}$ and p_{out} are given functions and the Einstein convention, where repeated indices imply summation, is used.

For optimization problems with equality constraints, it is appropriate to formulate the Lagrange function

$$L = F + \int_\Omega \lambda_i R_i^u \, d\Omega + \int_\Omega \lambda_p R^p \, d\Omega, \tag{8}$$

where for each flow variable u_i, $i = 1, 2$, and p we define so-called adjoint variables λ_i, $i = 1, 2$, and λ_p. The next approach is based on the method of CĂ©a, see [3].
Next, it is necessary to choose design variables $\mathbf{q} \in \mathbb{R}^{n_q}$. Complex shapes, such as a turbine blade, are suitably described using B-splines. This description is a linear combination of B-spline basis functions with coefficients known as control points, see [2]. For the selected solver, we choose as our design parameters the set of control points, specifically the coordinates of these control points. In the following text, the symbol q is used to represent an arbitrary element of the vector $\mathbf{q}$.

2 Objective function

The overall objective function, which is mentioned in (1), can be considered as an appropriate weighted combination of multiple components. In this text, we will introduce four components of the objective function F_1, F_2, F_3, F_4, so that:

$$F = w_1 F_1 + w_2 F_2 + w_3 F_3 + w_4 F_4. \tag{9}$$

1. The function F_1 quantifies the effect of the head. By optimizing this function, we achieve a minimal difference between the target head H_{tar} and the actual head H. The function F_1 is prescribed on the inflow and outflow part of the boundary, Γ_{in} and Γ_{out}. It is defined as follows:

$$F_1 = \frac{1}{2}\left(\frac{H - H_{tar}}{H_{tar}}\right)^2, \tag{10}$$

where the head H is defined as follows:

$$H = \frac{1}{\rho g S_{in}} \int_{\Gamma_{in}} p_{tot,in} \, d\Gamma - \frac{1}{\rho g S_{out}} \int_{\Gamma_{out}} p_{tot,out} \, d\Gamma, \tag{11}$$

$$\text{for} \quad p_{tot} = p_{stat} + \frac{1}{2}\rho v^2, \quad p_{stat} = \rho\, p, \quad v = \frac{Q}{S}, \tag{12}$$

where p_{stat} is static pressure and p kinematic pressure, further ρ denotes the density of the liquid, g gravitational acceleration, Q is the flow rate calculated from the velocity on the inlet boundary $\mathbf{u}_{in}$ and S is the area of the respective boundary segment.

2. The function F_2 is related to the efficiency of the water turbine. The ideal state is 100% efficiency, and therefore, we will minimize the deviation from this ideal state. Thus, we define the function F_2 as follows:

$$F_2 = 1 - \frac{M\omega}{Q\rho g H}, \tag{13}$$

where $\omega = \text{const.}$ denotes the angular velocity of the turbine shaft. The torque M, which acts on the turbine blade, i.e. Γ_{opt}, is defined as follows:

$$M = N \int_{\Gamma_{opt}} \mathbf{M} \cdot \mathbf{e} \, d\Gamma, \quad \text{where} \quad \mathbf{M} = \mathbf{r} \times \mathbf{F}, \quad \mathbf{F} = \mathbf{n}\, p_{stat} \tag{14}$$

and $\mathbf{e}$ is the direction of the axis of rotation, N is the number of blades, $\mathbf{F}$ denotes the force acting on the blade, $\mathbf{r}$ is the position vector perpendicular to the axis of rotation, and $\mathbf{n}$ is the normal vector pointing outward from the suction side. The above formulas are valid for 3D calculations. For our simplified 2D model, we choose $\mathbf{e} = (1, 0, 0)$, $\mathbf{r} = (0, 0, 1)$, $\omega = 1$ and $N = 5$

3. The function F_3 represents the pressure distribution on the blade. The optimization aims to match this distribution as closely as possible to the target pressure, p_{tar}. Hence, F_3 is defined as

$$F_3 = \frac{1}{2} \int_{\Gamma_{opt}} \frac{(p - p_{tar})^2}{p_{tar}^2} \, d\Gamma. \tag{15}$$

4. The final part of the objective function, F_4, minimizes the difference between the outflow boundary velocity and a given target outflow velocity $\mathbf{u}_{tar}$. This prevents undesirable turbulence behind the runner, thereby improving overall efficiency. Thus, F_4 is defined as:

$$F_4 = \frac{1}{2} \int_{\Gamma_{out}} \frac{||\mathbf{u} - \mathbf{u}_{tar}||^2}{||\mathbf{u}_{tar}||} \, d\Gamma. \tag{16}$$

3 Adjoint problem

To derive gradient, it is necessary to determine the total derivative of the Lagrange function (8) with respect to the shape parameters, i.e. $\frac{dL}{dq}$. Since the Lagrange function involves integrals, products, and quotients, the resulting expression will be highly complex, with some components being computationally infeasible to evaluate. However, it can be simplified by appropriately nullifying certain terms. This approach, detailed in [1], leads to the following adjoint problem

$$R_i^{\lambda} = -\frac{\partial \tau_{ij}^a}{\partial x_j} + \lambda_j \frac{\partial u_j}{\partial x_i} - u_j \frac{\partial \lambda_i}{\partial x_j} - \frac{\partial \lambda_p}{\partial x_i} = 0, \quad i = 1, 2, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^2, \tag{17}$$

$$R^{\lambda_p} = \frac{\partial \lambda_j}{\partial x_j} = 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^2 \tag{18}$$

with following boundary conditions

$$\lambda_i n_i = -(C_1 + C_3) \frac{\partial f_{1,in}(p)}{\partial p}, \qquad \lambda_i t_i = 0, \qquad \mathbf{x} \in \Gamma_{in}, \tag{19}$$

$$\lambda_i n_i = C_2 \frac{\partial f_{2,opt}(p)}{\partial p} - w_3 \frac{\partial f_{3,opt}(p)}{\partial p}, \qquad \lambda_i t_i = 0, \qquad \mathbf{x} \in \Gamma_{opt}, \tag{20}$$

$$\tau_{ij}^a n_j + u_j n_j \lambda_i + \lambda_p n_i = -w_4 \frac{\partial f_{4,out}(\mathbf{u})}{\partial u_i}, \qquad i = 1, 2, \qquad \mathbf{x} \in \Gamma_{out}, \tag{21}$$

$$\boldsymbol{\lambda}(\mathbf{x}) = \boldsymbol{\lambda}(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1, \tag{22}$$

$$\lambda_p(\mathbf{x}) = \lambda_p(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1,$$

$$\tau_{ij}^a(\mathbf{x}) = \tau_{ij}^a(T(\mathbf{x})), \qquad \mathbf{x} \in \Gamma_1,$$

where $\tau_{ij}^a = \nu \left(\frac{\partial \lambda_i}{\partial x_j} + \frac{\partial \lambda_j}{\partial x_i} \right)$ representing the adjoint stress tensor and

$$C_1 = \left(\frac{H}{H_{tar}} - 1 \right) \frac{w_1}{H_{tar}}, \quad C_2 = \frac{w_2 N \omega}{Q \rho g H}, \quad C_3 = \frac{w_2 N \omega}{Q \rho g H^2} \int_{\Gamma_{opt}} f_{2,opt}(p) d\Gamma.$$

4 Gradient

After adjoint problem formulation we obtain the expression for the gradient components in the form

$$\begin{aligned}\frac{dL}{dq} &= C_2 \int\limits_{\Gamma_{opt}} \left(\frac{\partial f_{2,opt}}{\partial x_j} n_j - \kappa_{opt} f_{2,opt} \right) \frac{dx_i}{dq} n_i d\Gamma + w_3 \int\limits_{\Gamma_{opt}} \left(\frac{\partial f_{3,opt}(p)}{\partial x_j} n_j - \right. \\ &- \left. \kappa_{opt} f_{3,opt}(p) \right) \frac{dx_i}{dq} n_i d\Gamma - \int\limits_{\Gamma_{opt}} \left(\tau_{ij}^a n_j + u_j n_j \lambda_i + \lambda_p n_i \right) \frac{\partial u_i}{\partial x_k} n_k \frac{dx_l}{dq} n_l d\Gamma \\ &- \int\limits_{\Gamma_{opt}} \frac{\partial \tau_{ij}}{\partial q} n_j \lambda_i d\Gamma + \int\limits_{\Gamma_{opt}} \left(\lambda_j R_j^u + \lambda_p R^p \right) \frac{dx_i}{dq} n_i \, d\Gamma, \end{aligned} \tag{23}$$

The numerical computation proceeds as follows: we set the initial shape of the blade, i.e., the boundary Γ_{opt}, and solve the primal problem (2) and (3) with boundary conditions (4), (5), (6), (7). This provides the state variables u_1, u_2 and p. The adjoint quantities λ_1, λ_2 and λ_p are obtained by solving the adjoint problem (17) and (18) with boundary conditions (19), (20), (21) and (22). Then, the gradient is computed using equation (23) and the shape of the blade is adjusted using any gradient-based method (here, for simplicity, the steepest descent method is used).

5 Conclusion

Our numerical experiments (not presented here) show, that the objective function as well as its individual components are decreasing, so this method shows great promise for the further development (especially into 3D and turbulent flow).

Acknowledgement: This work was supported by Technology Agency of the Czech Republic (TA CR) grant No. TK04020250.

References

[1] E.M. Papoutsis-Kiachagias, K.C. Giannakoglou: *Continuous Adjoint Methods for Turbulent Flows, Applied to Shape and Topology Optimization: Industrial Applications.* Archives of Computational Methods in Engineering 23, 2016, pp. 255–299, https://doi.org/10.1007/s11831-014-9141-9

[2] B. Bastl, M. Brandner, J. Egermaier, M. Michálková, E. Turnerová: *IgA-Based Solver for turbulence modelling on multipatch geometries.* Advances in Engineering Software 113, 2017, pp. 7–18, https://doi.org/10.1016/j.advengsoft.2017.06.012

[3] G. Allaire, Ch. Dapogny, F. Jouve: *Chapter 1 - Shape and topology optimization.* Handbook of Numerical Analysis: Geometric partial differential equations - part II, Volume 22, Elsevier, 2021, pp. 1–132, https://doi.org/10.1016/bs.hna.2020.10.004

Two types of preconditioning of discrete convection-diffusion equations with spectral estimates

L. Gaynutdinova, M. Ladecký, I. Pultarová, J. Zeman

Czech Technical University in Prague
University of Freiburg

1 Introduction

We focus on preconditioning of discretized differential operators and obtaining guaranteed bounds to the spectra of the resulting preconditioned matrices. We are concerned with two particular methods of precoditioning. The first type is performed by applying a discretized Green operator, which we call Green preconditioning. In the second method, a diagonal scaling is added, which we call Green-Jacobi preconditioning.

In recent years, new approaches yielding guaranteed bounds to eigenvalues of a preconditioned matrix by the Green matrix have been developed; see e.g. [3, 2]. They are based on local inspection of the data of the problem and of the Green operator. In this article, we adapt this technique to time-dependent PDEs, possibly with a convective term. While the Green preconditioner reduces spreading of the eigenvalues caused by differentiation, the Jacobi preconditioning can help to reduce the impact of highly contrasted data.

The goal of this paper is to apply Green and Green-Jacobi preconditioners to a model problem and provide lower and upper bounds to eigenalues of the preconditioned matrices. The model problem reads to solve

$$c_t \frac{\partial u}{\partial t} = \nabla \cdot a \nabla u + c_c\, b \cdot \nabla u + f \tag{1}$$

on a polygonal domain $\Omega \subset \mathbb{R}^2$ with homogeneous Dirichlet boundary conditions on $\partial\Omega$, where a is a point-wise positive constant or a symmetric and positive definite 2×2 matrix with every element essentially bounded in Ω. Vector b has zero divergence in Ω and $f \in L^2(\Omega)$. The constants c_t and c_c admit values of 0 or 1, indicating whether the corresponding term is present in the equation.

2 Preconditioning

Equation (1) is discretized by the finite element method (FEM) with triangular elements and piecewise linear basis functions. The time evolution is approximated by the implicit Euler method with the time step d_t. Thus we obtain a (set of) system(s) of linear equations of the form

$$\frac{c_t}{d_t}\mathsf{M}\mathsf{u} - \mathsf{K}_a\mathsf{u} - c_c\mathsf{C}\mathsf{u} = \frac{c_t}{d_t}\mathsf{M}\mathsf{v} + \mathsf{f}$$

where K_a is the stiffness matrix where the subscript a indicates the data of the diffusion term. The matrices M and C are the mass matrix and a nonsymmetric matrix corresponding to the convective term, respectively. Vector v represents a solution obtained in the previous time step. The Green and Green-Jacobi preconditioners considered here are

$$\mathsf{L}_p = \frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_p, \qquad \mathsf{L}_{p,\mathsf{D}} = \mathsf{D}_{\mathsf{AL}}{}^{\frac{1}{2}} \left(\frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_p \right) \mathsf{D}_{\mathsf{AL}}{}^{\frac{1}{2}},$$

respectively, where K_p is the stiffness matrix where the subscript p indicates the constant data $a(x) := p$, $x \in \Omega$, and

$$(\mathsf{D}_{\mathsf{AL}})_{ii} = \frac{(\mathsf{A}_a)_{ii}}{(\mathsf{L}_p)_{ii}}, \qquad \text{where} \quad \mathsf{A}_a = \frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_a.$$

3 Eigenvalue bounds

Once we have defined the Green and Green-Jacobi preconditioning matrices L_p and $\mathsf{L}_{p,\mathsf{D}}$, respectively, we aim to provide bounds to eigenvalues of the preconditioned matrices, namely the bounds to the spectra

$$\Lambda_p = \sigma\left(\mathsf{L}_p^{-1}\left(\frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_a - c_c\mathsf{C}\right)\right) \qquad \text{and} \qquad \Lambda_{p,\mathsf{D}} = \sigma\left(\mathsf{L}_{p,\mathsf{D}}^{-1}\left(\frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_a - c_c\mathsf{C}\right)\right),$$

respectively. Our estimating methods are based on pair-wise comparing local contributions to matrices $\frac{c_t}{d_t}\mathsf{M} - \mathsf{K}_a - c_c\mathsf{C}$ and $\mathsf{L}_{p,*}$. Providing the local matrices pair-wise share their kernels, we can obtain quite accurate bounds for every particular eigenvalue in $\Lambda_{p,*}$. However, the situations differ for Green and Green-Jacobi precoditioners, for different choices of c_c and for smooth or step function a. Depending whether $c_c = 0$ or 1, we call the bounds LRB (locally obtained real bounds) and LCB (locally obtained complex bounds), respectively. While LRB yields bounds for every particular eigenvalue, LCB only bounds all complex eigenvalues by a rectangle in the complex plain. We note that no satisfactory results are obtained from LRB in general for Green-Jacobi preconditioning if $c_t = 0$ and a is smooth. On the other hand, if a is a step function, we can obtain sensible bounds to most of the eigenvalues and a few additional outliers. The applicability of our methods is summarized in Table 1.

		smooth a		step a	
c_t	c_c	L_p	$\mathsf{L}_{p,\mathsf{D}}$	L_p	$\mathsf{L}_{p,\mathsf{D}}$
0	0	LRB	-	LRB	LRB+outliers
1	0	LRB	LRB	LRB	LRB
1	1	LCB	LCB	LCB	LCB

Table 1: Applicability of LRB and LCB estimates of spectra of preconditioned matrices for smooth and step function a.

Basics of the methods LRB and LCB for the Green preconditioner are introduced in [2] for the case $c_t = 0$ and both $c_c = 0$ or 1. The methods can be modified to cover the case $c_t = 1$ in a fairly straightforward way. The LRB and LCB estimates for the Green-Jacobi preconditioner including proofs of Lemmas 3.2 and 3.3 and other related issues will be presented in a paper currently being prepared for publication.

Remark 3.1. *A proof of the spectral bounds for Green-Jacobi preconditioner* $\mathsf{L}_{p,\mathsf{D}}$ *can be based on a spitting*

$$(\mathsf{D}_{\mathsf{AL}}^{\frac{1}{2}}\mathsf{L}_1\mathsf{D}_{\mathsf{AL}}^{\frac{1}{2}})^{-1}\mathsf{A}_a = \mathsf{M}_1 + \mathsf{M}_2 + \mathsf{M}_3$$

where $\mathsf{M}_1 \approx I$, $\|\mathsf{M}_2\|$ *is small, and* M_3 *has very low rank; see [1]. Usually, continuity of* a *is assumed to yield the spectrum close to one at least in asymptotic perspective [4]; for example* $a \in C^2(\Omega)$ *is assumed in [5]. For some structured data, the resulting spectra can be described exactly; see [5] for analysis of a 1D example with* $a(x) = \mathrm{e}^{kx}$, $k > 0$. *For nonsymmetric matrices, the convergence of preconditioned GMRES for a preconditioned problem depends on the number and sizes of the clusters of complex eigenvalues and on the norm of the matrix of eigenvectors [1].*

Lemma 3.2. *Consider equation* $-(au')' = f$ *on* $(0, 2\pi)$ *with* $a(x) = 1 + m + sin(kx)$, $m > 0$, $k \in \mathbb{N}$, *and periodic boundary conditions. Let the equation be considered in the weak form and uniformly discretized by FEM with* N *piece-wise linear basis functions. Then for any* $k \in \mathbb{N}$, $m > 0$ *and* $\delta > 0$ *there exists* $N \in \mathbb{N}$ *such that* $\Lambda_{p,\mathsf{D}} \subset (1 - \delta, 1 + \delta)$.

Lemma 3.3. *Consider* (1) *with* $c_c = 0$ *discretized by FEM with continuous piece-wise linear functions. Let* $\Omega_1 \subset \Omega$. *Let* S *be a set of such vertex numbers* k *that if* $\mathsf{A}_{kj} \neq 0$ *then* j *is also in* S. *Then if* $a(x_1, x_2)/p(x_1, x_2) = \alpha$ *in* Ω_1, *then there are* $\#S$ *eigenvalues equal to* α *in* $\Lambda_{p,\mathsf{D}}$.

4 Numerical experiments

We consider equation (1) with $\Omega = (0, 2\pi)^2 \subset \mathbb{R}^2$ and smooth or step function a,

$$a(x) = 1.1 - \sin(x_2) \qquad \text{or} \qquad a(x) = 1.1 - \text{sign}(\sin(x_2)),$$

respectively, $c = (1, 1)$, mildly oscillatory f. The weak form of the equation is discretized by FEM with $N = 2N_1N_2$ elements and $d_t = 0.5$. In the preconditioner, we use $p = 1$. We can see in Table 2 that for the step data a, the Green preconditioning seems to be sufficient, while for the smooth data, additional Jacobi preconditioning should be used. The spectra and their bounds are displayed in Figures 1 and 2 for smooth and step function a, respectively. Matlab code is available on https://github.com/LiyaGaynutdinova, section Time-Diffusion-Convection.

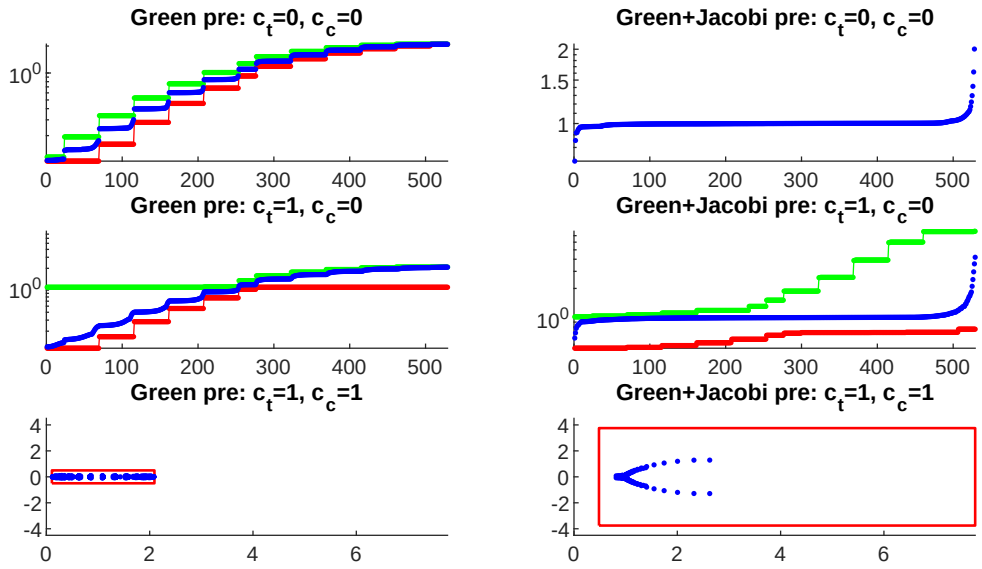


Figure 1: Smooth a: Λ_p and $\Lambda_{p,\mathsf{D}}$ (blue dots) and bounds (red and green lines), $N_1 = N_2 = 24$.

Acknowledgement: This work was supported by the CTU grant SGS24/001/OHK1/1T/11 (LG, IP) and by the European Commission (Marie-Curie 101106585 - microFFTTO (ML).

			smooth a			step a		
c_t	c_c	$N_1 = N_2$	-	L_p	$\mathsf{L}_{p,\mathsf{D}}$	-	L_p	$\mathsf{L}_{p,\mathsf{D}}$
0	0	16	112	37	12	159	3	16
		64	472	50	12	749	3	31
1	0	16	62	39	16	69	37	23
		64	271	47	19	300	40	38
1	1	16	73	46	22	95	60	33
		64	324	64	32	402	79	60

Table 2: Steps of preconditioned CG (for $c_c = 0$) and GMRES without restarting (for $c_c = 1$), respectively, for no, L_D and $\mathsf{L}_{p,\mathsf{D}}$ preconditioning, respectively.

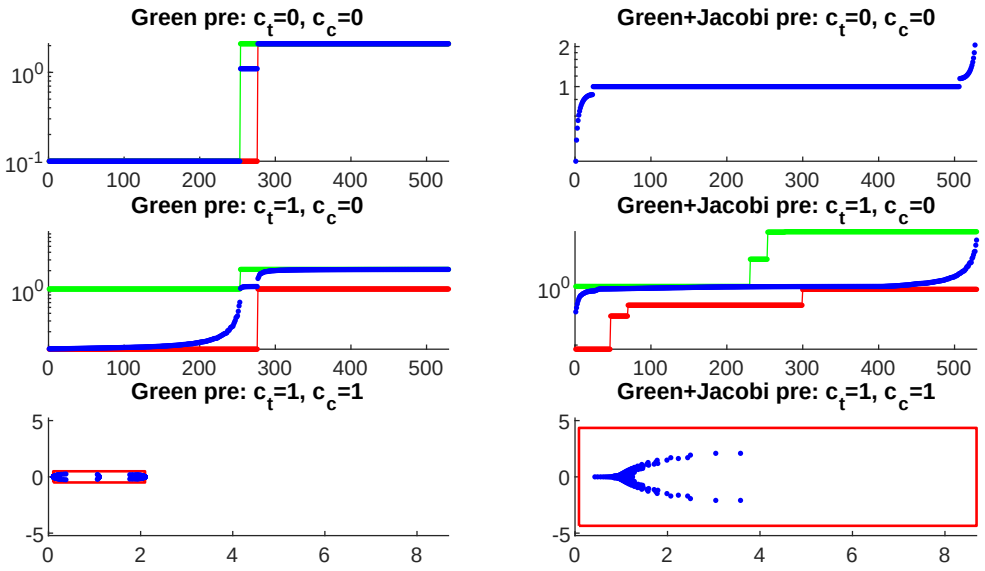


Figure 2: Step function a: Λ_p and $\Lambda_{p,\mathsf{D}}$ (blue dots) and bounds (red and green lines), $N_1 = N_2 = 24$.

References

[1] D. Bertacciny, G.H. Golub, S. Serra-Capizzao: *Spectral analysis of a preconditioned iterative method for the convection-diffusion equation*. SIAM Journal on Matrix Analysis and Applications 29(1), 2007, pp. 260–278.

[2] L. Gaynutdinova, M. Ladecký, I. Pultarová, M. Vlasák, J. Zeman: *Preconditioned discontinuous Galerkin method and convection-diffusion-reaction problems with guaranteed bounds to resulting spectra*. Numerical Linear Algebra with Applications 31(4), 2024, e2549.

[3] T. Gergelits, K.-A. Mardal, B.-F. Nielsen, Z. Strakoš: *Laplacian preconditioning of elliptic PDEs: localization of the eigenvalues of the discretized operator*. SIAM Journal on Numerical Analysis 57(3), 2019, pp. 1369–1394.

[4] J. Karatson: *Superlinear Krylov convergence under steamline diffusion FEM for convection-dominated elliptic operators*. Numerical Linear Algebra with Applications, 2024, early view, e2586.

[5] S. Serra: *The rate of convergence of Toeplitz based PC methods for second order nonlinear boundary value problems*. Numerische Mathematik 81, 1999, pp. 461–495.

A thermodynamically consistent full hydro-mechanical coupling

T. Ligurský

Institute of Geonics, Czech Academy of Sciences, Ostrava

1 Introduction

Swelling clays can be very useful as sealing and buffer materials, for instance in nuclear waste engineering barriers. To simulate possible behaviour of the clays, one needs an appropriate poromechanical model using convenient state variables and based on accessible measurement data. Especially, deformation and water content have to be coupled two-directionally since water affects deformation in the form of swelling and deformation affects water intake back.

Models combining movement of water in clays with their mechanical behaviour are often compiled in a straightforward manner: a mechanical model developed for a variably saturated soil is put together with a water retention curve introduced separately. Commonly, only an effect of the suction in a mechanical model is considered, but the effect of a mechanical model on water retention is disregarded in hydro–mechanical (HM) models. Even though the other direction of the coupling is taken into account to some extent in some models, the consistency of coupled HM models has often not been established rigorously on general thermodynamical principles, which form one of the cornerstones of reliable mathematical modelling. The applicability of such models is therefore limited, they may result in physically incorrect behaviour. On the other hand, profound thermodynamic considerations lead to models with very general couplings between state variables, which have not reached the level of applicability to practical problems [1].

To summarise, no model appears completely satisfactory for swelling clays yet. To progress in this direction, we shall focus on thermodynamically consistent full HM coupling as a basic part of such models. A simple thermodynamical poroelastoplastic framework will be introduced, and a consistent HM model with swelling developed according to it will be presented.

2 Thermodynamical framework for the unsaturated state

We consider non-stationary HM processes in a deformable swelling clay. The clay is viewed as a porous medium which is composed of solid clay minerals and a porous space. The small-strain framework is adopted and the sign convention with the stresses positive in tension and the pressures positive in compression is used. The total stress in the porous material is described by the Cauchy stress tensor $\boldsymbol{\sigma}$, whereas deformation of the solid skeleton is described by the linear strain tensor $\boldsymbol{\varepsilon} \equiv 1/2(\boldsymbol{\nabla u} + (\boldsymbol{\nabla u})^\top)$ with $\boldsymbol{u}$ standing for the displacement vector of the skeleton. The stress and strain tensors can be decomposed as:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_d - p\boldsymbol{I}, \quad \boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_d + \frac{1}{3}\varepsilon_v \boldsymbol{I},$$

where $p \equiv -1/3 \operatorname{tr} \boldsymbol{\sigma}$ denotes the total pressure, $\varepsilon_v \equiv \operatorname{tr} \boldsymbol{\varepsilon}$ is the volumetric strain (positive in dilation), and $\boldsymbol{\sigma}_d$, $\boldsymbol{\varepsilon}_d$ are the deviatoric stress and strain, respectively.

The first and second law of thermodynamics applied to an unsaturated deformable porous medium lead to the following Clausius–Duhem inequality in the context of small isothermal

transformations and negligible inertia forces and negligible effects of air [2, 3]:

$$\mathcal{D} = \mathcal{D}_{\text{sk}} + \mathcal{D}_{\text{w}} \geq 0 \tag{1}$$

with:

$$\mathcal{D}_{\text{sk}} = \boldsymbol{\sigma}_d : \dot{\boldsymbol{\varepsilon}}_d - p\dot{\varepsilon}_v - m_w \dot{g}_w - \dot{\Phi}, \quad \mathcal{D}_{\text{w}} = (-\nabla g_w + \boldsymbol{f}) \cdot \boldsymbol{w}_w. \tag{2}$$

Here g_w stands for the specific free enthalpy (also called the Gibbs potential) of water, m_w denotes the mass content of water per unit of initial volume, Φ the overall grand potential per unit of initial porous volume, $\boldsymbol{f}$ the body force density, $\boldsymbol{w}_w$ the mass flow vector of water relative to the skeleton, and a dot signifies the time derivative.

The overall dissipation $\mathcal{D}$ is split into two distinct contributions here: the dissipation $\mathcal{D}_{\text{sk}}$ related to the skeleton and the dissipation $\mathcal{D}_{\text{w}}$ due to relative motion of water with respect to the skeleton. Regarding very distinct nature of the dissipations $\mathcal{D}_{\text{sk}}$ and $\mathcal{D}_{\text{w}}$, we decouple unique inequality (1) into two separate inequalities as in [2]:

$$\mathcal{D}_{\text{sk}} \geq 0, \quad \mathcal{D}_{\text{w}} \geq 0. \tag{3}$$

Following [4], we introduce the notion of suction s in a swelling clay as a measure of the water free enthalpy:

$$s \equiv -\hat{\rho}_w (g_w - g_w^s). \tag{4}$$

Here $\hat{\rho}_w$ stands for the density of liquid water at reference conditions, and g_w^s denotes the specific free enthalpy of water in a clay sample that is in equilibrium with free moist air where the gas pressure equals the atmospheric pressure and the relative humidity equals 1. Then one can express the skeleton dissipation $\mathcal{D}_{\text{sk}}$ (2) as:

$$\mathcal{D}_{\text{sk}} = \boldsymbol{\sigma}_d : \dot{\boldsymbol{\varepsilon}}_d - p\dot{\varepsilon}_v + \frac{m_w}{\hat{\rho}_w}\dot{s} - \dot{\Phi}.$$

Plastic deformations cause plastic changes of the porosity, to which plastic changes of the water content can be related. Hence we shall consider the decomposition of the strains as well as of the water content into the elastic (superscript e) and plastic parts (superscript p):

$$\boldsymbol{\varepsilon}_d = \boldsymbol{\varepsilon}_d^e + \boldsymbol{\varepsilon}_d^p, \quad \varepsilon_v = \varepsilon_v^e + \varepsilon_v^p, \quad m_w = m_w^e + m_w^p.$$

Assuming no hysteretic effects, we split the grand potential Φ into two main parts in accordance with the additive character of energy-type quantities and the theory for the so-called 'decoupled materials' [4, 5]: (i) the potential Φ_s corresponding to the energy stored in the solid matrix during reversible mechanical processes, (ii) the potential Φ_w corresponding to water storage. To incorporate swelling into the model, we let the potential Φ_s depend on the suction. We take the following form of Φ precisely:

$$\Phi = \Phi_s(\boldsymbol{\varepsilon}_d^e, \varepsilon_v^e, s) + \Phi_w(s) + s\frac{m_w^p}{\hat{\rho}_w}.$$

Consequently the dissipation condition $\mathcal{D}_{sk} \geq 0$ from (3) leads to the following elastic state equations:

$$\boldsymbol{\sigma}_d = \frac{\partial \Phi_s}{\partial \boldsymbol{\varepsilon}_d^e}, \quad p = -\frac{\partial \Phi_s}{\partial \varepsilon_v^e}, \quad \frac{m_w^e}{\hat{\rho}_w} = \frac{\mathrm{d}\Phi_w}{\mathrm{d}s} + \frac{\partial \Phi_s}{\partial s}, \tag{5}$$

and the following condition for the dissipation related to skeleton plastic evolutions [3]:

$$\boldsymbol{\sigma}_d : \dot{\boldsymbol{\varepsilon}}_d^p - p\dot{\varepsilon}_v^p - s\frac{\dot{m}_w^p}{\hat{\rho}_w} \geq 0. \tag{6}$$

Plastic evolutions in soils are caused by irreversible relative sliding of solid grains forming the matrix, so the plastic volumetric strain is only due to plastic changes in the porous volume. This can be written as $\dot{\varepsilon}_v^p = \dot{\phi}^p$, where ϕ^p denotes the Lagrangian plastic porosity [6]. Consequently it seems quite natural to assume that it is possible to introduce a coefficient χ ranging from 0 to 1 such that [6]:

$$\frac{\dot{m}_w^p}{\hat{\rho}_w} = \chi\dot{\phi}^p = \chi\dot{\varepsilon}_v^p. \tag{7}$$

Then by introducing the effective pressure p' as:

$$p' \equiv p + \chi s,$$

one can rewrite inequality (6) into:

$$\boldsymbol{\sigma}_d : \dot{\boldsymbol{\varepsilon}}_d^p - p'\dot{\varepsilon}_v^p \geq 0.$$

This results in the effective stress $\boldsymbol{\sigma}' \equiv \boldsymbol{\sigma}_d - p'\boldsymbol{I}$ as a driving force of plastic strains and it leads to description of the plastic behaviour in terms of this stress.

Besides, the non-negativeness of the dissipation $\mathcal{D}_{\mathrm{w}}$ from (2) associated with water flow through the porous medium can be stated with the aid of (4) as:

$$\mathcal{D}_{\mathrm{w}} = (\nabla s + \hat{\rho}_w \boldsymbol{f}) \cdot \frac{\boldsymbol{w}_w}{\hat{\rho}_w} \geq 0. \tag{8}$$

3 A consistent poroelastoplastic model with swelling

Our consistent HM model is based on the Modified Cam-Clay one. We take the following stress–strain relationship in the unsaturated state:

$$\boldsymbol{\sigma}_d = \boldsymbol{\sigma}_d^0 + 2G\boldsymbol{\varepsilon}_d^e, \quad p = p_0 + \hat{p}\exp\left(-\frac{(1+e_0)\varepsilon_v^{e*}(\varepsilon_v^e, s)}{\kappa}\right) - \hat{p},$$

where G denotes the shear modulus of the skeleton, $\boldsymbol{\sigma}_d^0$ and p_0 the initial stress state, $\hat{p}$ a reference pressure, κ is a positive stiffness parameter and e_0 is the initial void ratio. In addition, ε_v^{e*} is defined by:

$$\varepsilon_v^{e*}(\varepsilon_v^e, s) := \varepsilon_v^e - \varepsilon_{sw}(\varepsilon_v^e, s)$$

with ε_{sw} standing for the swelling strain. To be able to treat even large swelling, we propose in particular:

$$\varepsilon_{sw}(\varepsilon_v^e, s) := (1 + \alpha_s(s)\varepsilon_v^e)\varepsilon_{sw}^c(s), \quad \alpha_s(s) := \alpha_{s0}\frac{s_0 - s}{s_0}$$

for a suitable function ε_{sw}^c and a parameter $\alpha_{s0} > 0$ with s_0 standing for the initial suction.

Concerning the plastic behaviour in the unsaturated case, we take the following yield function f:

$$f(\boldsymbol{\sigma}_d, p', p'_{\mathrm{co}}) = \frac{3}{2M^2}\boldsymbol{\sigma}_d : \boldsymbol{\sigma}_d + \left(p' - \frac{p'_{\mathrm{co}}}{2}\right)^2 - \left(\frac{p'_{\mathrm{co}}}{2}\right)^2,$$

where M denotes the slope of the critical state lines and p'_{co} the effective consolidation pressure. The plastic volume change ε_v^p is considered as a hardening variable, and the size of the yield surface depends also on the suction. We propose in particular:

$$p'_{\mathrm{co}}(\varepsilon_v^p, s) := p^0_{\mathrm{co}} \exp\left(-\frac{(1+e_0)(1-\alpha_s(s)\varepsilon^c_{sw}(s))\varepsilon_v^p}{\lambda-\kappa}\right)\zeta(s) + \chi s,$$

where p^0_{co} stands for the initial consolidation pressure at saturation, ζ is a function describing the dependence of the net consolidation pressure on the suction and λ is a stiffness parameter greater than κ.

Moreover, the associative flow rule is assumed, for the sake of simplicity:

$$\dot{\boldsymbol{\varepsilon}}_d^p = \dot{\Lambda}\frac{\partial f}{\partial \boldsymbol{\sigma}_d} = \dot{\Lambda}\frac{3}{M^2}\boldsymbol{\sigma}_d, \quad \dot{\varepsilon}_v^p = -\dot{\Lambda}\frac{\partial f}{\partial p'} = \dot{\Lambda}(p'_{\mathrm{co}} - 2p').$$

The plastic multiplier $\dot{\Lambda}$ satisfies the usual complementarity conditions:

$$\dot{\Lambda} \geq 0, \quad f \leq 0, \quad \dot{\Lambda} f = 0, \quad \dot{\Lambda}\dot{f} = 0.$$

A consistent hydraulic model is obtained consequently according to relations (5) and (7). Besides, dissipation condition (8) is satisfied by a convenient form of Darcy's law, for instance. And the overall poroelastoplastic constitutive model can be then extended continuously from the unsaturated state to the saturated one [3].

A complete HM model is obtained finally by inserting the resulting constitutive model and Darcy's law into the water mass balance and the equilibrium equation.

References

[1] V.-M. Pulkkanen: *A large deformation model for chemoelastic porous media – Bentonite clay in spent nuclear fuel disposal.* Ph.D. thesis, Aalto University, 2019, https://urn.fi/URN:ISBN:978-952-60-8372-8

[2] O. Coussy: *Poromechanics.* John Wiley & Sons, 2004, https://doi.org/10.1002/0470092718

[3] T. Ligurský: *A thermodynamically consistent fully coupled poroelastoplastic model for swelling clays.* Computers and Geotechnics 178:106903, 2025, https://doi.org/10.1016/j.compgeo.2024.106903

[4] A. Lempinen: *Coupling Swelling and Water Retention Processes in Compacted Bentonite.* Chemical Product and Process Modeling 6(1):4, 2011, https://doi.org/10.2202/1934-2659.1516

[5] I.F. Collins, G.T. Houlsby: *Application of thermomechanical principles to the modelling of geotechnical materials.* Proceedings of the Royal Society A 453(1964), London, 1997, pp. 1975–2001, https://doi.org/10.1098/rspa.1997.0107

[6] O. Coussy, J.-M. Pereira, J. Vaunat: *Revisiting the thermodynamics of hardening plasticity for unsaturated soils.* Computers and Geotechnics 37(1–2), 2010, pp. 207–215, https://doi.org/10.1016/j.compgeo.2009.09.003

UFO 2024
Interactive system for universal functional optimization

L. Lukšan, C. Matonoha, J. Vlček

Institute of Computer Science, Czech Academy of Sciences, Prague

1 Introduction

The UFO system was developed for solving both dense medium-size and sparse large-scale optimization problems. This system can be used for formulation and solution of particular optimization problems, for preparation of specialized optimization subroutines and for designing and testing new optimization methods.

The UFO system uses a macroprocessor with a special input language for generating the UFO control program. This means that arrays in the control program has (variable) dimensions induced by the problem solved and the problem description can have an arbitrary structure.

The UFO system can be used for solving various dense or sparse optimization problems:

- Unconstrained or box constrained optimization.
- Optimization with general linear constraints.
- Optimization with general nonlinear constraints.
- Optimization with complementarity constraints.
- Nonsmooth optimization.
- Global optimization.
- Solution to systems of nonlinear functional equations.
- Solution to systems of ordinary differential equations.

The objective function can have various forms:

- General objective function.
- Linear or quadratic objective function.
- Sum of squares or powers.
- Minimax criterion and l_1 or l_∞ norm.
- Integral criterion containing solution to ordinary differential equations.

The problem can be dense, sparse and partially separable. The UFO system contains many optimization methods that can be divided into the following classes:

- Heuristic methods for small-size problems.
- Conjugate gradient methods for large-scale problems.
- Variable metric methods that update an approximation of the Hessian matrix in every iteration.

- Modified Newton methods that use second order derivatives obtained either analytically or numerically or by automatic differentiation.
- Truncated Newton methods for large-scale problems based on difference approximations of directional derivatives.
- Combined Gauss-Newton and variable metric methods for nonlinear least squares.
- Quasi-Newton methods for dense nonlinear least squares and nonlinear equations.
- Simplex or interior point methods for linear and quadratic programming.
- Bundle methods for nonsmooth optimization.
- Recursive quadratic programming and interior point or nonsmooth equation methods for nonlinear programming.
- Primal interior point and smoothing methods for sparse generalized minimax problems.
- Various methods for global optimization.

These methods can be realized in various forms depending on the stepsize selection:

- Line search methods.
- General trust region methods.
- Cubic regularization methods.
- SQP filter methods (for nonlinear programming problems).

Moreover, various preconditioned iterative methods can be chosen for direction determination.

The system UFO contains several features advantageous for the user:

- Automatic differentiation.
- Facilities for checking problem descriptions.
- Collections of problems for testing optimization methods.
- Graphical environment.
- Interface to the CUTE testing collection.
- Creation of the performance profiles.

The use of the UFO system is demonstrated by the folloving transformer design [2]. We have to minimize objective function

$$\begin{aligned} F(x) &= \max_{1\le i\le 11} f_i(x), \\ f_i(x) &= \left|1 - 2\frac{v_1(x,t_i)}{w_1(x,t_i)+v_1(x,t_i)}\right|, \end{aligned}$$

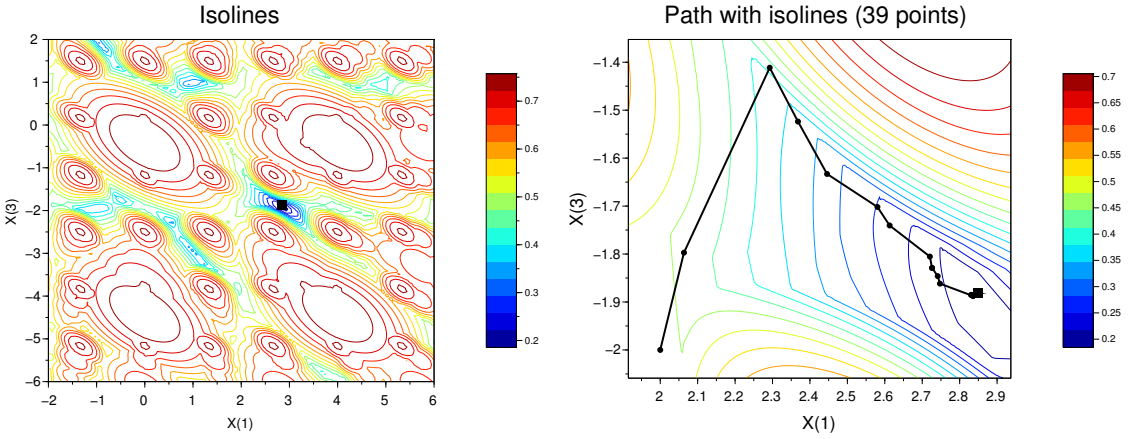
where $v_1(x,t_i)$ and $w_1(x,t_i)$ are complex numbers obtained recursively in such a way that $v_4(x,t_i)=1$, $w_4(x,t_i)=10$ and

$$\begin{aligned} v_k(x,t_i) &= \cos(\vartheta_i x_{2k-1}) v_{k+1}(x,t_i) + j\sin(\vartheta_i x_{2k-1})\frac{1}{x_{2k}} w_{k+1}(x,t_i), \\ w_k(x,t_i) &= \cos(\vartheta_i x_{2k-1}) w_{k+1}(x,t_i) + j\sin(\vartheta_i x_{2k-1}) x_{2k} v_{k+1}(x,t_i) \end{aligned}$$

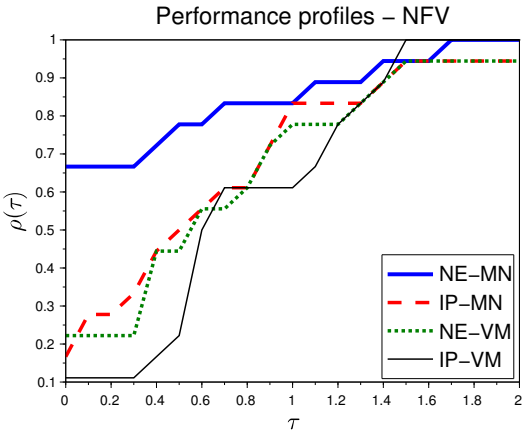
for $k=3,2,1$. Here $j=\sqrt{-1}$ is the imaginary unit and $\vartheta_i = (\pi/2)\, y_i$, $1\le i\le 11$, where the points y_i, $1\le i\le 11$, are shown in the UFO input file.

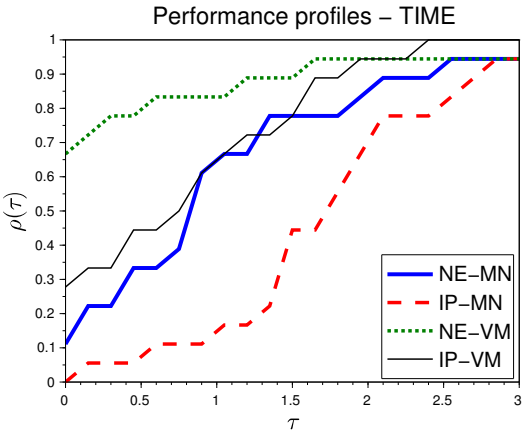
```
$SET(INPUT)
  $ADD(REAL,'\Y(11)')
  X(1)=2.0D0; X(2)=1.5D0; X(3)=-2.0D0
  X(4)=3.0D0; X(5)=0.8D0; X(6)= 6.0D0
  Y(1)=0.5D0; Y(2)=0.6D0; Y(3)=0.7D0; Y(4)=0.77D0
  Y(5)=0.9D0; Y(6)=1.0D0; Y(7)=1.1D0; Y(8)=1.23D0
  Y(9)=1.3D0; Y(10)=1.4D0; Y(11)=1.5D0
$ENDSET
$SET(FMODELA)
  $ADD(REAL,'\TH\CS\SN')
  $ADD(COMPLEX,'\V(4)\W(4)\C1\C2\C3')
  TH=0.5D0*Y(KA)*3.14159265358979324D0
  V(4)=CMPLX(1.0D0,0.0D0); W(4)=1.0D1*V(4)
  DO I=3,1,-1
  CS=COS(TH*X(2*I-1)); SN=SIN(TH*X(2*I-1))
  C1=CMPLX(CS,0.0D0); C2=CMPLX(0.0D0,(SN*X(2*I)))
  C3=CMPLX(0.0D0,(SN/X(2*I)))
  V(I)=C1*V(I+1)+C3*W(I+1); W(I)=C2*V(I+1)+C1*W(I+1)
  END DO
  FA=ABS(1.0D0-2.0D0*V(1)/(V(1)+W(1)))
$ENDSET
$NF=6; $NA=11; $NAL=0;
$MODEL='AM'
$CLASS='VM'; $MET=7
$GRAPH='E'; $ISO='Y'; $HIL='Y'; $PATH='Y'
$BATCH
$STANDARD
```

The results of the development are illustrated by using the following two pictures obtained by the UFO graphical environment.



The system UFO can also be used for testing and comparing various optimization methods. The next figures show performance profiles [1] for interior point [3] and nonsmooth equation [4] methods in the Newton (MN) and the variable metric (VM) realizations. These performance profiles were obtained using the UFO system.





Acknowledgement: This work was supported by the long-term strategic development financing of the Institute of Computer Science (RVO:67985807).

References

[1] E.D. Dolan, J.J. Moré: *Benchmarking optimization software with performance profiles.* Mathematical Programming 91, 2002, pp. 201–213.

[2] L. Lukšan, J. Vlček: *Test problems for nonsmooth unconstrained and linearly constrained optimization.* Technical Report V-798. ICS AS CR, Prague, 2000.

[3] L. Lukšan, C. Matonoha, J. Vlček: *Nonsmooth equation method for nonlinear nonconvex optimization.* In M. Křížek, P. Neittaanmäki, R. Glovinski, S. Korotov (eds.): Conjugate Gradient Algorithms and Finite Element Methods, Springer Verlag, Berlin, 2004, pp. 131–145.

[4] L. Lukšan, C. Matonoha, J. Vlček: *Interior point method for nonlinear nonconvex optimization.* Numerical Linear Algebra with Applications 11, 2004, pp. 431–453.

[5] L. Lukšan, M. Tůma, C. Matonoha, J. Vlček, N. Ramešová, M. Šiška, J. Hartman: *UFO 2024. Interactive System for Universal Functional Optimization.* Technical Report V-1289. ICS AS CR, Prague, 2024.

Accelerating parameter identification in bread baking simulations via model order reduction

A. Kovárnová, M. Isoz, T. Hlavatý, M. Sluková, T. Moucha

University of Chemistry and Technology, Prague
Institute of Thermomechanics, Czech Academy of Sciences, Prague

1 Introduction

Bread has been one of the principal nutriments for humans for thousands of years, with new bread recipes being introduced to the market to this day. To simulate the process of bread baking mathematically, one must take into account heat and mass transfer, evaporation and condensation of water, and deformation of the dough, see, e. g., [1]. Furthermore, published models rely on a large number of empirical parameters, some of which are only provided in the literature as a wide range of values [2].

Estimating correct values of unknown model parameters conveys evaluating the model for a large number of instances and comparing the results to experimental findings. The repeated simulations prove to be computationally expensive; however, computational requirements can be deflated by employing model order reduction (MOR).

The long-term goal of our project is to create a credible model of bread baking and use it to optimize the process from the point of view of energy costs while retaining all of the qualities of the bread. The short-term aim presented in this work is to identify the correct parameters of a preliminary model from the calculated evolution of the dough temperature. To speed up this identification, we utilize MOR; in particular, we combine proper orthogonal decomposition (POD) with interpolation through artificial neural networks (ANN), as done by Hesthaven and Ubbiali [3].

2 Methods

The preliminary model used focuses on internal heat and mass transfer inside the bread. External transport is included through effective transport coefficients. Furthermore, in the first approximation, we consider baking rigid dough, that is, there is no domain deformation.

Full order model – mass transfer The dough is approached as a three-phase pseudohomogeneous system comprising solid wheat (s), liquid water (l) and gas (g). Furthermore, the gas phase is composed of water vapor (v) and carbon dioxide (c). The mass transfer of liquid water is realized by evaporation and subsequent condensation. The mass balance of the liquid is then

$$\frac{\partial(\alpha^{\mathrm{l}}\rho^{\mathrm{l}})}{\partial t} = -\dot{m}^{\mathrm{ev}}, \qquad \dot{m}^{\mathrm{ev}} = k^{\mathrm{ev}}\left(p^{\mathrm{v},*} - p^{\mathrm{v}}\right). \tag{1}$$

Here and in all subsequent equations, α^j is the volume fraction and ρ^j the density of phase $j = \mathrm{l}, \mathrm{g}$. The rate of evaporation $\dot{m}^{\mathrm{ev}}$ is driven by the difference between the saturated vapor pressure $p^{\mathrm{v},*}$ and the local vapor pressure p^{v}. We call the proportionality constant k^{ev} the coefficient of evaporation and set it as one of the two parameters, whose values are sought in the next section.

In the gas phase, the gas is additionally transported along the pressure gradient according to the Darcy permeability law. Furthermore, carbon dioxide is produced via fermentation, $\dot{m}^{\mathrm{gen}}$,

$$\frac{\partial\left(\alpha^{\mathrm{g}}\rho^{\mathrm{g}}\right)}{\partial t}=\nabla\cdot\left(\rho^{\mathrm{g}}\frac{\kappa^{\mathrm{g}}}{\mu^{\mathrm{g}}}\nabla p^{\mathrm{g}}\right)+\dot{m}^{\mathrm{ev}}+\dot{m}^{\mathrm{gen}}_{\mathrm{CO_2}}, \tag{2}$$

where κ^{g} is the Darcy permeability constant, μ^{g} dynamic viscosity of the gas and p^{g} its pressure. Lastly, the mass balance of the water vapor is

$$\frac{\partial\left(\alpha^{\mathrm{g}}\rho^{\mathrm{g}}\omega^{\mathrm{v}}\right)}{\partial t}=\nabla\cdot\left(\rho^{\mathrm{g}}\omega^{\mathrm{v}}\frac{\kappa^{\mathrm{g}}}{\mu^{\mathrm{g}}}\nabla p^{\mathrm{g}}\right)+\nabla\cdot\left(\rho^{\mathrm{g}}D^{\mathrm{eff}}\nabla\omega^{\mathrm{v}}\right)+\dot{m}^{\mathrm{ev}}, \tag{3}$$

where ω^{v} is the mass fraction of water vapor in the gas and D^{eff} the effective diffusion coefficient of the water vapor in $\mathrm{CO_2}$.

Full order model – heat transfer Enthalpy in the system is generated by evaporation and transferred by conduction, with the governing equation being

$$\frac{\partial\left(\sum_j\rho^j\alpha^j c_p^j T\right)}{\partial t}=\nabla\cdot\left(\lambda^{\mathrm{eff}}\nabla T\right)-\dot{m}^{\mathrm{ev}}\Delta H^{\mathrm{ev}},\quad \lambda^{\mathrm{eff}}=\alpha^{\mathrm{s}}\lambda^{\mathrm{s}}+\alpha^{\mathrm{l}}\lambda^{\mathrm{l}}+\alpha^{\mathrm{g}}\lambda^{\mathrm{g}},\quad j=\{\mathrm{s},\mathrm{l},\mathrm{g}\}, \tag{4}$$

where T is temperature, λ^{eff} the effective thermal conductivity, calculated as weighted average of thermal conductivities of all the phases, and H^{ev} the evaporation enthalpy of water. The thermal conductivity λ^{s} was chosen as the second desired parameter.

Model order reduction A widely used family of approaches for MOR involves a reduced basis. The full order model (FOM) is evaluated for several parameter values, from which the basis is extracted, e.g., by proper orthogonal decomposition (POD) [4]. The approximation of the result for previously unseen parameter values is then sought as a superposition of the basis functions. The expansion coefficients can be determined by projection [5] or interpolation [3].

We employ a framework similar to Hesthaven and Ubbiali [3]. To obtain a suitable reduced basis, we use POD, and to create a time- and parameter-continuous model, we utilize regression by artificial neural networks (ANNs).

First, the temperature values are saved in a matrix of snapshots, $Q\in\mathbb{R}^{m\times N_t N_{\lambda^{\mathrm{s}}} N_{k^{\mathrm{ev}}}}$, where each column of the matrix corresponds to the temperature vector for one instance of time and the pair of parameters. Furthermore, m is the spatial dimension, N_t the number of saved times, $N_{\lambda^{\mathrm{s}}}$ the number of simulated parameters λ^{s} and $N_{k^{\mathrm{ev}}}$ the number of simulated parameters k^{ev}. We will denote the total number of snapshots $N=N_t N_{\lambda^{\mathrm{s}}} N_{k^{\mathrm{ev}}}$.

The reduced basis for this matrix is then created via POD, which corresponds to a truncated singular value decomposition, as

$$Q\approx Q^\ell=\Psi^\ell\Sigma^\ell(V^\ell)^{\mathrm{T}}=\Psi^\ell H^\ell,\quad \Psi^\ell=[\boldsymbol{\psi}_1,\ldots,\boldsymbol{\psi}_\ell],\quad H^\ell=[\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_\ell]^{\mathrm{T}}, \tag{5}$$

where $\Psi^\ell\in\mathbb{R}^{m\times\ell}$ is a matrix of the first ℓ left singular vectors $\boldsymbol{\psi}_r$ of Q, which are mutually orthonormal and corresponds to spatial modes (toposes), $\Sigma^\ell\in\mathbb{R}^{\ell\times\ell}$ contains first ℓ singular values and $(V^\ell)^{\mathrm{T}}\in\mathbb{R}^{\ell\times N}$ the first ℓ right singular vectors, which are again mutually orthonormal. We denote the matrix $\Sigma^\ell(V^\ell)^{\mathrm{T}}$ as H^ℓ and the vectors it contains as chronoses. The evolution of the temperature field is then reconstructed as a superposition of toposes $\boldsymbol{\psi}_r$, where the chronoses act as their time- and parameter-dependent amplitudes.

A single chronos $\boldsymbol{\eta}_r\in\mathbb{R}^N$ contains information on the mode behavior in discrete temporal and parametric values. We seek a continuous approximation of the chronoses with the help of curve fitting by ANNs. In particular, the architecture utilized was a multi-layer perceptron with 2 hidden layers made of 40 neurons and activation function sigmoid.

3 Results

The full order model was simulated by our in-house developed CFD solver built on the open source finite-volume C++ library OpenFOAM [6]. The loaf in question was rotationally symmetrical and, to reduce the required computational power, only a 2D wedge was simulated. The domain comprised 7920 finite volume cells and 90 temporal values, 5 values of λ^{s} and 5 values of k^{ev} were saved, leading to a 7920×2250 matrix of snapshots.

For parameter identification, we used experimental data from Zhang et al. [7], in particular the evolution of temperatures on the surface and in the center of the bread. The error between the model and the experiment is defined as

$$\varepsilon_{\mathrm{exp}} = \frac{1}{N_{t,\mathrm{exp}}} \left(\sum_{t=1}^{N_{t,\mathrm{exp}}} \frac{\left(T_{\mathrm{exp}}^{\mathrm{surf}}(t) - T^{\mathrm{surf}}(t)\right)^2}{\max_t \left(T_{\mathrm{exp}}^{\mathrm{surf}}(t)\right)} + \sum_{t=1}^{N_{t,\mathrm{exp}}} \frac{\left(T_{\mathrm{exp}}^{\mathrm{cent}}(t) - T^{\mathrm{cent}}(t)\right)^2}{\max_t \left(T_{\mathrm{exp}}^{\mathrm{cent}}(t)\right)} \right), \tag{6}$$

where temperatures with the subscript exp are from experiment and temperatures without subscript are simulated. The superscript surf denotes the surface values and cent the values in the center of the loaf.

The temperature values ranged from 299 to 420 K. To ensure that the POD modes were not completely dominated by the first mode describing the mean temperature fields, the matrix of snapshots was first scaled, so that the temperature values lied in $[0; 1]$ and the reduced-order model (ROM) output was rescaled back during evaluation.

The ROM was created from the first 10 modes, which corresponded to the relative error in the Frobenius norm $\|Q - Q^{\ell}\|_F / \|Q\|_F = 5.85 \cdot 10^{-7}$. The ROM was then used to minimize the error $\varepsilon_{\mathrm{exp}}$. The particular optimizer chosen was the Nelder-Mead algorithm, as implemented in the *scipy.optimize* Python library. The dependence of $\varepsilon_{\mathrm{exp}}$ on the parameters and the found optimum is depicted on Figure 1a.

The MOR framework includes neural networks that are initiated by random. To reduce the error caused by this randomness, 10 identical ANNs with different seeds used for the initial weight distribution were trained and the final results of the ROM were taken as an average. A plot of the mean value of the ROM output and 95 % confidence bands can be found in Figure 2. The optimum was taken as an average of the optima found by the individual ANNs and was located at $\lambda^{\mathrm{s}} = 0.792$, $k^{\mathrm{ev}} = 0.208$, producing error $\varepsilon_{\mathrm{exp}} = 0.0743$. The resulting temperature profiles can be seen in Figure 1b.

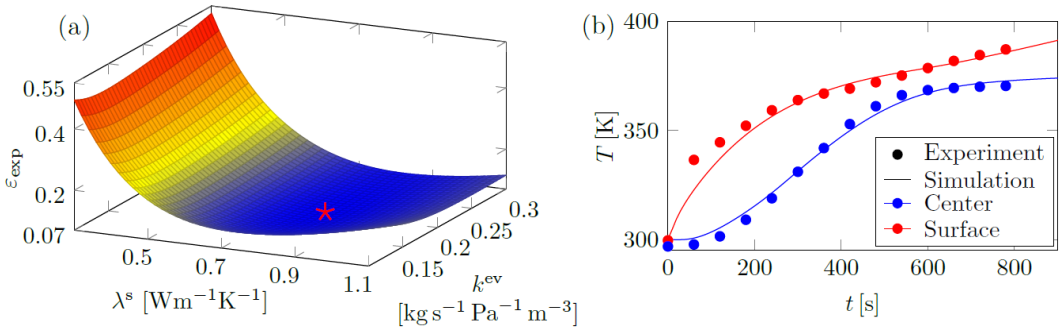


Figure 1: (a) Relative error $\varepsilon_{\mathrm{exp}}$ as a function of λ^{s} and k^{ev}. Prediction from one ANN. The red star corresponds to the optimum. (b) Comparison between experimental results and simulation for the optimal parameters.

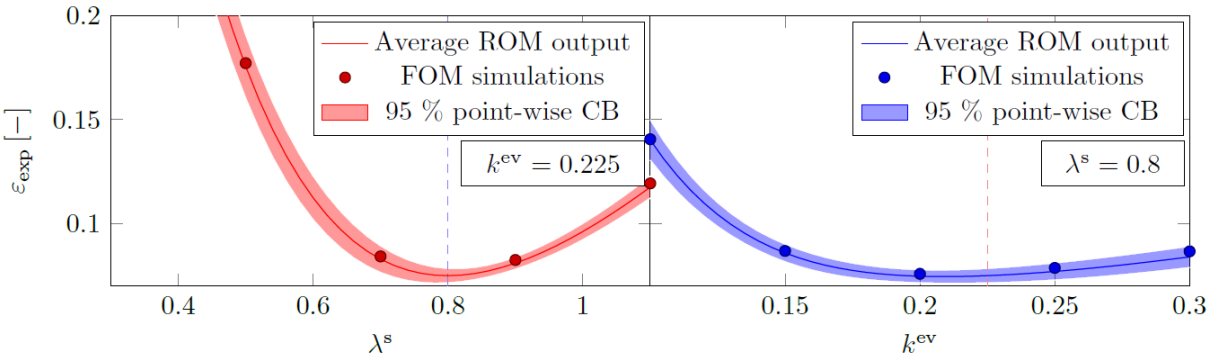


Figure 2: Slice through the ROM-computed error surfaces along $\lambda^{\mathrm{s}} = 0.8$ (left) and $k^{\mathrm{ev}} = 0.225$ (right), mean values and 95 % confidence bands. These values were chosen as they are in the middle of FOM sampling intervals, where the ANN inconsistencies are expected to be the highest. Real FOM values for comparison are given by darker dots.

The whole procedure of 10 mini-optimizations and averaging the results took 29.18 s on a Lenovo ThinkPad X1 Yoga 2nd with a 4-core CPU. On average, each mini-optimization comprised 62 function evaluations. Simulating the FOM for one pair of parameters required 24 minutes on the same computer, meaning cca 3000 times speedup of the optimization. Furthermore, 25 FOM simulations were needed for the construction of the ROM, meaning that the computational time for the ROM construction is less than half of the estimated necessary time for direct optimization.

4 Conclusion

In this contribution, we have performed model order reduction of a model describing baking of bread, and utilized the reduced-order model for identification of unknown model parameters. The framework consists of proper orthogonal decomposition and interpolation via artificial neural networks (ANNs). We have used 10 randomly initiated ANNs and averaged the results to reduce the ANN error. This framework offers cca 3000 times speedup in the online phase and more than 2 times speedup when taking both the ROM evaluation and construction into account.

Acknowledgments: The work was financially supported by the grant with No. QL24010110 of the National Agency for Agricultural Research (NAAR). The authors acknowledge the financial support provided by the Ministry of Education, Youth, and Sports of the Czech Republic via the project No. CZ.02.01.01/00/23_020/0008501 (METEX), co-funded by the European Union. The work was financially supported by the institutional support RVO:61388998.

References

[1] J. Zhang, A.K. Datta: *Mathematical modeling of bread baking process.* Journal of Food Engineering 75(1), 2006, pp. 78–89.

[2] C. Rask: *Thermal properties of dough and bakery products: a review of published data.* Journal of Food Engineering 9(3), 1989, pp. 167–193.

[3] J.S. Hesthaven, S. Ubbiali: *Non-intrusive reduced order modeling of nonlinear problems using neural networks.* Journal of Computational Physics 363, 2018, pp. 55–78.

[4] K. Pearson: *On lines and planes closest fit to systems of points in space.* The London, Edinburgh, and Dublin philosophical magazine and journal of science 2, 1901, pp. 559–572.

[5] R. Pinnau: *Model reduction via Proper Orthogonal Decomposition.* In W.H.A. Schilders, H.A. van der Vorst, J. Rommes (eds.): Model order reduction: theory, research aspects and applications, chapter 2, pages 95–109. Springer Berlin Heidelberg, Berlin, Germany, 2008.

[6] OpenCFD: *OpenFOAM: The Open Source CFD Toolbox.* User Guide. OpenCFD Ltd, UK, 2016.

[7] J. Zhang, A.K. Datta, S. Mukherjee: *Transport processes and large deformation during baking of bread.* AIChE Journal 51(9), 2005, pp. 2569–2580.

Lattice discrete particle model for 3D-printed alloy structures

J. Vorel, A. Jíra, J. Kruis

Department of Mechanics, Faculty of Civil Engineering, Czech Technical University in Prague

1 Introduction

In recent years, 3D printing technology of alloy structures has emerged and allowed the production of specific customised products. In the field of implants, it allows the production of patient-specific inserts, which can ensure better biocompatibility and also prevent so-called stress shielding, which contributes significantly to bone mass loss and creates excessive interface stresses that eventually induce interface debonding and implant loosening. A strong bond with the bone can be achieved by the combination of a porous exterior with a rigid implant core, particularly when customising implants for individual patients [1]. Moreover, it permits the required surface treatment, which is essential to the implant's successful osseointegration. This paper focuses on recent developments in a numerical model based on the lattice discrete particle model (LDPM) [2] to adequately capture the behaviour of 3D-printed alloy structures. LDPM can create and simulate the relevant material at the particle scale, considering the material's underlying mesostructure, including inevitable pores.

2 Lattice discrete particle model

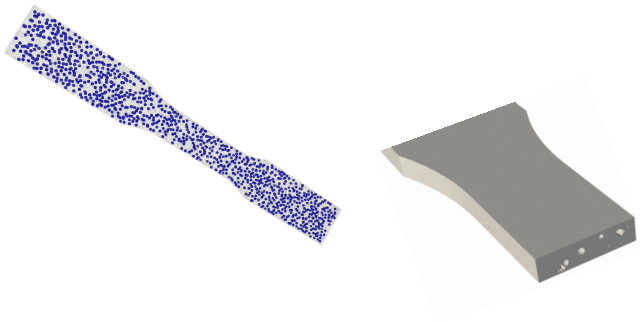
LDPM has proven its applicability for modelling of various materials, i.e., rocks [3] and concrete [2], as well as many engineering problems such as adhesive anchors, prestressed concrete beams, and fibre-reinforced polymer-concrete joints. The material is discretised as a collection of rigid entities, or cells, interacting across the defined facets that separate them. These facets, which are presumed to be between the neighbouring cells, may serve as surfaces for cracks [2]. In contrast to the first LDPM formulation, the current paper requires the 3D-printed metals' particle size distribution to be specified. Figure 1 displays the integrated defects and numerical model.

The model definition is based on stress and strain vectors defined on the facets. The rigid body kinetics is employed to describe the displacement vector associated with the facets [2]. However, the original formulation is not able to recover the full Poisson ratio range ($-1 < \nu < 0.5$) and is limited to $\nu < 0.25$. Therefore, the volumetric-deviatoric split introduced in the Microplane models [4] is considered. The volumetric-deviatoric split allows the recovery of the full Poisson ratio range needed for alloys and other materials. Because of the underlying tetrahedral mesh and corresponding facets Ω_e (see [2]) the volumetric (hydrostatic) strain is calculated as [5]

$$\varepsilon_{Vk} = \frac{1}{3\Omega_{e,0}} \sum_{m \in \mathcal{F}_e} \Gamma_m l_{ij} \varepsilon_{Nm}, \tag{1}$$

where $\Omega_{e,0}$ is the initial volume of the tetrahedral element, $\mathcal{F}_e$ is the set of facets belonging to one element, and Γ_m and l_{ij} are the facet area and distance of the adjacent nodes corresponding to the facet, respectively. ε_{Nm} stands for the normal strain component on the facet m. The normal deviatoric strain for facet k is written as

$$\varepsilon_{NDk} = \varepsilon_{Nk} - \varepsilon_{Vk}. \tag{2}$$



(a) (b)

Figure 1: Dog bone specimen: (a) full model with particle distribution; (b) part of the model with pores/voids.

Moreover, the shear (tangential) strain in the plane of the facet is written as $\varepsilon_{Tk} = (\varepsilon_{Mk}^2 + \varepsilon_{Lk}^2)^{1/2}$, where ε_{Mk} and ε_{Lk} are the shear components in the local coordinate system. The deviatoric strain is defined as $\varepsilon_{Dk} = (\varepsilon_{NDk}^2 + \varepsilon_{Tk}^2)^{1/2}$. The corresponding stress components then read

$$\sigma_V = E_V \varepsilon_N, \quad \sigma_{ND} = E_D \varepsilon_{ND}, \quad \sigma_M = E_D \varepsilon_M, \quad \sigma_L = E_D \varepsilon_L, \tag{3}$$

where $E_V = E/(1-\nu)$ and $E_D = E/(1+\nu)$ are the volumetric and deviatoric moduli, respectively, related to Young's modulus E. The constitutive material law defined on the facets is described in the following section. By imposing the equilibrium through the principle of virtual work, the internal work and nodal forces associated with the facet can be calculated [2]. Note that subscript k is omitted in the following text for readability.

3 LDPM for elastoplasticity

This section introduces a lattice discrete particle model for plasticity. The model is based on the volumetric-deviatoric split, and the approaches outlined in [6]. This equivalent stress-based material model is implemented in MARS software[1] and is also combined with isotropic damage. The model is defined by means of equivalent stress, σ^{eq}, and strain, $\varepsilon^{\mathsf{eq}}$. The equivalent strain has the form

$$\varepsilon^{\mathsf{eq}} = \sqrt{(\varepsilon_V + \alpha\varepsilon_{ND})^2 + \alpha\left(\varepsilon_M^2 + \varepsilon_L^2\right)} = \sqrt{\left(\varepsilon_N^{\mathsf{eq}}\right)^2 + \alpha\varepsilon_T^2}, \tag{4}$$

where $\varepsilon_N^{\mathsf{eq}} = \varepsilon_V + \alpha\varepsilon_{ND}$, α stands for the interaction coefficient. This definition of equivalent normal strain originates from the assumption that $\sigma_N = E_V \varepsilon_N^{\mathsf{eq}}$. Based on the principle of virtual power, we relate the stress components to the equivalent stress as

$$\sigma_N = \sigma^{\mathsf{eq}} \frac{\varepsilon_N^{\mathsf{eq}}}{\varepsilon^{\mathsf{eq}}}, \quad \sigma_M = \sigma^{\mathsf{eq}} \frac{\alpha\varepsilon_M}{\varepsilon^{\mathsf{eq}}}, \quad \sigma_L = \sigma^{\mathsf{eq}} \frac{\alpha\varepsilon_L}{\varepsilon^{\mathsf{eq}}}, \tag{5}$$

and

$$\sigma_V = \sigma^{\mathsf{eq}} \frac{\varepsilon_V}{\varepsilon^{\mathsf{eq}}}, \quad \sigma_{ND} = \sigma^{\mathsf{eq}} \frac{\alpha\varepsilon_{ND}}{\varepsilon^{\mathsf{eq}}}. \tag{6}$$

[1] https://www.es3inc.com/mars-solver/

E [GPa]	ν [-]	σ_{Y0} [MPa]	H [GPa]	κ_0 [-]	l_f [mm]
120	0.3	750	4.4	0.001	0.2

Table 1: Material properties used in current study for titanium alloy

By substituting Equations (5) and (6) into Equation (4), the effective stress is obtained in terms of normal and shear stresses

$$\sigma^{\mathsf{eq}} = \sqrt{\sigma_N^2 + \frac{\sigma_T^2}{\alpha}}, \quad \sigma_T = \sqrt{\sigma_M^2 + \sigma_L^2}. \tag{7}$$

If the elastic behaviour is assumed and taking into account Equations (5) and (6), the stresses are written as

$$\sigma_V = E^{\mathsf{eq}}\varepsilon_V, \quad \sigma_{ND} = \alpha E^{\mathsf{eq}}\varepsilon_{ND}, \quad \sigma_M = \alpha E^{\mathsf{eq}}\varepsilon_M, \quad \sigma_L = \alpha E^{\mathsf{eq}}\varepsilon_L, \tag{8}$$

where $E^{\mathsf{eq}} = \sigma^{\mathsf{eq}}/\varepsilon^{\mathsf{eq}} = E_V$ and thus $\alpha = E_D/E_V = 1 - 2\nu/1 + \nu$. This formulation covers the whole physical range of the Poisson ratio.

In this model, the yield condition is written as

$$f(\underline{\sigma}) = (\sigma^{\mathsf{eq}})^2 - [\sigma_Y(\kappa)]^2 = 0, \tag{9}$$

where κ is the hardening variable, which is related to the plastic multiplier λ through the relation $\dot{\kappa} = \dot{\lambda}$ and yield strength

$$\sigma_Y(\kappa) = \sigma_{Y0} + H\kappa, \tag{10}$$

where σ_{Y0} is the initial yield stress and H is the hardening modulus. When this condition is satisfied, yielding occurs. The radial return is performed on the equivalent stress if $f(\underline{\sigma}) > 0$.

If the elastoplastic model is combined with the isotropic damage, the final stresses are evaluated as

$$\sigma_N = (1-\omega)\sigma_N, \quad \sigma_M = (1-\omega)\sigma_M, \quad \sigma_L = (1-\omega)\sigma_L, \tag{11}$$

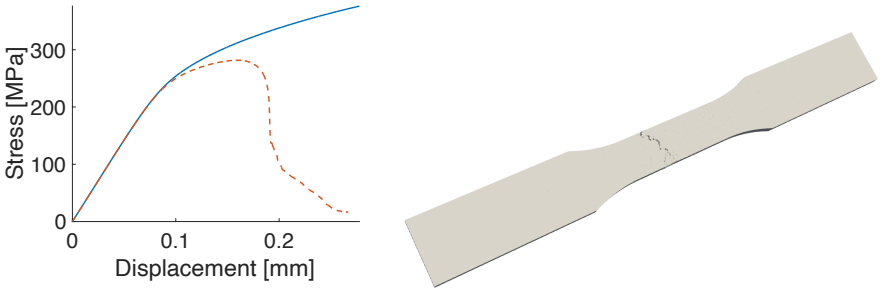
where ω is the damage parameter. The linear softening law is assumed and defined in the form

$$\omega = \frac{\kappa - \kappa_0}{\kappa_f - \kappa_0}, \tag{12}$$

where $\kappa_f = l_f/l_{ij}$, l_f is fracture opening, and κ_0 stands for the damage threshold (onset of damage).

4 Preliminary results

This section summarizes the preliminary findings of the above-described numerical model based on the equivalent stress and volume of pores equal to 15% assumed. The material model parameters used to simulate the uniaxial loading of dogbone specimens are summarised in Table 1. The results obtained for the model with and without damage are presented in Figure 2(a). As can be seen from the presented results, the inserted pores reduced the yield strength of the specimen compared to the raw material. Moreover, the material model with damage is capable of capturing the specimen's failure and crack evolution, see Figure 2(b).



(a) (b)

Figure 2: Uniaxial tension of dog bone specimen: (a) load-displacement diagram - model without damage (solid line); model with damage (dashed line); (b) overall view of the sample.

5 Conclusion

The paper introduces a novel lattice discrete particle model for 3D-printed titanium alloys. By removing cells from the computational model, the intrinsic porosity caused by the printing procedure is taken into account. It's important to note that the imperfections caused by the printing procedure significantly influence the specimen performance, particularly for smaller thicknesses near the printing limits. This research is significant as it presents a numerical model for a deeper understanding of the factors that affect the performance of 3D-printed titanium alloys.

Acknowledgement: The financial support provided by the GAČR grant No. 23-04971S is gratefully acknowledged. The authors would like to acknowledge ES3inc for providing access to MARS, a special-purpose computational framework for the Multiphysics Analysis of the Response of Structures.

References

[1] J.E. Bolander, J. Eliáš, G. Cusatis, K. Nagai: *Discrete mechanical models of concrete fracture.* Engineering Fracture Mechanics 257, paper 108030, 2021.

[2] G. Cusatis, D. Pelessone, A. Mencarelli: *Lattice discrete particle model (LDPM) for failure behavior of concrete. I: Theory.* Cement and Concrete Composites 33(9), 2011, pp. 881–890.

[3] S.E. Ashari, G. Buscarnera, G. Cusatis: *A lattice discrete particle model for pressure-dependent inelasticity in granular rocks.* International Journal of Rock Mechanics and Mining Sciences 91, 2017, pp. 49–58.

[4] I. Carol, Z.P. Bažant: *Damage and plasticity in microplane theory.* International Journal of Solids and Structures 34 (29), 1997, pp. 3807–3835.

[5] G. Cusatis, R. Rezakhani, E.A. Schauffert: *Discontinuous Cell Method (DCM) for the simulation of cohesive fracture and fragmentation of continuous media.* Engineering Fracture Mechanics 170, 2017, pp. 1–22, `https://doi.org/10.1016/j.engfracmech.2016.11.026`

[6] M. Brocca, Z.P. Bažant: *Microplane constitutive model and metal plasticity.* Applied Mechanics Reviews 53(10), 2000, pp. 265–281.

Title: Seminar on Numerical Analysis & Winter School

Proceedings of the conference SNA'25
Ostrava, January 27 – 31, 2025

Editors: Jiří Starý, Stanislav Sysala, Dalibor Lukáš

Published by: Institute of Geonics of the CAS, Ostrava

First electronic edition

Ostrava, 2025

ISBN 978-80-86407-88-3