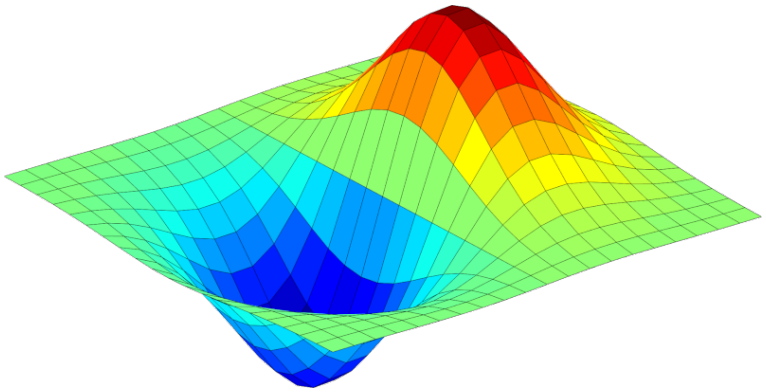
Institute of Geonics of the CAS, Ostrava

SNA'23

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 23 – 27, 2023

Programme committee:

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, 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
Marek Pecha	Institute of Geonics of the CAS, Ostrava
Jiří Starý	Institute of Geonics of the CAS, Ostrava
Stanislav Sysala	Institute of Geonics of the CAS, Ostrava
Dagmar Sysalová	Institute of Geonics of the CAS, Ostrava

Conference secretary:

Dagmar Sysalová	Institute of Geonics of the CAS, Ostrava

Institute of Geonics of the Czech Academy of Sciences
ISBN 978-80-86407-85-2

Preface

Seminar on Numerical Analysis 2023 (SNA'23) held in Ostrava is the sixteenth meeting in a series of SNA events. The previous meetings were held in Ostrava 2003, 2005, Monínec 2006, Ostrava 2007, Liberec 2008, Ostrava 2009, Nové Hrady 2010, Rožnov 2011, Liberec 2012, Rožnov 2013, Nymburk 2014, Ostrava 2015, 2017, 2019, and 2021 (online). SNA events help to join the Czech research community working in the field of numerical mathematics and computer simulations. The scope of the seminar ranges from mathematical modelling and simulation of challenging engineering problems, to methods of numerical mathematics, numerical linear algebra, and high-performance computing.

Going to the history, let us briefly remember two of the key persons of the SNA events, Professors Ivo Marek (1933–2017) and Radim Blaheta (1951–2022). Ivo Marek, a professor at the Charles University and the Czech Technical University in Prague, was for many decades one of the best-known Czech mathematicians, who contributed significantly to the development of computational methods and numerical analysis. At the same time, he was an excellent teacher. SNA 2003 was organized on the occasion of Ivo's seventieth birthday (meaning that he would be ninety this year) and followed on his previous organization effort. Ivo was also a member of Programme Committee of SNA events till 2017.

Radim Blaheta and Ivo Marek. SNA'13, Rožnov pod Radhoštěm.

Radim Blaheta, Ivo's student, was a well-known researcher, the head of a mathematical department and the director at the Institute of Geonics of the Czech Academy of Sciences. At the same time, he was a professor at the VŠB–Technical University of Ostrava. Radim was interested in numerical mathematics, computer sciences, geotechnical and environmental real-world problems and other scientific disciplines. He was an excellent organizer of many national and international scientific events. In particular, he was a co-founder of SNA and the main organizer of all these events held in Ostrava or Rožnov.

Ivo and Radim were very good friends with a very cordial and warm social nature, which significantly influenced the atmosphere of the SNA events. Unfortunatelly, they can no longer be with us, but we keep them in our minds and hearts. More information about them can be found in [1, 2].

This SNA'23 is dedicated to the memory of Prof. Radim Blaheta, who passed away last year. The conference has about 85 participants. The following researchers from abroad accepted our invitation: Prof. Svetozar Margenov (Bulgaria), Prof. Maya Neytcheva (Sweden), Prof. Peter Arbenz (Switzerland), Prof. János Karátson (Hungary), Prof. Yousef Saad (USA, on-line), Prof. Jan Mandel (USA, on-line), and Prof. Bedřich Sousedík (USA, on-line).

The scientific programme of SNA'23 includes the Winter school with the following tutorial lectures focused on selected important topics within the scope of the seminar:

- *M. Béreš*: Solution of PDEs with uncertainties in parameters by the stochastic Galerkin method with geotechnical applications
- *M. Ladecký*: Discrete Green's operator preconditioning: Theory and applications
- *S. Pozza*: Matrix decay phenomenon and its applications
- *J. Stebel*: Poroelasticity: Mathematical modelling, numerical solution and applications
- *Z. Strakoš*: Numerical approximation of the spectrum of self-adjoint operators and an unfinished chat with Radim Blaheta

The Winter school was suggested by Prof. Zdeněk Strakoš (other SNA co-founder) within SNA 2005. Then, it became a regular part of the SNA events. Therefore, we are very glad that Zdeněk has accepted our invitation to give one of tutorial lectures.

Beside the Winter school, SNA'23 includes 38 short oral presentations and 12 posters. Some of the contributions contain memories of Professors Radim Blaheta, Ivo Marek, and also Owe Axelsson, an excellent mathematician well-known over the whole world and our colleague for many years. Owe passed away last year, too.

Finally, let us wish all participants to enjoy the scientific programme, as well as the accompanying social events. We hope you find the lectures interesting and inspiring.

On behalf of the Programme and Organizing Committees of SNA'23,

Jiří Starý and Stanislav Sysala

References

[1] R. Blaheta, M. Tůma: *Professor Ivo Marek.* Applications of Mathematics **62**, 2017, pp. 719–721.

[2] S. Sysala: *Laudation for the 70th birthday of Professor Radim Blaheta.* Mathematics and Computers in Simulation **189**, 2021, pp. 3–4.

Extended Abstracts

Contents

Direct construction of reciprocal mass matrix and higher order finite element method

R. Cimrman[1], *R. Kolman*[1], *J.A. González*[2], *K.C. Park*[3]

[1] Institute of Thermomechanics of the CAS, Prague
[2] Escuela Técnica Superior de Ingeniería, Universidad de Sevilla
[3] Department of Aerospace Engineering Sciences, University of Colorado at Boulder

1 Introduction

When solving dynamical problems of computational mechanics, such as contact-impact problems or cases involving complex structures under fast loading conditions, explicit time-stepping algorithms are usually preferred over implicit ones [3, 1]. The explicit schemes, instead of solving a linear(ized) system with the full implicit scheme-related matrix (including stiffness matrix), require inverting only the effective mass matrix. To avoid solving a linear system in each time step, the explicit schemes are usually used with a mass matrix lumping technique resulting in a diagonal (lumped) mass matrix (LMM). Such a combination is efficient and moreover dispersion errors in wave propagation are partially eliminated by using the largest possible time step size that is allowed by a scheme's conditional stability. However, the lumping schemes for higher order FE approximations constitute an open problem, as demonstrated e.g. in [2], where the row-sum lumping algorithm [1] was used with the isogeometric analysis (IGA) and its accuracy was limited to second order even for higher orders of the NURBS basis. The row-sum lumping method in particular leads, for higher order approximations, to negative LMM entries/eigenvalues when used with the standard Lagrange polynomial or serendipity bases.

An alternative to lumping with advantageous properties, the reciprocal mass matrix (RMM) is an inverse mass matrix that has the same sparsity structure as the original consistent mass matrix (CMM), preserves the total mass, captures well the desired frequency spectrum and leads thus to efficient and accurate calculations. The RMM concept was introduced in [8], including selective mass scaling. In [4] a new RMM formulation via the method of Localized Lagrange multipliers (LLM) has been proposed, see [7] and [6]. This formulation was then extended for the IGA in [5]. The LLM-based RMM has the form:

$$\mathbf{M}^{-1} = \mathbf{A}^{-\mathrm{T}}\mathbf{C}\mathbf{A}^{-1} \,, \tag{1}$$

where $\mathbf{A}$ is a diagonal dual-basis projection matrix and $\mathbf{C}$ is an element-by-element assembled reciprocal mass matrix. Efficiently constructing the diagonal $\mathbf{A}$ is the key issue of the algorithm. It was shown in [4] that $\mathbf{A}$ corresponds to the row-sum lumped mass matrix. Thus, the challenges of higher order lumping apply also to the RMM.

2 The RMM Algorithm

The three-field variational form of the Hamilton's principle of the minimal action for constrained elastodynamics [4] is defined in terms of the displacement $\boldsymbol{u}$ and momentum $\boldsymbol{p} \equiv \rho\boldsymbol{v}$ fields and the Lagrange multipliers $\boldsymbol{\ell}$. After the FE discretization

$$\boldsymbol{u}(\boldsymbol{\xi}) = \mathbf{N}_u(\boldsymbol{\xi})\,\mathbf{u}, \qquad \boldsymbol{p}(\boldsymbol{\xi}) = \mathbf{N}_p(\boldsymbol{\xi})\,\mathbf{p}, \qquad \boldsymbol{\ell}(\boldsymbol{\xi}) = \boldsymbol{\delta}(\boldsymbol{\xi} - \boldsymbol{\xi}_i)\,\boldsymbol{\lambda} \,, \tag{2}$$

the following semi-discrete system is obtained:

$$\mathbf{A}\dot{\mathbf{p}} + \mathbf{B}\boldsymbol{\lambda} = \mathbf{f} - \mathbf{K}\mathbf{u} \equiv \mathbf{r}\,, \qquad \textit{Equilibrium equation} \tag{3}$$

$$\mathbf{A}^{\mathsf{T}}\dot{\mathbf{u}} - \mathbf{C}\mathbf{p} = \mathbf{0}\,, \qquad \textit{Momentum equation} \tag{4}$$

$$\mathbf{B}^{\mathsf{T}}\mathbf{u} = \bar{\mathbf{u}}\,, \qquad \textit{Boundary conditions} \tag{5}$$

where the projection matrix $\mathbf{A}$ and reciprocal mass matrix $\mathbf{C}$ are assembled from

$$\mathbf{A}_e = \int_{\Omega_e} \mathbf{N}_u^{\mathsf{T}}\, \mathbf{N}_p \, d\Omega\,, \qquad \mathbf{C}_e = \int_{\Omega_e} \tfrac{1}{\rho}\mathbf{N}_p^{\mathsf{T}}\, \mathbf{N}_p \, d\Omega\,, \tag{6}$$

$\mathbf{B}$ is the boundary assembly operator, $\mathbf{f}$ the external forces vector and $\mathbf{K}$ the stiffness matrix. By eliminating $\dot{\mathbf{p}} = \mathbf{C}^{-1}\mathbf{A}^{\mathsf{T}}\ddot{\mathbf{u}}$ from (3) we have

$$\left(\mathbf{A}\mathbf{C}^{-1}\mathbf{A}^{\mathsf{T}}\right)\ddot{\mathbf{u}} + \mathbf{B}\boldsymbol{\lambda} = \mathbf{r}\,, \tag{7}$$

which inspires the RMM in the form

$$\mathbf{M}^{-1} = \mathbf{A}^{-\mathsf{T}}\mathbf{C}\mathbf{A}^{-1}\,. \tag{8}$$

The local dual shape functions [4]

$$\mathbf{N}_p(\boldsymbol{\xi}) = \rho(\boldsymbol{\xi})\mathbf{N}_u(\boldsymbol{\xi})(\mathbf{M}_e^A)^{-1}\mathbf{M}_e^L \tag{9}$$

defined in terms of the averaged element mass matrix (AMM)

$$\mathbf{M}_e^A = \beta\mathbf{M}_e^L + (1-\beta)\mathbf{M}_e^C, \text{where } \beta \in [0,1]\,, \tag{10}$$

and the lumped element mass matrix $\mathbf{M}_e^L$ lead to the diagonal

$$\mathbf{A}_e = \int_{\Omega_e} \mathbf{N}_u^{\mathsf{T}}\mathbf{N}_p \, d\Omega = \Big(\int_{\Omega_e} \rho(\boldsymbol{\xi})\mathbf{N}_u^{\mathsf{T}}\mathbf{N}_u \, d\Omega\Big)\,(\mathbf{M}_e^A)^{-1}\mathbf{M}_e^L = \mathbf{M}_e^L \tag{11}$$

and $\mathbf{C}_e = \mathbf{A}_e^{\mathsf{T}}\mathbf{M}_e^{-1}\mathbf{A}_e$. After assembling $\mathbf{A} = \mathop{\mathsf{A}}_{e=1}^{N_e} \mathbf{A}_e$, $\mathbf{C} = \mathop{\mathsf{A}}_{e=1}^{N_e} \mathbf{C}_e$, the RMM can be easily computed using (8).

3 Example: Transient Simulation

The following example illustrates a single aspect of the RMM approach — the computational efficiency. It simulates an impact on a rigid wall of a 2D block domain (100×100 elements) of dimensions 5×5 mm^2, with the elastic properties $E = 200$ GPa, $\nu = 0.3$ and the density $\rho = 7800$ kg/m^3 and the initial velocity $\dot{\boldsymbol{u}} = [-1, 0]$ m/s.

A critical time step for each setting — the Lagrange bases of orders 1–3, AMM/RMM and $\beta \in \{0, 0.5, 1\}$ — is determined using an eigenvalue analysis of the maximum angular frequency $\omega_{\max}$ as $2/\omega_{\max}$. The explicit central difference method is used for the time integration. The y-axis displacement of the top-right corner is traced for $t \in [0, 3]$ μs, denoted below by $u_y(t)$.

The RMM results agree very well with AMM for the same settings of other parameters, as shown in Fig. 1. Timing results are summarized in Tab. 1 for the approximation orders 1–3. Note that AMM($\beta = 0$) is CMM, while AMM($\beta = 1$) is LMM and RMM($\beta = 1$) is equivalent to LMM.

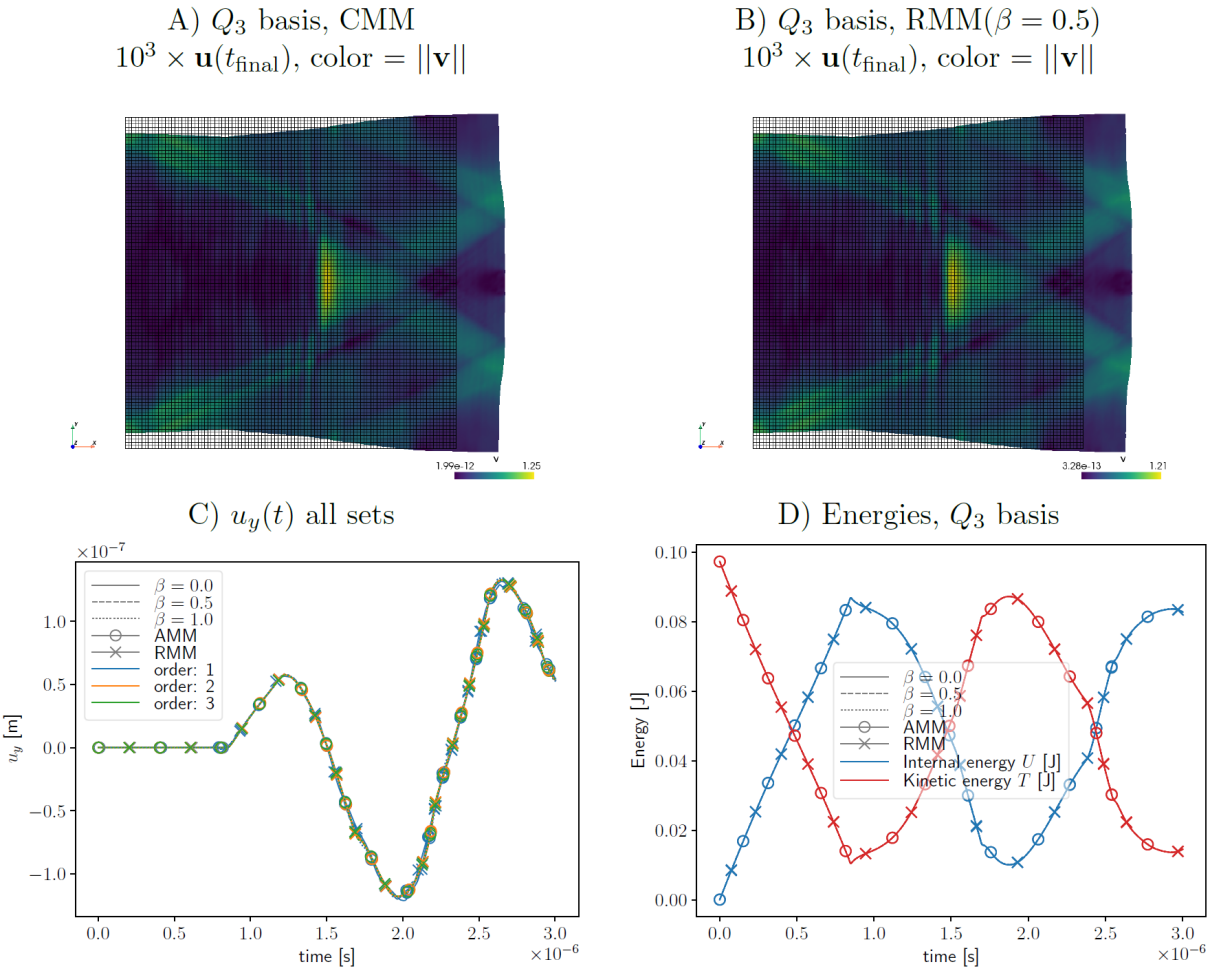


Figure 1: Impact problem. A), B) Final states for CMM resp. RMM($\beta = 0.5$) C) Top-right corner displacement time histories $u_y(t)$ for all parameter sets. D) The energies calculated with the approximation order 3.

The solutions are represented by the error defined as $||e|| \equiv (\int_0^{t_{\text{final}}} (u_y(t) - u_y^{ref}(t))^2 \, \mathrm{d}t)^{\frac{1}{2}}$, where the time histories $u_y(t)$, see Fig. 1-C, are compared to the reference solution, chosen arbitrarily[1] as the solution corresponding to the order 1, $\beta = 0$ and CMM. Other result columns show the critical time step, the total number of steps, the solution elapsed time t_{sol} and the ratios $\frac{t_{\text{sol}}^{\text{RMM}}}{t_{\text{sol}}^{\text{AMM}}}$.

4 Conclusion

We have presented the key notation and concepts of the RMM algorithm and outlined the lumping-related challenges for using the RMM algorithm with higher order FEM. The RMM algorithm's efficiency was numerically demonstrated. The conference presentation will focus on the practical usability of the RMM in connection with higher order FEM and illustrate the issues using numerical examples.

Acknowledgement: This work has been supported by the grant 23-06220S of the Czech Science Foundation.

[1] The purpose of this column is to compare the closeness of RMM and AMM solutions with other parameters being the same. The reference solution is not the exact solution.

order	#DOFs	beta	alg.	Δt [ns]	N_{step}	t_{sol} [s]	$\|\|e\|\|$	$\frac{t_{\text{sol}}^{\text{RMM}}}{t_{\text{sol}}^{\text{AMM}}}$ [1]
1	20301	0.0	AMM	4.3	694	7.1	0	-
			RMM	3.9	763	1.2	$3.86 \cdot 10^{-12}$	0.17
		0.5	AMM	6.9	437	4.6	$2.08 \cdot 10^{-12}$	-
			RMM	6.9	438	0.7	$1.47 \cdot 10^{-12}$	0.15
		1.0	AMM	8.4	357	0.5	$3.02 \cdot 10^{-12}$	-
			RMM	8.4	357	0.5	$3.02 \cdot 10^{-12}$	0.96
2	80601	0.0	AMM	1.9	1557	74.6	$2.55 \cdot 10^{-12}$	-
			RMM	1.8	1673	10.4	$2.55 \cdot 10^{-12}$	0.14
		0.5	AMM	2.7	1121	53.8	$2.54 \cdot 10^{-12}$	-
			RMM	2.7	1127	6.9	$2.54 \cdot 10^{-12}$	0.13
		1.0	AMM	3.1	960	4.7	$2.53 \cdot 10^{-12}$	-
			RMM	3.1	960	4.9	$2.53 \cdot 10^{-12}$	1.04
3	180901	0.0	AMM	1.1	2640	271.6	$2.54 \cdot 10^{-12}$	-
			RMM	1.1	2811	49.2	$2.46 \cdot 10^{-12}$	0.18
		0.5	AMM	1.6	1925	197.4	$2.54 \cdot 10^{-12}$	-
			RMM	1.6	1929	34.3	$2.53 \cdot 10^{-12}$	0.17
		1.0	AMM	1.7	1749	23.9	$2.55 \cdot 10^{-12}$	-
			RMM	1.7	1749	24.1	$2.55 \cdot 10^{-12}$	1.01

Table 1: Comparison of solution times.

References

[1] T. Belytschko, W.K. Liu, B. Moran: *Nonlinear Finite Elements for Continua and Structures.* John Wiley and Sons, Chichester, 2000.

[2] J.A. Cottrell, A. Reali, Y. Bazilevs, T.J.R. Hughes: *Isogeometric analysis of structural vibrations.* Computer Methods in Applied Mechanics and Engineering, 195(41-43):5257–5296, August 2006.

[3] M. Dokainish, K. Subbaraj: *A survey of direct time-integration methods in computational structural dynamics – I. Explicit methods.* Computers & Structures, 32(6):1371–1386, 1989.

[4] J.A. González, R. Kolman, S.S. Cho, C.A. Felippa, K.C. Park: *Inverse mass matrix via the method of localized lagrange multipliers.* International Journal for Numerical Methods in Engineering, 113(2):277–295, 2018.

[5] J.A. González, J. Kopačka, R. Kolman, S.S. Cho, K.C. Park: *Inverse mass matrix for isogeometric explicit transient analysis via the method of localized Lagrange multipliers.* International Journal for Numerical Methods in Engineering, 117(9):939–966, 2019.

[6] J.A. González, K.C. Park, C. Felippa, R. Abascal: *A formulation based on localized lagrange multipliers for bem-fem coupling in contact problems.* Computational Methods in Applied Mechanics and Engineering, 197:623–640, 2008.

[7] K.C. Park, C.A. Felippa, U.A. Gumaste: *A localized version of the method of lagrange multipliers and its applications.* Computational Mechanics, 24(6):476–490, 2000.

[8] A. Tkachuk, M. Bischoff: *Direct and sparse construction of consistent inverse mass matrices: general variational formulation and application to selective mass scaling.* International Journal for Numerical Methods in Engineering, 101(6):435–469, 2015.

Block preconditioning with approximate inner solvers for incompressible flow problems based on IgA discretization

J. Egermaier, H. Horníková

Department of Mathematics, Faculty of Applied Sciences, University of West Behemia, Pilsen

1 Introduction

We focus on efficient numerical solution of the steady incompressible Navier–Stokes equations (NSE) using our in-house solver based on the isogeometric analysis (IgA) approach. The B-spline/NURBS discretization basis has several specific properties different from standard finite element basis, most importantly a higher interelement continuity leading to denser matrices. Our aim is also to developed efficient solver of these systems by a preconditioned Krylov subspace method. Based on our comparison of ideal versions of several state-of-the-art block preconditioners for linear systems arising from the IgA discretization of the incompressible NSE [2], suitable candidates have been selected. In this contribution, we focus on efficient approximate solvers suitable for solving subsystems within these preconditioners.

2 Numerical model and block preconditioning

The mathematical model on bounded domain $\Omega \subset \mathbf{R}^d$ is based on the incompressible Navier–Stokes equations together with boundary conditions

$$\begin{aligned} -\nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{0} \quad \text{in } \Omega, \\ \nabla\cdot\mathbf{u} &= 0 \quad \text{in } \Omega, \end{aligned} \tag{1}$$

where $\mathbf{u}$ is the flow velocity, p is the kinematic pressure and ν is the kinematic viscosity. The nonlinear problem (1) is linearized by Picard method and discretized using isogeometric analysis approach, see [1] for details. We limit ourselves to the B-spline discretization basis in this work. The discretization, similarly to finite element method, leads to a sparse non-symmetric linear system of saddle-point type

$$\begin{bmatrix} \mathbf{F} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \tag{2}$$

where $\mathbf{F}$ is block diagonal with the diagonal blocks consisting of the discretization of the viscous term and the linearized convective term, $\mathbf{B}^T$ and $\mathbf{B}$ are discrete gradient and negative divergence operators, respectively. Krylov subspace methods are the most commonly used in similar applications and can be very efficient if combined with a good preconditioning technique. Since our matrices are non-symmetric, we choose a Krylov subspace method GMRES.

In contrast of standard finite element method, the B-spline basis is generally of higher continuity across the element boundaries. This leads to denser matrices, which makes the linear system more expensive to solve. We are interested in the convergence behavior of the preconditioned GMRES with several block preconditioners based on the decomposition

$$\begin{bmatrix} \mathbf{F} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{B}\mathbf{F}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{F} & \mathbf{0} \\ \mathbf{0} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{F}^{-1}\mathbf{B}^T \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \tag{3}$$

where $\mathbf{S} = -\mathbf{B}\mathbf{F}^{-1}\mathbf{B}^T$ is the Schur complement, which is approximated in different ways. The tested preconditioners are LSC (Least-Squares Commutator), PCD (Pressure Convection-Diffusion), and MSIMPLER (Modified Semi-Implicit Method for Pressure Linked Equations). An overview of these preconditioners can be found e.g. in [3].

3 Numerical experiments

We present comparisons of the iterative solution of the linear systems mentioned above for a well known benchmark problem of flow over the backward facing step in 2D. The linear systems are solved with the GMRES method with no restarts preconditioned with the LSC, PCD and MSIMPLER preconditioners. The main attention is devoted to the efficiency of approximate inner solvers for solving subsystems within these preconditioners. We compare the convergence of GMRES for different configurations of solvers, discretizations with various degree and continuity, different values of viscosity and different mesh refinements.

We consider inner solvers based on geometric multigrid and compare several approaches based on different smoothers - "standard" Gauss–Seidel, Macro Gauss–Seidel, SCMS (Subspace Corrected mass smoother) and ILUT (incomplete LU decomposition with double threshold strategy). We also include an inner solver based on several iterations of a preconditioned Krylov subspace method in the comparison. This approximate solvers are compared with respect to the influence on GMRES convergence and also with respect to the efficiency of computation.

4 Conclusion

It turns out that the multigrid method with the ILUT smoother is an effective approximate solver of systems with the matrix $\mathbf{F}$. This method can also be used for solving systems within Schur complement approximations, but even in cases of higher degree of discretization and low continuity, a higher fill factor is needed and thus the method is not so efficient. Efficiency of other smoothers depends on the chosen discretization. The PCD preconditioner turns out to be very robust considering the possibilities of using different approximate solvers.

Acknowledgement: This work has been supported by Technology agency of the Czech Republic by the grant TK04020250.

References

[1] B. Bastl, M. Brandner, J. Egermaier, K. Michálková, E. Turnerová: *Isogeometric analysis for turbulent flow*. Mathematics and Computers in Simulation 145, 2018, pp. 3–17.

[2] H. Horníková, C. Vuik, J. Egermaier: *A comparison of block preconditioners for isogeometric analysis discretizations of the incompressible Navier–Stokes equations*. In: International Journal for Numerical Methods in Fluids 93, 2021, pp. 1788–1815

[3] A. Segal, M. Rehman, C. Vuik: *Preconditioners for incompressible Navier–Stokes solvers*. Numerical Mathematics: Theory, Methods and Applications 3, 2010, pp. 245–275.

Guaranteed lower bounds to effective PDE parameters

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

Czech Technical University in Prague

1 Introduction

We are interested in computing guaranteed lower bounds of the effective (homogenized) coefficients of steady state heat and linear elasticity operators defined in 2D or 3D domains. Specifically, we compare several finite element (FE) approximation spaces of the respective dual problems. For the heat equation defined on 3D domains, we show that these spaces can be built from the standard Lagrange as well as from the Nédélec [1, 3, 4] FE basis functions. However, the latters yield larger systems of linear equations structured less favorably for FFT based preconditioners. We also show that Lagrange FEs cannot be used for linear elasticity dual problem; instead we employ the Bogner-Fox-Schmit functions [5].

2 Heat equation, duality, Lagrange and Nédélec FE fields

Solving the homogenized (effective) conductivity reads to find $C^{\text{eff}} \in \mathbb{R}^{d\times d}$ such that

$$\alpha^T C^{\text{eff}} \alpha = \frac{1}{|Y|} \min_{u\in H^1_{\text{per}}} \int_Y (C(\alpha+\nabla u), \alpha+\nabla u)_{\mathbb{R}^d} \,\mathrm{d}x, \tag{1}$$

where $\alpha \in \mathbb{R}^d$, $Y \subset \mathbb{R}^d$ is a rectangle or a hexahedron, $d = 2$ or 3, respectively, and $C : Y \to \mathbb{R}^{d\times d}$; see e.g. [6]. The variational approach and any FE discretization of (1) naturally yield an upper bound to C^{eff} which would be guaranteed if the solution was exact. The lower bounds to C^{eff} can be obtained by solving the dual problem: find $C^{\text{eff,dual}}$ such that

$$\beta^T C^{\text{eff,dual}} \beta = \frac{1}{|Y|} \min_{v\in H^{\text{div},0}_{\text{per,mean},0}} \int_Y (C^{-1}(\beta+v), \beta+v)_{\mathbb{R}^d} \,\mathrm{d}x, \tag{2}$$

holds for any $\beta \in \mathbb{R}^d$. Since $(C^{\text{eff,dual}})^{-1} = C^{\text{eff}}$ and any FE discretization yields an upper bound to $C^{\text{dual,eff}}$, we thus obtain a desired lower bound to C^{eff}. Provided that $\beta = C^{\text{eff}}\alpha$, the minimizers $u^0 \in H^1_{\text{per}}$ of (1) and $v^0 \in H^{\text{div},0}_{\text{per,mean},0}$ of (2), respectively, are connected via

$$C^{\text{eff}}\alpha + v^0 = C(\alpha + \nabla u^0), \quad \alpha \in \mathbb{R}^3. \tag{3}$$

Derivation of the dual problem as well as choices of approximation spaces and practical implementations of numerical methods were presented in [3, 2, 6]. The trivial upper and lower bounds are obtained by setting $u = 0$ and $v = 0$ in (1) and (2), respectively,

$$\left(\frac{1}{|Y|}\int_Y C^{-1}\,\mathrm{d}x\right)^{-1} \le C^{\text{eff}} \le \frac{1}{|Y|}\int_Y C\,\mathrm{d}x. \tag{4}$$

For numerical solutions, (3) does not hold[2]. To obtain an approximation to v^0 from u^0 we must first get a conforming approximation $\widetilde{v^0} \in H^{\text{div},0}_{\text{per,mean},0}$ to v^0 and then minimize (2) with respect to 1D subspace of $H^{\text{div},0}_{\text{per,mean},0}$ generated by $\widetilde{v^0}$.

[2] If u^0 is a FE minimizer of (1), then v^0 may not be conforming, because it only fulfills $\int_Y \nabla\phi \cdot v^0\,\mathrm{d}x = \int_Y \nabla\phi \cdot C(\alpha+\nabla u^0)\,\mathrm{d}x = 0$ for all FE basis functions ϕ, but not for all functions from H^1_{per}.

To obtain an approximate solution of (1) we usually use the FE method with Lagrange basis functions, i.e. continuous and piecewise linear functions on a triangular or tetrahedral mesh. Assuming element-wise constant data, we can compute the stiffness matrix and the right hand side vector exactly. The smallest guaranteed upper bound to C^{eff} can then be obtained from the exact solution of the linear system. To solve (2) numerically, we can use Lagrange FEs again. Assuming element-wise constant data, we can obtain the matrix and the right hand side exactly, and thus we get guaranteed lower bounds to C^{eff}. Note that piece-wise linear and continuous fields are sufficient to yield conforming approximation of $H^{\text{div},0}_{\text{per,mean},0}$ (by their partial derivatives) defined on 3D domains. Details of this approach as well as the theoretical background can be found e.g. in [2]. First order Nédélec FE fields defined on triangles [3, 4] are usually used to approximate H^{div} spaces. In 2D, these fields have six DOFs in every element, and the zero divergence condition imposes an additional restriction which can be enforced by a projection or using the Lagrange multipliers. Periodic boundary conditions and zero average condition must also by applied. These additional requirements prevent obtaining the system that is favorable for FFT-based preconditioners.

Based on the above conclusions, we can obtain the lower and upper bounds to the effective conductivity C^{eff} in a few different ways. In particular, we can compare:

- upper bounds $C_n^{\text{eff,U}}$ obtained by discretizing (1) by using Lagrange FEs
- lower bounds $C_n^{\text{eff,L,Ned}}$ obtained by discretizing (2) by using Nédélec FEs
- lower bounds $C_n^{\text{eff,L,Lag}}$ obtained by discretizing (2) by using Lagrange FEs
- trivial lower bounds $C_n^{\text{eff,L,triv}}$ obtained by using $v = 0$ in (2), i.e. using the leftmost term in (4)
- lower bounds $C_n^{\text{eff,L,proj}}$ obtained by getting the nearest (in L^2 sense) conforming field $\widetilde{v}_n^0$ to the solution of discretized (3)
- lower bounds $C_n^{\text{eff,L,proj,optim}}$ obtained by optimizing (2) in the 1D space generated by $\widetilde{v}_n^0$

For this set of lower and upper bounds to C^{eff} we have the following inequalities. The inequalities between matrices are defined as

$$A \leq B \quad \Longleftrightarrow \quad \alpha^T A \alpha \leq \alpha^T B \alpha,\ \alpha \in \mathbb{R}^d.$$

Lemma 1. The following inequalities hold true

$$\begin{array}{rrcl} & C_n^{\text{eff,L,triv}} \leq C_n^{\text{eff,L,Ned}} & \leq & C_n^{\text{eff,U}} \\ C_n^{\text{eff,L,proj}} \leq C_n^{\text{eff,L,proj,optim}} & \leq C_n^{\text{eff,L,Lag}} & \leq & C_n^{\text{eff,U}} \\ & C_n^{\text{eff,L,triv}} & \leq & C_n^{\text{eff,L,proj,optim}}. \end{array}$$

Proof. The proof can be found in the manuscript which is currently being prepared.

Example 1. Let us consider $Y = (-\pi, \pi) \times (-\pi, \pi)$, and the mesh $n_1 = n_2 = 16$ or 32. Let the conductivity tensor be

$$C(x_1, x_2) = \begin{pmatrix} 4.1 + \text{sign}(\sin(2x_1)\cos(2x_2)) & -2 - \text{sign}(\cos(2x_2)) \\ -2 - \text{sign}(\cos(2x_2)) & 5 + \text{sign}(\sin(2x_1)\cos(2x_2)) \end{pmatrix}.$$

The obtained elements of lower and upper bounds to C^{eff} are summarized in the following table, where the columns L-triv, L-proj, L-proj-optim, L-Lag, L-Ned, and U-Lag correspond to $C_n^{\text{eff,L,triv}}$, $C_n^{\text{eff,L,proj}}$, $C_n^{\text{eff,L,proj,optim}}$, $C_n^{\text{eff,L,Lag}}$, $C_n^{\text{eff,L,Ned}}$, $C_n^{\text{eff,U}}$, respectively. The approximate inequality symbol $\lesssim$ denotes that the inequality is not guaranteed, but it is likely because one approximation space has more DOFs that the other.

$n_1 = n_2 = 16$	L-triv	$\lesssim$	L-proj	$\leq$	L-proj-optim	$\leq$	L-Lag	$\lesssim$	L-Ned	$\leq$	U-Lag
$(C_n^{\text{eff}})_{1,1}$	3.3681		3.6737		3.6738		3.7038		3.7557		3.8052
$(C_n^{\text{eff}})_{2,2}$	4.2476		4.7911		4.7915		4.8221		4.8578		4.9050
$(C_n^{\text{eff}})_{1,2}$	-2.4175		-2.0826		-2.0804		-2.0702		-2.0541		-2.0278
$n_1 = n_2 = 32$	L-triv	$\lesssim$	L-proj	$\leq$	L-proj-optim	$\leq$	L-Lag	$\lesssim$	L-Ned	$\leq$	U-Lag
$(C_n^{\text{eff}})_{1,1}$	3.3681		3.7259		3.7259		3.7364		3.7576		3.7769
$(C_n^{\text{eff}})_{2,2}$	4.2476		4.8339		4.8339		4.8445		4.8596		4.8777
$(C_n^{\text{eff}})_{1,2}$	-2.4175		-2.0661		-2.0660		-2.0597		-2.0530		-2.0421

Example 2. Let us consider $Y = (-\pi, \pi) \times (-\pi, \pi)$, and the mesh $n_1 = n_2 = 16$ or 32. Let the coductivity tensor be

$$C(x_1, x_2) = (1 + 0.45\,(\text{sign}(\sin(2x_1)) + \text{sign}\,(\cos(2x_2)))\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

The obtained lower and upper bounds to C^{eff} are shown in the following table.

$n_1 = n_2 = 16$	L-triv	$\lesssim$	L-proj	$\leq$	L-proj-optim	$\leq$	L-Lag	$\lesssim$	L-Ned	$\leq$	U-Lag
$(C_n^{\text{eff}})_{1,1}$	0.3193		0.7587		0.7607		0.7662		0.7708		0.7765
$(C_n^{\text{eff}})_{2,2}$	0.3193		0.7587		0.7607		0.7662		0.7708		0.7765
$(C_n^{\text{eff}})_{1,2}$	0		0.0000		0.0000		0.0000		0.0000		0.0000
$n_1 = n_2 = 32$	L-triv	$\lesssim$	L-proj	$\leq$	L-proj-optim	$\leq$	L-Lag	$\lesssim$	L-Ned	$\leq$	U-Lag
$(C_n^{\text{eff}})_{1,1}$	0.3193		0.7672		0.7674		0.7696		0.7712		0.7732
$(C_n^{\text{eff}})_{2,2}$	0.3193		0.7672		0.7674		0.7696		0.7712		0.7732
$(C_n^{\text{eff}})_{1,2}$	0		0.0000		0.0000		0.0000		0.0000		0.0000

3 Linear elasticity, duality and Bogner-Fox-Schmit functions

Homogenized stiffness tensor $C^{\text{eff}} \in \mathbb{R}^{3\times 3}$ can be obtained as

$$\alpha^T C^{\text{eff}} \alpha = \frac{1}{|Y|} \min_{u \in (H^1_{\text{per}})^2} \int_Y (C(\alpha + \partial u), \alpha + \partial u)_{L^2(Y)}\, \mathrm{d}x,$$

where $\alpha \in \mathbb{R}^3$, $Y \subset \mathbb{R}^2$ is a rectangle,

$$\partial u = \begin{pmatrix} \frac{\partial u_1}{\partial x_1} \\ \frac{\partial u_2}{\partial x_2} \\ \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \end{pmatrix}, \quad C = \begin{pmatrix} c_{11} & c_{12} & 0 \\ c_{12} & c_{22} & 0 \\ 0 & 0 & c_{33} \end{pmatrix},$$

and the stiffness tensor $C : Y \to \mathbb{R}^{3\times 3}$ is positive definite, measurable and bounded uniformly in Y. The dual space to the fields ∂u in $(L^2(Y))^3$ can be arbitrarily closely approximated by the fields

$$v = \left(\frac{\partial^2 \varphi}{\partial x_2^2}, \frac{\partial^2 \varphi}{\partial x_1^2}, -\frac{\partial^2 \varphi}{\partial x_1 \partial x_2} \right),$$

where φ are the BFS functions [5].

Lemma 2. If φ is Y-periodic, then each component of v has zero mean in Y and fulfills periodic boundary conditions on ∂Y.
Proof. The proof can be found in the manuscript which is currently being prepared.

Example 3. Let us consider the linear elasticity problem with the stiffness tensor

$$C = \begin{pmatrix} 7+\mathrm{sign}(s_1 s_2) & -2-\mathrm{sign}(s_2) & 0 \\ -2-\mathrm{sign}(s_2) & 7+\mathrm{sign}(s_1 s_2) & 0 \\ 0 & 0 & 3+2\mathrm{sign}(s_2)\mathrm{sign}(s_1) \end{pmatrix},$$

where $s_j = \sin(2x_j)$, $j = 1, 2$, $x \in Y = (-\pi, \pi) \times (-\pi, \pi)$. The following triples of matrices are obtained as (from left to the right) a trivial lower bound corresponding to (4), the lower bound obtained by applying the BFS FEs, and the upper bounds obtained from Lagrange FEs, respectively, for the mesh of $n_1 \times n_2$ nodes, where $n_1 = n_2 = 8$

$$\begin{pmatrix} 6.844 & -2.044 & 0 \\ -2.044 & 6.844 & 0 \\ 0 & 0 & 1.667 \end{pmatrix} \leq \begin{pmatrix} 6.893 & -2.003 & -0.000 \\ -2.003 & 6.893 & 0.000 \\ -0.000 & 0.000 & 2.065 \end{pmatrix} \leq \begin{pmatrix} 6.950 & -1.998 & -0.005 \\ -1.998 & 6.950 & -0.005 \\ -0.005 & -0.005 & 2.618 \end{pmatrix},$$

or $n_1 = n_2 = 16$

$$\begin{pmatrix} 6.844 & -2.044 & 0 \\ -2.044 & 6.844 & 0 \\ 0 & 0 & 1.667 \end{pmatrix} \leq \begin{pmatrix} 6.897 & -2.001 & 0.000 \\ -2.001 & 6.897 & 0.000 \\ -0.000 & 0.000 & 2.089 \end{pmatrix} \leq \begin{pmatrix} 6.921 & -1.996 & -0.004 \\ -1.996 & 6.921 & -0.004 \\ -0.004 & -0.004 & 2.332 \end{pmatrix}.$$

The inequalities among the matrices are obviously satisfied.

Acknowledgement: The authors thank Petr Mayer for helpful comments on BFS FEs. This research has been performed in the Center of Advanced Applied Sciences (financially supported by the European Regional Development Fund, project No. CZ.02.1.01/0.0/0.0/16_019/0000778) (LG, ML, IP, JZ). This work has been further supported by GAČR project No. 22-35755K (LG), by CTU SGS project No. SGS22/002/OHK1/1T/11 (LG) and SGS22/004/OHK1/1T/11 (ML).

References

[1] P.G. Ciarlet: *The Finite Element Method for Elliptic Problems*. Stud. Math. Appl. 4, North Holland, Amsterdam, 1978.

[2] L. Gaynutdinova, M. Ladecký, A. Nekvinda, I. Pultarová, J. Zeman: *Efficient numerical method for reliable upper and lower bounds to homogenized parameters*. Submitted. ArXiv:2208.09940.

[3] J. Haslinger, I. Hlaváček: *Convergence of a finite element method based on the dual variational formulation*. Applications of Mathematics 21 (1), 1976, pp. 43–65.

[4] J.C. Nédélec: *Mixed finite elements in* $\mathbb{R}^3$. Numerische Mathematik 35, 1980, pp. 315–341.

[5] J. Valdman: *MATLAB Implementation of C1 Finite Elements: Bogner-Fox-Schmit Rectangle*. Parallel Processing and Applied Mathematics, 13th International Conference, PPAM 2019, Bialystok, Poland, September 8â€“11, 2019, Revised Selected Papers, Part II.

[6] J. Vondřejc, J. Zeman, I. Marek: *Guaranteed upper-lower bounds on homogenized properties by FFT-based Galerkin method*. Computer Methods in Applied Mechanics and Engineering 297, 2015, pp. 258–291.

Development, validation, and application of a solver for non-isothermal non-adiabatic packed bed reactors

T. Hlavatý, M. Isoz, M. Khýr

University of Chemistry and Technology, Prague
Institute of Thermomechanics of the CAS, Prague

1 Introduction

Heterogeneous catalysis presents an essential chain segment in industrial chemical processes ranging from oxychlorination of ethylene conducted in a tubular reactor packed with Rashig rings to conversion of pollutant exhaust gases in automobile catalytic filters, see Fig. 1. In fact, heterogeneous catalysis contributes to producing more than 80 % of all chemical products in the world [1].

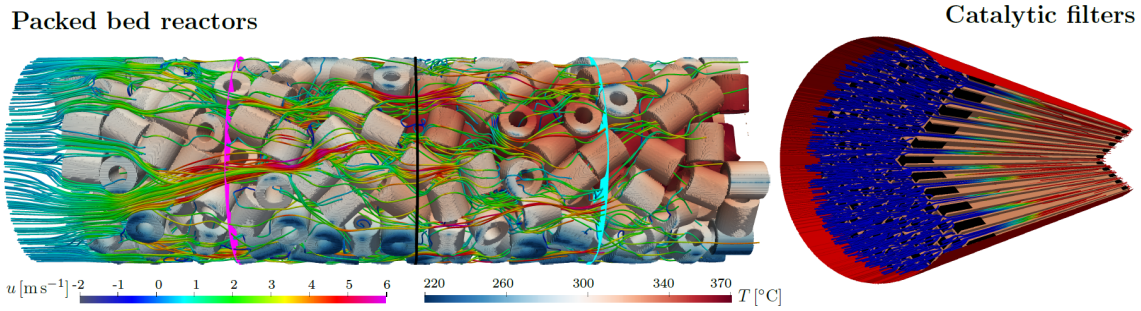


Figure 1: Real-life flow, reaction and temperature in a packed bed reactors and a catalytic filter.

Having in mind the long term goal of speeding-up the design phase and enabling optimization of heterogeneously catalyzed reactors, we aim to develop a reliable high-fidelity numerical solver for modeling of such processes. The present contribution is focused on a description of the solver fundamental working principles, verification of each of the solver components and on presenting an application of the solver on an industrially relevant case of ethylene oxichlorination performed in a tubular reactor packed with Raschig rings coated by CuCl2 catalyst.

2 Methods

Solver description The present solver is implemented within the OpenFOAM framework [2]. As such, it is based on a numerical solution of the problem governing equations via the finite volume (FV) method. The governing equations are written for non-isothermal heterogeneously catalyzed flow of a compressible Newtonian fluid inside an open bounded domain Ω split into free channels (ch) and porous media (pm) coated by a catalyst, i.e., $\Omega = \Omega_{\text{ch}} \cup \Omega_{\text{pm}}$, $\Omega_{\text{ch}} \cap \Omega_{\text{pm}} = \emptyset$ [3].

Marking u the fluid velocity, p pressure, y_i molar fraction of the i-th specie, μ dynamic viscosity, M^{g} gas molar mass, R^{g} universal gas constant, and T temperature, the model equations are

$$\begin{aligned} \partial_t(\rho u) + \nabla \cdot [\rho u \otimes u] - \nabla \cdot [\mu (\nabla u + (\nabla u)^{\intercal})] &= -\nabla p + \rho s \\ \partial_t(\rho) + \nabla \cdot [\rho u] &= 0 \end{aligned}, \quad \rho = \frac{p M^{\text{g}}}{\text{R}^{\text{g}} T}, \tag{1}$$

$$\partial_t(c_{\text{T}}\, y_i) + \nabla \cdot [u c_{\text{T}} y_i] - \nabla \cdot \left[D_i^{\text{eff}} \nabla (c_{\text{T}} y_i) \right] = r_i, \quad i = 1, \ldots, m, \quad c_{\text{T}} = \frac{\rho}{M^{\text{g}}}, \tag{2}$$

$$\partial_t(\rho\, c_{\text{p}}\, T) + \nabla \cdot [\rho u c_{\text{p}} T] - \nabla \cdot \left[\lambda^{\text{eff}} \nabla T \right] = s^h, \tag{3}$$

where equations (1) represent the momentum and mass balance, respectively. Equation (2) stands for the i-th specie balance and equation (3) gives the enthalpy balance. The density ρ is linked with T and p via the ideal gas state equation. Furthermore, the source term ρs in the momentum balance $(1)_1$ expresses the additional resistance to the flow in Ω_{pm}, computed from the Darcy permeability model.

Reactions in the balance equations for individual reactive species present in the system (2) are assumed to occur solely in Ω_{pm}. Therefore, we set $r_i = \varphi \nu_i r^c$, where φ is volumetric fraction of the catalytic material, ν_i is the stoichiometric coefficient and r^c the reaction rate computed from kinetics. A standard Fickian diffusion of y_i is assumed. The effective diffusivity coefficient D_i^{eff} in (2) is varied within Ω, (i) in the free channels, the diffusion coefficient is calculated from the Fuller equation ($D_i^{\mathrm{eff}} = D_i^{\mathrm{free}}$), (ii) in the porous media, the free diffusion coefficient is lowered by porosity ε and tortuosity τ ($D_i^{\mathrm{eff}} = D_i^{\mathrm{free}} \cdot \varepsilon/\tau$, $\varepsilon < 1$, $\tau > 1$).

Finally, the enthalpy balance (3) is formulated with T as the primary variable, all the mechanical heat sources neglected, and the reference temperature $T_{\mathrm{ref}} = 0$ °C. The specific gas heat capacity c_{p} and effective thermal diffusivity λ^{eff} are treated as temperature-dependent. Furthermore, λ^{eff} in the free channels corresponds to the gas thermal diffusivity, while in the porous media, it is a convex combination of the gas and the solid material thermal diffusivity. Lastly, s^h is the enthalpy source term caused by the reaction heat.

Overall solution algorithm The system (1)-(3) is solved in a segregated manner via methods based on the standard SIMPLE, PISO, and PIMPLE algorithms, see, e.g., [4]. However, the standard pressure-based $p-u$ coupling had to be extended by including the species balances (2) and the enthalpy balance (3). The discrete version of (1)-(3) is

$$\begin{aligned} M(\boldsymbol{u}^o, \boldsymbol{\rho}^o, \boldsymbol{T}^o)\,\boldsymbol{u}^n + N\boldsymbol{p}^o &= \boldsymbol{s_f}(\boldsymbol{T}^o)\,, \\ N^\intercal D^{-1}(\boldsymbol{u}^o, \boldsymbol{\rho}^o, \boldsymbol{T}^o) N\boldsymbol{p}^n &= N^\intercal D^{-1}(\boldsymbol{u}^o, \boldsymbol{\rho}^o, \boldsymbol{T}^o)\,[\boldsymbol{s_f}(\boldsymbol{T}^o) - H(\boldsymbol{u}^o, \boldsymbol{\rho}^o)\boldsymbol{u}^o]\,, \end{aligned} \tag{4}$$

$$P_i(\boldsymbol{u}^o, \boldsymbol{\rho}^o)\boldsymbol{y}_i^n = \boldsymbol{r}_i(\boldsymbol{\rho}^o, \{\boldsymbol{y}_j^o\}_{j=1}^m, \boldsymbol{T}^o), \quad i = 1, \ldots, m\,, \tag{5}$$

$$Q(\boldsymbol{u}^o, \boldsymbol{\rho}^o, \boldsymbol{T}^o)\boldsymbol{T}^n = \boldsymbol{s_h}(\{\boldsymbol{y}_j^o\}_{j=1}^n, \boldsymbol{T}^o)\,, \tag{6}$$

where $M = D + H$, N, P, and Q are matrices arising from FV discretization of the governing equations, discrete counterparts of functions from (1)-(3) are written in bold font and the old and new values are denoted by superscripts o and n, respectively. Note that apart from the dependence on temperature and density, equations (4) represent the standard SIMPLE $p-u$ coupling. During the solver implementation and testing, it was found out that to improve the system convergence, it is profitable to solve the equations as $(4)_1 \rightarrow (5) \rightarrow (6) \rightarrow (4)_2$, i.e., to encapsulate the species and enthalpy balances between the momentum predictor and pressure equation.

3 Results

Verification of inner mass and heat transport in a spherical particle For the case of a single reactive specie, first order reaction and a spherical catalyst particle, the internal transport represented by (2) and (3) can be simplified to [5]

$$\tilde{y}'' + (2/\tilde{r})\tilde{y}' = \phi^2 \tilde{y} \exp\left[\gamma\beta(1-\tilde{y})/(1+\beta(1-\tilde{y}))\right], \quad \tilde{y}'(0) = 0\,,\ \tilde{y}(1) = 1\,, \quad \eta = (3/\phi)y'(1) \tag{7}$$

where $\tilde{y} = \tilde{y}(\tilde{r})$ and $\tilde{r}$ stand, in order, for dimensionless concentration and radial coordinate, ϕ is the Thiele modulus, and β and γ are parameters. In Fig. 2, we compare the effectiveness factor η obtained from $(7)_4$ to the one computed from simulation results. Note the steady states multiplicity for $\phi = 0.4$ achievable by setting a different initial condition.

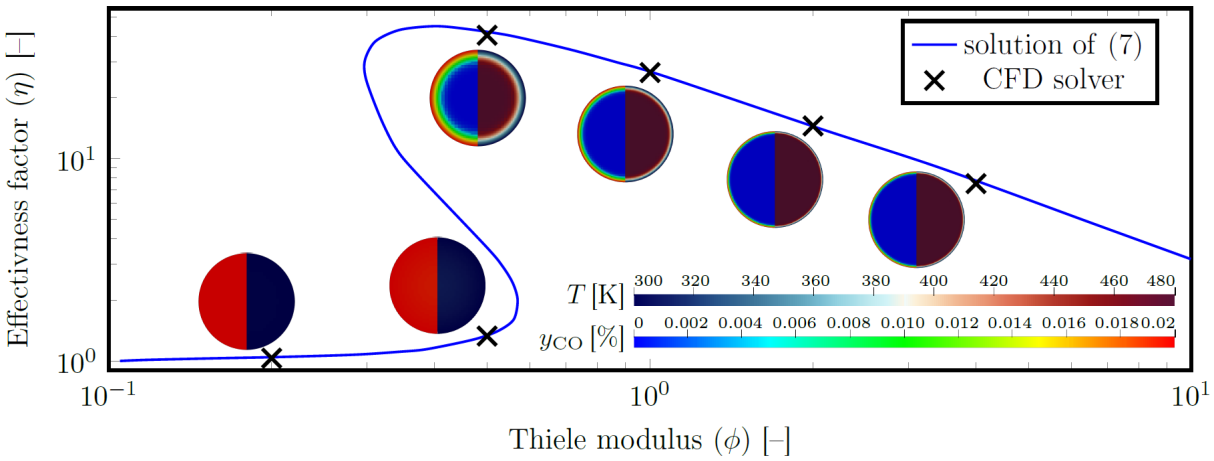


Figure 2: Comparison of the effectiveness factor obtained by the solution of the (7) with the solution obtained by newly developed solver.

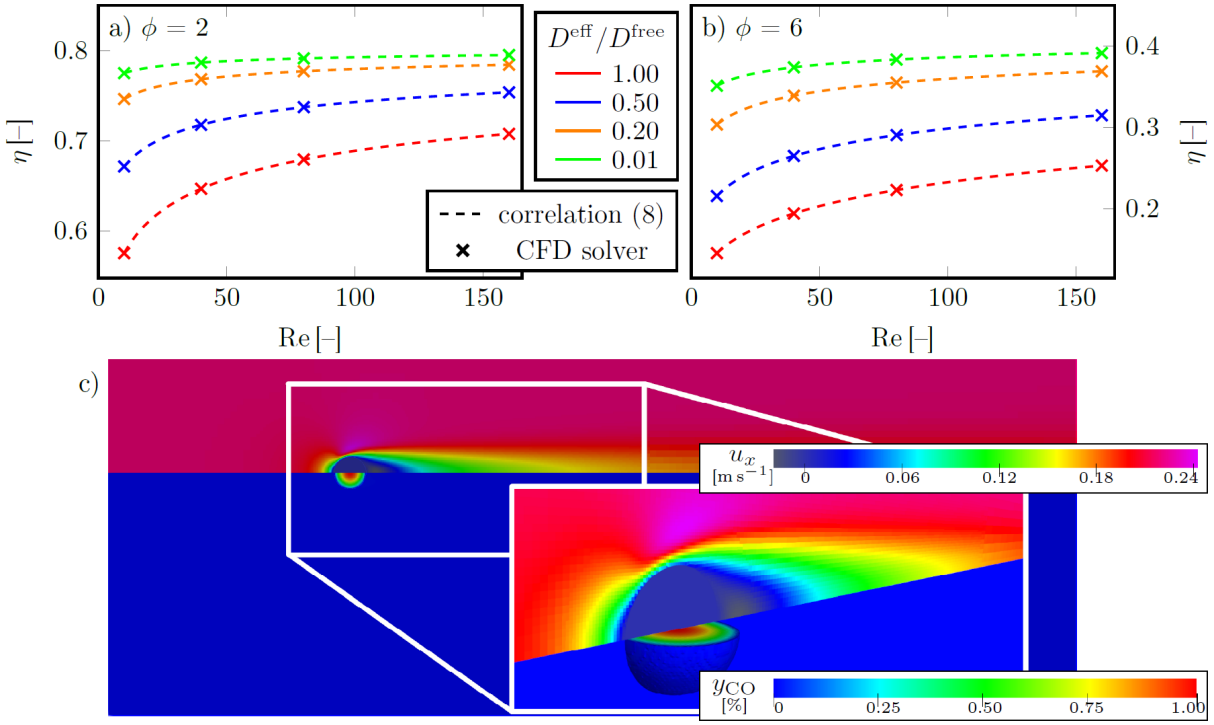


Figure 3: a, b) Comparison of the effectiveness factor (η) dependence on the Reynolds number (Re) for correlation and simulation, c) stream-wise velocity component u_x, and CO molar fraction contours on the longitudinal slice through the domain.

Verification of conjugated inner and outer mass transport The previously verified inner mass transport can be combined with an outer mass transport. In particular, we study the same reaction occurring in the same catalyst particle as above, but placed in a flowing gas. Such a system is well described by available empirical correlations. Specifically, we use approach described in [6], (i) the Sherwood number (Sh) is computed from the Frössling correlation $(8)_1$, (ii) the

mass transfer coefficient (k_m) is evaluated using the computed Sh $(8)_2$, and (iii) the effectiveness factor (η^corr) is computed from the analytical solution $(8)_3$. This correlated effectiveness factor is then compared with the one obtained directly from the simulation data by numerically evaluating the relation $(8)_4$.

$$\mathrm{Sh} = 2.0 + 0.6\,\mathrm{Re}^{\frac{1}{2}}\,\mathrm{Sc}^{\frac{1}{3}}\,,\; k_\mathrm{m} = \frac{\mathrm{Sh}\,D^\mathrm{free}}{2\,R_\mathrm{sp}}\,,\; \eta^\mathrm{corr} = \frac{3}{\phi^2}\frac{\phi\mathrm{coth}\phi - 1}{1 + \frac{D^\mathrm{eff}\,(\phi\mathrm{coth}\phi - 1)}{k_\mathrm{m}\,R_\mathrm{sp}}}\,,\; \eta^\mathrm{sim} = \frac{\iiint_V kc_s \mathrm{d}V}{\frac{4}{3}R^3 kc_0} \quad (8)$$

In Fig. 3a and b, the correlated effectiveness factors are compared with the simulation results for cases with Thiele modulus $\phi = 2$ and 6, respectively. Different ratios of the diffusivity inside the particle (D^eff) and the free diffusivity (D^free), typical for real life applications, were studied, c.f. different colors in Fig. 3a and b. Finally in Fig. 3c, the stream-wise velocity component u_x and CO molar fraction y_CO contours on a longitudinal slice through the domain are presented.

Application to ethylene oxichlorination As an example of a real-life application of the developed solver, we studied oxichlorination of ethylene performed in an industrial tubular reactor. The reactor itself comprises horizontal tubes of i.d. 28.5 mm and o.d. 31.8 mm cooled by water evaporation inside the inter-pipe space. The reaction occurs in the gas phase and is heterogeneously catalyzed by $CuCl_2$. The reactor tubes are packed with Raschig rings of characteristic dimension $d_\mathrm{rash} = 5$ mm coated by the catalyst. A numerical simulation of the complete reactor tube of length $l \approx 3.5$ m is computationally unfeasible and we focused on a 10 cm long section of the tube placed at 1.4 meters from the inlet. Illustration of the model geometry and qualitative results is given in the left hand side of Fig. 1.

4 Conclusion

The present work is a part of an ongoing research aimed at development of numerical methods for studies of heterogeneously catalyzed non-isothermal reactive flows. We presented fundamental working principles of a custom solver based on a segregated solution of the flow governing equations. Furthermore, several verification tests were outlined as well as a proof-of-concept application of the solver to simulate a semi-industrial scale model of an ethylene oxichlorination reactor. With respect to the model fidelity, the new solver corresponds to a direct numerical simulation. As such, it can provide detailed data on the studied processes. However, it is costly to evaluate. In the future work, we plan to analyze the potential for application of methods of model order reduction to enable parametric studies or optimizations based on the presented solver results.

References

[1] X. Hu, A.C.K. Yip: *Heterogeneous Catalysis: Enabling a Sustainable Future.* Frontiers in Catalysis, 1:667675, 2021.

[2] OpenCFD: *OpenFOAM: The Open Source CFD Toolbox.* User Guide Version 1.4, OpenCFD Limited. Reading UK, 2007.

[3] T. Hlavatý, M. Isoz, P. Kočí: *Developing a Coupled CFD Solver for Mass, Momentum and Heat Transport in Catalytic Filters.* In D. Šimurda, T. Bodnár (eds.): Proceedings of the conference Topical Problems of Fluid Mechanics 2022, 02 2022, pp. 79–86.

[4] F. Moukalled, M. Darwish, L. Mangani: *The finite volume method in computational fluid dynamics: an advanced introduction with OpenFOAM and Matlab*. Springer-Verlag, Berlin, Germany, 1 edition, 2016.

[5] P.B. Weisz, J.S. Hicks: *The behaviour of porous catalyst particles in view of internal mass and heat diffusion effects*. Chemical Engineering Science 17(4), 1962, pp. 265–275.

[6] V. Chandra, E.A.J. Peters, J.A.M. Kuipers: *Direct numerical simulation of a non-isothermal non-adiabatic packed bed reactor*. Chemical Engineering Journal, 385:123641, 2020.

Numerical valuation of the investment project flexibility: a comparison of European, Bermudan and American option styles

J. Hozman, T. Tichý

Technical University of Liberec
VŠB – Technical University of Ostrava

1 Introduction

Real option pricing is an essential issue of modern investment theory, recognizing important qualitative and quantitative characteristics of some of the intrinsic attributes of the investment opportunities, namely, irreversibility of investments, choice of timing and last but not least uncertainty over the future rewards from investments, see [2]. Since the real options approach interprets the flexibility value, embedded in a project, as an option premium, various option pricing techniques are widely used to valuate flexibility, see [7] for a brief overview. Among them the contingent claims analysis [1] enjoys greater interest in the real options valuation, based on formulation by a partial differential equation (PDE) comparing the change in option/project values with the change in the value of a suitably constructed portfolio of trading assets.

The real option value depends on several features. Among the most essential ones we can classify the way, in which we specify the time, at which a given option can be exercised, i.e., *(i)* only at the date of expiration, *(ii)* at finite discrete set of dates up to expiration or *(iii)* at any time between the purchase and the date to expiration. The problem we face is described by PDEs of the Black-Scholes type, equipped with the terminal condition enforced at time instants resulting from the specific option style considered. Since the early exercise is allowed, closed-form pricing formulae are not available in general and the valuation should rely on numerical approaches that take into account properties of a differential operator in particular PDE models as well as an early exercise constraint.

2 PDE models for real options valuation

We briefly recall PDE models for valuation of project flexibility from [6]. At first it is necessary to describe the value of the project itself and then we are able to find the value of its flexibility by solving the relevant PDEs that link both option and project values. We assume that fluctuations in project values are tracked back to uncertainty via the output price P, driven by a geometric Brownian motion with a drift factor $r-\delta$ (i.e., risk-free interest rate $r>0$ and mean convenience yield $\delta>0$) and volatility $\sigma>0$. Further, we denote by $V_0(P,t)$ and $V_1(P,t)$, for current price P and time t (in years), the value of the project having no options and with the embedded option allowing the particular action at a prespecified time T, respectively. Let $T^*>T$ be the maximum lifetime of both projects, when both projects are already worthless, i.e., $V_1(P,T^*)=V_0(P,T^*)=0$, $P>0$. Next, in line with [1], the value functions V_0 and V_1 at times $t\in[T,T^*)$ and for $P>0$ are characterized as solutions of a couple of deterministic backward PDEs:

$$\frac{\partial V_i}{\partial t}+\underbrace{\frac{1}{2}\sigma^2P^2\frac{\partial^2V_i}{\partial P^2}+(r-\delta)P\frac{\partial V_i}{\partial P}-rV_i}_{\mathcal{L}_{\mathrm{BS}}(V_i)}=-\varphi_i,\qquad i=0,1, \tag{1}$$

where $\varphi_0(P,t)$ and $\varphi_1(P,t)$ represent (after-tax) cash flow rates associated with the given project function.

Subsequently, we introduce the flexibility value function $F(P,t)$ as the difference $V_1(P,t)-V_0(P,t)$ for all $t \in [0,T)$, that represents the value added to the project function. On the expiration date (i.e., $t=T$), the flexibility value is simply given by

$$F(P,T) = \Pi(P) \equiv \max(V_1(P,T) - V_0(P,T) - \mathcal{K}, 0), \qquad P > 0, \tag{2}$$

where $\mathcal{K}$ is implementation costs (if positive) or disinvestment costs (if negative). The function Π plays the role equivalent to a payoff function with strike $\mathcal{K}$, well-known from financial options pricing. Within the timeline $[0,T)$, taking into account an equivalence of cash flow rates, the value function F satisfies the governing equation with the same Black-Scholes differential operator as in (1), but with zero right-hand side.

The possibility to realize the embedded flexibility known as *European* constraint is too restrictive and the choice of timing is more general in practice. Therefore, the *Bermudan* option style is partially interesting, allowing exercise at one of the finite discrete times $\mathcal{B} = \{k\Delta\}_{k=1}^{\lfloor T/\Delta \rfloor} \cup \{T\}$, where $0 < \Delta < T$. This early exercise restricted to certain dates imposes an additional constraint that $F(P,t) \geq \Pi(P)$ for any $t \in \mathcal{B}$. Moreover, as $\Delta \to 0+$, we obtain *American* option style under constraint $F(P,t) \geq \Pi(P)$ for any $t \in [0,T)$.

There are several approaches how to handle the early exercise feature, among the widely used ones, just penalty techniques [9] allow us to formulate the pricing problem as follows

$$\frac{\partial F}{\partial t} + \mathcal{L}_{\mathrm{BS}}(F) + q_F = 0, \quad P \in (0,\infty),\ t \in [0,T), \tag{3}$$

where the penalty term q_F is defined to ensure Bermudan constraint by using conditions

$$q_F(P,t) = 0, \ \text{ for } t \in [0,T)\backslash\mathcal{B}, \qquad q_F(P,t) \begin{cases} = 0, & \text{if } F(P,t) > \Pi(P), \\ > 0, & \text{if } F(P,t) = \Pi(P), \end{cases} \text{ for } t \in \mathcal{B}. \tag{4}$$

In line with the above, for American constraint, we simply require

$$q_F(P,t) = 0, \text{ if } \ F(P,t) > \Pi(P), \qquad q_F(P,t) > 0, \text{ if } \ F(P,t) = \Pi(P), \ \ t \in [0,T). \tag{5}$$

Note, if we put $q_F(P,t) = 0$, for all $P > 0$ and $t \in [0,T)$, in the case of a European exercise right, the penalty approach can be unified for all three option styles considered.

The localization of governing equations (1) and (3) to a bounded interval $\Omega = (0, P_{\max})$ is necessary for the subsequent numerical treatment. Therefore we have to impose project as well as option values at both endpoints $P = 0$ and $P = P_{\max}$. The project values are estimated by the net present value approach for the given cash flow rates as follows

$$V_i(z,t) = \int_t^{T^*} \varphi_i(z,\xi) e^{-r(\xi - t)} \mathrm{d}\xi, \ \ z \in \{0, P_{\max}\}, t \in [T, T^*),\ i = 0, 1. \tag{6}$$

Considering real put options, the exercise rights lead to a couple of Dirichlet boundary conditions in the form

$$\begin{aligned}
&F(0,t) = e^{-r(T-t)}\Pi(0), && F(P_{\max},t) = 0, \ \ t \in [0,T), \ \ \text{(European)} \\
&F(0,t) = \begin{cases} e^{-r(T_{\mathrm{B}}-t)}\Pi(0), & \text{if } t \in [0,T)\backslash\mathcal{B}, \\ \Pi(0), & \text{if } t \in \mathcal{B}, \end{cases} && F(P_{\max},t) = 0, \ \ t \in [0,T), \ \ \text{(Bermudan)} \\
&F(0,t) = \Pi(0), && F(P_{\max},t) = 0, \ \ t \in [0,T), \ \ \text{(American)}
\end{aligned} \tag{7}$$

where $T_{\mathrm{B}} \in \mathcal{B}$ is the smallest value that satisfies $t < T_{\mathrm{B}}$.

3 Numerical approach

Since the governing equations (1) and (3) are closely related to the class of convection-diffusion problems and exhibit a hyperbolic behaviour as $|r-\delta| \gg \sigma^2$, the proposed valuation methodology is based on discontinuous Galerkin (DG) method, successfully used in the field of financial option pricing, see, e.g., [4] and [5]. The numerical solution is constructed as a composition of piecewise polynomial, generally discontinuous, functions on finite element mesh without any requirements on the continuity of the solution across the partition nodes, see [8]. From this point of view, this discontinuous framework is a very promising numerical tool in financial engineering, suitable for problems for which other techniques fail or have computational difficulties.

With respect to the space-time domain of governing equations, the numerical treatment consists of two consecutive phases — spatial semi-discretization and temporal discretization. Within the first phase, at each time instant, we construct the semi-discrete solution, defined using the variational formulation and represented by the system of ODEs. The second phase is then devoted to the discretization with respect to the time coordinate by implicit Euler scheme, having no restrictive condition on the length of the time step, that results into a linear algebraic problem (with a sparse matrix) at each time level.

In summary, determining the value of flexibility of an investment project at the present time $t = 0$ consists of two consecutive problems. First, we solve a pair of PDEs (1) with homogeneous terminal conditions to obtain the project values V_0 and V_1 at $t = T$ that we use in the construction of the terminal value of the embedded flexibility (2). Consequently, we solve the problem (3) under the given exercise constraint to obtain the present value of flexibility, i.e., real option value F at $t = 0$. This procedure is known as backward induction.

In conclusion, we briefly present the numerical experiment, performed on the reference data from [6]. We consider an option on ownership transfer of half of the project for $\mathcal{K} = -10^{10}$ USD under various exercise rights with parameters $T^* = 75.8$, $T = 1$, $\Delta = 0.25$ (i.e., a quarter of a year), $\sigma = 0.3$, $r = 0.06$ and $\delta = 0.02$. The corresponding cash flow rates are linked by $\varphi_1(P,t) = \frac{1}{2}\varphi_0(P,t)$, where $\varphi_0(P,t) = 0.07\, e^{0.007t}\left(0.95\, P - 25\, e^{0.005t}\right)$, for $P \in \Omega$ and $t \in [T, T^*)$. The numerical scheme is implemented in the solver Freefem++ [3] with the time step corresponding to $T/100$ and with piecewise quadratic approximations on the uniformly partitioned grid (with mesh size $P_{\max}/100$) of Ω for $P_{\max} = 80$.

Figure 1 records differences between flexibility values at present time ($t = 0$) for every two of all exercise rights considered. One can easily observe that an early exercise feature increases value of the project flexibility. More precisely, the intuitive relationship $F_{\mathrm{Am}} \geq F_{\mathrm{Be}} \geq F_{\mathrm{Eu}}$ is well resolved and meets the expectations of financial practitioners as a financially meaningful result.

Acknowledgement: Both authors were supported through the Czech Science Foundation (GAČR) under project 22-17028S. Furthermore, the second author also acknowledges the support provided within SP2023/001, an SGS research project of VŠB-TU Ostrava. The support is greatly acknowledged.

References

[1] F. Black, M. Scholes: *The pricing of options and corporate liabilities.* Journal of Political Economy **81**, 1973, pp. 637–659.

[2] A. Dixit, R. Pindyck: *Investment Under Uncertainty.* Princeton, Princeton University Press, 1994.

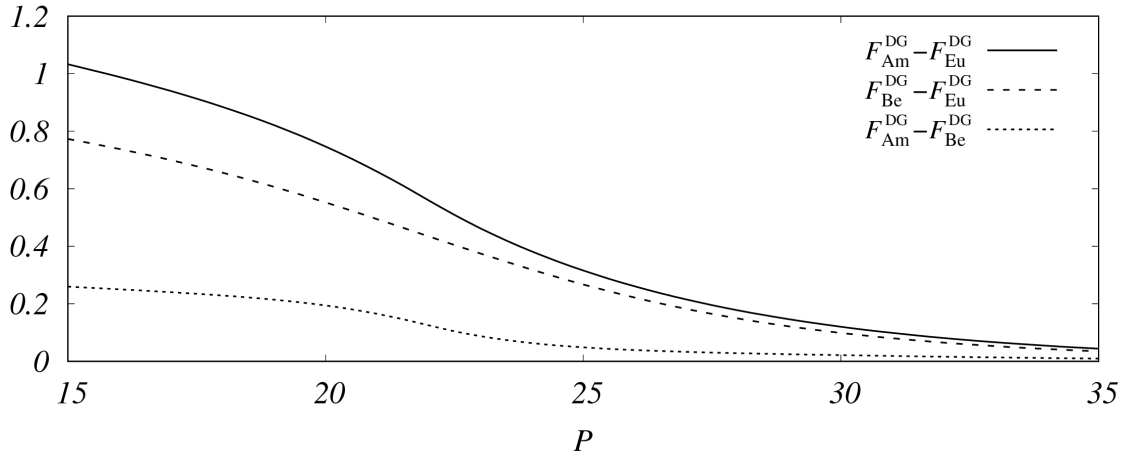


Figure 1: The differences between option values (in 10^9 USD) under European (Eu), Bermudan (Be) and American (Am) constraints.

[3] F. Hecht: *New development in FreeFem++*. Journal of Numerical Mathematics **20**, No. 3-4, 2012, pp. 251–265.

[4] J. Hozman, T. Tichý: *DG framework for pricing European options under one-factor stochastic volatility models*. Journal of Computational and Applied Mathematics **344**, 2018, pp. 585–600.

[5] J. Hozman, T. Tichý: *The discontinuous Galerkin method for discretely observed Asian options*. Mathematical Methods in the Applied Sciences **43**, 2020, pp. 7726–7746.

[6] N. Li, S. Wang, S. Zhang: *Pricing options on investment project contraction and ownership transfer using a finite volume scheme and an interior penalty method*. Journal of Industrial & Management Optimization **16**, 2020, pp. 1349–1368.

[7] J. Mun: *Real Options Analysis: Tools and Techniques for Valuing Strategic Investments and Decisions*. John Wiley & Sons, Inc., Hoboken, 2002.

[8] B. Riviére: *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*. SIAM, Philadelphia, 2008.

[9] R. Zvan, P.A. Forsyth, K.R. Vetzal: *Penalty methods for American options with stochastic volatility*. Journal of Computational and Applied Mathematics **91**, 1998, pp. 199–218.

A note on Clifford Algebras

D. Janovská

University of Chemistry and Technology, Prague

1 Introduction

Many research fields, such as physics, robotics, machine vision and computer graphics, rely on geometric models of external world. In these applications geometric objects and their transformations are traditionally formalized using linear algebra, i.e. all geometric concepts have to be represented by vectors and matrices. Compared with traditional linear algebra, geometric algebra (also known as Clifford algebra) is a powerful mathematical tool that offers a natural and direct way to model geometric objects and their transformations. Geometric entities, such as lines, planes and volumes, become basic elements of the algebra and can be manipulated by a rich set of algebraic operators that have a direct geometric significance.

Clifford algebra was invented by W. Clifford [1]. In his research, he combined Hamilton's quaternions [3] and Grassmann's exterior algebra [4]. Further development of the theory of Clifford algebras is associated with a number of famous mathematicians and physicists R. Lipschitz, T. Vahlen, E. Cartan, E. Witt, C. Chevalley, M. Riesz, and others. Dirac equation [5] had a great influence on the development of Clifford algebra. Nowadays Clifford algebra is used in different branches of modern mathematics, physics, geometry, computer science, mechanics, robotics, signal and image processing.

2 What is a Clifford algebra?

Clifford algebras are defined on vector spaces on which a non-degenerate quadratic form Q has been chosen. Different quadratic forms will generally lead to different algebras. The most important property of the quadratic form is its signature: the numbers of positive and negative eigenvalues. We will be concerned only with real Clifford algebras.

In any case, these are the geometric ideas behind Clifford's algebras, what makes them so attractive.

Remark: Signature of quadratic form Q.
Let us have the vector space $\mathbb{R}^n$ on which is given a non-degenerate quadratic form Q,

$$Q(x) = x^{\mathrm{T}} Q x.$$

Let us note that the notation Q will be also used for the symmetric $n \times n$ matrix. For Q symmetric, its eigenvalues are real and the fact that Q is non-degenerate means that Q has p positive and q negative eigenvalues with $p + q = n$. This pair (p, q) is called the signature of Q and is the only important property of Q in defining the associated Clifford algebra.
This algebra will be usually denoted as $\mathcal{C}\ell(p, q)$.

Choose an orthonormal basis $e_1, e_2, \ldots, e_n \in \mathbb{R}^n$. Here orthonormal means relative to Q so that we require

$$\begin{aligned} e_i^{\mathrm{T}} Q e_j &= 0,\ i \neq j\,, \\ e_i^{\mathrm{T}} Q e_i &= \begin{cases} +1, & i = 1 \ldots p \\ -1 & i = p+1 \ldots n\,. \end{cases} \end{aligned}$$

Definition: Let n be a natural number and E be a linear space of dimension 2^n over the field of real numbers $\mathbb{R}$ with the basis enumerated by the ordered multiindices with a length between 0 and n:

$$e, e_{a_1}, e_{a_1 a_2}, \cdots, e_{1\ldots n}, \quad \text{where} \quad 1 \leq a_1 < a_2 < \cdots < a_n \leq n.$$

Let us introduce the operation of multiplication on E:

- with the properties of distributivity, associativity:

$$\begin{aligned} U(\alpha V + \beta W) &= \alpha UV + \beta UV, \quad U, V.W \in E, \quad \alpha, \beta \in \mathbb{R} \\ (\alpha U + \beta V)W &= \alpha UW + \beta VW, \quad U, V.W \in E, \quad \alpha, \beta \in \mathbb{R} \\ U(VW) &= (UV)W, \quad U, V, W \in E\,. \end{aligned}$$

- e is the identity element: $\quad Ue = eU = U, \quad U \in E.$

- $e_a,\ a = 1, \ldots, n$ are generators:

$$e_{a_1} e_{a_2} \cdots e_{a_n} = e_{a_1 \ldots a_n}, \quad 1 \leq a_1 < a_2 < \cdots < a_n \leq n$$

- generators satisfy $\quad e_a e_b + e_b e_a = 2\eta_{ab} e$, where

$$\eta = \|\eta_{ab}\| = \mathrm{diag}(\underbrace{1, \ldots, 1}_{p}, \underbrace{-1, \ldots, -1}_{q}, \underbrace{0, \ldots, 0}_{r}), \quad p + q + r = n \tag{1}$$

  is a diagonal matrix with p times 1, q times -1, and r times 0 on the diagonal.

The linear space E with such operation of multiplication is called real Clifford algebra $C\ell_{p,q,r}$.

Any element of the real Clifford algebra $C\ell_{p,q,r}$ has the form

$$U = ue + \sum_{a=1}^{n} u_a e_a + \sum_{a<b} u_{ab} e_{ab} + \cdots + u_{1\ldots n} e_{1\ldots n} \tag{2}$$

where $u, u_a, u_{ab}, \ldots, u_{1\ldots n} \in \mathbb{R}$ are real numbers.

2.1 Examples in small dimensions

- $C\ell_0$
  In the case of $C\ell_0$, arbitrary Clifford algebra element has the form $U = ue$, where $e^2 = e$. We obtain the isomorphism $C\ell_0 \cong \mathbb{R}$.

- $C\ell_1$
  In the case of $C\ell_1$, arbitrary Clifford algebra element has the form $U = ue + u_1 e_1$, where e is the identity element and $e_1^2 = e$. We obtain the isomorphism with double numbers: $C\ell_1 \cong \mathbb{R} \oplus \mathbb{R}$.

- $C\ell_{0,1}$
  In the case of $C\ell_{0,1}$, arbitrary Clifford algebra element has the form $U = ue + u_1e_1$, where e is the identity element and $e_1^2 = -e$. We obtain the isomorphism with complex numbers: $C\ell_{0,1} \cong \mathbb{C}$.

- $C\ell_{0,2}$
  In the case of $C\ell_{0,2}$, arbitrary Clifford algebra element has the form $U = ue + u_1e_1 + u_2e_2 + u_{12}e_{12}$. We can verify the following relations:

$$\begin{aligned} (e_1)^2 &= (e_2)^2 = -e \\ (e_{12})^2 &= e_1e_2e_1e_2 = -e_1e_1e_2e_2 = -e \\ e_1e_2 &= -e_2e_1 = e_{12}, \qquad e_2e_{12} = -e_{12}e_2 = e_1 \\ e_{12}e_1 &= -e_1e_{12} = e_2. \end{aligned}$$

  Using the following substitution [6]

$$e_1 \to \mathrm{i}, \quad e_2 \to \mathrm{j}, \quad e_{12} \to \mathrm{k}$$

  where i, j, and k are imaginary units of quaternions, we obtain the isomorphism $C\ell_{0,2} \cong \mathbb{H}$.

3 Clifford algebras, Clifford numbers and Clifford vectors

Definition: Let $n \in \mathbb{N} := \{1, 2, \ldots\}$. A real Clifford algebra, denoted by C_n is the $\nu := 2^{n-1}$-dimensional vector space $\mathbb{R}^\nu$ with an additional, associative, multiplication $\mathbb{R}^\nu \times \mathbb{R}^\nu \to \mathbb{R}^\nu$ generated by an $(n-1)$-dimensional vector space over $\mathbb{R}$ with a basis $e_1, e_2, \ldots, e_{n-1}$, the elements of which satisfy the following rules of multiplication:

$$e_j^2 = -1, \quad e_je_k = -e_ke_j, \quad j \neq k, \ 1 \leq j, \ k \leq n-1. \tag{3}$$

The basis of C_n consists of all products

$$\{e_{j_1}e_{j_2}\cdots e_{j_k} : 1 \leq j_1 < j_2 < \cdots < j_k \leq n-1, \quad 0 \leq k \leq n-1\}. \tag{4}$$

The empty product ($k = 0$) defines the multiplicative identity of C_n, denoted by 1.

Since there are $\binom{n-1}{k}$ products in (4) with exactly k factors the total number of products (including the empty one) is $\nu := 2^{n-1}$ which is the dimension of C_n.

Definition: Let $\nu := 2^{n-1}$ and $1, i_1, i_2, \ldots, i_{\nu-1}$ be any fixed enumeration of the basis elements of C_n. Then $a \in C_n$ has the unique representation $a = a_0 + \sum_{j=1}^{\nu-1} a_j i_j$ and $||a|| = \sqrt{\sum_{j=0}^{\nu-1} |a_j|^2}$ will be called the Euclidean norm of a. The elements in C_n will be called Clifford numbers. If they belong to the subalgebra C_{n-1} spanned by the products of $e_1, e_2, \ldots, e_{n-1}$ they will be called Clifford vectors [2].

The construction which is given in (3) and (4) says whether a given $\mathbb{R}^n$ algebra is a Clifford algebra or not.

As an example let us ask whether the algebra of coquaternions H_{coq} is a Clifford algebra. In H_{coq} we have $(1^2, \mathrm{i}^2, \mathrm{j}^2, \mathrm{k}^2) = (1, -1, 1, 1)$. If we put $e_1 = \mathrm{i}, e_2 = \mathrm{j}, e_3 = \mathrm{k}$, then apparently the first part of (3) is not true. Thus, H_{coq} is not a Clifford algebra. That implies, that not all $\mathbb{R}^n$ algebras are Clifford algebras.

4 Clifford algebras in $\mathbb{R}^8$

As a model of the $\mathbb{R}^8$ algebra we will use for our investigation the following C_4 algebra. The units are arranged in such a way that the first four components represent the quaternionic part.

Table 1: Multiplication table for canonical unit vectors in $\mathbb{R}^8$.

i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
i_2	$-i_1$	i_4	$-i_3$	i_6	$-i_5$	i_8	$-i_7$
i_3	$-i_4$	$-i_1$	i_2	i_7	$-i_8$	$-i_5$	i_6
i_4	i_3	$-i_2$	$-i_1$	i_8	i_7	$-i_6$	$-i_5$
i_5	$-i_6$	$-i_7$	i_8	$-i_1$	i_2	i_3	$-i_4$
i_6	i_5	$-i_8$	$-i_7$	$-i_2$	$-i_1$	i_4	i_3
i_7	i_8	i_5	i_6	$-i_3$	$-i_4$	$-i_1$	$-i_2$
i_8	$-i_7$	i_6	$-i_5$	$-i_4$	i_3	$-i_2$	i_1

Our algebra is $\mathbb{R}^8$ as ordinary normed vector space with an additional multiplication. We assume that the eight canonical unit vectors, denoted here by $i_j, j = 1, 2, \ldots, 8$ can be multiplied by using the corresponding Table 1. This multiplication is derived from the Clifford algebra C_4. It is an associative multiplication and thus, not the multiplication of the octonions. It extends the multiplication of the real numbers $\mathbb{R}$, the complex numbers $\mathbb{C}$, and the quaternions $\mathbb{H}$.

In addition to the actual problems in Clifford algebras, we plan to study numerical problems such as linear equations with algebraic coefficients, finding the zeros of polynomials with algebraic coefficients, or matrix problems where matrices have algebraic coefficients as entries.

References

[1] W. Clifford: *Application of GrassmannÂ's Extensive Algebra.* American Journal of Mathematics 1, 1878, 350358.

[2] S. Franchini, G. Vassallo, F. Sorbello: *A brief introduction to Clifford algebra.* Universita degli studi di Palermo, Technical Report N. 2/2010.

[3] W. Hamilton: *On Quaternions, or on a New System of Imaginaries in Algebra.* Philosophical Magazine, 1844.

[4] H. Grassmann: *Die Lineale Ausdehnungslehre, ein neuer Zweig derMathematik [The Theory of Linear Extension, a New Branch of Mathematics]*, 1844.

[5] P. Dirac: *The Quantum Theory of the Electron.* Proc. Roy. Soc. Lond. A117, 1928, 610Â–624.

[6] D. Shirokov: *Clifford Algebras and Their Applications to Lie Groups and Spinors*, 2018. DOI:10.7546/giq-19-2018-11-53.

Multi-time step methods for lattice discrete particle models

J. Kruis

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

In memory of professor Radim Blaheta.

1 Introduction

This paper concerns with the second order time dependent problems which emerge in connection with the use of lattice discrete particle models (LDPM). One of such models is described in [1] and [2]. The LDPM require very short time steps and therefore an explicit time integration methods are used [3]. The explicit methods use a diagonal matrix and therefore the solution of a system of algebraic equations is very easy. Unfortunately, the explicit methods are conditionally stable which requires the time step to be smaller than a limit value. The limit length of the time step is usually not known or its determination is a very computationally demanding task and therefore only estimates are used [4].

There are many problems, where the use of a fine mesh on the whole domain and a short time step during the whole time period studied are inefficient. The finite element mesh (or the finite difference grid) has to be adaptably refined which enables different time steps with respect to the element sizes. Another possibility is to use a fixed fine mesh while a combination of explicit and implicit time integration methods is applied or different lengths of the time step are used in different parts of the domain solved.

The use of different lengths of time steps is called multi-time step methods or sub-cycling. A new opportunity for the multi-time step methods emerged in connection with parallel computers and domain decomposition methods [5]. In the case of a multi-time step method, one subdomain can be integrated with a shorter time step than the neighboring subdomain and the continuity on the interface becomes more complicated. But many time steps can be saved and therefore, shorter computation time is obtained.

2 Sub-cycling methods for second order problems

Linear second order problem has the form

$$\boldsymbol{M}\ddot{\boldsymbol{u}}(t) + \boldsymbol{C}\dot{\boldsymbol{u}}(t) + \boldsymbol{K}\boldsymbol{u}(t) = \boldsymbol{f}(t), \tag{1}$$

where $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{C}$ is the damping matrix, $\boldsymbol{K}$ is the stiffness matrix, $\ddot{\boldsymbol{u}}(t)$ is the vector of acceleration, $\dot{\boldsymbol{u}}(t)$ is the vector of velocity, $\boldsymbol{u}(t)$ is the vector of displacement and $\boldsymbol{f}(t)$ is the vector of prescribed forces. The mass matrix $\boldsymbol{M}$ can be diagonalized. If the damping matrix $\boldsymbol{C}$ is assumed to be proportional to the mass matrix only, it can be diagonal too.

Nodal unknowns stored in the vector $\boldsymbol{u}$ in (1) are split into the block $\boldsymbol{u}^{(S)}$, where short time step Δt is used and the block $\boldsymbol{u}^{(L)}$ with long time step $m\Delta t$. The system (1) can be rewritten in the form

$$\begin{pmatrix} \boldsymbol{M}^{(L)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}^{(S)} \end{pmatrix} \begin{pmatrix} \ddot{\boldsymbol{u}}^{(L)} \\ \ddot{\boldsymbol{u}}^{(S)} \end{pmatrix} + \begin{pmatrix} \boldsymbol{C}^{(L)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}^{(S)} \end{pmatrix} \begin{pmatrix} \dot{\boldsymbol{u}}^{(L)} \\ \dot{\boldsymbol{u}}^{(S)} \end{pmatrix} +$$
$$+ \begin{pmatrix} \boldsymbol{K}^{(L)} & \boldsymbol{K}^{(LS)} \\ \boldsymbol{K}^{(SL)} & \boldsymbol{K}^{(S)} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{(L)} \\ \boldsymbol{u}^{(S)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}^{(L)} \\ \boldsymbol{f}^{(S)} \end{pmatrix}. \tag{2}$$

In this paper, the vectors connected with long step have one subscript

$$\boldsymbol{u}_k^{(L)} = \boldsymbol{u}^{(L)}(t_k) = \boldsymbol{u}^{(L)}(km\Delta t) \tag{3}$$

but the vectors connected with short step have two subscripts

$$\boldsymbol{u}_{k,j}^{(S)} = \boldsymbol{u}^{(S)}(t_{k,j}) = \boldsymbol{u}^{(L)}((km+j)\Delta t). \tag{4}$$

2.1 Explicit method

Belytschko and Lu proposed the following algorithm for solution of the equation of motion (1) in [6]. Newmark method with parameters $\beta = 0$ and $\gamma = 0.5$ leads to expressions

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \Delta t\dot{\boldsymbol{u}}_k + \frac{1}{2}\Delta t^2\ddot{\boldsymbol{u}}_k, \tag{5}$$

$$\dot{\boldsymbol{u}}_{k+1} = \dot{\boldsymbol{u}}_k + \frac{1}{2}\Delta t(\ddot{\boldsymbol{u}}_k + \ddot{\boldsymbol{u}}_{k+1}). \tag{6}$$

Constant velocity in the group L between time instants t_k and t_{k+1} led to numerical difficulties. Therefore the vector of acceleration in the group L is fixed and the following relationships are valid

$$\ddot{\boldsymbol{u}}_{k,j+1}^{(L)} = \ddot{\boldsymbol{u}}_{k,j}^{(L)} = \ddot{\boldsymbol{u}}_k^{(L)}, \tag{7}$$

At time t_k, the vectors $\boldsymbol{u}_{k,0}^{(S)}$, $\dot{\boldsymbol{u}}_{k,0}^{(S)}$, $\ddot{\boldsymbol{u}}_{k,0}^{(S)}$, $\boldsymbol{u}_k^{(L)}$, $\dot{\boldsymbol{u}}_k^{(L)}$ and $\ddot{\boldsymbol{u}}_k^{(L)}$ are known. m short steps in the form

$$\left(\boldsymbol{M}^{(S)} + \frac{1}{2}\Delta t\boldsymbol{C}^{(S)}\right)\ddot{\boldsymbol{u}}_{k,j+1}^{(S)} = \left(\boldsymbol{M}^{(S)} - \frac{1}{2}\Delta t\boldsymbol{C}^{(S)} - \frac{1}{2}\Delta t^2\boldsymbol{K}^{(S)}\right)\ddot{\boldsymbol{u}}_{k,j}^{(S)} + \boldsymbol{f}_{k,j+1}^{(S)} -$$
$$-\boldsymbol{f}_{k,j}^{(S)} - \Delta t\boldsymbol{K}^{(S)}\dot{\boldsymbol{u}}_{k,j}^{(S)} - \Delta t\boldsymbol{K}^{(SL)}\dot{\boldsymbol{u}}_k^{(L)} - j\Delta t^2\boldsymbol{K}^{(SL)}\ddot{\boldsymbol{u}}_k^{(L)} - \frac{1}{2}\Delta t^2\boldsymbol{K}^{(SL)}\ddot{\boldsymbol{u}}_k^{(L)}. \tag{8}$$

are performed and the vector $\boldsymbol{u}_{k+1,0}^{(S)} = \boldsymbol{u}_{k,m}^{(S)}$ becomes to be known. Then, one long time step in the form

$$\left(\boldsymbol{M}^{(L)} + \frac{m\Delta t}{2}\boldsymbol{C}^{(L)}\right)\ddot{\boldsymbol{u}}_{k+1}^{(L)} = \left(\boldsymbol{M}^{(L)} - \frac{m\Delta t}{2}\boldsymbol{C}^{(L)} - \frac{m^2\Delta t^2}{2}\boldsymbol{K}^{(L)}\right)\ddot{\boldsymbol{u}}_k^{(L)} +$$
$$+\boldsymbol{f}_{k+1}^{(L)} - \boldsymbol{f}_k^{(L)} - m\Delta t\boldsymbol{K}^{(L)}\dot{\boldsymbol{u}}_k^{(L)} - \boldsymbol{K}^{(LS)}\left(\boldsymbol{u}_{k+1}^{(S)} - \boldsymbol{u}_k^{(S)}\right). \tag{9}$$

is performed and the vector $\boldsymbol{u}_{k+1}^{(L)}$ is known.

2.2 Implicit Method

For simplicity, an undamped linear system in the form

$$\boldsymbol{M}\ddot{\boldsymbol{u}} + \boldsymbol{K}\boldsymbol{u} = \boldsymbol{f} \tag{10}$$

is assumed. In paper [7], the problem is split into two parts. Part L is integrated with long time step $m\Delta t$ while part S with short time step Δt. The system (10) can be written in the form

$$\boldsymbol{M}^{(L)}\ddot{\boldsymbol{u}}_{k+1}^{(L)} + \boldsymbol{K}^{(L)}\boldsymbol{u}_{k+1}^{(L)} = \boldsymbol{f}_{k+1}^{(L)} - \left(\boldsymbol{B}^{(L)}\right)^T\boldsymbol{\lambda}_{k+1}, \tag{11}$$

$$\boldsymbol{M}^{(S)}\ddot{\boldsymbol{u}}_{k,j+1}^{(S)} + \boldsymbol{K}^{(S)}\boldsymbol{u}_{k,j+1}^{(S)} = \boldsymbol{f}_{k,j+1}^{(S)} - \left(\boldsymbol{B}^{(S)}\right)^T\boldsymbol{\lambda}_{k,j+1}, \tag{12}$$

$$\boldsymbol{B}^{(L)}\dot{\boldsymbol{u}}_{k,j+1}^{(L)} + \boldsymbol{B}^{(S)}\dot{\boldsymbol{u}}_{k,j+1}^{(S)} = \boldsymbol{0}, \tag{13}$$

where (11) is the equation of motion of the part L, (12) is the equation of motion of the part S and new condition (13) has to be added which expresses continuity along the interface between parts L and S. Condition (13) enforces continuity of velocities. The terms $\left(\boldsymbol{B}^{(S)}\right)^T \boldsymbol{\lambda}$ and $\left(\boldsymbol{B}^{(L)}\right)^T \boldsymbol{\lambda}$ express internal forces between parts L and S. In equations (11-13), $\boldsymbol{B}^{(S)}$ and $\boldsymbol{B}^{(L)}$ are the Boolean matrices and $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers which represent internal nodal forces in this case. The Lagrange multipliers, $\boldsymbol{\lambda}$, and velocities, $\dot{\boldsymbol{u}}^{(L)}$, are not evaluated between times t_k and t_{k+1} and therefore, their values are interpolated in the form

$$\boldsymbol{\lambda}_{k,j} = \left(1 - \frac{j}{m}\right) \boldsymbol{\lambda}_k + \frac{j}{m} \boldsymbol{\lambda}_{k+1}, \tag{14}$$

$$\dot{\boldsymbol{u}}^{(L)}_{k,j} = \left(1 - \frac{j}{m}\right) \dot{\boldsymbol{u}}^{(L)}_k + \frac{j}{m} \dot{\boldsymbol{u}}^{(L)}_{k+1}. \tag{15}$$

The Newmark method [3] is used for time integration of (11-13). The displacements and velocities in the part L are in the form

$$\begin{aligned} \boldsymbol{u}^{(L)}_{k+1} &= \boldsymbol{u}^{(L)}_k + m\Delta t \dot{\boldsymbol{u}}^{(L)}_k + (m\Delta t)^2 \left(\frac{1}{2} - \beta\right) \ddot{\boldsymbol{u}}^{(L)}_k + (m\Delta t)^2 \beta \ddot{\boldsymbol{u}}^{(L)}_{k+1} = \\ &= \boldsymbol{p}^{(L)}_k + (m\Delta t)^2 \beta \ddot{\boldsymbol{u}}^{(L)}_{k+1}, \qquad (16) \\ \dot{\boldsymbol{u}}^{(L)}_{k+1} &= \dot{\boldsymbol{u}}^{(L)}_k + m\Delta t(1-\gamma) \ddot{\boldsymbol{u}}^{(L)}_k + m\Delta t \gamma \ddot{\boldsymbol{u}}^{(L)}_{k+1}, \qquad (17) \end{aligned}$$

where $\boldsymbol{p}^{(L)}_k$ is the predictor

In the paper [7], the following decomposition of the acceleration is introduced

$$\ddot{\boldsymbol{u}}^{(L)}_k = \ddot{\boldsymbol{w}}^{(L)}_k + \ddot{\boldsymbol{q}}^{(L)}_k, \tag{18}$$

$$\ddot{\boldsymbol{u}}^{(S)}_{k,j} = \ddot{\boldsymbol{w}}^{(S)}_{k,j} + \ddot{\boldsymbol{q}}^{(S)}_{k,j}. \tag{19}$$

At time t_k the vectors $\boldsymbol{u}^{(S)}_{k,0}$, $\dot{\boldsymbol{u}}^{(S)}_{k,0}$, $\ddot{\boldsymbol{u}}^{(S)}_{k,0}$, $\boldsymbol{u}^{(L)}_k$, $\dot{\boldsymbol{u}}^{(L)}_k$ and $\ddot{\boldsymbol{u}}^{(L)}_k$ are known. From the equation

$$\left(\boldsymbol{M}^{(L)} + (m\Delta t)^2 \beta^{(L)} \boldsymbol{K}^{(L)}\right) \ddot{\boldsymbol{w}}^{(L)}_{k+1} = \boldsymbol{f}^{(L)}_{k+1} - \boldsymbol{K}^{(L)} \boldsymbol{p}^{(L)}_k, \tag{20}$$

the vector $\ddot{\boldsymbol{w}}^{(L)}_{k+1}$ is determined. Then m steps are performed, where the vector $\ddot{\boldsymbol{w}}^{(S)}_{k,j+1}$ is computed from

$$\left(\boldsymbol{M}^{(S)} + \Delta t^2 \beta^{(S)} \boldsymbol{K}^{(S)}\right) \ddot{\boldsymbol{w}}^{(S)}_{k,j+1} = \boldsymbol{f}^{(S)}_{k,j+1} - \boldsymbol{K}^{(S)} \boldsymbol{p}^{(S)}_{k,j}, \tag{21}$$

the vector $\boldsymbol{\lambda}_{k,j+1}$ is computed from

$$\begin{aligned} &\left(m\Delta t \gamma^{(L)} \boldsymbol{B}^{(L)} \left(\boldsymbol{M}^{(L)} + (m\Delta t)^2 \beta^{(L)} \boldsymbol{K}^{(L)}\right)^{-1} \left(\boldsymbol{B}^{(L)}\right)^T + \right. \qquad (22) \\ &\left. + \Delta t \gamma^{(S)} \boldsymbol{B}^{(S)} \left(\boldsymbol{M}^{(S)} + \Delta t^2 \beta^{(S)} \boldsymbol{K}^{(S)}\right)^{-1} \left(\boldsymbol{B}^{(S)}\right)^T\right) \boldsymbol{\lambda}_{k,j+1} = \boldsymbol{B}^{(L)} \dot{\boldsymbol{w}}^{(L)}_{k,j+1} + \boldsymbol{B}^{(S)} \dot{\boldsymbol{w}}^{(S)}_{k,j+1}. \end{aligned}$$

and the vector $\ddot{\boldsymbol{q}}^{(S)}_{k,j+1}$ is computed from

$$\left(\boldsymbol{M}^{(S)} + \Delta t^2 \beta^{(S)} \boldsymbol{K}^{(S)}\right) \ddot{\boldsymbol{q}}^{(S)}_{k,j+1} = -\left(\boldsymbol{B}^{(S)}\right)^T \boldsymbol{\lambda}_{k,j+1}. \tag{23}$$

After m steps, the vector $\boldsymbol{u}^{(S)}_{k+1,0} = \boldsymbol{u}^{(S)}_{k,m}$ is known and the vector $\ddot{\boldsymbol{q}}^{(L)}_{k+1}$ is obtained from

$$\left(\boldsymbol{M}^{(L)} + (m\Delta t)^2 \beta^{(L)} \boldsymbol{K}^{(L)}\right) \ddot{\boldsymbol{q}}^{(L)}_{k+1} = -\left(\boldsymbol{B}^{(L)}\right)^T \boldsymbol{\lambda}_{k+1}. \tag{24}$$

3 Conclusion

Explicit and implicit versions of time integration algorithms enabling multi-time step approach, also called sub-cycling, are summarized in this contribution. A sub-cycling method will be used in connection with lattice discrete particle models which are extremely computationally demanding.

Acknowledgement: This outcome has been achieved with the financial support of the Grant Agency of the Czech Republic, project No. 21-28525S. The financial support is gratefully acknowledged.

References

[1] G. Cusatis, D. Pelessone, A. Mencarelli: *Lattice Discrete Particle Model (LDPM) for failure behavior of concrete I: Theory.* Cement & Concrete Composites, Vol. 33, 2011, pp. 881–890.

[2] G. Cusatis, A. Mencarelli, D. Pelessone, J. Baylot: *Lattice Discrete Particle Model (LDPM) for failure behavior of concrete I: Calibration and validation.* Cement & Concrete Composites, Vol. 33, 2011, pp. 891–905.

[3] K.J. Bathe: *Finite Element Procedures.* Prentice Hall, New Jersey, 1996.

[4] T. Belytschko, T.J.R. Hughes: *Computational methods for transient analysis.* Series Computational methods in mechanics, Vol. 1, North-Holland, Amsterdam, New York, Oxford, 1983.

[5] J. Kruis: *Domain Decomposition Methods for Distributed Computing.* Saxe-Coburg Publications, Kippen, Stirling, Scotland, 2006.

[6] T. Belytschko, Y.Y. Lu: *Explicit multi-time step integration for first and second order finite element semidiscretization.* Computer Methods in Applied Mechanics and Engineering, Vol. 108, 1993, pp. 353–383.

[7] A. Gravouil, A. Combescure: *Multi-time-step explicit-implicit method for non-linear structural dynamics.* International journal for numerical methods in engineering, Vol. 50, 2001, pp. 199–225.

Model order reduction of transport-dominated systems with rotations using shifted proper orthogonal decomposition and artificial neural networks

A. Kovárnová, M. Isoz

University of Chemistry and Technology, Prague
Institute of Thermomechanics of the CAS, Prague

1 Introduction

Transport-dominated systems and particle-laden flows are pervasive in both industrial and engineering practice. There are two predominant approaches for their simulations. The first one, *Eulerian-Eulerian*, approximates all particles by one Eulerian continuum and describes the whole system as a two-phase flow of interpenetrating continua. This approach is computationally feasible even for large systems; however, it requires a numerous empirical parameters. The second one, *Eulerian-Lagrangian*, describes only the flow as a Eulerian continuum and couples it with a Lagrangian description of individual particles, where each particle is characterized by its own set of Netwon's equations of motion. This approach does not need as many parameters; on the other hand, it tends to be computationally expensive, which complicates its usage in optimization or system control, where models needs to be evaluated repeatedly with only small changes in system parameters.

In order to reduce the computational costs of repeated model evaluations, various methods of model order reduction (MOR) have been introduced. The principal idea of MOR is to replace the original complicated model with a lower-dimensional, less computationally expensive, surrogate, while also preserving the most important properties of the original system. In this work, we focus on a-posteriori or data-driven methods in which the surrogate is constructured based on available full order model (FOM) results.

A common data-driven MOR approach in the field of computational fluid dynamics is the proper orthogonal decomposition (POD) combined with the Galerkin projection (used e.g. in [1]). POD takes as an input the matrix of snapshots of the original full order model $Y = (y_{ij}) = (y(x_i, t_j))$, $Y \in \mathbb{R}^{m\times n}$, decomposes it via singular value decomposition (SVD) and approximates it as e.g.

$$Y \approx Y^\ell = \sum_{r=1}^{\ell} \psi_r \otimes \eta_r = \Psi^\ell H^\ell\,, \quad \Psi^\ell = [\psi_1, \ldots, \psi_\ell] \in \mathbb{R}^{m\times\ell}\,, \; H^\ell = [\eta_1, \ldots, \eta_n] \in \mathbb{R}^{\ell\times n}\,, \quad (1)$$

where $\{\psi_r\}_{r=1}^{\ell}$ are stationary spatial modes, *toposes*, and $\{\eta_r\}_{r=1}^{\ell}$ are their time-dependent amplitudes, *chronoses*. The matrix Y can then be approximated via superposition of the first ℓ stationary spatial modes, $\ell \ll n$. However, this approach tends to be ineffective for transport-dominated systems, as a great number of spatial modes is needed to approximate Y with a sufficient accuracy [2].

Several methods to mitigate this issue have been introduced – one of the approaches is to apply a time-dependent transport operator that shifts the data into another frame of reference and compensates for the transport. In the present work, we used the shifted proper orthogonal decomposition (sPOD) [2], a method that is able to treat systems with multiple different types of transport by sorting the data into several co-moving frames of reference.

However, sPOD is only able to provide data in discrete time-steps. Common projection methods used to generate a time-continuous reduced order model (ROM), cannot be used for sPOD as there is no single projector. Instead, in this work artificial neural networks were used as a data-driven interpolator between the chronoses $\{\eta_r\}_{r=1}^{\ell}$ to provide a time-continuous model. The method is illustrated on an example from computational fluid dynamics. In particular, we have chosen a system where the transport in question is rotation, as rotation until recently posed challenges for the method.

2 Methods

As was said above, POD is based on SVD of the matrix of snapshots. Therefore, the relative importance of the individual modes corresponds to the rate of the singular value decay, i.e. the faster the decay is, the lesser number of modes is needed for a sufficient approximation.

The singular value decay for transport-dominated systems is usually extremely slow, as the positions of their predominant spatial structures are time-dependent. To illustrate this, let us a consider a single Gaussian pulse travelling with velocity c along the domain, see Fig. 1, then let us discretise it and save the data into the matrix $Y = [\boldsymbol{f}(\boldsymbol{x} - ct_1), \dots, \boldsymbol{f}(\boldsymbol{x} - ct_n)]$, $t_1 = 0$, $t_n = L$, $n = 185$. The singular values decay for this matrix is remarkably slow; however, if we apply a time-dependent transport operator $\mathcal{T}^{-\Delta^t}(\boldsymbol{f}(t,x)) := \boldsymbol{f}(t, x + ct)$, here $\Delta^t := ct$, which transforms it into a stationary wave, there is only one mode with a non-zero singular value.

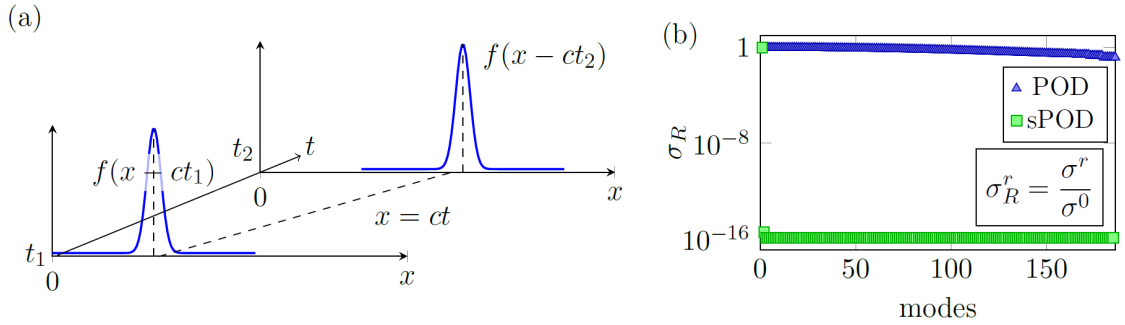


Figure 1: (a) Gaussian pulse, (b) singular value decay of the original travelling wave vs. of a stationary wave created by applying the transport operator.

In general, after applying the transport operators, equation (1) transforms into

$$Y^{\ell} \approx \sum_{k=1}^{N_\mathrm{f}} \mathcal{T}^{\Delta_k^t}\left(\Psi_k^{\ell_k} H_k^{\ell_k}\right) = \sum_{k=1}^{N_\mathrm{f}} \mathcal{T}^{\Delta_k^t}\left(\sum_{r=1}^{\ell_k} \psi_r^k \otimes \eta_r^k\right), \tag{2}$$

where N_f is the number of frames of reference, i.e., the number of different transports in the system, and ℓ_k the number of modes preserved in each frame. The exact algorithm used to sort the data into frames is outside the scope of this paper and can be found in [2].

So far, sPOD has been used on systems with rectangular spatial domain. In these systems, there are no problems with implementation of translation – the domain can be assumed to be periodic, in other words, the information that would otherwise be transported outside of the domain can be saved on the other side of the domain. On the other hand, the implementation of rotation is problematic, since the information in the corners of the domain travels outside and is lost, see Fig. 2. Therefore, the domain needs to be extended and padded with zeros before transport operator is applied and the decomposition is done. The sPOD accuracy remains the same, as the zeros have no effect on the resulting singular value decay.

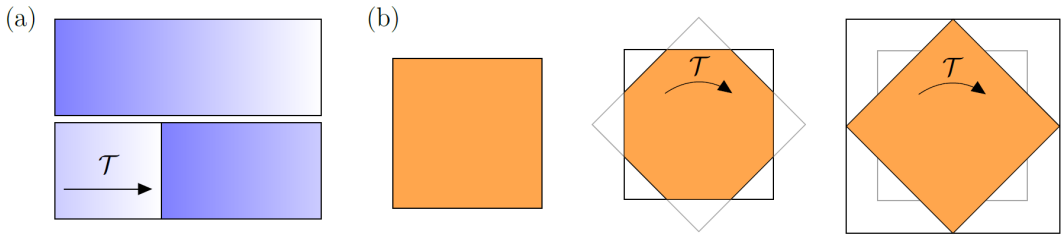


Figure 2: (a) Translation, (b) rotation. The domain needs to be extended (see right), otherwise some information is lost.

Temporal interpolation POD is commonly combined with various projection methods, e.g. the Galerkin projection, to obtain a time-continuous ROM. However, projection methods require (i) only one projector and (ii) the model to be described by one consistent system of differential equations. Shifted POD does not yield a single projector usable for (i), and Eulerian-Lagrangian descriptions of particle-laden flows are not expressed as (ii).

In this contribution, artificial neural networks were used for temporal interpolation, as they do not suffer from the shortcomings of projection methods. In particular, we utilized a single-layer perceptron, as it is the universal interpolator. The resulting method, sPODIANN (sPOD with interpolation via artificial neural networks) is, apart from the required knowledge of transport operators, purely data-driven.

3 Results

The presented method was illustrated on CFD-DEM models prepared with the open-source C++ library OpenFOAM combined with the openHFDIB-DEM solver [3]. This solver works with the immersed boundary method, where the presence of a particle in a cell is indicated via the λ field, $\lambda = 0$ for cells inside the fluid, $\lambda = 1$ inside the solid body and $\lambda \in (0; 1)$ at the solid-fluid interface.

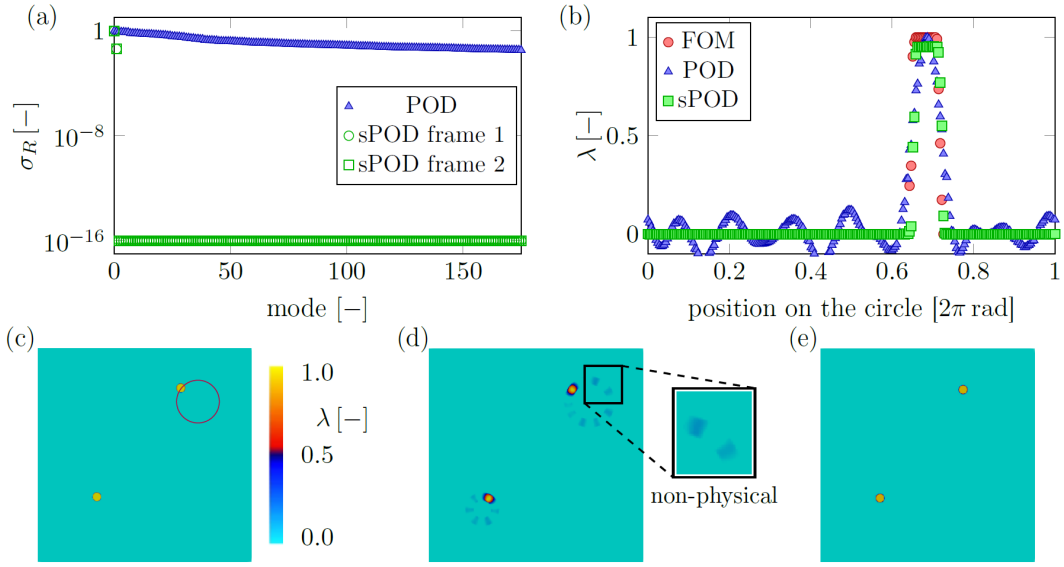


Figure 3: Two discs, λ (a) Singular value decay for POD and both sPOD frames, (b) a slice through λ field along the circle on (c) for FOM, POD and sPOD reconstruction, (c) – (e) λ field at $t = 1.42$ for FOM, POD (24 modes) and sPOD ($1 \cdot 2$ modes).

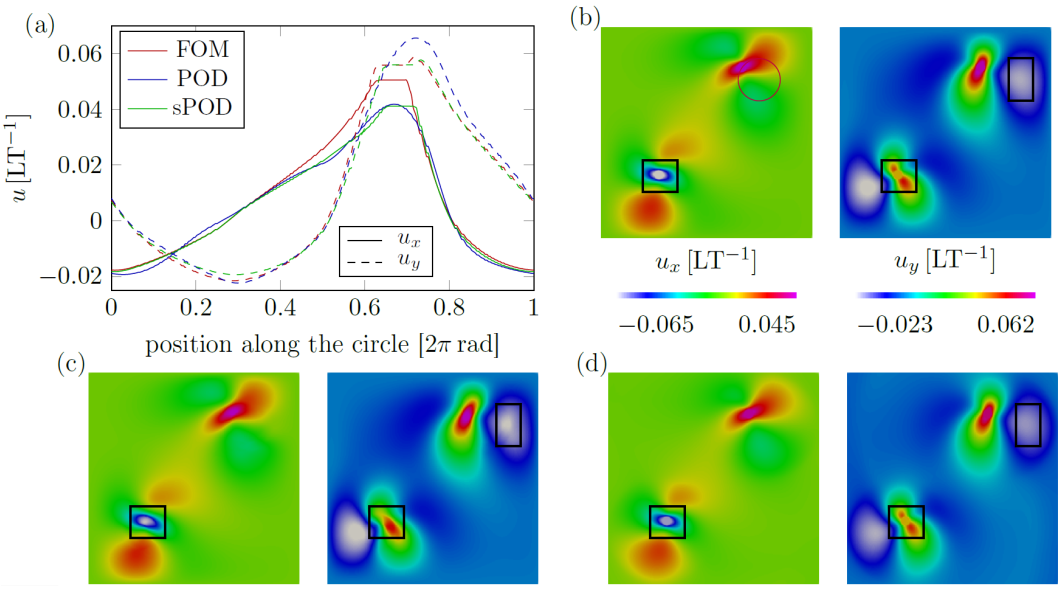


Figure 4: Two discs, velocity. (a) Velocities along the circle on (b) for FOM, POD and sPOD reconstruction, (b) – (d) components of the velocity field at $t = 1.42$ for FOM, POD (12 modes) and sPOD ($2 \cdot 2$ modes). Note the areas with the most distinct improvement of sPOD vs. POD in the black rectangles.

We present an sPODIANN based ROM of two discs, $d = 8$ mm, moving along circular trajectories inside a square domain, $L = 0.2$ m, filled with a fluid. The first sphere travels along a circle $S_1 = (0.15, 0.15)$, $r_1 = 0.02\,\mathrm{m}$ with a prescribed angular velocity $\omega_1 = 1.2\pi\,\mathrm{s}^{-1}$, the second one along a circle $S_2 = (0.05, 0.05)$, $r_2 = 0.012\,\mathrm{m}$ with $\omega_2 = 2\pi\,\mathrm{s}^{-1}$. The linear velocity is identical for both spheres. The reduction was performed on the λ field (see Fig. 3) and velocity field generated by the movement (Fig. 4), in both the cases sPODIANN outperformed its POD analogue.

4 Conclusion

In this contribution, we presented a framework for model order reduction that combines shifted proper orthogonal decomposition, a method for MOR of data with dominant transport, and combined it with interpolation via artificial neural networks to obtain a time-continuous model. We have illustrated the method functionality on a CFD-DEM model, where it outperformed a standard POD-based method. The long-term goal of this research is to extend the framework for parametrized systems.

Acknowledgement: The work was financially supported by the institutional support RVO:61388998.

References

[1] S. Chaturantabut, D.C. Sorensen: *Nonlinear model reduction via discrete empirical interpolation.* SIAM J. Sci. Comput., 32:2737-2764, 2010.

[2] J. Reiss: *Optimization-based modal decomposition for systems with multiple transports.* SIAM J. Sci. Comput., 43:A2079-A2101, 2021.

[3] M. Isoz, M. Kotouč Šourek, O. Studeník, P. Kočí: *Hybrid fictitious domain-immersed boundary solver coupled with discrete element method for simulations of flows laden with arbitrarily-shaped particles.* Comp. & Fluids, 244:105538, 2022.

Implementation of wall functions into a hybrid fictitious domain-immersed boundary method

L. Kubíčková, M. Isoz

University of Chemistry and Technology, Prague
Institute of Thermomechanics of the CAS, Prague

1 Introduction

Computational fluid dynamics (CFD) is an established numerical tool that is nowadays used in a wide range of applications for designing and testing new devices and components. However, the speed, accuracy and stability of every CFD simulation relies on the quality of the computational mesh and mesh generation (meshing) is considered to be one of the biggest bottlenecks of CFD [1]. Especially when CFD is used in geometry optimizations, meshing represents a rather big obstacle in improvement of effectiveness and robustness of optimization methods.

The mesh-related difficulties can be evaded by utilization of an immersed boundary method (IBM). In IBM, the complex geometry-conforming meshes are replaced by simple ones where the geometry is represented by an indicator scalar field (λ) and adjustment of governing equations [2]. Usage of IBM in geometry optimizations allows for a substantial speed-up since the geometry changes are reflected only in the indicator scalar field allowing the mesh to be the same throughout the whole optimization [3, 4].

However, almost all the IBM applications have been in the low-Reynolds number regime [5] where the boundary layer is rather wide, can be resolved and the boundary conditions on the immersed walls can be satisfied using simple linear or quadratic interpolation [2]. Nonetheless, optimizations of components under real-life conditions require the IBM to be able to handle problems of turbulent flows where the boundary layer thins out. A trivial solution is to locally refine the mesh. However, mesh refinement is linked to higher computational costs that, beyond some threshold, become unbearable, particularly in optimizations [3].

More affordable and optimization-oriented solution is to implement techniques already developed for geometry-conforming meshes, i.e., the Reynolds averaged simulation (RAS) approach comprising Reynolds averaged Navier-Stokes equations, turbulence closure models and wall functions. So far, there were several attempts on coupling of RAS modeling with IBM [1, 5, 6] reporting acceptable accuracy yet not sufficient robustness for employment in automated geometry optimization.

In the present contribution, we report on a research progress on implementation of the RAS approach into our custom immersed boundary method variant, the hybrid fictitious domain-immersed boundary method (HFDIB) [2], where we focus on the method effectiveness and robustness. The HFDIB-RAS approach comprises the k-ω turbulence model and wall functions for velocity and the closure variables (k and ω). The framework is implemented in OpenFOAM [7] and it is general enough for extension by any one or two equation RAS model. The method behavior is presented on several verification tests and the results show qualitative agreement with a standard CFD solver. Eventually, the goal is to use the HFDIB-RAS approach in our topology optimization framework [8].

2 Methods

First, let Ω be a finite volume computational mesh, and Ω_s and Ω_f its parts immersed in solid and fluid, respectively. In the HFDIB-RAS approach, the solid phase Ω_s is projected into Ω by a scalar field λ and adjustment of the governing equations. In each cell $\Omega_P \in \Omega$, the λ field is defined as

$$\lambda = \begin{cases} 0 & \text{if } \Omega_P \in \Omega_\mathrm{f} \\ 1 & \text{if } \Omega_P \in \Omega_\mathrm{s} \\ \tilde{\lambda} \in (0,1) & \text{if } \Omega_P \in \Gamma_\mathrm{sf} \end{cases}, \quad \Omega = \Omega_\mathrm{f} \cup \Omega_\mathrm{s} \cup \Gamma_\mathrm{sf}, \quad \tilde{\lambda} = 0.5 \left[1 - \tanh\left(\frac{y_\perp}{\overline{V}^{\frac{1}{3}}} \right) \right] \tag{1}$$

where Γ_sf marks the fluid-solid interface, $\overline{V}$ is the average cell volume in Ω and $y_\perp$ is the perpendicular distance from P, the center of Ω_P, to the solid surface. For the purpose of the correct boundary layer modeling, we had to further sort the cells into (i) in-solid cells having $\lambda \geq 0.5$, i.e. $P \in \Omega_\mathrm{s}$, (ii) boundary cells having either $\lambda \in (0, 0.5)$ or $\lambda = 0$ and an in-solid cell as a vertex neighbor and (iii) free stream cells that comprised the rest of the cells.

The turbulent flow in Ω is then described by Reynolds averaged Navier-Stokes equations with an additional force term $\boldsymbol{f}_\mathrm{ib}$. For incompressible and isothermal flow of a Newtonian fluid, the equations have the form of

$$\begin{gathered} \mathcal{M}(\boldsymbol{u}) = -\nabla\tilde{p} + \boldsymbol{f}_\mathrm{ib} \\ \nabla \cdot \boldsymbol{u} = 0 \end{gathered}, \qquad \begin{gathered} \mathcal{M}(\boldsymbol{u}) = \nabla \cdot (\boldsymbol{u} \otimes \boldsymbol{u}) - \nabla \cdot \left[\nu_\mathrm{eff} \left(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\mathrm{T} \right) \right] \\ \boldsymbol{f}_\mathrm{ib} = \alpha(\lambda) \left(\mathcal{M}(\boldsymbol{u}_\mathrm{ib}) + \nabla\tilde{p} \right) \end{gathered} \tag{2}$$

where $\tilde{p}$ and $\boldsymbol{u}$ are averaged kinematic pressure and velocity, respectively. The ν_eff is the effective viscosity and it is given by $\nu_\mathrm{eff} = \nu + \nu_\mathrm{t}$ where ν is the fluid viscosity and ν_t is turbulence viscosity. Turbulence viscosity is computed via a suitable turbulence closure model. First, we chose to implement the $k - \omega$ turbulence model in which $\nu_\mathrm{t} = k/\omega$ with k being the turbulence kinetic energy and ω the specific rate of dissipation of k. The two closure variables, k and ω, are acquired by solving their conservation equations, for details see [9].

The source term $\boldsymbol{f}_\mathrm{ib}$ in (2) enforces the prescribed boundary conditions on Γ_sf. The scope of effect of $\boldsymbol{f}_\mathrm{ib}$ is determined by a scalar field α, where $\alpha = 1$ for in-solid and boundary cells, and $\alpha = 0$ for free-stream cells. Furthermore, similar source term has to be added to the k conservation equation where it depends on α and k_ib. The ω conservation equation was not extended but an additional source term, but to enforce the boundary conditions the ω_ib values are required, as well.

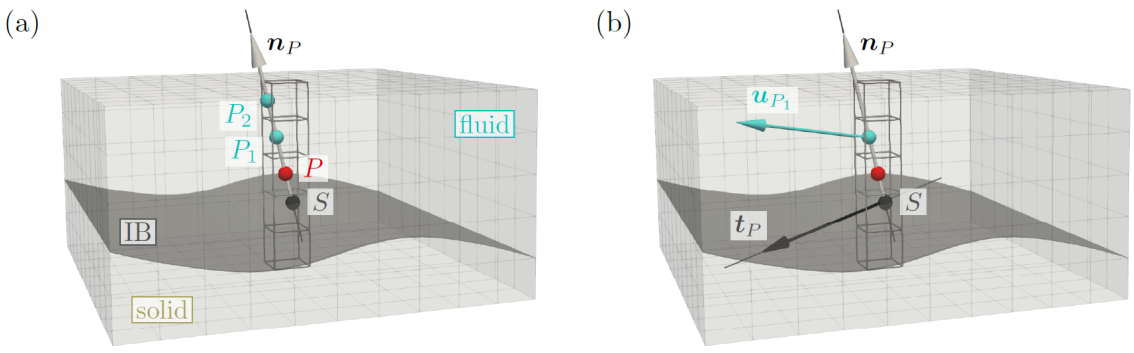


Figure 1: Data required by the HFDIB-RAS method. (a) Interpolation points (P_1 and P_2) for a boundary cell with center P and surface point S. The $\boldsymbol{n}_P$ is a unit vector normal to the surface computed as $\boldsymbol{n}_P = -(\nabla\lambda)_P / \|(\nabla\lambda)_P\|$. (b) Projection of the surface unit tangential vector $\boldsymbol{t}_P$, which is computed from the $\boldsymbol{u}_{P_1}$ as $\boldsymbol{t}_P = \boldsymbol{u}_{t,P_1} / \|\boldsymbol{u}_{t,P_1}\|$ where $\boldsymbol{u}_{t,P_1} = \boldsymbol{u}_{P_1} - (\boldsymbol{n}_P \cdot \boldsymbol{u}_{P_1})\boldsymbol{n}_P$.

The values of $\boldsymbol{u}_{\mathrm{ib}}$, k_{ib} and ω_{ib} are estimated via wall functions prior to the solution of (2) and they are enforced in cells with $\alpha = 1$. The values ω_{ib} are enforced by a direct modification of the ω equation matrix for the purposes of simulation stability [9]. For $\boldsymbol{u}$ and k, the preset values are enforced by the aforementioned source term addition and modifications of the outer solver loop [2].

A correct setting of the enforced values is important to satisfy the boundary conditions at the fluid solid interface and for a physical simulation of the boundary layer behavior. For the in-solid cells, $\boldsymbol{u} = \boldsymbol{0}$ and $k = 0$ are enforced. Furthermore, ω is set to be the maximum of the ω field from the previous iteration. In boundary cells, the value in the cell center is unknown and must be interpolated using values from the free stream cells and value at the surface computed via wall functions, which we implemented in a switch-based variant similarly to OpenFOAM [7]. The exact interpolation type is chosen according to the mesh capability to resolve the fluid boundary layer.

The resolution of the boundary layer was be described by a dimensionless indicator y^+ [9], which is defined as $y^+ = \left(y^{\perp} u_\tau\right)/\nu$ with u_τ being the friction velocity. Based on it, it can be said whether the boundary cell center is inside the viscous sublayer, buffer layer or in the logarithmic region of the boundary layer. Consequently, for cells in the viscous sublayer, we use a quadratic interpolation and in the logarithmic region, we use a custom logarithmic interpolation. The buffer layer is treated either as a viscous sublayer of as a logarithmic region based on a constant switch value [7]. The interpolation stencil and local coordinate system for a boundary cell are illustrated in Fig. 1.

3 Results

The HFDIB-RAS capabilities were tested on several verification tests, which were designed to show the research progress. In every test, we compared our results to a standard OpenFOAM solver, simpleFoam [7]. Velocity profiles from six such tests are depicted in Fig. 2 and 3. The results show a good qualitative agreement between simpleFoam and the HFDIB-RAS approach that holds for a wide range of flow Reynolds numbers. However, further work on the solution accuracy is required.

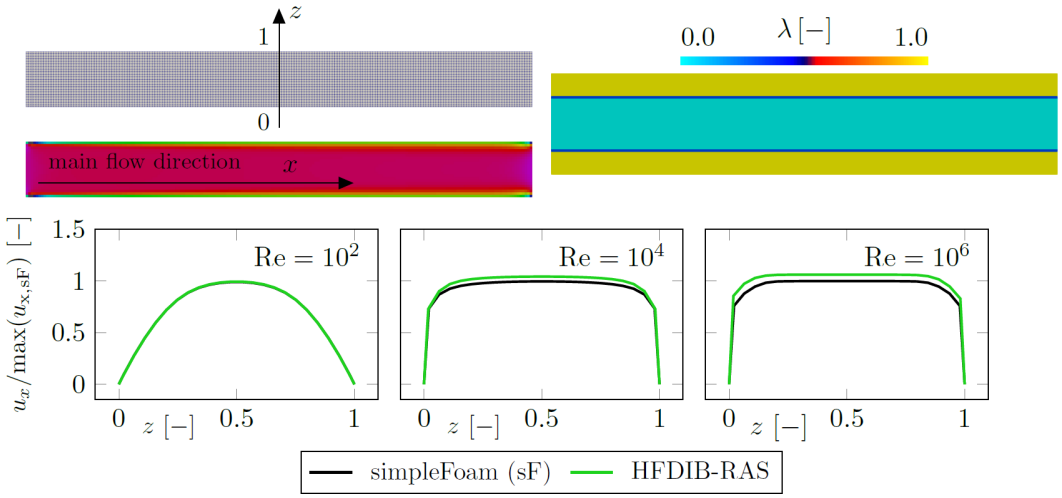


Figure 2: Comparison of velocity profiles in a straight pipe along the z-line for different flow Reynolds numbers.

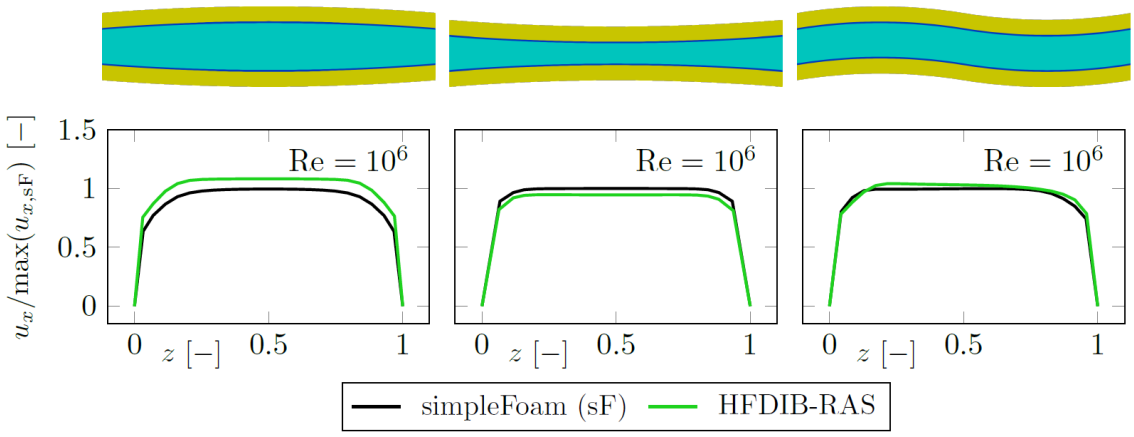


Figure 3: Comparison of velocity profiles in bent pipes along the z-line for $\mathrm{Re} = 10^6$.

4 Conclusion

In this contribution, we present development of the HFDIB-RAS approach that is designed to allow affordable turbulence modeling via our custom immersed boundary method variant. The approach combines modified version of Reynolds averaged Navier-Stokes equations, the $k - \omega$ closure turbulence model and switch-based wall functions. Verification tests showed that results acquired via HFDIB-RAS method are in a good qualitative agreement with standard geometry-conforming simulations. Nevertheless, the future development shall focus on improvement of the solution accuracy.

Acknowledgement: The work was supported by the institutional support RVO:61388998.

References

[1] F. Capizzano: *A Turbulent Wall Model for Immersed Boundary Methods.* In 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 01 2010.

[2] M. Isoz, M. Kotouč Šourek, O. Studeník, P. Kočí: *Hybrid fictitious domain-immersed boundary solver coupled with discrete element method for simulations of flows laden with arbitrarily-shaped particles.* Computers and Fluids, 244:105538, 2022.

[3] L. Kubíčková, M. Isoz, J. Haidl: *Increasing Ejector Efficiency via Diffuser Shape Optimization.* In Proceedings of Topical Problems of Fluid Mechanics 2021. IT CAS, 02 2021, pp. 79–86

[4] S. Kubo, A. Koguchi, K. Yaji, T. Yamada, K. Izui, S. Nishiwaki: *Level set-based topology optimization for two dimensional turbulent flow using an immersed boundary method.* Journal of Computational Physics, 446:110630, 2021.

[5] G. Kalitzin, G. Iaccarino: *Turbulence modeling in an immersed-boundary RANS method.* Center for Turbulence Research Annual Research Briefs, 01 2002, pp. 415–426.

[6] N. Troldborg, N.N. Sørensen, F. Zahle: *Immersed boundary method for the incompressible Reynolds Averaged Navier-Stokes equations.* Computers and Fluids, 237:105340, 2022.

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

[8] L. Kubíčková, M. Isoz: *Hybrid Fictitious Domain-Immersed Boundary Method in CFD-Based Topology Optimization.* In Proceedings of Topical Problems of Fluid Mechanics 2022. IT CAS, 02 2022, pp. 119–126.

[9] D.C. Wilcox: *Turbulence modeling for CFD.* DCW Industries, USA, 3 edition, 2006.

On thermodynamically consistent coupling of the Barcelona Basic Model with a hydraulic model for unsaturated soils

T. Ligurský

Institute of Geonics of the CAS, Ostrava

1 Introduction

The Barcelona Basic Model (BBM) is a constitutive model describing elastoplastic behaviour of unsaturated soils. It was proposed originally in [1] as a mechanical model, without taking into account hydraulic processes in the soils. Recently, this model has been exploited widely in some engineering applications where hydraulic effects have been captured as well, for instance, in modelling of clay-based barriers for nuclear waste disposal. However, surprisingly little work has been done in verification of basic thermodynamical principles of coupled hydro–mechanical (HM) models incorporating BBM to the best knowledge of the author. The aim of the lecture is thus to present a theoretical analysis of thermodynamically consistent coupling of BBM with a hydraulic model.

2 Formulation of BBM

One considers a soil as a porous solid with pores filled by water and gas. The formulation of BBM in [1] treats the porous solid as an isotropic material under triaxial stress conditions and the small-strain assumption. Non-linear poroelasticity and non-associated plasticity are adopted. Two yield surfaces are introduced: a loading–collapse and a suction–increase one. In this presentation, the latter yield surface will be omitted, for simplification. The sign convention with the stresses positive in tension and the pressures positive in compression will used.

Deformation of the porous solid is described by the strain tensor $\boldsymbol{\varepsilon} \equiv 1/2\big(\boldsymbol{\nabla u} + (\boldsymbol{\nabla u})^\top\big)$, where $\boldsymbol{u}$ denotes the displacement vector of the solid. The strain $\boldsymbol{\varepsilon}$ is decomposed additively into the elastic (superscript el) and plastic (superscript p) part:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^{el} + \boldsymbol{\varepsilon}^{p}.$$

The independent stress variables are the net stress $\boldsymbol{\sigma}'$ and the suction s defined by:

$$\boldsymbol{\sigma}' \equiv \boldsymbol{\sigma} + p_g \boldsymbol{I}, \qquad s \equiv p_g - p_w,$$

where $\boldsymbol{\sigma}$ stands for the total stress and p_g and p_w are the pressures of gas and water in the pores. The net pressure p' and the volumetric strain ϵ_v are introduced as:

$$p' \equiv -\frac{1}{3}\operatorname{tr}\boldsymbol{\sigma}', \qquad \epsilon_v \equiv -\operatorname{tr}\boldsymbol{\varepsilon},$$

whereas the deviatoric stress q and deviatoric strain ϵ_q are given by:

$$q = -(\sigma_1 - \sigma_3), \qquad \epsilon_q = -\frac{2}{3}(\varepsilon_1 - \varepsilon_3),$$

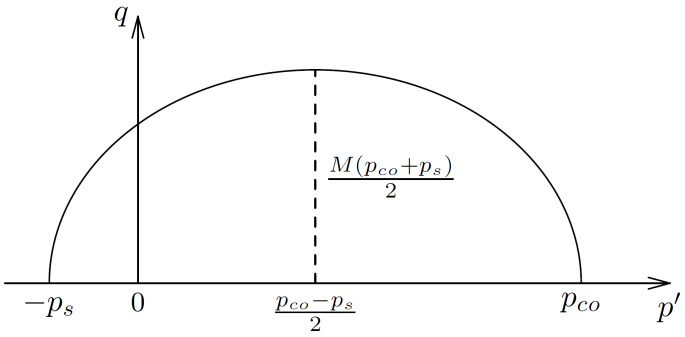


Figure 1: Yield surface $f = 0$ in the (p', q)-plane.

with σ_1 and σ_3 and ε_1 and ε_3 being principal stresses and strains. (Note that ϵ_v is positive in contraction.)

The elastic behaviour of the porous solid is described as follows:

$$d\epsilon_v^{el} = \frac{\kappa}{1+e_0}\frac{dp'}{p'} + \frac{\kappa_s}{1+e_0}\frac{ds}{s+p_{atm}}, \qquad d\epsilon_q^{el} = \frac{dq}{3\mu},$$

where d denotes the differential operator with respect to time, κ and κ_s are elastic stiffness parameters, μ is the shear modulus, e_0 stands for the initial void ratio and p_{atm} for the atmospheric pressure.

The loading–collapse yield function f can be defined by (see Figure 1):

$$f(p', q, s, p_{co}^*) := \left(p' - \frac{p_{co} - p_s}{2}\right)^2 + \frac{q^2}{M^2} - \left(\frac{p_{co} + p_s}{2}\right)^2,$$

where

$$p_s = ks$$

and the net consolidation pressure p_{co} at the current suction is related to the consolidation pressure p_{co}^* at saturated conditions and a reference pressure p_{ref} by:

$$p_{co} = p_{ref}\left(\frac{p_{co}^*}{p_{ref}}\right)^{\frac{\lambda(0)-\kappa}{\lambda(s)-\kappa}}$$

with:

$$\lambda(s) = \lambda(0)\big((1-r)e^{-\beta s} + r\big).$$

Here M, k, $\lambda(0)$, r and β are material parameters.

The non-associated potential g is introduced by:

$$g(p', q, s, p_{co}^*) := \left(p' - \frac{p_{co} - p_s}{2}\right)^2 + \frac{\alpha q^2}{M^2} - \left(\frac{p_{co} + p_s}{2}\right)^2, \tag{1}$$

where α is an additional parameter. The flow rule for strains then reads as:

$$d\epsilon_v^p = d\lambda\frac{\partial g}{\partial p'} = 2d\lambda\left(p' - \frac{p_{co} - p_s}{2}\right), \qquad d\epsilon_q^p = d\lambda\frac{\partial g}{\partial q} = 2d\lambda\frac{\alpha q}{M^2},$$

where the plastic multiplier $d\lambda$ satisfies the usual complementarity conditions:

$$d\lambda \geq 0, \qquad f \leq 0, \qquad d\lambda \cdot f = 0.$$

Finally, the hardening law is expressed by:

$$\frac{dp_{co}^*}{p_{co}^*} = \frac{1+e_0}{\lambda(0)-\kappa}d\epsilon_v^p.$$

3 Hydraulic behaviour

Let n denote the Eulerian porosity of the porous solid: $n\,\mathrm{d}\Omega_t$ is the volume of the porous space in an arbitrary infinitesimal volume $\mathrm{d}\Omega_t$ in the deformed current configuration. The Lagrangian porosity ϕ refers the current porous volume in $\mathrm{d}\Omega_t$ to the corresponding initial infinitesimal volume $\mathrm{d}\Omega_0$: $\phi\,\mathrm{d}\Omega_0 = n\,\mathrm{d}\Omega_t$. The Eulerian (usual) saturation s_f related to pore fluid f ($f = w$ for water and $f = g$ for gas) is defined with regard to the current porous volume: $\phi s_f\,\mathrm{d}\Omega_0$ is the current volume occupied by fluid f in the current porous volume $\phi\,\mathrm{d}\Omega_0$. Finally, one introduces the Lagrangian saturation S_f, $f = w, g$, with regard to the initial porous volume by [2]:

$$\phi s_f = \phi_0 S_f + \varphi_f, \qquad S_w + S_g = 1,$$

where ϕ_0 denotes the initial Lagrangian porosity and φ_f represents deformation of the porous volume that is occupied by fluid f currently.

Hydraulic processes in a soil can be described by the saturation S_w and the partial pore deformation φ_w. The latter can be split into the elastic and plastic part in an analogous way as the strain:

$$\varphi_w = \varphi_w^{el} + \varphi_w^{p}.$$

4 Thermodynamically consistent HM coupling

Coupling between BBM and a hydraulic model is thermodynamically consistent when the dissipation associated with the overall porous system is non-negative. Following [3, 4, 5, 6] and assuming an incompressible solid matrix, negligible hysteretic effects on the water retention curve and isothermal evolutions, one can show that the dissipation is non-negative if:

(a) There exist

(i) an elastic energy F depending on the elastic strains ϵ_v^{el}, ϵ_q^{el} and the suction s;

(ii) a locked energy Z depending on the plastic strain ϵ_v^{p} and the suction s;

(iii) an energy U of the interfaces between the solid, water and gas which depends on s

such that:

$$p' = \frac{\partial F}{\partial \epsilon_v^{el}}, \qquad q = \frac{\partial F}{\partial \epsilon_q^{el}}, \qquad \varphi_w^{el} = \frac{\partial F}{\partial s} + \frac{\partial Z}{\partial s}, \qquad p_{co} = \frac{\partial Z}{\partial \epsilon_v^{p}}, \qquad S_w = \frac{\mathrm{d}U}{\mathrm{d}s}. \tag{2}$$

(b) A flow rule for φ_w^p is such that:

$$p' d\epsilon_v^p + q d\epsilon_q^p - s d\varphi_w^p - p_{co} d\epsilon_v^p \geq 0. \tag{3}$$

In particular, let us take the flow rule from [6]:

$$d\varphi_w^p = 0.$$

When the plastic part of BBM is supplemented by this relation, it can be shown that requirement (3) is always satisfied under the condition $\alpha \geq 1/2$, where α is the parameter from (1).

Besides, the last equality in (2) leads to a water retention curve in the classical form $s \mapsto S_w(s)$, which may be an arbitrary function.

Further, the first, second and fourth equality in (2) and the constitutive relationships in BBM determine the energies F and Z completely up to a function of s. The third equality in (2) then leads to:

$$\varphi_w^{el} = \varphi_{wp'}^{el}(\epsilon_{vp'}^{el}, s) + \varphi_{wp}^{el}(\epsilon_v^p, s) + \varphi_{ws}^{el}(s), \tag{4}$$

where $\varphi^{el}_{wp'}$ is a certain exponential function of the elastic volumetric strain $\epsilon^{el}_{vp'}$ in response to the net pressure p':

$$\epsilon^{el}_{vp'} \equiv \epsilon^{el}_{v} - \frac{\kappa_s}{1+e_0} \ln \frac{s + p_{atm}}{s_0 + p_{atm}}$$

(this is given from $\partial F/\partial s$), and φ^{el}_{wp} is a certain exponential function of the plastic strain ϵ^{p}_{v} (given by $\partial Z/\partial s$). Only the last function φ^{el}_{ws} in (4), which should correspond to deformation of the pore space occupied by water in response to the suction s, is not determined by the thermodynamical relations, and it can be chosen arbitrarily according to physical experiments.

Unfortunately, the exponential dependencies of $\varphi^{el}_{wp'}$ and φ^{el}_{wp} appear to be physically problematic in the expression (4) for φ^{el}_{w}, as it will be demonstrated in the lecture.

5 Conclusion

It is possible to couple BBM with a hydraulic model in a thermodynamically consistent way, in principle (at least when the suction–increase yield surface is omitted). However, the resulting coupling does not seem to be completely physically consistent. More generally, thermodynamical restrictions on HM coupling of poromechanical models using the net stress and suction as independent stress variables appear to be too strict and problematic to be fulfilled entirely.

On the other hand, mechanical models based on a Bishop-type (also termed effective) stress look to be more amenable to thermodynamically consistent HM coupling according to [5]. Thus one can conclude that these appear to be more suitable for coupled HM modelling from this point of view.

Acknowledgement: This work has been supported by European Union's Horizon 2020 research and innovation programme under grant agreement number 847593 and by The Czech Radioactive Waste Repository Authority (SÚRAO) under grant agreement number SO2020-017.

References

[1] E.E. Alonso, A. Gens, A. Josa: *A constitutive model for partially saturated soils.* Géotechnique 40(3), 1990, pp. 405–430.

[2] O. Coussy: *Mechanics and Physics of Porous Solids.* John Wiley & Sons, 2010.

[3] O. Coussy: *Poromechanics.* John Wiley & Sons, 2004.

[4] S. Samat, J. Vaunat, A. Gens: *A thermomechanical framework for modeling the response of unsaturated soils.* In: D.G. Toll, C.E. Augarde, D. Gallipoli, S.J. Wheeler (eds.): Unsaturated Soils: Advances in Geo-Engineering, CRC Press, 2008, pp. 547–552.

[5] S. Samat: *Thermomechanical modelling of ground response under environmental actions.* PhD thesis, Universitat Politècnica de Catalunya, Barcelona, 2016.

[6] O. Coussy, J.-M. Pereira, J. Vaunat: *Revisiting the thermodynamics of hardening plasticity for unsaturated soils.* Computers and Geotechnics 37, 2010, pp. 207–215.

Post-buckling solution for nonlinear beam developed by D.Y. Gao

J. Machalová, H. Netuka

Department of Mathematical Analysis and Applications of Mathematics
Faculty of Science, Palacký University, Olomouc

1 Introduction

In this contribution, we will consider the nonlinear beam of length L with only axial loading, i.e. without vertical loading. The focus of our interest will be on the so-called buckling phenomenon. In our context, buckling is a sudden change in the shape of a given beam at a certain critical value of compressive axial load.

The phenomenon was seriously investigated for the first time by Leonhard Euler (1707 â€“ 1783). In 1757 he derived the formula, nowadays known as the Euler formula, which gives the maximum axial load that a long slender column (i.e. beam in vertical position) can carry without buckling. Unfortunately, the exact shape of the buckled column cannot be achieved within Euler's theory, which is based on the following linear equation

$$EI\,w'''' + P w'' = 0, \qquad \text{in } (0,\mathrm{L}), \tag{1}$$

where E is Young's modulus, I area moment of inertia, w transversal displacement function and P external axial force, compressive if $P > 0$. For this linear model Euler proposed a solution using eigenvalues and eigenfunctions, which is still widely used but has qualitative character only.

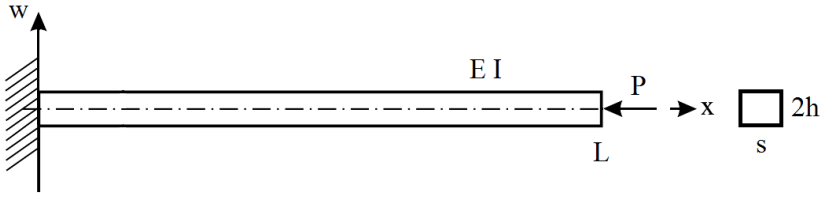


Figure 1: Beam with axial load

Nonlinear model developed by D.Y. Gao in [1]

$$EI\,w'''' - E\alpha\,(w')^2 w'' + \lambda\,w'' = 0 \qquad \text{in } (0,\mathrm{L}), \tag{2}$$

is intended for a full post-buckling analysis. Here, $\alpha = 3\mathrm{hs}(1-\nu^2)$, where ν is the Poisson's ratio, 2h indicates beam's height, s its width, and in addition, because we assume a rectangular cross-section, $I = \frac{2}{3}\,\mathrm{h}^3\mathrm{s}$, see Fig. 1. Finally, λ, is a constant that depends on the axial load P (see [1]). Therefore, it can be expressed as $\lambda = \mu P$, $\mu > 0$. The authors have already presented this model several times at previous seminars, see e.g. [2], or in papers, see e.g. [3], however buckling was not considered there.

In a mathematical sense, buckling is a bifurcation in the solution to the equations of static equilibrium. Beyond the bifurcation point, we enter the post-buckling region with multiple solutions. It is well known that the total potential energy of the beam becomes nonconvex there. See, e.g. [4].

2 Determination of the critical load

Obviously, the key problem is to establish a formula for (2) which will give the relation to Euler's formula for (1). For this purpose, let us consider the variational formulation of (2)

$$\text{find } w \in V: \quad a(w,v) - \lambda b(w,v) + \pi(w,v) = 0 \qquad \forall v \in V, \tag{3}$$

where

$$a(w,v) = EI \int_0^{\mathrm{L}} w''v'' \mathrm{d}x, \quad b(w,v) = \int_0^{\mathrm{L}} w'v' \mathrm{d}x, \quad \pi(w,v) = \frac{1}{3} E\alpha \int_0^{\mathrm{L}} (w')^3 v' \mathrm{d}x,$$

and V is the space of admissible displacements, which satisfies $H_0^2((0,\mathrm{L})) \subseteq V \subset H^2((0,\mathrm{L}))$ and contains the constraints on stable boundary conditions. The associated energy functional Π now reads as

$$\Pi(v) = \frac{1}{2} a(v,v) - \frac{1}{2} \lambda b(v,v) + \frac{1}{4} \pi(v,v), \qquad v \in V. \tag{4}$$

Its convexity or non-convexity depends, of course, on the value λ.

The useful and commonly used tool in the case of the linear equation (1) is the Rayleigh quotient

$$R(v) = \frac{a(v,v)}{b(v,v)} = \frac{\int_0^{\mathrm{L}} EI\,(v'')^2 \mathrm{d}x}{\int_0^{\mathrm{L}} (v')^2 \mathrm{d}x}, \qquad v \in V, \tag{5}$$

as for the Euler critical load, we have

$$P_{cr}^E = \lambda_{cr}^E = \min_{\substack{v \in V \\ v \neq 0}} R(v).$$

Although we cannot use a direct analogy for the equation (2), we are able to generalize (5) to the nonlinear Rayleigh quotient for $u, v \in V$ as follows:

$$\mathcal{R}_u(v) = \frac{a(v,v) + \frac{1}{3}\int_0^{\mathrm{L}} E\alpha\,(u')^2 (v')^2 \mathrm{d}x}{b(v,v)}.$$

Then the critical load for the considered nonlinear beam (2) can be defined as

$$\lambda_{cr} = \min_{u \in V} \min_{\substack{v \in V \\ v \neq 0}} R_u(v) = \min_{u \in V} \min_{\substack{v \in V \\ v \neq 0}} \left(R(v) + \frac{1}{3} \frac{\int_0^{\mathrm{L}} E\alpha\,(u')^2 (v')^2 \mathrm{d}x}{\int_0^{\mathrm{L}} (v')^2 \mathrm{d}x} \right).$$

From this we immediately obtain the first important result

$$\lambda_{cr} = \lambda_{cr}^E = P_{cr}^E.$$

Next, we have to answer the question of how many solutions the problem (3) has for $\lambda > \lambda_{cr}$. At $\lambda = \lambda_{cr}$ there is a bifurcation point. For this purpose we need to examine properties of the functional (4) and its stationary points, as they are solutions of (3). The first examination states that this functional is coercive and continuous for any λ. Furthermore, it is obvious that the function $w_0 = 0$ is its stationary point and it holds

$$\Pi(w) = \begin{cases} 0 & \text{if } w = w_0 = 0, \\ -\frac{1}{4}\pi(w_1, w_1) < 0 & \text{for any other stationary point } w_1. \end{cases}$$

The last trivial observation says that if w_1 is a stationary point, then $-w_1$ is also a stationary point. Then it can be proved that the problem has exactly 3 solutions in $(\lambda_{cr}, \overline{\lambda})$ for some $\overline{\lambda} > \lambda_{cr}$. The situation is shown in the Fig. 2, where $w_{max} = \max_{x \in [0,\mathrm{L}]} w(x)$. This is the second important result.

As for the functional (4), the situation is illustrated by a schematic diagram on Fig. 3, which corresponds to the so-called double-well potential known from quantum mechanics.

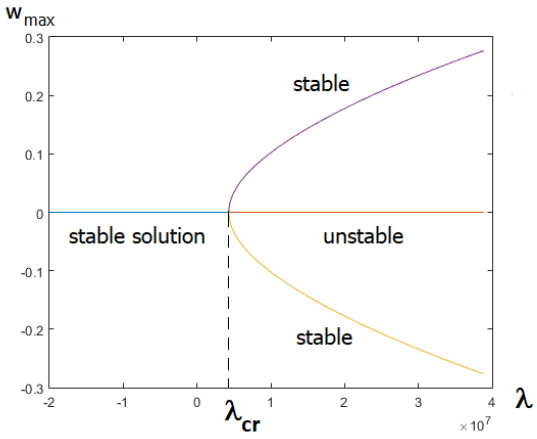


Figure 2: Pitchfork diagram of buckling solutions

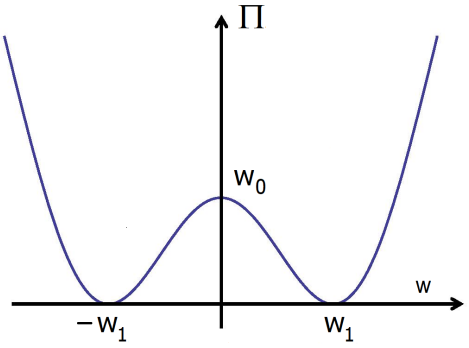


Figure 3: Cross-section of energy functional (scheme)

3 Analysis of computational results

Because of the analysis of buckling problems, many numerical experiments have been performed using the finite element method [5]. The iteration process was built on 32 beam elements, each with 4 DOF, all elements having the same length. Four different boundary value problems were tested. As input parameters were considered: the elastic constants E and ν together with the geometric values h, s and L.

It is interesting but not surprising that for the axial load λ whose value is close to λ_{cr}, the shape of the deflection curve $w(x)$ is almost identical to the shape of the Euler solution. For λ far from λ_{cr}, the shapes remain similar but different. See Fig. 4, where the cantilever beam with input data $E = 2.1 \cdot 10^{11}\,[\mathrm{Pa}]$, $\nu = 0.3$, $\mathrm{h} = 0.05\,[\mathrm{m}]$, $\mathrm{s} = 0.1\,[\mathrm{m}]$, $\mathrm{L} = 1\,[\mathrm{m}]$, is presented. Note that from the calculation we get $\lambda_{cr} = 4.317951960 \cdot 10^6\,[N]$.

Many properties of the buckling problem cannot be derived analytically due to the nonlinear character of this problem. Therefore, data analysis must help us to get the results we want. Of course, this requires a very large number of calculated examples and many hours to analyze them. Nevertheless, the results are very satisfactory because they are valid regardless of the boundary conditions. The authors present them here in the form of the Tab. 1, where C is a real constant. Note that here we use the concept of "relative values" λ defined by

$$\lambda_r = \frac{\lambda}{\lambda_{cr}} = \frac{P}{P_{cr}}.$$

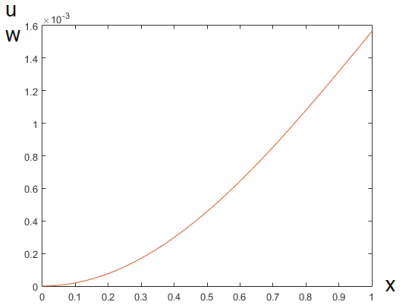
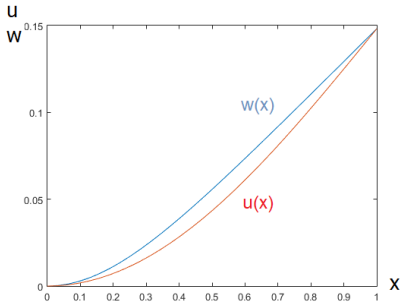


Figure 4: $\lambda = 1.001\lambda_{cr}$



$\lambda = 8\lambda_{cr}$

Table 1: Dependence of the deflection and functional on the change of parameter.

Parameter	change	λ_{cr}^{new}	w_{max}^{new} for the same values λ_r	Π^{new}
E	CE	$C\lambda_{cr}$	w_{max}	$C\,\Pi$
ν	$C\nu$	λ_{cr}	$\left(\frac{1-\nu^2}{1-(C\nu)^2}\right)^{1/2} w_{max}$	$\frac{1-\nu^2}{1-(C\nu)^2}\,\Pi$
h	Ch	$C^3\lambda_{cr}$	Cw_{max}	$C^5\,\Pi$
s	Cs	$C\lambda_{cr}$	w_{max}	$C\,\Pi$
L	CL	$\frac{1}{C^2}\lambda_{cr}$	w_{max}	$\frac{1}{C^3}\,\Pi$

Acknowledgement: This work has been supported by the grant CZ.02.1.01/0.0/0.0/17_049/0008408 Hydrodynamic design of pumps.

References

[1] D.Y. Gao: *Nonlinear elastic beam theory with application in contact problems and variational approaches.* Mechanics Research Communications, 23 (1), 1996, pp. 11–17.

[2] H. Netuka, J. Machalová: *Gao beam: From definition to contact problems.* SNA'19, Ostrava, January 21 - January 25, 2019.

[3] J. Machalová, H. Netuka: *Control variational method approach to bending and contact problems for Gao beam.* Applications of Mathematics, Vol. 62, No. 6, 2017, pp. 661–677.

[4] Z.P. Bazant, L. Cedolin: *Stability of Structures: Elastic, Inelastic, Fracture and Damage Theories.* World Scientific Publishing Co., 2010.

[5] J.N. Reddy: *An Introduction to Nonlinear Finite Element Analysis.* Oxford University Press, Oxford, 2004.

Building a fuel moisture model for the coupled fire-atmosphere model WRF-SFIRE from data: From Kalman filters to recurrent neural networks

J. Mandel[1], *J. Hirschi*[1], *A.K. Kochanski*[2], *A. Farguell*[2], *J. Haley*[3]
D.V. Mallia[4], *B. Shaddy*[5], *A.A. Oberai*[5], *K.A. Hilburn*[3]

[1]University of Colorado Denver, Denver, CO
[2]San José State University, San José, CA
[3]Colorado State University, Fort Collins, CO
[4]University of Utah, Salt Lake City, UT
[5]University of Southern California, Los Angeles, CA

Dedicated to the memory of Professor Radim Blaheta

1 Introduction

The WRF-SFIRE modeling system [4, 5] couples Weather Research Forecasting (WRF) model with a wildfire spread model and a fuel moisture content (FMC) model. The FMC is an important factor in wildfire behavior, as it underlies the diurnal variability and different severity of wildfires. The FMC model uses atmospheric variables (temperature, relative humidity, rain) from Real-Time Mesoscale Analysis (RTMA) to compute the equilibrium FMC and then runs a simple time-lag differential equation model of the time evolution of the FMC. In the *learning phase*, the model assimilates [9] FMC observations from sensors on Remote Automated Weather Stations (RAWS) [6], using the augmented extended Kalman filter. In the *forecast phase*, the model runs from the atmospheric state provided by WRF without the Kalman filter, since the sensor data are still in future and not known (Fig. 1).

We seek to improve the accuracy of both the FMC model and of the data assimilation. The time-lag model represents the FMC in a wood stick by a single number, while more accurate models use multiple layers [8] or a continous radial profile [7]. Also, the Kalman filter assumes Gaussian probability distributions and a linear model, while more sophisticated data assimilation methods can represent more general distributions and allow nonlinear models. It is, however, unclear how much additional sophistication is worthwhile given the available data. Thus, we want to build a model together with data assimilation directly from data instead. We propose to use a Recurrent Neural Network (RNN) for this.

2 The FMC model with Kalman filter

We briefly describe the model from [4] with data assimilation from [9]. For simplicity, we consider here only the situation at a single RAWS location, without rain, and with a single fuel class with 10h time lag. See [4, 9] for details, references, and a more general case.

The FMC $m(t)$ in wood is the mass of water as % of the mass of dry wood, and it changes with time t and atmospheric conditions. A simple empirical model of the evolution of $m(t)$ in a wood stick in constant atmospheric conditions is the stick losing water if $m\,(t) > E_d$, the drying equilibrium, and gaining water if $m\,(t) < E_w$, the wetting equilbrium, with a characteristic time constant T given by the stick diameter ($T = 10$h for 10h fuel). The values of E_w and E_d, $E_w < E_d$, are computed from atmospheric conditions, namely relative humidity and temperature. We add

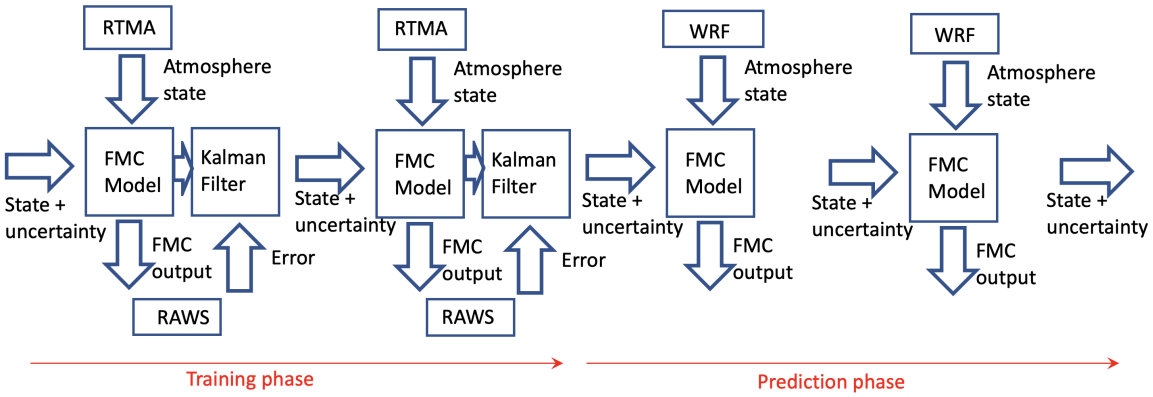


Figure 1: Data flow in the existing model with Kalman filter.

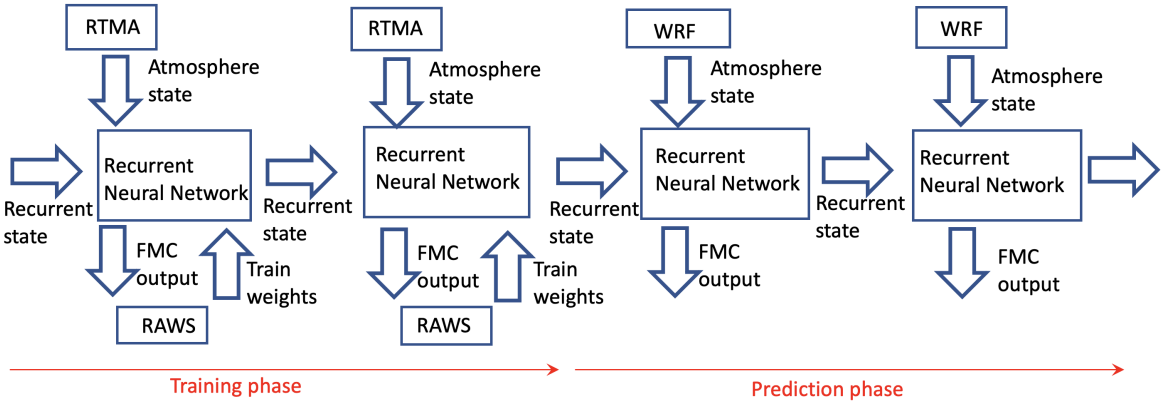


Figure 2: Data flow with the Recurrent Neural Network.

to both a correction ΔE, assumed constant in time and to be identified from data. This gives a system of differential equation on the interval $[t_k, t_{k+1}]$ for the augmented state $u = (m, \Delta E)$ of dimension 2,

$$\frac{dm}{dt} = \frac{E_w + \Delta E - m(t)}{T} \text{ if } m(t_k) < E_w + \Delta E, \quad \frac{dm}{dt} = \frac{E_d + \Delta E - m(t)}{T} \text{ if } m(t_k) < E_d + \Delta E,$$
$$\frac{dm}{dt} = 0 \text{ if } E_w + \Delta E \le m(t_k) \le E_d + \Delta E, \quad \frac{d\Delta E}{dt} = 0.$$

We apply the extended Kalman filter to the evolution $u(t_k) \mapsto u(t_{k+1})$ with the observations $m(t_k) = d(t_k)+$noise.

The FMC Model and Kalman filter blocks in the data flow in Fig. 1 implement a nonlinear operator. Since the operator is the same at all times t_k, it is applied recurrently. We seek to replace this operator by a Neural Network (NN) (Fig. 2), which then becomes a RNN.

3 Recurrent Neural Network

Filtering as an application of NNs is now classical. E.g., RNN was trained to match the Kalman filter [1] and synthetizing neural filters [3] estimate both an optimal filter and a model. For contemporary RNN basics, see, e.g., [2, Ch. 15]. Our goal is to build a RNN in the context of current high-performance high-level software, such as Keras, to translate a time series of the atmospheric data in the form of features E_d, E_w to a time series of FMC values m.

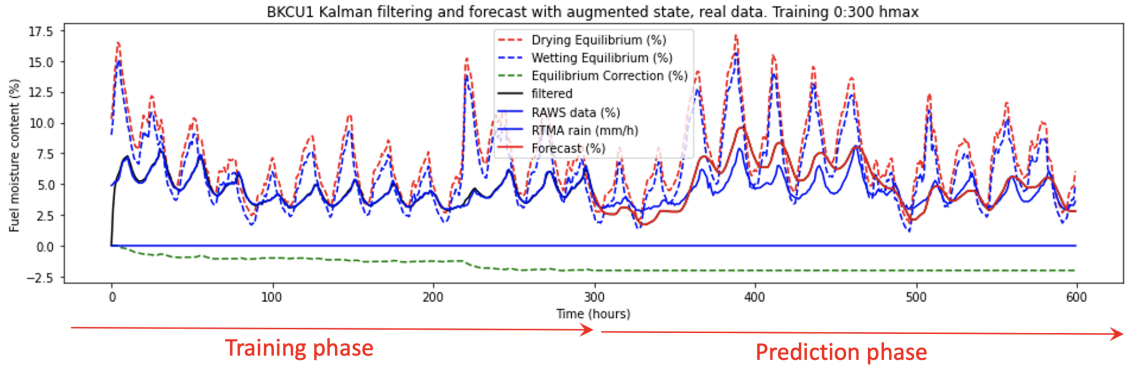


Figure 3: Learning and forecast with the time-lag model and Kalman filter. The equilibrium correction ΔE stabilizes in the training. Note a large prediction error from 300 to 600 hours.

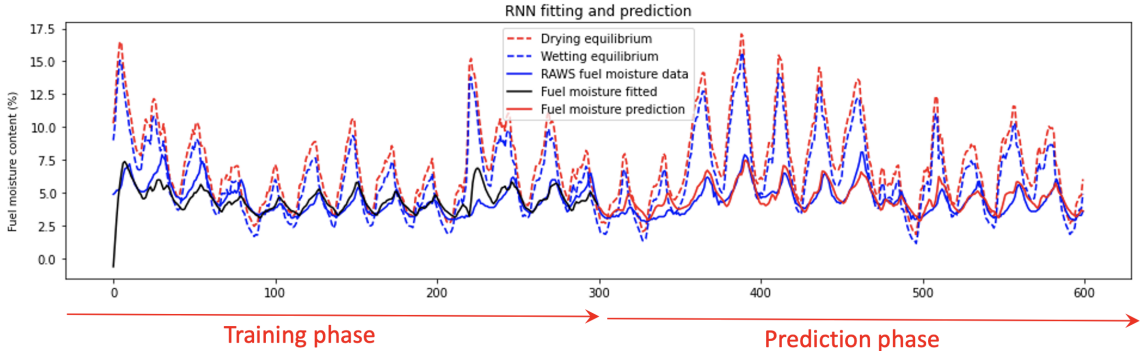


Figure 4: Training and prediction with the Recurrent Neural Network.

Training RNNs is known to be tricky. One reason is that computing the gradient of the loss function by back propagation uses the chain rule applied to the NN operator composed with itself many times, which results in "vanishing" or "exploding" gradients. To overcome this, we train a stateful RNN model [2, p. 532] and limit the number of times the NN operator is composed with itself to a small number of s timesteps. In each batch, the built-in stochastic gradient (SG) optimizer in Keras is presented with a sequence of training samples, each of the form of a short sequence of $(\text{input}_{k+1},\ldots,\text{input}_{k+s})$ and (target_{k+s}), $k = 1, 2, \ldots$. The inputs are the features (E_d, E_w), and the targets are the observations from the RAWS FMC sensors. The NN operator is applied to the recurrent state $(\text{hidden}_{k+1},\ldots,\text{hidden}_{k+s})$ and the input to produce the new recurrent state $(\text{hidden}_{k+2},\ldots,\text{hidden}_{k+s+1})$ and (output_{k+s}) which is compared with (target_{k+s}) to compute a contribution to the loss function and its gradient. After the RNN is trained, the optimized weights are copied to an identical stateless NN model [2, p. 534], which is then used for the evaluation of the NN operator in the prediction phase.

Though this procedure is commonly used, it did not work well in this application and the resulting forecast was much worse than when using the extended KF. However, it is straightforward to implement a version of the Euler method for the time-lag differential equation $dm/dt = (E - m)/T$ by a single neuron with linear activation and a suitable choice of weigths,

$$m_{k+1} = e^{-\Delta t/T} m_k + \left(1 - e^{-\Delta t/T}\right) E_k, \qquad \Delta t = t_{k+1} - t_k.$$

Here, m_k is the hidden state, also copied to the output, and E_k is the input. The single neuron RNN worked well on synthetic examples, so we used a hidden layer with linear activation, pre-trained by using the initial weights above.

Our final network had one hidden layer of 6 neurons pre-trained as above, and one input and one output neuron, also with linear activation. We chose the dimension of the hidden state 6 to accommodate different time scales. The training used $s = 5$ timesteps. The resulting prediction (Fig. 4) was better than from the differential equation model with extended KF (Fig. 3).

4 Conclusion

We have used batch training of a stateful RNN with linear activation and initial weights chosen to make the RNN an exact model in a special case. A hidden layer of several neurons initialized to the same weights then produced a better prediction than a differential equation model with extended KF. Exploiting this principle with more general activations, such as RELU, may enable switching between different behaviours, such as drying, wetting, or rain, or quantification of uncertainty in future.

Acknowledgement: This work was partially supported by NASA grants 80NSSC19K1091, 80NSSC22K1717, and 80NSSC22K1405.

References

[1] J.P. DeCruyenaere, H.M. Hafez: *A comparison between Kalman filters and recurrent neural networks*, in [Proceedings 1992] IJCNN International Joint Conference on Neural Networks, Vol. 4, IEEE, 1992, pp. 247–251.

[2] A. Géron: *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly, 2nd ed., 2019.

[3] J.T.-H. Lo: *Synthetic approach to optimal filtering.* IEEE Transactions on Neural Networks 5, 1994, pp. 803–811.

[4] J. Mandel, S. Amram, J. Beezley, G. Kelman, A.K. Kochanski, V.Y. Kondratenko, B.H. Lynn, B. Regev, M. Vejmelka: *Recent advances and applications of WRF-SFIRE.* Natural Hazards and Earth System Science 14, 2014, pp. 2829–2845.

[5] J. Mandel, M. Vejmelka, A.K. Kochanski, A. Farguell, J.D. Haley, D.V. Mallia, K. Hilburn: *An interactive data-driven HPC system for forecasting weather, wildland fire, and smoke*, in 2019 IEEE/ACM HPC for Urgent Decision Making (UrgentHPC), Supercomputing 2019, Denver, CO, USA, IEEE, 2019, pp. 35–44.

[6] National Wildfire Coordinating Group, *NWCG Standards for Fire Weather Stations, PMS 426-3*, March 2019. `https://www.nwcg.gov/publications/426-3`, retrieved December 2022.

[7] R.M. Nelson Jr.: *Prediction of diurnal change in 10-h fuel stick moisture content.* Canadian Journal of Forest Research 30, 2000, pp. 1071–1087.

[8] D.W. Van der Kamp, R.D. Moore, I.G. McKendry: *A model for simulating the moisture content of standardized fuel sticks of various sizes.* Agricultural and Forest Meteorology 236, 2017, pp. 123–134.

[9] M. Vejmelka, A.K. Kochanski, J. Mandel: *Data assimilation of dead fuel moisture observations from remote automatic weather stations.* International Journal of Wildland Fire 25, 2016, pp. 558–568.

Bohl-Marek decomposition applied to a class of biochemical networks with conservation properties

Š. Papáček[1], *C. Matonoha*[2], *J. Duintjer Tebbens*[2,3]

[1] Institute of Information Theory and Automation of the CAS, Prague
[2] Institute of Computer Science of the CAS, Prague
[3] Faculty of Pharmacy, Charles University, Hradec Králové

1 Introduction

This study presents an application of one special technique, further called as Bohl-Marek decomposition, related to the mathematical modeling of biochemical networks with mass conservation properties. We continue in direction of papers devoted to inverse problems of parameter estimation for mathematical models describing the drug-induced enzyme production networks [3]. However, being aware of the complexity of general physiologically based pharmacokinetic (PBPK) models, here we focus on the case of enzyme-catalyzed reactions with a substrate transport chain [5]. Although our ultimate goal is to develop a reliable method for fitting the model parameters to given experimental data, here we study certain numerical issues within the framework of optimal experimental design [6]. Before starting an experiment on a real biochemical network, we formulate an optimization problem aiming to maximize the information content of the corresponding experiment. For the above-sketched optimization problem, the computational costs related to the *two formulations of the same biochemical network*, being (i) the classical formulation $\dot{x}(t) = Ax(t) + b(t)$ and (ii) the 'quasi-linear' Bohl-Marek formulation $\dot{x}_M(t) = M(x(t))\, x_M(t)$, can be determined and compared.

2 Problem formulation

The system of differential equations describing the processes under study is described in Tab. 1. It can be systematically derived using the so-called stoichiometric matrix $S \in \mathbb{R}^{n\times q}$, where q is the number of reactions (including the transport of species).

Table 1: Description of the transport and reaction processes defining the network.

Description of the related process	**Chem. notation**	**Param.**
R_0: Substrate X_{ext} dosing (model input)	$\emptyset \to X_{ext}$	$u(t)$
R_1: Substrate transport between compartments	$X_{ext} \rightleftharpoons X_{int}$	k_0
R_2: Enzyme E binds to substrate,	$X_{int} + E \rightleftharpoons C$	k_1
formation of a complex C		
R_3: Reverse reaction to R_2		k_{-1}
R_4: Complex breaks down into E plus	$C \to E + P$	k_2
a product P – altered substrate molecule		

The vector of changes in species concentrations $x \in \mathbb{R}^n$ is then described as a linear transformation of the reaction rate vector $\nu \in \mathbb{R}^q$:

$$\dot{x}(t) = S\, \nu(x, p), \tag{1}$$

where

$$S = \begin{array}{c} \begin{matrix} R_1 & R_2 & R_3 & R_4 \end{matrix} \\ \begin{pmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 0 & -1 & 1 & 1 \\ 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{array}, \qquad \nu = \begin{pmatrix} k_0\ (x_1 - x_2) \\ k_1\ x_2\ x_3 \\ k_{-1}\ x_4 \\ k_2\ x_4 \end{pmatrix}, \qquad p = \begin{pmatrix} k_0 \\ k_1 \\ k_{-1} \\ k_2 \end{pmatrix}. \tag{2}$$

Reaction networks frequently possess subsets of reactants that remain constant at all times, i.e., they are referred to conserved species. Generally, there exists a conservation matrix Γ (with dimension $h \times n$), where the rows represent the linear combination of species (reactants), which are constant in time. It can be solved explicitly for large systems ($0 = \Gamma\ S$). For our case of S in form (2), the conservation property reads

$$x_3 + x_4 = e_0, \quad x_1 + x_2 + x_4 + x_5 = u_0. \tag{3}$$

Consequently, here

$$\Gamma = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \end{pmatrix}. \tag{4}$$

The existence of two relations (3) signifies not only the possibility to reduce the number of state variables, but also induces the reformulation of the governing equations for species concentration using negative M-matrices, see (9). For instance, using (2), we get the resulting ODE system in the usual form

$$\dot{x}(t) = Ax(t) + b(x(t)), \tag{5}$$

with the constant matrix (the linear part of the system)

$$A = \begin{pmatrix} -k_0 & k_0 & 0 & 0 & 0 \\ k_0 & -k_0 & 0 & k_{-1} & 0 \\ 0 & 0 & 0 & k_{-1} + k_2 & 0 \\ 0 & 0 & 0 & -(k_{-1} + k_2) & 0 \\ 0 & 0 & 0 & k_2 & 0 \end{pmatrix} \tag{6}$$

and the vector representing nonlinear, e.g. bilinear, parts

$$b(x(t)) = \begin{pmatrix} u(t) \\ -k_1 \cdot x_2(t) \cdot x_3(t) \\ -k_1 \cdot x_2(t) \cdot x_3(t) \\ k_1 \cdot x_2(t) \cdot x_3(t) \\ 0 \end{pmatrix}. \tag{7}$$

The initial conditions are

$$x(0) = \begin{pmatrix} u(t_0) & 0 & e_0 & 0 & 0 \end{pmatrix}^T. \tag{8}$$

The ODE system (5) is nonlinear because of the bilinear terms and time-varying dosing function $u(t)$. Nevertheless, thanks to the conservation properties (3), there exists an alternative, a quasi-linear approach representing (in some sense) linearization of originally non-linear system (5) with the block diagonal system matrix of a special form (negative M-matrix). However, the system matrix dimension (order) has to be bigger because of the repeated presence of some state

variables (as it is shown in the next section). To the best of our knowledge, this approach was proposed by Bohl and Marek [1, 2] and further extended into the control theory framework by Marek [4].

Theorem (Bohl-Marek decomposition): When the conservation equations of a system of ODEs contain all variables, then the system can be decomposed into coupled, quasi-linear subproblems.

Sketch of the proof: Knowing that all state variables are involved in the conservation properties, the rate of change of the sum of certain variables (in the left hand side of a corresponding ODE) must be zero. Consequently the corresponding part of column sums also must be zero. Finally, the ODE can be reassembled in blocks with desired special structure of M-matrices.

Here in our case study, the state variables are listed in two subsets $\{x_3, x_4\}$ and $\{x_1, x_2, x_4, x_5\}$, and thus the non-linear ODEs (5) can be represented as a linear system with the system matrix of a special form, a negative M-matrix. Let these two subsets of state variables be assembled and merged together as follows

$$\tilde{x}(t) = \begin{pmatrix} x^1(t) \\ x^2(t) \end{pmatrix}, \quad x^1(t) = \begin{pmatrix} x_3(t) \\ x_4(t) \end{pmatrix}, \quad x^2(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_4(t) \\ x_5(t) \end{pmatrix}.$$

Then the ODE system for modified state variable vector $\tilde{x}(t)$ is

$$\frac{\mathrm{d}\tilde{x}(t)}{\mathrm{d}t} = M\tilde{x}(t), \tag{9}$$

with the block diagonal system matrix M of a special form

$$M = \begin{pmatrix} -k_1 \cdot x_2 & k_{-1} + k_2 & 0 & 0 & 0 & 0 \\ k_1 \cdot x_2 & -(k_{-1} + k_2) & 0 & 0 & 0 & 0 \\ 0 & 0 & -k_0 & k_0 & 0 & 0 \\ 0 & 0 & k_0 & -k_0 - k_1 \cdot x_3 & k_{-1} & 0 \\ 0 & 0 & 0 & k_1 \cdot x_3 & -(k_{-1} + k_2) & 0 \\ 0 & 0 & 0 & 0 & k_2 & 0 \end{pmatrix}. \tag{10}$$

The initial conditions are

$$x(0) = \begin{pmatrix} e_0 & 0 & u(t_0) & 0 & 0 & 0 \end{pmatrix}^T.$$

3 Parameter estimation and experimental design

The quality of parameter estimation is usually measured by the squared error functional

$$J = \int_{t_0}^{t_f} \left(z_m(t) - z(p; u(t); t)\right)^2 \mathrm{d}t, \tag{11}$$

where $z(p; u(t); t) \in \mathbb{R}^{n_{out}}$ is the output vector, $z_m(t) \in \mathbb{R}^{n_{out}}$ are (continuous) measured data, $p \in \mathbb{R}^q$ is a parameter vector, e.g., $p = (k_0, k_1, k_{-1}, k_2)^T$, and $u(t)$ is the control input.

Here, in order to maximize the information content of the corresponding experiment, we formulate the optimal control problem, e.g., we look for an optimal impuls input $u(t_i)$

$$\max_{\text{admissible } u(t)} \|\mathcal{F}(p_0)\|. \tag{12}$$

If the quantity $\|\mathcal{F}(p_0)\|$, being evaluated at p_0, is the determinant of the Fisher information matrix, i.e., $\|\mathcal{F}(p_0)\| \equiv \det(\mathcal{F}(p_0))$, we speak about a D-criterion. Note that the key role in evaluation of $\mathcal{F}$ plays the sensitivity matrix $\chi = \frac{\partial z(p_0;u(t);t)}{\partial p} \in \mathbb{R}^{n_{out} \times q}$ because $\mathcal{F} = \chi^T \chi \in \mathbb{R}^{q \times q}$.

4 Conclusion

As a proof of concept, we took the case of enzyme-catalyzed reactions with a substrate transport chain, see [5] for parameter values. For two above introduced model formulations, i.e. the classical formulation (5) and the 'quasi-linear' Bohl-Marek formulation (9), and based on the different impuls controls $u(t_i)$ – the same dosis of substrate in different time instants t_i, one can calculate (numerically) parameter sensitivities, i.e. the partial derivatives of the output vector $z(p; u(t); t)$ with respect to individual model parameters. Afterwards, comparing $\|\mathcal{F}(p_0)\|$, the optimal control input maximizing the information content can be selected. Eventually, the computational costs related to both formulations (5) and (9) can be compared as well.

Acknowledgement: The work of Š. Papáček was supported by the Czech Science Foundation through the research grant No. 21-03689S. The work of Ctirad Matonoha was supported by the long-term strategic development financing of the Institute of Computer Science (RVO:67985807).

References

[1] E. Bohl, I. Marek: *Existence and Uniqueness Results for Nonlinear Cooperative Systems.* In: I. Gohberg, H. Langer (eds): Linear Operators and Matrices. Operator Theory: Advances and Applications, Vol. 130. Birkhäuser, Basel, 2002.

[2] E. Bohl, I. Marek: *Input-output systems in biology and chemistry and a class of mathematical models describing them.* Appl. Math. 50, 2005, pp. 219–245.

[3] J. Duintjer Tebbens, C. Matonoha, A. Matthios, Š. Papáček: *On parameter estimation in an* in vitro *compartmental model for drug-induced enzyme production in pharmacotherapy.* Applications of Mathematics, 64, 2019, pp. 253–277.

[4] I. Marek: *On a Class of Stochastic Models of Cell Biology: Periodicity and Controllability.* In: R. Bru, S. Romero-Vivó (eds.): Positive Systems. Lecture Notes in Control and Information Sciences, Vol. 389, Springer, Berlin, Heidelberg, 2009.

[5] C. Matonoha, Š. Papáček, V. Lynnyk: *On an optimal setting of constant delays for the D-QSSA model reduction method applied to a class of chemical reaction networks.* Applications of Mathematics, 67, 2022, pp. 831–857.

[6] A. Strouwen, B.M. Nicolaï, P. Goos: *Optimizing oxygen input profiles for efficient estimation of Michaelis-Menten respiration models.* Food and Bioprocess Technology 12, (5), 2019, pp. 769–780.

Matrix decay phenomenon and its applications

S. Pozza

Charles University, Prague

1 Introduction

In matrix computation, it is common to divide matrices into dense and sparse categories. Even though such categories are not precisely defined, we can think of a sparse matrix as one whose number of zero elements is large enough to be conveniently exploitable and a dense one as a matrix that is not sparse. It is important to note that the notion of sparsity does not consider the magnitude of the nonzero elements. This can be an issue since, in many applications, one has to deal with dense matrices in which most elements are so close to zero to being negligible. These matrices are close to being sparse in the sense that they are sparse upon truncation. Moreover, the nonnegligible elements are usually localized in some part of the matrix, and the magnitude of the other elements tends to decay to zero as we move away from them. Localization can be exploited for linear system solvers, precondition construction, eigenvalue problems, matrix function approximation, and many other numerical linear algebra problem; see, for instance, [1] and references therein.

As an example, consider the tridiagonal (diagonally dominant) matrix A in Figure 1. Its inverse is dense; however, the magnitude of its elements quickly decays to zero as we move away from the diagonal, as visible in the right-hand plot of Figure 1. Therefore, the inverse can be considered banded upon truncation; [4]

$$A = \begin{bmatrix} 3 & 1 & 0 & \cdots & 0 \\ 1 & 3 & 1 & & \vdots \\ 0 & 1 & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & 3 \end{bmatrix}$$

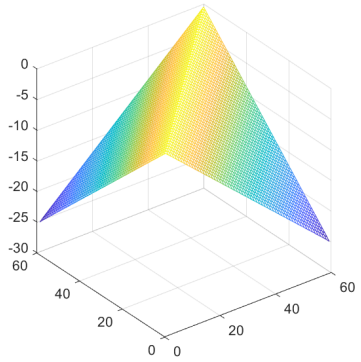


Figure 1: Left: tridiagonal 60×60 matrix A. Right: magnitude of A^{-1} elements in logarithmic scale.

We first present the decay phenomenon for function of banded and sparse matrices (Sections 2 and 3). Then we discuss its role in Krylov subspace methods (Section 4). Finally, we show the connection between the decay elements in the inverse of banded matrices and a new numerical method for the solution of linear ODEs (Section 5).

2 Functions of banded matrices

For the sake of simplicity, assume that A is a matrix with upper and lower bandwidth b. Then A^k has (upper and lower) bandwidth kb, for $k = 0, 1, \ldots$. Now, consider a matrix function defined by the series $f(A) = \sum_{j=0}^{\infty} \alpha_k A^k$, with $\alpha_k \in \mathbb{C}$ (for instance, $\alpha_k = 1/k!$ defines the matrix exponential $\exp(A)$ and $\alpha_k = \alpha^k$, with $|\alpha|$ small enough, the resolvent $(I - \alpha A)^{-1}$); see, e.g., [6]. Under certain conditions on α_k and on A, the elements in $\alpha_k A^k$ converge to zero. This, together with the fact that A^k has bandwidth kb, explains the observed decay phenomenon. Using polynomial approximation, it is possible to produce a-priori bounds for the magnitude of the off-diagonal elements of $f(A)$, which usually take the form

$$|f(A)_{i,j}| \leq K\rho^{|i-j|}, \tag{1}$$

with $K > 0$ and $0 < \rho < 1$ determined by f and A. Such bounds usually depend on i) the (upper and lower) bandwidth of A, ii) the properties of the function f, iii) the spectral properties of A (e.g., spectral sets such as the spectral interval, the field of values, and the pseudospectrum). For more information, see, e.g., [1, 2, 8].

3 Functions of sparse matrices

The analysis of the decay phenomenon for banded matrices can be extended to sparse matrices by using graph theory (e.g., [1, 2, 3]). Let $G = (V, E)$ be the graph induced by the sparsity pattern of A, i.e., the graph with nodes $V = \{1, \ldots, N\}$, where N is the size of A, and edges $(i, j) \in E$ if and only if $A_{ij} \neq 0$ (assuming A elements are from $\{0, 1\}$, A is the adjacency matrix of G). The length of the shortest walk from a node i to a node j is called the (geodesic or shortest-path) distance in G from i to j and is denoted by $\mathsf{d}_G(i, j)$. The following property holds

$$(A^k)_{ij} = 0, \text{ for every } k < \mathsf{d}_G(i, j). \tag{2}$$

Analogously to the banded case in Section 2, it is possible to use (2) in combination with polynomial approximation to devise a-priori decay bounds for $f(A)$. In this case, the bounds take the form

$$|f(A)_{i,j}| \leq K\rho^{\mathsf{d}_G(i,j)},$$

with $K > 0$ and $0 < \rho < 1$ determined by f and A. Note that the distance from the diagonal of formula (1) is replaced here by the geodesic distance; see, e.g., [1, 3].

Vice versa, it is also possible to exploit the decay phenomenon to analyze the stability of network centrality measures upon edge perturbation as done in [9]. Indeed, popular and effective measures of the importance of a node or a set of nodes in a graph are defined in terms of suitable entries of functions of matrices $f(A)$, with A the adjacency matrix of the graph $G = (V, E)$. Let us add, remove or simply modify the edges in the set δE, obtaining $\widetilde{G} = (V, \widetilde{E})$, with $\widetilde{E} \subset E \cup \delta E$ and with adjacency matrix $\widetilde{A} = A + \delta A$. In [9], bounds for the quantity $|f(A)_{k,\ell} - f(A + \delta A)_{k,\ell}|$ are provided which enlightens the dependency on the distance that separates either k or ℓ from the nodes touched by the edges in δE.

4 Decay phenomenon in Krylov subspace methods

Given a matrix $A \in \mathbb{R}^{N\times N}$ and a vector $\boldsymbol{v} \neq 0$, Arnoldi's method produces the orthogonal matrix $U_m = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$, basis of the (polynomial) Krylov subspace

$$\mathcal{P}_m(A, \boldsymbol{v}) := \operatorname{span}\{\boldsymbol{v}, A\,\boldsymbol{v}, \ldots, A^{m-1}\,\boldsymbol{v}\}.$$

Starting with $\boldsymbol{u}_1 = \boldsymbol{v}/\|\boldsymbol{v}\|$, Arnoldi's method is a Gram-Schmidt orthogonalization process defined by the recurrences

$$t_{j+1,j}\boldsymbol{u}_{j+1} = A\boldsymbol{u}_j - \sum_{i=1}^{j} t_{i,j}\boldsymbol{u}_i, \quad t_{i,j} = \boldsymbol{u}_i^* A\boldsymbol{u}_j, \quad t_{j+1,j} = \|\boldsymbol{u}_{j+1}\|, \quad j = 1, \ldots, m.$$

The recurrences can be rewritten in the matrix form $AU_m = U_m T_m + t_{m+1,m}\boldsymbol{u}_{m+1}\boldsymbol{e}_m^T$, with T_m the $m \times m$ upper Hessenberg matrix with entries $t_{i,j}$ ($\boldsymbol{e}_m$ the mth vector of the canonical basis). Note that by orthogonality, we get $T_m = U_m^* A\, U_m$. The matrix T_m plays two roles in the algorithm: i) it represents the orthogonalization process (coefficients $t_{i,j}$), ii) it represents the action of A in the Krylov subspace $\mathcal{P}_m(A, \boldsymbol{v})$, i.e., $U_m T_m U_m^* = U_m U_m^* A\, U_m U_m^*$. The matrix T_m can be used for matrix-function approximation in the formula

$$f(A)\boldsymbol{v} \approx U_m f(T_m)\boldsymbol{e}_1;$$

see, e.g., [6]. Since T_m is banded in its lower part, it is possible to derive a-priori decay bound for $f(T_m)$. Such decay bounds can be used, e.g., for devising relaxed approaches (inexact Arnoldi) and producing stopping criteria for iterative solvers in matrix function evaluations and matrix equation problems; see, e.g., [5, 8].

4.1 Rational Krylov subspace method

Setting $\boldsymbol{\sigma} = [\sigma_1, \ldots, \sigma_{m-1}]$ with $\sigma_j \notin \lambda(A)$, the rational Krylov subspace is defined as

$$\mathcal{K}_m(A, \boldsymbol{v}, \boldsymbol{\sigma}) := \operatorname{span}\left\{\boldsymbol{v}, (A - \sigma_1 I)^{-1}\,\boldsymbol{v}, \ldots, \prod_{j=1}^{m-1}(A - \sigma_j I)^{-1}\,\boldsymbol{v}\right\}.$$

The rational Krylov subspace method (RKSM) produces the orthogonal matrix $V_m = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m]$ basis of $\mathcal{K}_m(A, \boldsymbol{v}, \boldsymbol{\sigma})$. RKSM is based on the Gram-Schmidt orthogonalization recurrences:

$$h_{j+1,j}\boldsymbol{v}_{j+1} = (A - \sigma_j I)^{-1}\boldsymbol{v}_j - \sum_{i=1}^{j} h_{i,j}\boldsymbol{v}_i, \quad h_{i,j} = \boldsymbol{v}_i^*(A - \sigma_j I)^{-1}\boldsymbol{v}_j, \quad h_{j+1,j} = \|\boldsymbol{v}_{j+1}\|,$$

for $j = 1, \ldots, m$, which can be rewritten in the matrix form

$$A\, V_m H_m = V_m K_m - h_{m+1,m}(A - \sigma_m I)\boldsymbol{v}_{m+1}\boldsymbol{e}_m^T,$$

with H_m the Hessenberg matrix with entries $h_{i,j}$, and $K_m = (I + H_m \operatorname{diag}(\sigma_1, \ldots, \sigma_m))$. The reduced-order matrix is defined as

$$J_m := V_m^* A V_m = K_m\, H_m^{-1} - h_{m+1,m} V_m^*(A - \sigma_m I)\,\boldsymbol{v}_{m+1}\boldsymbol{e}_m^T\, H_m^{-1},$$

which is the projection of A onto $\mathcal{K}_m(A, \boldsymbol{v}, \boldsymbol{\sigma})$. The matrix function $f(J_m)$ can be used in matrix function and in matrix equation approximations (see references in [7]). Despite the fact that J_m is, generally, not banded, it is still possible to derive a-priori decay bounds for $f(J_m)$ exploiting i) the hidden sparsity structure of J_m, that is a consequence of the orthogonalization process, ii) rational function approximation, iii) the domain of analyticity of f, iv) the field of values of A. See [7] for more information.

5 Decay phenomenon and ODEs

In the new method for the solution of a linear ODE presented in [10], the ODE solution $y(t) \in \mathbb{C}$, with t in a bounded interval I, is given in terms of the coefficients of a truncated Legendre polynomial expansion, i.e., $y(t) \approx \sum_{j=0}^{M-1} u_j p_j(t)$, with $p_j(t)$ Legendre polynomials. In particular, the vector of the coefficients $\boldsymbol{u} = [u_j]_{j=0}^{M-1}$ is defined by the expression

$$\boldsymbol{u} = H(I - F)^{-1}\boldsymbol{e}_1,$$

with H, F banded matrices, and $\boldsymbol{e}_1$ the first canonical vector. Since $y(t)$ is a smooth function, we expect $|u_j|$ to exponentially decay as j increases (for M large enough). This is reflected by the fact that $(I - F)^{-1}\boldsymbol{e}_1$ is the first column of the resolvent of F whose elements are also expected to decay in magnitude under certain conditions on F.

Acknowledgement: This work was supported by Charles University research programs UNCE/SCI/023 and PRIMUS/21/SCI/009.

References

[1] M. Benzi: *Localization in Matrix Computations: Theory and Applications.* In: M. Benzi, V. Simoncini (eds.): Exploiting Hidden Structure in Matrix Computations: Algorithms and Applications, Lecture Notes in Mathematics, Vol. 2173, 2016, pp. 211–317.

[2] M. Benzi, P. Boito: *Decay properties for functions of matrices over C-algebras.* Linear Algebra Appl. 456, 2014, pp. 174–198.

[3] M. Benzi, P. Boito, N. Razouk: *Decay properties of spectral projectors with applications to electronic structure.* SIAM Rev. 55, 2013, pp. 3–64.

[4] S. Demko, W.S. Moss, P.W. Smith: *Decay rates for inverses of band matrices.* Math. Comp. 43, 1984, pp. 491–499.

[5] S. Güttel, M. Schweitzer: *A comparison of limited-memory Krylov methods for Stieltjes functions of Hermitian matrices.* SIAM J. Matrix Anal. Appl. 42, 2021, pp. 83–107.

[6] N.J. Higham: *Functions of Matrices.* SIAM, 2008.

[7] S. Pozza, V. Simoncini: *Functions of rational Krylov space matrices and their decay properties.* Num. Math. 148, 2021, pp. 99–126.

[8] S. Pozza, V. Simoncini: *Inexact Arnoldi residual estimates and decay properties for functions of non-Hermitian matrices.* BIT Num. Math. 59(4), 2019, pp. 969–986.

[9] S. Pozza, F. Tudisco: *On the Stability of Network Indices Defined by Means of Matrix Functions.* SIAM J. Matrix Anal. Appl. 39(4), 2013, pp. 1521–1546.

[10] S. Pozza, N. Van Buggenhout: *The *-product approach for linear ODEs: a numerical study of the scalar case.* In: Programs and Algorithms of Numerical Mathematics Proceedings of Seminar 21, Jablonec nad Nisou, 2022, to appear.

Numerical approximation of the spectrum of self-adjoint operators, operator preconditioning and an unfinished talk with Radim Blaheta

Z. Strakoš

Charles University, Prague

Many phenomena are mathematically expressed in terms of eigenvalues and eigenvectors of matrices and operators. Besides standard physical and engineering examples of waves and vibrations, they are essential also, e.g., in the mathematical foundations of quantum mechanics, which pioneered its use in spectral representations of Hermitian/self-adjoint operators.

Consider a real symmetric n by n matrix G. It can be considered as a mapping that takes a vector u in the n-dimensional Euclidean space and maps it to a vector Gu in the same space. Basic linear algebra results state that there are exactly n vectors in this space that remain essentially unchanged when mapped by G, except for multiplication by a real number, i.e., $Gu_i = \lambda_i u_i, i = 1, 2, \ldots, n$. Moreover, these vectors are orthogonal, their normalized versions form an orthonormal basis and the matrix can be written as $G = \sum_{i=1}^{n} \lambda_i u_i u_i^T$, which is called the spectral decomposition of G. This result can be generalized to operators defined on an infinite dimensional real Hilbert space V. Indeed, any self-adjoint operator $\mathcal{G} : V \to V$ can be expressed in terms of the Riemann-Stieltjes integral as

$$\mathcal{G} = \int \lambda \, dE(\lambda), \quad \text{i.e.,} \quad (\mathcal{G}\psi, \phi) = \int \lambda \, d(E(\lambda)\psi, \phi) \text{ for all } \psi, \phi \in V,$$

where the spectral function $E(\lambda)$ of $\mathcal{G}$ represents a family of orthogonal projections (projection-valued measure), analogous to $\{u_i u_i^T, i = 1, \ldots, n\}$ for symmetric matrices.d

When such an infinite dimensional operator is discretized, as, e.g., when solving boundary value problems for partial differential equations, we should be concerned with the interplay between the eigenvalues of the matrices arising from discretizations and *the whole spectrum* of the associated infinite-dimensional operator. This issue is not only of theoretical interest, but it is also important for efficient numerical computations. Such consideration must naturally include the continuous part of the spectrum of $\mathcal{G}$. This contribution presents some recent results in this direction.

The presented results were obtained jointly with Tomáš Gergelits, Kent-André Mardal and, in particular, Bjørn Fredrik Nielsen. But they are much in line with many discussions that we had together with Radim Blaheta over several decades and that were for me always useful and very pleasant. Due to involvement in many other projects and duties we have not transformed them into a real joint project that would end up in a joint paper. Our intention to change this will remain unfinished. But our interaction have definitely been for me very rewarding, professionally and even more personally.

Improving computational efficiency of contact solution in fully resolved CFD-DEM simulations with arbitrarily-shaped solids

O. Studeník, M. Kotouč Šourek, M. Isoz

University of Chemistry and Technology, Prague
Institute of Thermomechanics of the CAS, Prague

1 Introduction

The primary motivation for this work is the increasing demand for an efficient and accurate description of systems with a solid phase dispersed in a fluid. These processes are vastly spread in industry and nature, and the most common practice is describing them with experimental data or empirical correlations [1]. At smaller scales, their direct simulation is possible via computational fluid dynamics (CFD) coupled with the discrete element method (DEM). Still, the vast majority of present-time CFD-DEM solvers neglect the particles shapes and approximate them as spheres [2]. This approach is beneficial with respect to computational efficiency. However, the realistic particle shape has to be considered in numerous applications, see, e.g., deposition of an active catalytic layer into a monolith structure [3]. The true particle shape might be approximated utilizing a cluster of spheres or a single particle with a complex surface defined by a triangulation of the real particle surface. In the present contribution, we concentrate on the latter and apply the *soft*DEM approach with the overlap computation stemming from [4], which is based on the so-called overlap volume. The overlap volume computation for a contact between two arbitrarily-shaped solids is costly. Previously, we proposed an efficient algorithm for the overlap volume computation in CFD-DEM solvers called *virtual mesh* [5]. Presently, we further evaluate the new algorithm accuracy and focus on its parallelization. In the following, we briefly outline the fundamental principles of our in-house CFD-DEM solver [6] and hint at the currently tested approach to parallelization of both its CFD and DEM parts. The discussion of principles is complemented by a few illustrative results.

2 Methods

Overall, our goal is to couple the Eulerian description of a fluid flow with particle transport model implemented within the Lagrangian framework. In particular, let us have a computational domain $\Omega \subset \mathbb{R}^3$ divided into solid part $\Omega_\mathrm{s}(t)$ and fluid part $\Omega_\mathrm{f}(t)$. In Ω, we study flow and transport of solids. The flow is solved as Eulerian in the whole Ω. Considering an incompressible Newtonian fluid, the flow is governed by the standard Navier-Stokes equations

$$\left.\begin{aligned}\mathcal{M}(\boldsymbol{u}) &= -\nabla\tilde{p} + \boldsymbol{f}_\mathrm{ib}\\ \nabla\cdot\boldsymbol{u} &= 0\end{aligned}\right\}, \quad \mathcal{M}(\boldsymbol{u}) = \frac{\partial\boldsymbol{u}}{\partial t} + \nabla\cdot(\boldsymbol{u}\otimes\boldsymbol{u}) - \nabla\cdot(\nu\nabla\boldsymbol{u}) \tag{1}$$

where $\boldsymbol{u}$ is the fluid velocity, ν kinematic viscosity, and $\tilde{p}$ kinematic pressure. The forcing term $\boldsymbol{f}_\mathrm{ib}$ is constructed in a way that it generates a fictitious representation of Ω_s inside Ω. The specific used immersed boundary method is described in detail in [6].

The main focus of the present contribution lies in the DEM (Lagrangian) part of the presented CFD-DEM solver. First, Ω_s is split into individual solid bodies $\Omega_\mathrm{s} = \bigcup_{i=1}^{N_\mathcal{B}} \mathcal{B}_i$ and the movement

of $\mathcal{B}_i$ is governed by

$$m_i \frac{\mathrm{d}^2 \boldsymbol{x}_i}{\mathrm{d}t^2} = \boldsymbol{f}_\mathrm{g} + \boldsymbol{f}_\mathrm{d} + \boldsymbol{f}_\mathrm{c}\,, \quad I_i \frac{\mathrm{d}\boldsymbol{\omega}_i}{\mathrm{d}t} = \boldsymbol{t}_\mathrm{g} + \boldsymbol{t}_\mathrm{d} + \boldsymbol{t}_\mathrm{c}\,, \tag{2}$$

where m_i is the mass of $\mathcal{B}_i$ and $\boldsymbol{x}_i(t)$ its centroid position at time t. Next, $\boldsymbol{\omega}_i$ is the body angular velocity and I_i is the matrix of its inertial moments. In this work, we consider $\mathcal{B}_i$ to be affected by gravity (g), drag (d), and contact (c) with other bodies or solid parts of Ω boundary. Thus, $\boldsymbol{f}$ and $\boldsymbol{t}$ in (2) represent the forces and torques acting on $\mathcal{B}_i$, respectively.

The computational efficiency of any CFD-DEM solver is strongly affected by the efficiency of contact solution, i.e., by the efficiency of $\boldsymbol{f}_\mathrm{c}$ and $\boldsymbol{t}_\mathrm{c}$ evaluation. In the present work, we consider systems with arbitrarily-shaped solids defined by a triangulation of their surface $\partial\mathcal{B}_i$, $i = 1, \ldots, N_\mathcal{B}$. Now, let us consider contact between two elasto-plastic bodies $\mathcal{B}_i$ and $\mathcal{B}_j$. Provided the bodies are of arbitrary shape, the normal contact force $\boldsymbol{f}_\mathrm{c}^n$ acting on the bodies is computed as [6],

$$\boldsymbol{f}_\mathrm{c}^n = \left(\frac{Y^* V_{ij}^\mathrm{o}}{\ell_\mathrm{c}} + \gamma^* \sqrt{\frac{Y^* M^*}{\ell_\mathrm{c}^3}} \frac{\mathrm{d}V_{ij}^\mathrm{o}}{\mathrm{d}t} \right) \boldsymbol{n}\,, \quad \ell_\mathrm{c} = 4\frac{r_i r_j}{r_i + r_j}\,, \quad V_{ij}^\mathrm{o} = \int_{\mathcal{B}_i \cap \mathcal{B}_j} \lambda \, dV\,, \tag{3}$$

where Y stands for Young's modulus, γ represents the damping coefficient, M represents the mass of the colliding pair, $\boldsymbol{n}$ is the contact normal, ℓ_c represents characteristic length of contact, with r being the distance between particle's centroid and the center of contact. By $\mathcal{B}_i \cap \mathcal{B}_j$ we mark *yet undefined computational cells* shared between the bodies $\mathcal{B}_i$ and $\mathcal{B}_j$. Finally, the material properties marked with $*$ denote the harmonic average of the material properties of individual solids. For a detailed description of the model (3) and its physical motivation, see [4].

Numerical solution of the governing equation The complete CFD-DEM solver is implemented in the C++ library OpenFOAM [7]. The Eulerian part (1) of the coupled system (1) and (2) is discretized via the finite volume method (FVM), providing a computational mesh $\Omega^h = \Omega_\mathrm{s}^h \cup \Omega_\mathrm{f}^h$. Hence, each body $\mathcal{B}_i$ now has its FV-discretized counterpart $\mathcal{B}_i^h$ spanning a finite number of cells, which can be used to evaluate (3). However, utilizing Ω^h for the DEM-related computations is inefficient as (i) DEM requires a significantly finer mesh than CFD, and (ii) FV mesh usable for CFD computations needs to carry a number of variables that are not required for DEM and as such, it is an unnecessarily complex object.

To overcome the issues with computational efficiency of contact solution between arbitrarily-shaped solids, we proposed a *virtual mesh*, which is local to each potential contact between solids $\mathcal{B}_i$ and $\mathcal{B}_j$ and used to (i) identify the contact and to, (ii) evaluate key parameters $(V_{ij}^\mathrm{o}, \boldsymbol{n})$ in relation (3) in an efficient manner, for details see [5]. However, the selected approach poses an interesting challenge to code parallelization. The first level of parallelization (L_1) concerns mainly the flow and the selected approach is inherited from OpenFOAM and based on the domain decomposition. Still, the contact evaluation, which is based on the *virtual mesh*, is (almost) independent on the decomposed Ω^h. Thus, it is profitable to construct a second parallelization level (L_2), where not the FV cells, but individual contacts are distributed between the computational resources.

3 Results

To evaluate the *virtual mesh* algorithm accuracy and efficiency, let us concentrate on a simple elasto-plastic collision between perfect spheres. Two tests were designed to (i) study the evolution

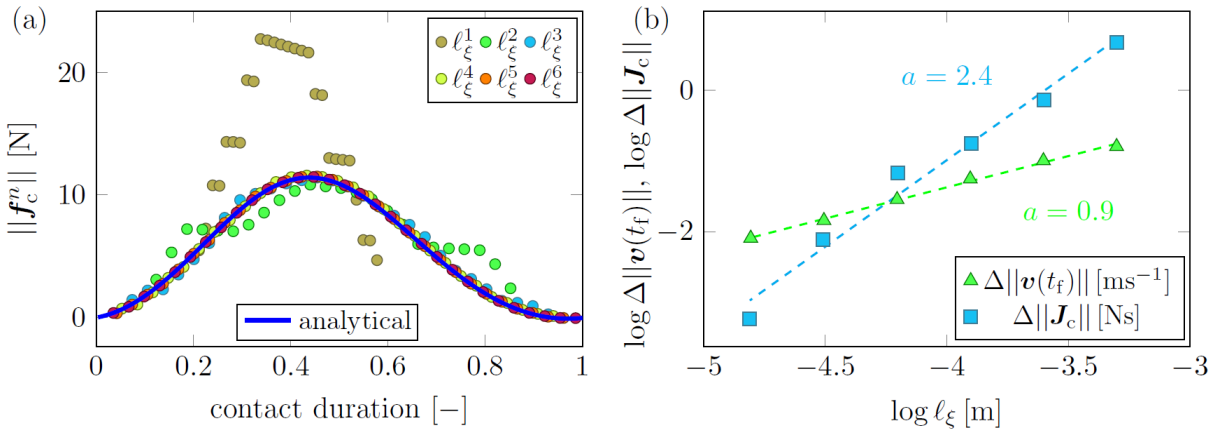


Figure 1: Convergence test results (a) evolution of the normal contact force magnitude during the collision. (b) Rate of convergence of the end time velocity $||\boldsymbol{v}(t_f)||$ and the normal contact force impulse $||\boldsymbol{J}_c||$ towards the analytical sphere results.

of normal contact force $\boldsymbol{f}_c^n$ and to evaluate the algorithm single-core efficiency, and to (ii) test the current code parallelization. In the first test, we consider one pair of identical spherical particles projected onto hexahedral CFD computational mesh with the particle diameter d_s spanning over 20 CFD cells. Initially, the sphere centers are $1.5\,d_s$ apart. The top sphere moves towards the fixed bottom sphere with the initial velocity of $\boldsymbol{v}_i = (0, -1, 0)^T\,\mathrm{m\,s^{-1}}$ and the y axis connecting the sphere centers. The material properties are rubber-like, i.e. $Y = 0.1\,\mathrm{GPa}$, $\rho = 1000\,\mathrm{kg\,m^{-3}}$, and the damping coefficient is set to $\gamma = 2.5$. The second test comprises simultaneously evaluated eight pairs of contacts identical to the first test. In the following, the tests are referred to as "single colliding pair" and "multi-colliding pairs", respectively. All the numerical computations are focused solely on the DEM behavior with the time integration step $\Delta t = 5\,\mu\mathrm{s}$.

Convergence of virtual mesh to perfect geometry The *virtual mesh* is designed for arbitrarily shaped solids represented by a triangulated surface. However, to discard the possible error caused by an inaccurate geometry, we work solely with perfect spheres. In the *virtual mesh*, the CFD cells are divided to $n_\xi = 2^{3\,\mathrm{LV}}$ *sub-volumes* ξ, which are used to numerically evaluate $(3)_3$. The *virtual mesh* results are compared to the ones obtained from *soft*DEM leveraging *analytical* $(3)_3$ evaluation available for perfect spheres. The independent variable for the test is the length of the side ℓ of the *sub-volume* ξ at the refinement level $\mathrm{LV} = 1, \ldots, 6$; ℓ_ξ^{LV}. The comparison of $\|\boldsymbol{f}_c^n\|$ evolution during the contact given in Fig. 1a shows that for $\mathrm{LV} \geq 4$ the $\|\boldsymbol{f}_c^n\|$ evolution is qualitatively indistinguishable from the *analytical* results. Quantitatively, the observed rate of convergence for magnitudes of the terminal particle velocity $||\boldsymbol{v}(t_f)||$ and the normal contact force impulse $||\boldsymbol{J}_c||$ is approximately 0.9 and 2.4, respectively, see Fig. 1b.

Single-core virtual mesh efficiency Now, let us compare computational costs of three approaches applicable for accurate contact solution for the spherical particles in CFD-DEM, (i) analytical solution, (ii) proposed *virtual mesh* (VM) and (iii) adaptive CFD mesh refinement (AMR) in the vicinity of contact. The tests were performed for the same settings of spatial refinement as in the former test. However, the analytical solution is not affected by the spatial refinement as it is mesh-independent. The AMR results are presented only up to refinement level $\mathrm{LV} = 3$, as the higher settings are computationally unfeasible due to the required time and disk capacity. For detailed results, see Fig. 2a.

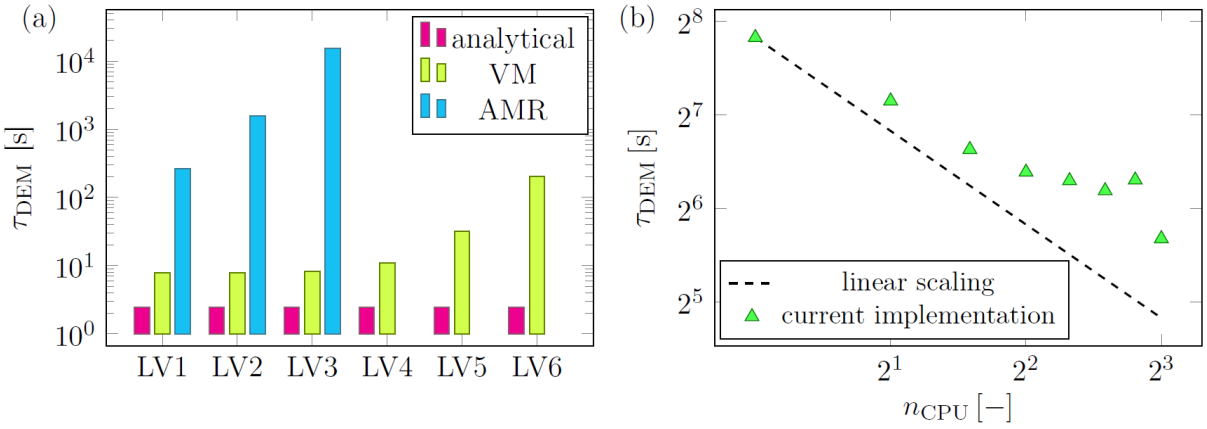


Figure 2: (a) Single colliding pair, comparison of computational times for selected approaches (i) analytic solution (ii) *virtual mesh* (VM) and (iii) adaptive mesh refinement (AMR), with different levels of refinement. (b) Multi-colliding pairs, scaling of our current DEM implementation.

Parallelization and speed-up of contact solution The last test is an efficiency study of our current parallelization implementation for the contact solution with *virtual mesh*. The used test is the multi-colliding pairs one with the level of refinement LV = 5. The test was evaluated using $1, \ldots, 8$ CPUs. The test results are given in Fig. 2b. The presented data combine the original L_1 OpenFOAM parallelization with a custom L_2 approach designed for *virtual mesh*. For the L_2 approach, the maximum reasonable number of CPUs for 8 multi-colliding pairs is 8 as each CPU is assigned a single contact pair.

4 Conclusion

In this paper, we presented new results for the *virtual mesh* extension to our in-house developed CFD-DEM solver that were focused on the improvement of accuracy and computational efficiency for the collision of arbitrarily shaped solids. The results show satisfactory accuracy with a significant increase in time efficiency compared to alternative methods such as adaptive CFD mesh refinement. However, the *virtual mesh* time requirements are still high compared to analytical approaches applicable to spherical particles. A method to increase the *virtual mesh* is efficient utilization of parallel architectures, which, despite being a work in progress, shows promising results.

Acknowledgement: The work was supported by the institutional support RVO:61388998 and by the Czech Science Foundation (GA 22-12227S).

References

[1] X. Kang, Z. Xia, J. Wang, W. Yang: *A novel approach to model the batch sedimentation and estimate the settling velocity, solid volume fraction, and floc size of kaolinite in concentrated solutions.* Colloids and Surfaces A: Physicochemical and Engineering Aspects, 579:123647, 2019.

[2] H. Ma, L. Zhou, Z. Liu, M. Chen, X. Xia, Y. Zhao: *A review of recent development for the CFD-DEM investigations of non-spherical particles.* Powder Technology, 412, 2022.

[3] M. Blažek, M. Žalud, P. Kočí, A. York, C.M. Schlepütz, M. Stampanoni, V. Novák: *Wash-coating of catalytic particulate filters studied by time-resolved X-ray tomography.* Chemical Engineering Journal, 409:128057, 2021.

[4] J. Chen: *Understanding the Discrete Element Method: Simulation of Non-Spherical Particles for Granular and Multi-Body Systems.* PhD thesis, 2012.

[5] O. Studeník, M. Kotouč Šourek, M. Isoz: *Octree-Generated Virtual Mesh for Improved Contact Resolution in CFD-DEM Coupling.* In D. Šimurda, T. Bodnár (eds.): Proceedings of the conference Topical Problems of Fluid Mechanics 2022, pp. 151–158, 02 2022.

[6] M. Isoz, M. Kotouč Šourek, O. Studeník, P. Kočí: *Hybrid fictitious domain-immersed boundary solver coupled with discrete element method for simulations of flows laden with arbitrarily shaped particles.* Computers and Fluids, 244:105538, 2022.

[7] OpenCFD: *OpenFOAM: The Open Source CFD Toolbox.* User Guide Version 1.4, OpenCFD Limited. Reading UK, 2007.

L^2 stability of macroscopic traffic flow models on networks using numerical fluxes at junctions

L. Vacek[1], *V. Kučera*[1], *C.-W. Shu*[2]

[1] Faculty of Mathematics and Physics, Charles University, Prague
[2] Division of Applied Mathematics, Brown University, Providence

1 Introduction

Modelling of traffic flows will have an important role in the future. With a rising number of cars on the roads, we must optimize the traffic situation. That is the reason we started to study traffic flows. We can model real traffic situations and optimize e.g. the timing of traffic lights or local changes in the speed limit. The benefits of modelling and optimization of traffic flows are both ecological and economical.

Let us have a road and an arbitrary number of cars. We would like to model the movement of cars on our road. There are two main ways how to describe traffic flow, *microscopic models* and *macroscopic models*. We choose the macroscopic approach, where we view our traffic situation as a continuum and study the density of cars in every point of the road. This model is described by partial differential equations.

2 Macroscopic traffic flow models

Our work [1] describes the numerical solution of traffic flows on networks. Using macroscopic models, it is possible to make simulations on big networks with a large number of cars. These models are described by partial differential equations in the form of conservation laws:

$$\frac{\partial}{\partial t}\rho\left(x,t\right)+\frac{\partial}{\partial x}Q\left(x,t\right)=0, \tag{1}$$

where $\rho\left(x,t\right)$ and $Q\left(x,t\right)$ are the unknown traffic density and traffic flow at position x and time t, respectively. Equation (1) must be supplemented by the initial condition $\rho(x,0)=\rho_0(x)$ and $Q(x,0)=Q_0(x)$ and an inflow boundary condition. We have only one equation (1) for two unknowns. Thus, we use the *Lighthill–Whitham–Richards model* (abbreviated LWR) where $Q\left(x,t\right)$ is taken as the equilibrium flow $Q_e(\rho(x,t))$, cf. [1].

Following [2], we consider a complex *network* represented by a directed graph. The graph is a finite collection of directed edges, connected together at vertices. Each vertex has a finite set of incoming and outgoing edges. On each road (edge) we consider the LWR model, while at junctions (vertices) we consider a *Riemann solver*.

3 Discontinuous Galerkin method

Due to the character of equation (1), we can expect discontinuity of the traffic density $\rho\left(x,t\right)$. Therefore, for the numerical solution of our models, we choose the *discontinuous Galerkin* (DG) method, which is essentially a combination of finite volume and finite element techniques, cf. [3].

Consider an interval $\Omega = (a, b)$. Let $\mathcal{T}_h$ be a partition of $\overline{\Omega}$ into a finite number of intervals (elements). We denote the set of all boundary points of all elements by $\mathcal{F}_h$. We seek the numerical solution in the space of discontinuous piecewise polynomial functions $S_h = \{v;\ v|_K \in P^p(K),\ \forall K \in \mathcal{T}_h\}$, where $P^p(K)$ denotes the space of all polynomials on K of degree at most $p \in \mathbb{N}$. For a function $v \in S_h$ we denote the *jump* in the point s as $[v]_s = v^{(L)}(s) - v^{(R)}(s)$, where we use the notation of spatial limits $v^{(L)}(s) := \lim_{x \to s_-} v(x)$ and $v^{(R)}(s) := \lim_{x \to s_+} v(x)$.

The DG formulation of equation (1) then reads: Find $\rho_h : [0, T] \to S_h$ such that

$$\int_\Omega (\rho_h)_t \varphi \,\mathrm{d}x - \sum_{K \in \mathcal{T}_h} \int_K Q_e(\rho_h) \varphi_x \,\mathrm{d}x + \sum_{s \in \mathcal{F}_h} H(\rho_h^{(L)}, \rho_h^{(R)})\, [\varphi]_s = \int_\Omega g\varphi \,\mathrm{d}x,$$

for all $\varphi \in S_h$. In the boundary terms on $\mathcal{F}_h$ we use the approximation $Q_e(\rho_h) \approx H(\rho_h^{(L)}, \rho_h^{(R)})$, where H is a *numerical flux*. We use the *Godunov flux*, cf. [4]:

$$H(u_h^{(L)}, u_h^{(R)}) = \begin{cases} \min_{u_h^{(L)} \le u \le u_h^{(R)}} f(u), & \text{if } u_h^{(L)} < u_h^{(R)}, \\ \max_{u_h^{(R)} \le u \le u_h^{(L)}} f(u), & \text{if } u_h^{(L)} \ge u_h^{(R)}. \end{cases} \tag{2}$$

For our purposes, we use an alternative form of the Godunov numerical flux. Let the convective flux f have a global maximum at u_* and f is non–decreasing on the interval $(-\infty, u_*]$ and non–increasing on $[u_*, \infty)$. Then the Godunov numerical flux is defined as

$$H^{God}\left(u^-, u^+\right) = \min\left\{f_{in}(u^-), f_{out}(u^+)\right\}, \tag{3}$$

where

$$f_{in}(u^-) = \begin{cases} f(u^-), & \text{if } u^- < u_*, \\ f(u_*), & \text{if } u^- \ge u_*, \end{cases} \qquad f_{out}(u^+) = \begin{cases} f(u_*), & \text{if } u^+ \le u_*, \\ f(u^+), & \text{if } u^+ > u_*. \end{cases}$$

This can be interpreted as the maximal possible flow through the common boundary, where f_{in} is the maximal possible inflow from the left element and f_{out} is the maximal possible outflow to the right element. Formulas (2) and (3) are equivalent in the case of the convective flux f defined above.

4 Implementation

For time discretization of the DG method we use the *Adams–Bashforth methods*. As a basis for S_h, we use *Legendre polynomials*. We use *Gauss–Legendre quadrature* to evaluate integrals over elements.

Because we calculate physical quantities (density and velocity), the result must be in some interval, e.g. $\rho \in [0, \rho_{\max}]$. Thus, we use *limiters* in each time step to obtain the solution in the admissible interval. Here it is important not to change the total number of cars, which is fulfilled by complying with the relevant CFL condition. Following [4], we also apply limiting to treat spurious oscillations near discontinuities and sharp gradients in the numerical solution.

All the above was performed on networks. Thus, we had to deal with the problem of boundary conditions at the junctions. In [1] we introduce our own approach to boundary conditions at junctions, which uses special numerical flux choices. This approach is new and the behavior of the resulting model can be interpreted as the introduction of turning lanes in front of the

junction. This is a different approach to the models in [2] and [5], which correspond to single–lane roads where overtaking is prohibited. Moreover, the presented construction of the traffic flux at junctions allows the simulation of arbitrary traffic light combinations instead of only full red/green lights as in [2] and [5].

We proved several important properties of our proposed numerical scheme, such as a discrete analogue to the Rankine–Hugoniot conditions for the numerical fluxes at the junction, conservation property of the DG scheme and traffic distribution error, cf. [1, Lemma 2, Theorems 1 and 2]. In our further researched we proved L^2 stability of the solution and derived estimates of the discretisation error. These new results are the subject of a subsequent paper which is in preparation.

5 Conclusion

Our ongoing work deals with the numerical solution of macroscopic traffic flow models on network using the discontinuous Galerkin method. We briefly described an overview of our paper [1] and discussed the theoretical results we obtained. On individual roads, we use the Godunov numerical flux, while on junctions, we construct a new numerical flux based on the preferences of drivers. We show several important properties of our proposed numerical scheme, such as the Rankine–Hugoniot conditions, conservation property of the DG scheme, L^2 stability, and discretisation and traffic distribution errors. In the paper, we compare our approach with the paper [5] by Čanić, Piccoli, Qiu and Ren, where Runge–Kutta methods are used along with a different choice of numerical fluxes at junctions. We discuss the differences between the two approaches, where that of [5] corresponds to single–lane roads with a strict enforcement of a priory traffic distribution, while the presented approach corresponds to having dedicated turning–lanes and/or flexibility of the drivers' preferences in extreme situations such as congestion.

Acknowledgement: The short lecture by L. Vacek is supported by the project Cooperatio, of the Faculty of Mathematics and Physics, Charles University. The joint research of L. Vacek and C.–W. Shu was conducted under the auspices of the Brown–Charles Memorandum of Understanding.

References

[1] L. Vacek, V. Kučera: *Discontinuous Galerkin method for macroscopic traffic flow models on networks*. Comm. Appl. Math. Comput. (submitted), arXiv: 2011.10862.

[2] M. Garavello, B. Piccoli: *Traffic flow on networks*, AIMS Series on Applied Mathematics, 2006.

[3] V. Dolejší, M. Feistauer: *Discontinuous Galerkin Method – Analysis and Applications to Compressible Flow*. Springer, Heidelberg, 2015.

[4] C.-W. Shu: *Discontinuous Galerkin methods: general approach and stability*. Numerical solutions of partial differential equations, Birkhäuser Basel, 201, 2009.

[5] S. Čanić, B. Piccoli, J. Qiu, T. Ren: *Runge–Kutta Discontinuous Galerkin Method for Traffic Flow Model on Networks*. Journal of Scientific Computing 63, 2014.

Comparison of different approaches to determination of resonant frequencies of coupled vibro-acoustic systems

J. Valášek, J. Hubálek

Faculty of Mechanical Engineering, Czech Technical University in Prague

1 Introduction

This contribution deals with resonant frequency determination of the coupled vibro-acoustic problem. This problem is motivated by popular relaxing technique used by voice professionals – phonation into tubes or straws of various dimensions, see [3]. The enlargement of vocal tract by an additional tube, i.e. prolongation of the acoustic resonator, should result to the decrease of acoustic resonant frequencies, here to the proximity of the vocal folds vibrational spectrum. Contrary to this purely acoustic theory, the first acoustic resonance frequency was substantially higher, see measurements [3]. The same reference explained it satisfactory by relatively strong interaction between acoustics and mechanically compliant vocal folds with the aid of a simplified 1D model. Here we develop 2D model and we present two approaches of resonant frequencies determination – the modal analysis and the transfer function approach.

2 Mathematical model

Let us consider a two-dimensional vibro-acoustic problem in domain Ω which is composed of elastic structure domain Ω^s (vocal folds) and acoustic domain Ω^a. The acoustic domain represents human vocal tract of the length L_1 together with a thin tube inserted into mouth of the length L_2 and diameter S_2, see Figure 1. The following (disjoint) parts of boundary $\partial\Omega$ are considered: Γ^a_{Dir}, Γ^s_{Dir} and Γ^a_{Neu} and the common interface Γ_{W}.

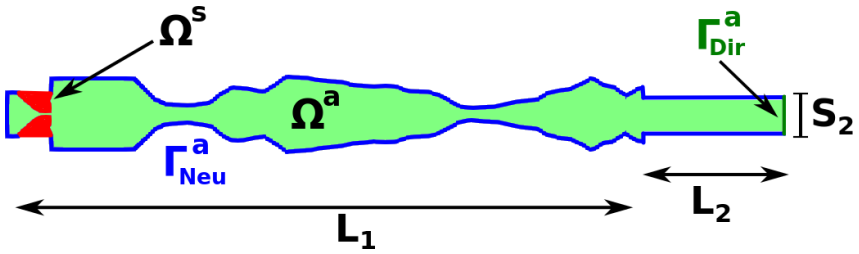


Figure 1: Scheme of considered vibro-acoustic domain Ω consisting of structure domain Ω^s and acoustic domain Ω^a together with marked boundaries of $\partial\Omega^a$.

As we are interested in frequency spectra the vibro-acoustic problem is described in frequency domain, i.e. all involved quantities depend on the spatial coordinates x and angular frequency ω, e.g. $u(x,\omega)$. Such description can be obtained by Fourier transform of the corresponding equations in time domain, see e.g. [1].

Acoustics. The sound propagation through homogeneous medium is governed in frequency domain by the Helmholtz equation for acoustic potential $\phi^a(x,\omega)$, see [1],

$$-\frac{\omega^2}{c^2}\phi^a - \Delta\phi^a = f^a(x,\omega), \qquad \text{in } \Omega^a, \tag{1}$$

where c is the speed of sound and function f^a describes possible generic sound sources. Acoustic potential is related to acoustic velocity and pressure by relations: $\mathbf{v}^a = -\nabla\phi^a$, $p^a = i\omega\rho^a\,\phi^a$, where ρ^a is constant air density and $i^2 = -1$. Two types of boundary conditions are here regarded

$$\text{a) } \phi^a(x,\omega) = 0 \quad \text{for } x \in \Gamma^a_{\text{Dir}}, \qquad \text{b) } \frac{\partial\phi^a}{\partial\mathbf{n}}(x,\omega) = 0 \quad \text{for } x \in \Gamma^a_{\text{Neu}}, \tag{2}$$

where vector $\mathbf{n} = (n_j)$ is unit outer normal to $\partial\Omega^a$. Condition (2 a) models the free end of tube while condition (2 b) represents fully reflecting walls, see [1].

Elastic structure. The elastic structure displacement $\mathbf{u}(x,\omega) = (u_1, u_2)$ is modelled by

$$\omega^2\rho^s u_i + \frac{\partial\tau^s_{ij}}{\partial x_j} = f^s_i(x,\omega), \qquad \text{in } \Omega^{\text{s}}, \quad (\text{i} = 1,2), \tag{3}$$

where ρ^s is the structure density, τ^s_{ij} denote the components of the Cauchy stress tensor and functions f^s_i stay for a possible elastic volume force source. The elastic body is here considered isotropic thus the stress tensor components can be written with help of the Hooke's law and Lamé coefficients λ^s, μ^s as

$$\tau^s_{ij} = \lambda^s \text{div } \mathbf{u}\,\delta_{ij} + 2\mu^s e^s_{ij}(\mathbf{u}), \tag{4}$$

where $e^s_{ij}(\mathbf{u}) = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right)$ is the small strain tensor, [1]. The elastic body is fixed on the boundary Γ^s_{Dir}

$$\mathbf{u}(x,\omega) = \mathbf{0}, \qquad \text{for } x \in \Gamma^s_{\text{Dir}}. \tag{5}$$

Vibro-acoustic coupling. The common vibro-acoustic coupling is used, i.e. the continuity of normal velocities and normal stresses along interface Γ_{W} is prescribed. It leads to the boundary condition for acoustic potential ϕ^a of the form

$$\frac{\partial\phi^a}{\partial\mathbf{n}}(x,\omega) = -i\omega\ \mathbf{u}(x,\omega)\cdot\mathbf{n}, \qquad x \in \Gamma_{\text{W}}, \tag{6}$$

where unit outer normal $\mathbf{n} = (n_j)$ to Γ_{W} points from Ω^s to Ω^a. i.e. it represents the acoustic emission given by normal acceleration of vibrating surface Γ_{W}, [1].

Further, the boundary condition prescribed for elastic body reads

$$\tau^s_{kl}(x,\omega)\,n_l = -i\omega\rho^a\ \phi^a(x,\omega)\,n_l, \qquad x \in \Gamma_{\text{W}}. \tag{7}$$

3 Numerical modelling

The FEM is used for spatial discretization of both subproblems (1) and (3). The Lagrange finite elements of the first order is chosen in both cases.

Acoustics. The weak formulation of problem (1) together with conditions (2) and (6) in functional space $X = \{f \in H^1(\Omega^a) | f = 0 \text{ on } \Gamma^{\mathrm{a}}_{\mathrm{Dir}}\}$ reads: find $\phi^a \in X$ that

$$-\frac{\omega^2}{c^2}\left(\phi^a, \eta\right)_{\Omega^a} + \left(\nabla\phi^a, \nabla\eta\right)_{\Omega^a} - i\omega\left(\mathbf{u}\cdot\mathbf{n}, \eta\right)_{\Gamma_{\mathrm{W}}} = \left(f^a, \eta\right)_{\Gamma_{\mathrm{W}}} \tag{8}$$

is satisfied for any $\eta \in X$ and by $(\cdot,\cdot)_{\mathcal{D}}$ is denoted the scalar product of functions from $L^2(\mathcal{D})$.

Finite element approximation ϕ^a_h of sought exact ϕ^a can be expressed as a linear combination of basis functions η_j from FE space $\mathbf{X}_h$, i.e. $\phi^a_h(x,\omega) = \sum_{j=1}^{N} \alpha^a_j(\omega)\eta_j(x)$. It leads to linear algebraic system of equations for unknown vector $\boldsymbol{\alpha}^a = (\alpha^a_j)$ for given $\omega \in \mathbb{R}$

$$-\frac{\omega^2}{c^2}\mathbb{M}^a\boldsymbol{\alpha}^a + \mathbb{K}^a\boldsymbol{\alpha}^a - i\omega\mathbb{C}^s\boldsymbol{\alpha}^s = \mathbf{b}^a(\omega), \tag{9}$$

where vector $\boldsymbol{\alpha}^s$ denotes unknowns of elastic part of the problem. The elements of matrices $\mathbb{M}^a = (m^a_{ij}), \mathbb{K}^a = (k^a_{ij})$ and $\mathbb{C}^s = (c^s_{ij})$ and the right hand side (RHS) vector $\mathbf{b}^a(\omega) = (b^a_j)$ are given by

$$m^a_{ij} = (\eta_j, \eta_i)_{\Omega^a}, \quad k^a_{ij} = (\nabla\eta_j, \nabla\eta_i)_{\Omega^a}, \quad c^s_{ij} = \left(\boldsymbol{\psi}_j\cdot\mathbf{n}, \eta_i\right)_{\Gamma_{\mathrm{W}}}, \quad b^a_j = \left(f^a(\omega), \eta_j\right)_{\Omega^a}, \tag{10}$$

where $\boldsymbol{\psi}_j$ denotes FE basis functions of the structural FE approximation space $\mathbf{V}_h$. Since the acoustic and the structure meshes are chosen to be consistent across the interface Γ_{W} no special treatment of the coupling matrices is needed.

Elastic structure. The standard weak formulation of (3) together with considered boundary conditions (5) and (7) leads to problem – find $\mathbf{u} \in \mathbf{V}$ such that

$$-\omega^2\rho^s\left(\mathbf{u}, \boldsymbol{\psi}\right)_{\Omega^s} + \left(\lambda^s(\operatorname{div}\mathbf{u})\,\mathbb{I} + 2\mu^s\mathbf{e}^s(\mathbf{u}), \mathbf{e}^s(\boldsymbol{\psi})\right)_{\Omega^s} + i\omega\rho^a\left(\phi^a\mathbf{n}, \boldsymbol{\psi}\right)_{\Gamma_{\mathrm{W}}} = \left(\mathbf{f}^s, \boldsymbol{\psi}\right)_{\Omega^s}, \tag{11}$$

holds for all $\boldsymbol{\psi} \in \mathbf{V} = V \times V$, $V = \{f \in H^1(\Omega^s) | f = 0 \text{ on } \Gamma^s_{\mathrm{Dir}}\}$.

The same discretization procedure by the FEM yields

$$-\omega^2\mathbb{M}^s\boldsymbol{\alpha}^s + \mathbb{K}^s\boldsymbol{\alpha}^s + i\omega\rho^a\mathbb{C}^a\boldsymbol{\alpha}^a = \mathbf{b}^s(\omega), \tag{12}$$

where matrices $\mathbb{M}^s$ and $\mathbb{K}^s$ are the mass and stiffness matrices, respectively. The components of right hand side vector are $b^s_j(\omega) = \left(\mathbf{f}^s, \boldsymbol{\psi}_j\right)_{\Omega^s}$ and the elements of coupling matrix are given by $c^a_{ij} = \left(\eta_j\,\mathbf{n},\ \boldsymbol{\psi}_i\right)_{\Gamma_{\mathrm{W}}}$.

Numerical solution of coupled vibro-acoustic problem. Collecting all terms of (9) and (12) in one system depending on the given $\omega \in \mathbb{R}$ yields

$$\left(-\omega^2\begin{pmatrix}\frac{1}{c^2}\mathbb{M}^a & 0\\ 0 & \rho^s\mathbb{M}^s\end{pmatrix} + i\omega\begin{pmatrix}0 & -\mathbb{C}^s\\ \rho^a\mathbb{C}^a & 0\end{pmatrix} + \begin{pmatrix}\mathbb{K}^a & 0\\ 0 & \mathbb{K}^s\end{pmatrix}\right)\begin{pmatrix}\boldsymbol{\alpha}^a\\ \boldsymbol{\alpha}^s\end{pmatrix} = \begin{pmatrix}\mathbf{b}^a\\ \mathbf{b}^s\end{pmatrix}. \tag{13}$$

In order to obtain frequency spectra two approaches can be followed. First, problem (13) assuming zero RHS vector represents the generalized eigenvalue problem which can be solved by e.g. mathematical library ARPACK. Second, the transfer function approach based on the computation of system response on the unit forcing at the given frequency is considered. The frequency spectrum is then obtained by substituting discrete values from the interval of frequency interest, see e.g. [1].

4 Numerical experiments

The considered vocal tract (VT) model with $L_1 = 0.1931\,\mathrm{m}$ is based on vowel [u:] of [2] and additional tube of dimensions $L_2 = 0.264\,\mathrm{m}$, $S_2 = 6.77\,\mathrm{mm}$, see [3]. The speed of sound is chosen as $c_0 = 343\,\mathrm{m/s}$ and density is $\rho^a = 1.2\,\mathrm{kg/m^3}$. The vocal fold (VF) geometry and material settings are overtaken from [4], i.e. density is chosen as $\rho^s = 1020\,\mathrm{kg/m^3}$.

First, the modal analysis of the coupled system (13) is performed, see Figure 2. Two obtained eigenfrequencies of the coupled system with acoustic meaning (under 800 Hz) are identical in this case with the first two modes of purely acoustic system composed of VT and tube and without coupling to the elastic VF. Thus this approach does not provide us relevant results (with current implementation in program Octave and its eigenvalue procedure).

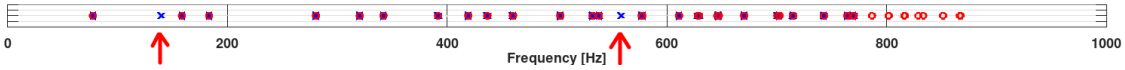


Figure 2: The first fifty eigenmodes of coupled system (blue crosses) compared with purely structural eigenfrequencies (red circles). Two highlighted frequencies with acoustic meaning are 140 Hz and 560 Hz.

Second, the transfer function approach is utilized. The unit forcing is prescribed in the closest vertices to point $[0.0065, 0]$m (acoustics) and $[0.006, \pm 0.002]$m (structure). Then for frequency in the range of $50 - 1000\,\mathrm{Hz}$ the problem (13) is solved. The max norm of acquired solutions are plotted in Figure 3. By further analysis of spatial distributions of solutions at selected frequencies can be found that the sought eigenfrequencies of the coupled system are: 305 and 690 Hz. For details see upcoming poster.

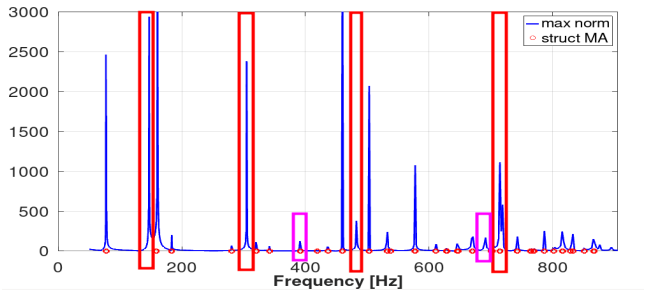


Figure 3: The response of coupled system (blue) to unit forcing at the chosen position and the given frequency. The purely structural frequencies are included for comparison (red circles). The highlighted frequencies without structural origin are: $146, 305, 390, 483, 690$ and 720 Hz.

5 Conclusion

The mathematical model of the vibro-acoustic problem in frequency domain motivated by phonation into tubes was described. The finite element method is used for the numerical solution of this problem. Two different approaches how to determine the resonant frequencies of the coupled system is presented. The modal analysis in this case failed probably due to wrong internal setting of the eigenvalue solver in Octave. The transfer function approach pointed on many frequencies

not related to the structure spectrum. Based on the spatial configuration of the investigated solutions the first two eigenfrequencies of the coupled system with an acoustic importance were identified. In accordance with reference [3] the first (acoustic) eigenfrequency of coupled system 305 Hz is significantly raised compared to the same setting considering purely acoustic problem ($F_1 = 139.8$ Hz).

Acknowledgement: This work has been supported by the Grant No. SGS22/148/OHK2/3T/12 of the Grant Agency of the CTU in Prague.

References

[1] M. Kaltenbacher: *Numerical simulation of mechatronic sensors and actuators: FEM for multiphysics*. Springer, 2015.

[2] B.H. Story, I.R. Titze, E.A. Hoffman: *Vocal tract area functions from magnetic resonance imaging*. The Journal of the Acoustical Society of America **100**(1), 537, 1996.

[3] J. Horáček, V. Radolf, A.M. Laukkanen: *Low frequency mechanical resonance of the vocal tract in vocal exercises that apply tubes*. Biomedical Signal Processing and Control **37**, 39, 2017.

[4] J. Valášek, M. Kaltenbacher, P. Sváček: *On the application of acoustic analogies in the numerical simulation of human phonation process*. Flow, Turbulence and Combustion **102**(1), 129, 2019.

Title: Seminar on Numerical Analysis & Winter School

Proceedings of the conference SNA'23
Ostrava, January 23 – 27, 2023

Editors: Jiří Starý, Stanislav Sysala, Dagmar Sysalová

Published by: Institute of Geonics of the CAS, Ostrava

First electronic edition

Ostrava, 2023

ISBN 978-80-86407-85-2