BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 69(4), 2021, Article number: e137125
DOI: 10.24425/bpasts.2021.137125

Three-dimensional simulations of the airborne COVID-19 pathogens using the advection-diffusion model and alternating-directions implicit solver

Marcin ŁOŚ [1], Maciej WOŹNIAK [1], Ignacio MUGA [2], and Maciej PASZYNSKI [1*]

[1] AGH University of Science and Technology, Faculty of Computer Science, Electronics and Telecommunications, al. Mickiewicza 30, 30-059 Krakow, Poland
[2] Instituto de Matemáticas, Pontificia Universidad Católica de Valparaíso, Chile

Abstract. In times of the COVID-19, reliable tools to simulate the airborne pathogens causing the infection are extremely important to enable the testing of various preventive methods. Advection-diffusion simulations can model the propagation of pathogens in the air. We can represent the concentration of pathogens in the air by "contamination" propagating from the source, by the mechanisms of advection (representing air movement) and diffusion (representing the spontaneous propagation of pathogen particles in the air). The three-dimensional time-dependent advection-diffusion equation is difficult to simulate due to the high computational cost and instabilities of the numerical methods. In this paper, we present alternating directions implicit isogeometric analysis simulations of the three-dimensional advection-diffusion equations. We introduce three intermediate time steps, where in the differential operator, we separate the derivatives concerning particular spatial directions. We provide a mathematical analysis of the numerical stability of the method. We show well-posedness of each time step formulation, under the assumption of a particular time step size. We utilize the tensor products of one-dimensional B-spline basis functions over the three-dimensional cube shape domain for the spatial discretization. The alternating direction solver is implemented in C++ and parallelized using the GALOIS framework for multi-core processors. We run the simulations within 120 minutes on a laptop equipped with i7 6700 Q processor 2.6 GHz (8 cores with HT) and 16 GB of RAM.

Key words: COVID-19; pathogen spread; isogeometric analysis; implicit dynamics; advection-diffusion; parallel alternating directions solver.

1. INTRODUCTION

According to the World Health Organization [1], COVID-19 is mainly spread through particles containing virus material exhaled by infected people through speech breathing or spread by sneezing or coughing.

How these particles spread, how their concentration changes with the distance from the speaking or sneezing person changes, and how the air movement influences these concentrations is in general unknown and can be measured experimentally, which is very difficult to estimate by using computer simulations.

Thus, in the days of COVID-19, computer simulations can help in making decisions such as the required distance in public places, the need to wear masks covering the mouth and nose, made of specific materials, the permissible time of stay with a person infected in one room, etc. This paper focuses on advection-diffusion equations, which can model the propagation of pathogens in the air. We can represent the concentration of pathogens in the air by "contamination" propagating from the source by advection and diffusion mechanisms. The first term describes the spread of particles forced by the air movements, and the second one models the spontaneous propagation of pathogen particles in the air. Unfortunately, the three-dimensional time-dependent advection-diffusion equations are complicated to simulate. This is related to the high computational cost of the three-dimensional problem as well as the instabilities of the numerical methods. As the remedy to the first problem, we propose the alternating directions implicit solver. We also focus on the mathematical analysis of the simulation that presents the necessary conditions for its stability.

The alternating directions implicit solver (ADI) was originally proposed for performing finite-difference simulations of time-dependent problems on regular grids. The first papers concerning the ADI method were published in 1960 [2–5]. The ADI with finite difference method is still popular for fast solutions of different classes of problems with finite difference method [6, 7]. The method introduces intermediate time steps in its basic version, and the differential operator is split into sub-operators, containing only the x, y, z derivatives. The time integration scheme involves sub-steps with only one sub-operator on the left-hand side and the other sub-operators on the right-hand side, acting on the previous sub-step solutions. As a result of this direction splitting, after the discretization of the linear equations system, we deal only with derivatives in one direction while the rest of the operator is on the right-hand side. If derived on the regular three-dimensional grid, the

*e-mail: paszynsk@agh.edu.pl

Manuscript submitted 2020-11-17, revised 2021-02-21, initially accepted for publication 2021-02-25, published in August 2021

resulting system of linear equations has a Kronecker product structure, and it can be factorized in a linear $\mathcal{O}(N)$ computational cost.

The isogeometric analysis (IGA) [8] is a modern method for performing finite element method (FEM) simulations with B-splines and NURBS. The IGA-FEM has multiple applications for simulations of time-dependent problems, including wind turbine aerodynamics [9], turbulent flow simulations [10], phase field phase-separation simulations [11–14], incompressible hyper-elasticity [16–19] and tumor growth simulations [20, 21]. The computational cost of the factorization of a system of linear equations resulting from IGA discretization with standard multi-frontal solvers is $\mathcal{O}\left(N^2p^3\right)$, and the constant in front of the computational complexity can be reduced using some special techniques [22].

The direction splitting method has been used to solve the isogeometric L2 projection problem over regular grids with tensor product B-spline basis functions [23–25] in a linear $\mathcal{O}(N)$ computational cost. The direction splitting, in this case, is possible by exploiting the Kronecker product structure of the Gram matrix resulting from a discretization over the patch of finite elements with tensor product structure of the B-spline basis functions. This direction splitting for isogeometric L2 projections were applied for performing simulations of explicit dynamics [20, 26–29], where each time step is computed as the solution of isogeometric L2 projection of the previous time step solution, plus the changes enforced by the physics of the computational problem, computed from the differential operator application to the previous time-step solution, plus the right-hand side.

Recently, in the conference proceedings [30], we extended the explicit dynamics method [20, 26–29] into the implicit dynamics solver. The solver was applied for the advection-diffusion problem. It employed the isogeometric analysis approach, stabilized with the residual minimization method. The residual minimization method stabilizes the finite element simulations by considering a Petrov-Galerkin formulation with different trial and test spaces. The stabilization is obtained by increasing the dimension of the test space. The residual minimization stabilization is required when there is a large contrast between the advection and diffusion terms, which is not the case in the airborne pathogen simulations.

The novelties of this paper with regard to our previous work, are the following. In this paper, we provide the mathematical analysis on the well-posedness of the weak formulation for the advection-diffusion problem, for each substep of the split operator, under the zero Neumann boundary assumption condition. We are not aware of any other paper proving the well-posedness in this case. The general Dirichlet boundary condition case was analyzed in [31]. We also find the relation between the time-step size and the coercivity constant, thus the necessary condition for the non-stationary simulations' stability.

The structure of this paper is the following. We start in Section 2 with the derivation of the isogeometric alternating direction implicit method for the advection-diffusion problem. We show the strong form in Section 2.1, weak form in Section 2.2, we prove the well-posedness in Section 2.3, and we introduce the direction splitting in Section 2.4. Next, in Section 3, we present numerical results and the MATLAB code finding the relation between the time-step size and the coercivity constant.

Next, in Section 3, we present the linear computational cost numerical results. We summarize the paper with conclusions in Section 4. We also present in Section 5 an Appendix explaining how to perform linear computational cost factorization with Kronecker product matrices.

2. ISOGEOMETRIC ALTERNATING DIRECTIONS IMPLICIT METHOD FOR ADVECTION-DIFFUSION PROBLEM

In this section, partial derivatives with respect to a given variable s will be denoted by ∂_s. Additionally, a time-dependent function w captured at a given time-step τ, will be denoted by w_τ.

2.1. Strong formulation

Let $\Omega \subset \mathbb{R}^3$ be a cube with $\Gamma := \partial\Omega$ and let $T > 0$ be some final lapse of time. As usual, the standard $L^2(\Omega)$ inner product will be denoted by $(\cdot,\cdot)$. Let $f = f(x, y, z, t)$ be a source function, let $\vec{\beta} = \left(\beta^x(x, y, z, t), \beta^y(x, y, z, t), \beta^z(x, y, z, t)\right)$ be an advection field, and let the (positive) diffusion coefficients be denoted by

$$\begin{pmatrix} K^x(x,y,z,t) & 0 & 0 \\ 0 & K^y(x,y,z,t) & 0 \\ 0 & 0 & K^z(x,y,z,t) \end{pmatrix},$$

$\forall (x, y, z) \in \overline{\Omega}\ \ \forall t \in (0, T]$. Additionally, for any $t \in (0, T]$, we require that $f(\cdot, t) \in L^2(\Omega)$, $\beta(\cdot, t) \in [L^\infty(\Omega)]^3$, and $K(\cdot, t) \in [L^\infty(\Omega)]^9$.

Our model problem will be the advection-diffusion equation. We seek for the pollutant concentration field $u \in C^1\left(0, T; C^2(\Omega)\right)$ such that

$$\partial_t u - \nabla \cdot (K\nabla u) + \vec{\beta} \cdot \nabla u = f, \quad \text{in } (0,T) \times \Omega, \tag{1}$$

with homogeneous Neumann boundary condition and zero initial state, i.e.,

$$\begin{cases} (K\nabla u) \cdot n = 0 & \forall (x, y, z, t) \in \Gamma \times (0, T), \\ u(x, y, z, 0) = 0 & \forall (x, y, z) \in \Omega. \end{cases} \tag{2}$$

This three-dimensional time-dependent problem can be treated like a four-dimensional problem with Dirichlet boundary condition on one face of the hypercube, free boundary condition on the opposite face, and Neumann boundary condition on other faces. For this reason, it has a unique solution. The mathematical proof of the solution uniqueness for a similar setup for the heat transfer problem is presented in [26].

2.2. Weak formulation

In order to derive the weak formulation of (1, 2), we multiply by test functions and integrate by parts to get:

$$(\partial_t u, v) + (K\nabla u, \nabla v) + \left(\vec{\beta}\cdot\nabla u, v\right) = (f, v),$$
$$\forall v \in K^1\Omega. \tag{3}$$

where we have used the zero Neumann boundary condition in (2).

Now, we apply the alternating direction method concerning time. We split the advection and diffusion operators into three sub-operators, the first one with the derivatives in the x direction, the second one with the derivatives in the y direction, and the third one with the derivatives in the z direction. We introduce three intermediate time steps of size $\tau > 0$ each. In the first sub-step, we keep on the left-hand side the sub-operator with z derivatives, and we put the x and y derivatives on the right-hand side, i.e.,

$$\begin{aligned}\left(u_{t+\frac{1}{3}}, v\right) + \tau\left(\partial_z u_{t+\frac{1}{3}}, K^z_{t+\frac{1}{3}}\partial_z v + \beta^z_{t+\frac{1}{3}} v\right) &= (u_t + \tau f_t, v)\\ &- \tau\left(\partial_x u_t, K^x_t \partial_x v + \beta^x_t v\right)\\ &- \tau\left(\partial_y u_t, K^y_t \partial_y v + \beta^y_t v\right).\end{aligned} \tag{4}$$

In the second sub-step we keep on the left-hand side the sub-operator with y derivatives, and we put the x and z derivatives on the right-hand side, i.e.,

$$\begin{aligned}\left(u_{t+\frac{2}{3}}, v\right) + \tau\left(\partial_y u_{t+\frac{2}{3}}, K^y_{t+\frac{2}{3}}\partial_y v + \beta^y_{t+\frac{2}{3}} v\right) &= \left(u_{t+\frac{1}{3}} + \tau f_{t+\frac{1}{3}}, v\right)\\ &- \tau\left(\partial_x u_{t+\frac{1}{3}}, K^x_{t+\frac{1}{3}}\partial_x v + \beta^x_{t+\frac{1}{3}} v\right)\\ &- \tau\left(\partial_z u_{t+\frac{1}{3}}, K^z_{t+\frac{1}{3}}\partial_z v + \beta^z_{t+\frac{1}{3}} v\right).\end{aligned} \tag{5}$$

In the third sub-step we keep on the left-hand side the sub-operator with x derivatives, and we put the y and z derivatives on the right-hand side, i.e.,

$$\begin{aligned}(u_{t+1}, v) + \tau\left(\partial_x u_{t+1}, K^x_{t+1}\partial_x v + \beta^x_{t+1} v\right) &= \left(u_{t+\frac{2}{3}} + \tau f_{t+\frac{2}{3}}, v\right)\\ &- \tau\left(\partial_y u_{t+\frac{2}{3}}, K^y_{t+\frac{2}{3}}\partial_y v + \beta^y_{t+\frac{2}{3}} v\right)\\ &- \tau\left(\partial_z u_{t+\frac{2}{3}}, K^z_{t+\frac{2}{3}}\partial_z v + \beta^z_{t+\frac{2}{3}} v\right).\end{aligned} \tag{6}$$

Thus, in the three sub-steps, we have sub-problems with identical structures. The fact that we keep on the left-hand side only derivatives in one direction results in a Kronecker product structure of the linear system of equations after discretization with isogeometric analysis over a cube shape domain. This will result later in a linear $\mathcal{O}(N)$ computational cost of the solver.

2.3. Well-posedness

Let us focus now on the well-posedness of the weak sub-problems (4), (5) and (6). Within this Subsection 2.3, the Cartesian system of coordinates in $\mathbb{R}^3$ will be denoted by (x_1, x_2, x_3) instead of (x, y, z).

For $i = 1, 2, 3$, let V_i be the space of functions $v \in L^2(\Omega)$ such that the distributional derivative $\partial_{x_i} v$ is also in $L^2(\Omega)$. This space will be endowed with the scalar product:

$$(u, v)_{V_i} = (u, v) + (\partial_{x_i} u, \partial_{x_i} v), \qquad \forall u, v \in V_i. \tag{7}$$

It is easy to verify that V_i is a Hilbert space. The induced norm on V_i will be denoted by $\|\cdot\|_{V_i}$.

Given $w \in H^2(\Omega)$ (representing the solution from the previous time-step t) we can rewrite our sub-problems (4)–(6) in the following general form:

$$\text{Find } u \in V_i \text{ such that} \quad b(u, v) = l_w(v), \quad \forall v \in V_i, \tag{8}$$

with

$$\begin{aligned} b(u, v) &:= (u, v) + \tau\left(\partial_{x_i} u, K^i_{t+\tau}\partial_{x_i} v + \beta^i_{t+\tau} v\right)\\ l_w(v) &:= (w + \tau f_t, v) + \tau\left(K^j_t \partial^2_{x_j} w - \beta^j_t \partial_{x_j} w, v\right)\\ &\quad + \tau\left(K^k_t \partial^2_{x_k} w - \beta^k_t \partial_{x_k} w, v\right),\end{aligned} \tag{9}$$

where we have assumed that K is constant in the integration by parts, and $i \neq j \neq k$, and where we have integrated by parts back in the right-hand side $l_w(v)$, to ensure the well-posedness of the problem. Namely, both the bilinear form b and the linear form l are continuous with respect to the norm induced by the inner product (7) in V_i.

Theorem 1. For any $t \in (0, T - \tau]$, assume that there exists a constant $\eta^i_t > 0$ such that $K^i_{t+\tau} \geq \eta^i_t > \tau/4 \|\beta^i(\cdot, t+\tau)\|^2_{L^\infty}$ for almost every $x \in \Omega$. Then the bilinear form

$$b(u, v) = (u, v) + \tau\left(\partial_{x_i} u, K^i_{t+\tau}\partial_{x_i} v + \beta^i_{t+\tau} v\right) \tag{10}$$

is coercive on V_i, and thus problem (8) is well-posed for any $w \in H^2(\Omega)$.

Proof. For any $u \in V_i$, first observe that $\left(K^i_{t+\tau}\partial_{x_i} u, \partial_{x_i} u\right) \geq \eta^i_t \|\partial_{x_i} u\|^2_{L^2}$. Next, notice that:

$$\begin{aligned}(\beta^i_{t+\tau}\partial_{x_i} u, u) &\geq -|(\beta^i_{t+\tau} u, \partial_{x_i} u)|\\ &\geq -\|\beta^i_{t+\tau}\|_{L^\infty}\|u\|_{L^2}\|\partial_{x_i} u\|_{L^2}\\ &\geq -\|\beta^i_{t+\tau}\|_{L^\infty}\left(\frac{\|u\|^2_{L^2}}{2\varepsilon} + \frac{\varepsilon}{2}\|\partial_{x_i} u\|^2_{L^2}\right),\end{aligned}$$

where we have used Hölder inequality and Young's inequality with $\varepsilon > 0$. Thus, we have

$$
\begin{aligned}
b(u,u) &= \|u\|_{L^2}^2 + \tau(K^i_{t+\tau}\partial_{x_i}u, \partial_{x_i}u) + \tau(\beta^i_{t+} \\
&\geq \|u\|_{L^2}^2 + \tau\eta^i_t\|\partial_{x_i}u\|_{L^2}^2 \\
&- \tau\|\beta^i_{t+\tau}\|_{L^\infty}\left(\frac{\|u\|_{L^2}^2}{2\varepsilon} + \frac{\varepsilon}{2}\|\partial_{x_i}u\|_{L^2}^2\right) \qquad (11)\\
&= \left(1 - \frac{\tau\|\beta^i_{t+\tau}\|_{L^\infty}}{2\varepsilon}\right)\|u\|_{L^2}^2 \\
&+ \tau\left(\eta^i_t - \frac{\varepsilon}{2}\|\beta^i_{t+\tau}\|_{L^\infty}\right)\|\partial_{x_i}u\|_{L^2}^2.
\end{aligned}
$$

To get coercivity, we need to show the positivity of the last two expressions on the right-hand side of (11). If $\|\beta^i_{t+\tau}\|_{L^\infty} = 0$, then the result is trivial. Otherwise, take any $\delta > 0$ such that $\frac{\tau}{4}\|\beta^i_{t+\tau}\|^2_{L^\infty} < \delta < \eta^i_t$ and let $\varepsilon = 2\delta\|\beta^i_{t+\tau}\|^{-1}_{L^\infty}$. Then,

$$
1 - \frac{\tau\|\beta^i_{t+\tau}\|_{L^\infty}}{2\varepsilon} = 1 - \frac{\tau\|\beta^i_{t+\tau}\|^2_{L^\infty}}{4\delta} > 0
$$

and

$$
\eta^i_t - \frac{\varepsilon}{2}\|\beta^i_{t+\tau}\|_{L^\infty} = \eta^i_t - \delta > 0.
$$

Finally, the well-posedness of problem (8) is a consequence of the Lax-Milgram Theorem for coercive and continuous bilinear forms. □

Remark 1. The message from Theorem 1 is that, to ensure solvability, we need to pick a *small enough* time-step $\tau > 0$ fulfilling the requirement:

$$
K^i_{t+\tau} \geq \eta^i_t > \frac{\tau}{4}\|\beta^i(\cdot, t+\tau)\|^2_{L^\infty}, \qquad \text{for some} \quad \eta^i_t > 0.
$$

However, too small values of $\tau > 0$ will compromise the stability of the problem (same as in reaction-dominated diffusion problems).

Remark 2. On the discrete level, the integration back by parts requires using higher order B-spline basis functions.

2.4. Direction splitting with isogeometric analysis discretization

Let us perform now the discretization of the weak sub-problem using B-spline basis functions from the isogeometric analysis [8].

Let N_x, N_y and N_z be positive integers. On each axis we consider a basis of one-dimensional B-splines of order $p \in \mathbb{N}$. Thus, on the x-axis we consider the basis $\{B^x_{i;p}\}_{i=1,\ldots,N_x}$; on the y-axis we consider the basis $\{B^y_{j;p}\}_{j=1,\ldots,N_y}$; and on the z-axis we consider the basis $\{B^z_{k;p}\}_{k=1,\ldots,N_z}$. Our general discrete space V_h will consist in the tensor product of the previous B-splines. This is,

$$
V_h =: \operatorname{span}\{B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z)\}.
$$

Let us proceed with the discrete counterpart of the weak sub-problems (4), (5) and (6). To exemplify, we take the x-sub-problem. The discrete counterpart of the weak sub-problem (6) reads as follows:

$$
\begin{cases}
\text{Given } w_h \in V_h, \text{ find } u_h \in V_h \text{ such that:} \\
b^x(u_h, v_h) = l^x_{w_h}(v_h), \quad \forall v_h \in V_h,
\end{cases} \qquad (12)
$$

where

$$
b^x(u_h, v_h) := (u_h, v_h) + \tau\left(\partial_x u_h, K^x_{t+\tau}\partial_x v_h + \beta^x_{t+\tau}v_h\right)
$$

and

$$
\begin{aligned}
l^x_{w_h}(v_h) := (w_h + \tau f_t, v_h) + \tau\left(K^y_t\partial^2_y w_h - \beta^y_t\partial_y w_h, v_h\right) + \\
+ \tau\left(K^z_t\partial^2_z w_h - \beta^z_t\partial_z w_h, v_h\right).
\end{aligned}
$$

Remark 3. Let $V_x := \left\{v \in L^2(\Omega) : \partial_x v \in L^2(\Omega)\right\}$ endowed with the topology induced by the norm $\|\cdot\|^2_{V_x} := \|\cdot\|^2_{L^2} + \|\partial_x\cdot\|^2_{L^2}$. Clearly $V_h \subset V_x$. By Theorem 1 the (continuous) bilinear form $b^x(\cdot,\cdot)$ is coercive in V_x provided that t, τ, β^x and K^x satisfy the requirements of the aforementioned Theorem. If the case, the discrete problem (12) is also coercive (thus, well-posed) whenever V_h is endowed with the norm $\|\cdot\|_{V_x}$.

On the other hand, notice that $w_h \in V_h$ represents the discrete solution of a previous time-step. That solution requires supporting second order weak-derivatives to make l_{w_h} a continuous functional on V_x. For that reason we need the order of the B-splines to be $p \geq 2$.

Since the solution $u_h \in V_h$ of problem (12) is of the form

$$
u_h = \sum_{i,j,k} u_{ijk}B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z), \qquad (13)
$$

the problem (12) translates into a $N_xN_yN_z \times N_xN_yN_z$ linear system whose unknowns are the scalars $\{u_{ijk}\}$ and the coefficients of the associated matrix are:

$$
\begin{cases}
B^x_{lmnijk} = b^x\left(B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z),\ B^x_{l;p}(x)B^y_{m;p}(y)B^z_{n;p}(z)\right), \\
\forall i,l = 1,\ldots,N_x \quad \forall j,m = 1,\ldots,N_y \quad \forall k,n = 1,\ldots,N_z.
\end{cases} \qquad (14)
$$

On the other hand, the coefficients of the right-hand side vector are given by:

$$
\begin{cases}
l^x_{w_h}\left(B^x_{l;p}(x)B^y_{m;p}(y)B^z_{n;p}(z)\right), \\
\forall l = 1,\ldots,N_x \quad \forall m = 1,\ldots,N_y \quad \forall n = 1,\ldots,N_z.
\end{cases}
$$

Let us focus on the particular entries of the left-hand side matrix (14) discretized with tensor product of one dimensional B-spline basis functions:

$$
\begin{aligned}
&B^x_{lmnijk} \\
&= \left(B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z),\ B^x_{l;p}(x)B^y_{m;p}(y)B^z_{n;p}(z)\right) \\
&+\tau\left(\partial_x B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z), K^x_{t+\tau}\partial_x B^x_{l;p}(x)B^y_{m;p}(y)B^z_{n;p}(z)\right) \qquad (15)\\
&+\tau\left(\partial_x B^x_{i;p}(x)B^y_{j;p}(y)B^z_{k;p}(z), \beta^x_{t+\tau}B^x_{l;p}(x)B^y_{m;p}(y)B^z_{n;p}(z)\right).
\end{aligned}
$$

We perform the direction splitting for each sub-problem. Namely, we express the matrix entries as the multiplication of three one-dimensional integrals, the first one with B-spline basis functions along the x direction, the second one with B-splines along the y direction, and the third one with B-splines along the z direction. For the sub-problem corresponding to $i = 1$ we get

$$\begin{aligned} B^x_{lmnijk} &= \left(B^x_{i;p}(x), B^x_{l;p}(x)\right)\left(B^y_{j;p}(y), B^y_{m;p}(y)\right) \\ &\times \left(B^z_{k;p}(z), B^z_{n;p}(z)\right) \\ &+\tau\left(\partial_x B^x_{i;p}(x), K^x_{t+\tau}\partial_x B^x_{l;p}(x)\right)\left(B^y_{j;p}(y), B^y_{m;p}(y)\right) \\ &\times \left(B^z_{k;p}(z), B^z_{n;p}(z)\right) \\ &+\tau\left(\partial_x B^x_{i;p}(x), \beta^x_{t+\tau} B^x_{l;p}(x)\right)\left(B^y_{j;p}(y), B^y_{m;p}(y)\right) \\ &\times \left(B^z_{k;p}(z), B^z_{n;p}(z)\right) \\ &= \left[\left(B^x_{i;p}(x), B^x_{l;p}(x)\right)\right. \\ &\left. + \tau\left(\partial_x B^x_{i;p}(x), K^x_{t+\tau}\partial_x B^x_{l;p}(x) + \beta^x_{t+\tau} B^x_{l;p}(x)\right)\right] \\ &\times \left(B^y_{j;p}(y), B^y_{m;p}(y)\right)\left(B^z_{k;p}(z), B^z_{n;p}(z)\right) \end{aligned}$$

$$B^x = C \otimes D \otimes E. \tag{16}$$

In our model we assume that $K^i_{t+\tau}$ and $\beta^i_{t+\tau}$ are constant in a given time moment. Hence, our matrices at every time step are decomposed into three Kronecker product sub-matrices $B^x = C \otimes D \otimes E$. Namely, each sub-problem is equivalent to the three one-dimensional problems with multiple right-hand sides and can be factorized with a linear $\mathcal{O}(N)$ computational cost. We refer to the Appendix for more details.

3. NUMERICAL RESULTS

In this section, we propose a three-dimensional model of coughing based on advection-diffusion equation. We describe the three-dimensional numerical simulation of coughing with an advection-diffusion model over a 3D cube shape domain representing the area of $500\times500\times500$ cm.

$$\frac{du}{dt} - \nabla\cdot(K\nabla u) + \beta\cdot\nabla u = f. \tag{17}$$

In our equation we use the diffusion of the air

$$K = (0.25, 0.25, 0.25)\,[\mathrm{cm}^2/\mathrm{s}]. \tag{18}$$

The advection-diffusion model requires assuming the velocity of the air β as enforced by the coughing. The β is assumed to be given by $\beta = (w(t), 0, 0)$ $w(t) = 80$ [cm/s] for $t \in (0, 0.2)$ [s] and $w(t) = 80 - 80\ (t - 0.2)$ [cm/s] for $t \in (0.2, 0.3)$ [s] and $w(t) = 0$ for $t > 0.3$ [s]. In other words, we assume the air velocity of 80 [cm/s] for 0.2 [s] that later goes down to 0 within 0.1 [s].

The source function models the volume of released particles potentially with pathogens. The source is defined as a ball located at $(1, 2.5, 2.5)$ [m] with radius $r = 0.15$ [m]. It is amplified by $s(t) = max(1 - t/2)^2, 0)$, which is a function decreasing in quadratic way from 1 to 0 over a time interval $(0, 0.2)$ [s]. The concentration inside the initial ball is defined as $\phi(r^2)$, where $r = \min(\|\mathbf{x} - (1.0, 2.5, 2.5)\|/0.15)$, $\phi(t) = (t-1)^2(t+1)^2$. The initial state is defined as the zero pathogens concentration in the entire domain.

The computational domain unit is centimeter [cm], the wind velocity β is given in centimeters per second [cm/s], and the diffusion coefficient K is given in square centimeters per second [cm^2/s]. The units for the solution are then kilograms per cube centimeter [kg/cm^3].

We recall Theorem 1, where we provide sufficient conditions for the coercivity of the bilinear form. This allows us to estimate the coercivity constant that enters the Céa's Lemma, which in this case, is a function of the time-step size τ. To avoid instabilities, we want this constant to be as far away as possible from zero.

We have derived a simple MATLAB code to estimate the coercivity constant as a function of the time-step size τ. The code allows selecting the time-step size that provides the maximum coercivity constant. In the MATLAB code, we consider 0.1 [s] for the time unit and centimeter for the dimension unit. Thus, the diffusion in the horizontal direction z since it has the minimal value of `K=2.5` [cm^2]/(0.1 s) affecting most of the stability of the simulation. We estimate also the L^∞ norm of the advection function to obtain `beta=8` [cm/0.1 s]. We plot the relationship between the time step size `tau` and the coercivity constant `y` in a range going from 0 to the maximum admissible value of `4*K/beta*^2`, with the span of 100 points.

```
K=2.5; [cm^2/0.1s]
beta = 8; [cm/0.1s]
tau = linspace(0,4*K / beta^2,1);
y = K*tau + 1 - sqrt((1- K*tau).^2 +
  tau.*tau*beta^2);
plot(tau,y)
```

We have generated the plot presented in Fig. 1. The optimal value of τ corresponds to the maximum value of the coercivity constant. We have read the optimal $\tau = 0.1$ [0.1 s] $= 0.1\,s$. As a result of that, we have obtained a stable simulation. We perform 500-time steps (which results in 5 s of the total time), and we present the snapshots of the simulation in time steps 10, 20, 40, 60, 100, 200, 350, 500 in Fig. 2, 3.

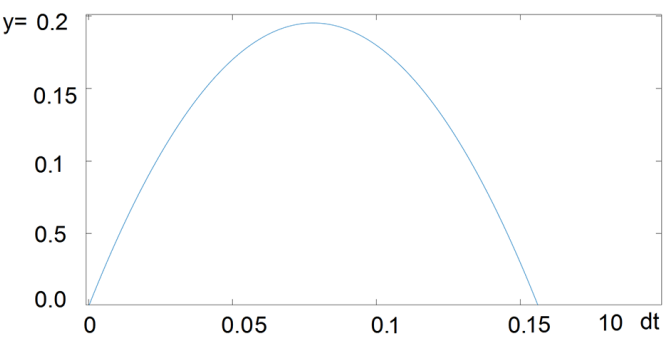

Fig. 1. Optimal values of time step size

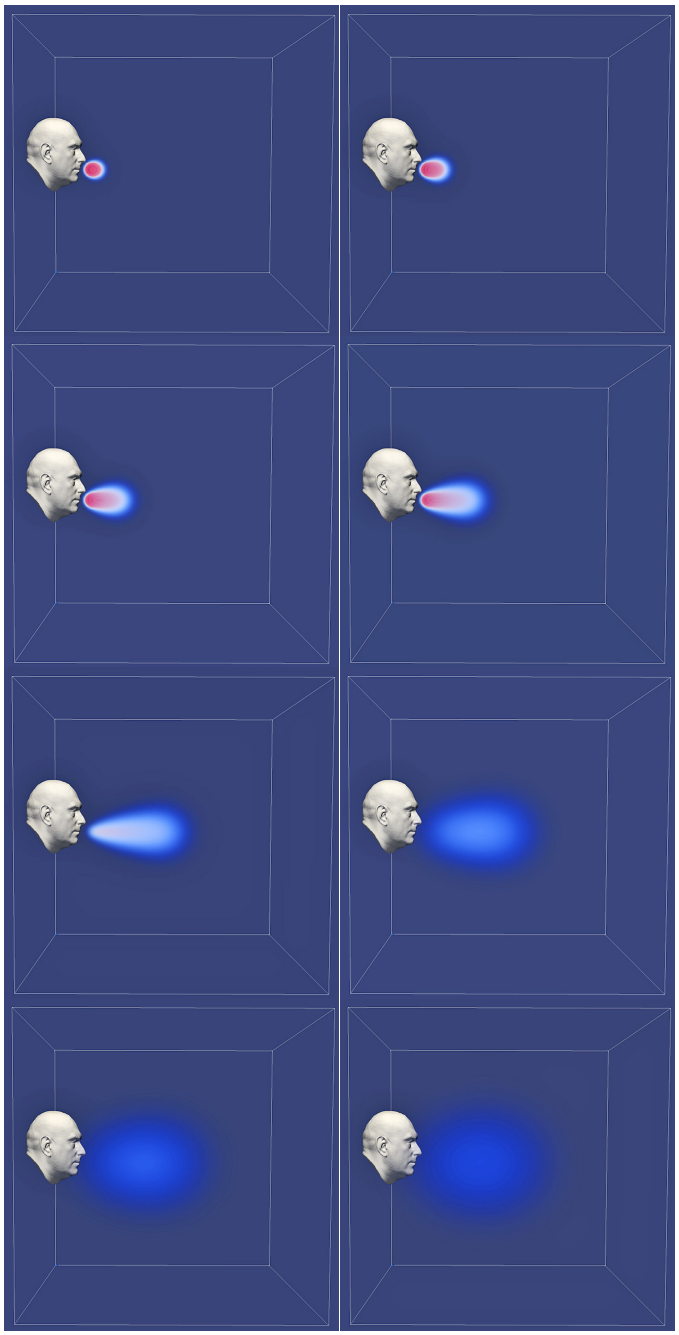

Fig. 2. Distribution of the density of pathogens at the time steps 10, 20, 40, 60, 100, 200, 350, 500 of the simulation performed with quadratic B-splines over 100×100×100 mesh. Without face mask

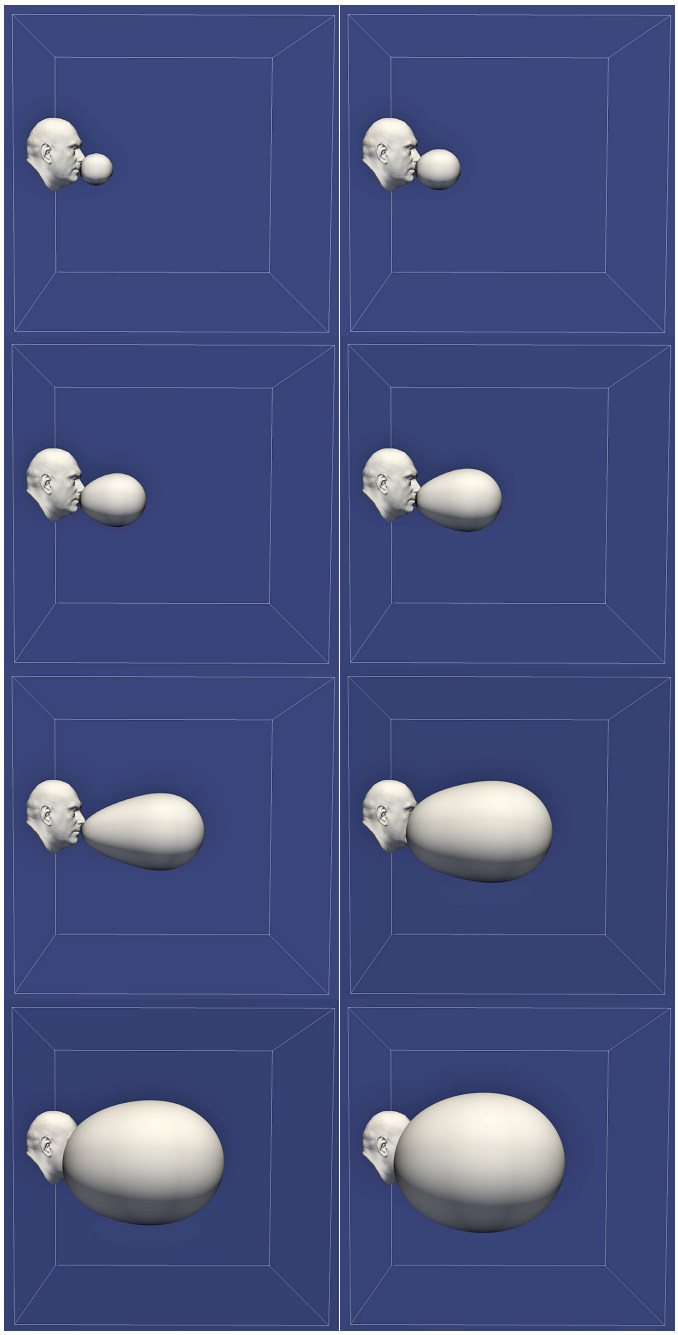

Fig. 3. Contour of the area of pathogens spread at the time steps 10, 20, 40, 60, 100, 200, 350, 500 of the simulation performed with quadratic B-splines over 100×100×100 mesh. Without face mask

For the simulation purposes, we use the IGA-ADS parallel code [28], with destination the shared memory Linux cluster nodes. It utilizes the GALOIS framework [32] for the parallelization. The total simulation time was 120 minutes on a laptop with i7 6700Q processor 2.6 GHz (8 cores with HT) and 16 GB of RAM. Note that the alternating direction implicit solver uses a direct solver. It utilizes the Kronecker product structure of the matrix. It provided *exact (up to the floating-point representation and round-off errors)* numerical Gaussian factorization of the linear system of linear equations. In this sense, we do not present the iterations or the ADI solver's convergence since it is executed only once. In other words, we can perform 500 Gaussian eliminations, each with 1,000,000 unknowns, on a laptop with eight cores, with the implicit method, within 2 hours.

4. CONCLUSIONS

In this paper, we analyzed the advection-diffusion model for the simulations of airborne pathogens mathematically. Simulating the propagation of pathogens in the air may have critical

applications in preventing the spread of the COVID-19 epidemic. In particular, the possibility of simulating such propagation enables experimental verification of preventive methods such as social distancing or covering the mouth and nose. We focused on an isogeometric finite element method simulations of the advection-diffusion problem with alternating directions implicit solver, resulting in ultra-fast, linear $\mathcal{O}(N)$ computational cost of the simulator. The application of B-spline basis functions for approximation results in the smooth, higher-order approximation of the solution. The method has been verified on the three-dimensional advection-diffusion problem. We have shown that the problem is well-posed under the assumption of zero Neumann boundary condition, and we derive the formula to find a stable time step size. Our future work will involve the extension of the model to more complicated geometries. In particular, we plan to study the airborne pathogen in a realistic geometry setup. It is a common misunderstanding that the alternating-direction implicit solver is limited to simple tensor product geometries. Indeed, as it has been recently shown in [33], it can be used for simulations in extremely complicated geometries.

ACKNOWLEDGEMENTS

The work was supported by the National Science Centre, Poland, grant no. grant no. 2017/26/M/ ST1/ 00281.

5. APPENDIX: LU FACTORIZATION OF KRONECKER PRODUCT MATRICES

Let us assume that we want to LU factorize the system of linear equations

$$\mathbf{M}\mathbf{x} = \mathbf{b} \tag{19}$$

with $\mathbf{M} = \mathbf{A} \otimes \mathbf{B}$, where $\mathbf{A}$ is $n \times n$, $\mathbf{B}$ is $m \times m$. From the definition of the Kronecker product we have

$$\mathbf{M} = \mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} \mathbf{A}B_{11} & \mathbf{A}B_{12} & \cdots & \mathbf{A}B_{1m} \\ \mathbf{A}B_{21} & \mathbf{A}B_{22} & \cdots & \mathbf{A}B_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}B_{m1} & \mathbf{A}B_{m2} & \cdots & \mathbf{A}B_{mm} \end{bmatrix}. \tag{20}$$

The right-hand side and the solution vectors are partitioned into m blocks of size n each

$$\begin{aligned} \mathbf{x}_i &= (x_{i1}, \ldots, x_{in})^{\mathrm{T}} \\ \mathbf{b}_i &= (b_{i1}, \ldots, b_{in})^{\mathrm{T}}. \end{aligned} \tag{21}$$

We can rewrite our system as a block matrix equation:

$$\begin{cases} \mathbf{A}B_{11}\mathbf{x}_1 + \mathbf{A}B_{12}\mathbf{x}_2 + \cdots + \mathbf{A}B_{1m}\mathbf{x}_m = \mathbf{b}_1 \\ \mathbf{A}B_{21}\mathbf{x}_1 + \mathbf{A}B_{22}\mathbf{x}_2 + \cdots + \mathbf{A}B_{2m}\mathbf{x}_m = \mathbf{b}_2 \\ \quad\vdots \qquad\quad \vdots \quad \vdots \qquad\quad \vdots \\ \mathbf{A}B_{m1}\mathbf{x}_1 + \mathbf{A}B_{m2}\mathbf{x}_2 + \cdots + \mathbf{A}B_{mm}\mathbf{x}_m = \mathbf{b}_m \end{cases}.$$

Now, we factor out $\mathbf{A}$:

$$\begin{cases} \mathbf{A}\left(B_{11}\mathbf{x}_1 + B_{12}\mathbf{x}_2 + \cdots + B_{1m}\mathbf{x}_m\right) = \mathbf{b}_1 \\ \mathbf{A}\left(B_{21}\mathbf{x}_1 + B_{22}\mathbf{x}_2 + \cdots + B_{2m}\mathbf{x}_m\right) = \mathbf{b}_2 \\ \quad\vdots \qquad\quad \vdots \quad \vdots \qquad\quad \vdots \\ \mathbf{A}\left(B_{m1}\mathbf{x}_1 + B_{m2}\mathbf{x}_2 + \cdots + B_{mm}\mathbf{x}_m\right) = \mathbf{b}_m \end{cases}. \tag{22}$$

We multiply by $\mathbf{A}^{-1}$ and define $\mathbf{y}^i = \mathbf{A}^{-1}\mathbf{b}^i$. We have here one 1D problem $\mathbf{A}\,\mathbf{y}_i = \mathbf{b}_i$ with multiple right-hand sides

$$\begin{cases} A_{11}y^{1i} + A_{12}y^{2i} + \cdots + A_{1n}y^{ni} = b_{1i} \\ A_{21}y^{1i} + A_{22}y^{2i} + \cdots + A_{2n}y^{ni} = b_{2i} \\ \quad\vdots \qquad\quad \vdots \quad \vdots \qquad\quad \vdots \\ A_{m1}y^{1i} + A_{m2}y^{2i} + \cdots + A_{nn}y^{ni} = b_{ni} \end{cases}. \tag{23}$$

Finally, in our family of problems

$$\begin{cases} B_{11}\mathbf{x}_1 + B_{12}\mathbf{x}_2 + \cdots + B_{1m}\mathbf{x}_m = \mathbf{y}_1 \\ B_{21}\mathbf{x}_1 + B_{22}\mathbf{x}_2 + \cdots + B_{2m}\mathbf{x}_m = \mathbf{y}_2 \\ \quad\vdots \qquad\quad \vdots \quad \vdots \qquad\quad \vdots \\ B_{m1}\mathbf{x}_1 + B_{m2}\mathbf{x}_2 + \cdots + B_{mm}\mathbf{x}_m = \mathbf{y}_m \end{cases} \tag{24}$$

we consider each component of $\mathbf{x}_i$ and $\mathbf{y}_i$ to get a family of linear systems

$$\begin{cases} B_{11}x^{1i} + B_{12}x^{2i} + \cdots + B_{1m}x^{mi} = y_{1i} \\ B_{21}x^{1i} + B_{22}x^{2i} + \cdots + B_{2m}x^{mi} = y_{2i} \\ \quad\vdots \qquad\quad \vdots \quad \vdots \qquad\quad \vdots \\ B_{m1}x^{1i} + B_{m2}x^{2i} + \cdots + B_{mm}x^{mi} = y_{mi} \end{cases} \tag{25}$$

for each $i = 1, \ldots, n$. We have another 1D problem here with multiple right-hand sides $\mathbf{B}\,\mathbf{x}_i = \mathbf{y}_i$.

In the case of finite element method computations with matrices (16), the systems of linear equations (23) and (25) have multi-diagonal matrices with $2p + 1$ diagonals. This is because the one-dimensional B-splines of order p span over $p + 1$ elements, the rows and columns in the one-dimensional matrices are related to particular B-splines, the entries are non-zero only if the B-splines from rows and columns overlap, and the maximum span of overlaping B-splines is thus $2p + 1$. The method generalizes into three-dimensions. Thus, these matrices can be factorizes in $\mathcal{O}(N_x p_x + N_y p_y + N_z p_z)$ where N_x, N_y, N_z are the number of B-splines along x, y, and z axis, p_x, p_y, p_z are the B-splines orders. We have $N_y N_z$, $N_x N_z$, and N_x, N_y right-hand sides in steps representing the factorizations

along x, y, and z directions. Thus the overall computational cost is $\mathcal{O}(N_x N_y N_z p_x + N_x N_y N_z p_y + N_x N_y N_z p_z) = \mathcal{O}(Np)$ since $N_x N_y N_z = N$ and $p = max\{p_x, p_y, p_z\}$.

REFERENCES

[1] "Coronavirus disease (COVID-19): How is it transmitted?". [Online] Available: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub/q-a-detail/q-a-how-is-covid-19-transmitted.

[2] D.W. Peaceman and H.H. Rachford Jr., "The numerical solution of parabolic and elliptic differential equations", *J. Soc. Ind. Appl. Math*., vol. 3, no. 1, pp. 28–41, 1955.

[3] J. Douglasand and H. Rachford, "On the numerical solution of heat conduction problems in two and three space variables", *Trans. Am. Math. Soc*., vol. 82, no. 2, pp. 421–439, 1956.

[4] E.L. Wachspress and G. Habetler, "An alternating-direction-implicit iteration technique", *J. Soc. Ind. Appl. Math*., vol. 8, no. 2, pp. 403–423, 1960.

[5] G. Birkhoff, R.S. Varga, and D. Young, "Alternating direction implicit methods", *Adv. Comput*., vol. 3, pp. 189–273, 1962.

[6] J.L. Guermond and P. Minev, "A new class of fractional step techniques for the incompressible Navier-Stokes equations using direction splitting", *C.R. Math*., vol. 348, pp. 581–585, 2010.

[7] J.L. Guermond, P. Minev, and J. Shen, "An overview of projection methods for incompressible flows", *Comput. Methods Appl. Mech. Eng*., vol. 195, pp. 6011–6054, 2006.

[8] J.A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Unification of CAD and FEA*, John Wiley and Sons, 2009.

[9] M.-C. Hsu, I. Akkerman, and Y. Bazilevs, "High-performance computing of wind turbine aerodynamics using isogeometric analysis", *Comput. Fluids*, vol. 49, pp. 93–100, 2011.

[10] K. Chang, T.J.R. Hughes, and V.M. Calo, "Isogeometric variational multiscale large-eddy simulation of fully-developed turbulent flow over a wavy wall", *Comput. Fluids*, vol. 68, pp. 94–104, 2012.

[11] L. Dedè, T.J.R. Hughes, S. Lipton, and V.M. Calo, "Structural topology optimization with isogeometric analysis in a phase field approach", *USNCTAM2010, 16th US National Congree of Theoretical and Applied Mechanics*, 2010.

[12] L. Dedè, M.J. Borden, and T.J.R. Hughes, "Isogeometric analysis for topology optimization with a phase field model", *Arch. Comput. Methods Eng*., vol. 19, pp. 427–465, 2012.

[13] H. Gómez, V.M. Calo, Y. Bazilevs, and T.J.R. Hughes, "Isogeometric analysis of the {Cahn-Hilliard} phase-field model", *Comput. Methods Appl. Mech. Eng*., vol. 197, pp. 4333–4352, 2008.

[14] H. Gómez, T.J.R. Hughes, X. Nogueira, and V.M. Calo, "Isogeometric analysis of the isothermal Navier-Stokes-Korteweg equations", *Comput. Methods Appl. Mech. Eng*., vol. 199, pp. 1828–1840, 2010.

[15] R. Duddu, L. Lavier, T.J.R. Hughes, and V.M. Calo, "A finite strain Eulerian formulation for compressible and nearly incompressible hyper-elasticity using high-order NURBS elements", *Int. J. Numer. Methods Eng*., vol. 89, pp. 762–785, 2012.

[16] S. Hossain, S.F.A. Hossainy, Y. Bazilevs, V.M. Calo, and T.J.R. Hughes, "Mathematical modeling of coupled drug and drug-encapsulated nanoparticle transport in patient-specific coronary artery walls", *Comput. Mech*., vol. 49, pp. 213–242, 2012.

[17] Y. Bazilevs, V.M. Calo, Y. Zhang, and T.J.R. Hughes, "Isogeometric fluid-structure interaction analysis with applications to arterial blood flow", *Comput. Mech*., vol. 38, pp. 310–322, 2006.

[18] Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. Hughes, A. Reali, and G. Scovazzi, "Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows", *Comput. Methods Appl. Mech. Eng*., vol. 197, pp. 173–201, 2007.

[19] V.M. Calo, N. Brasher, Y. Bazilevs, and T.J.R. Hughes, "Multiphysics Model for Blood Flow and Drug Transport with Application to Patient-Specific Coronary Artery Flow", *Comput. Mech*., vol. 43, pp. 161–177, 2008.

[20] M. Łoś, M. Paszyński, A. Kłusek, and W. Dzwinel, "Application of fast isogeometric L2 projection solver for tumor growth simulations", *Comput. Methods Appl. Mech. Eng*., vol. 316, pp. 1257–1269, 2017.

[21] M. Łoś, A. Kłusek, M. Amber Hassam, K. Pingali, W. Dzwinel, and M. Paszyński, "Parallel fast isogeometric L2 projection solver with GALOIS system for 3D tumor growth simulations", *Comput. Methods Appl. Mech. Eng*., vol. 343, pp. 1–22, 2019.

[22] A. Paszyńska, K. Jopek. M. Woźniak, and M. Paszyński, "Heuristic algorithm to predict the location of C0 separators for efficient isogeometric analysis simulations with direct solvers", *Bull. Pol. Acad. Sci. Tech. Sci*., vol. 66, no. 6, pp. 907–917, 2018.

[23] L. Gao and V.M. Calo, "Fast Isogeometric Solvers for Explicit Dynamics", *Comput. Methods Appl. Mech. Eng*., vol. 274, pp. 19–41, 2014.

[24] L. Gao and V.M. Calo, "Preconditioners based on the alternating-direction-implicit algorithm for the 2D steady-state diffusion equation with orthotropic heterogeneous coefficients", *J. Comput. Appl. Math*., vol. 273, pp. 274–295, 2015.

[25] L. Gao, "Kronecker Products on Preconditioning", PhD. Thesis, King Abdullah University of Science and Technology, 2013.

[26] M. Łoś, M. Woźniak, M. Paszyński, L. Dalcin, and V.M. Calo, "Dynamics with Matrices Possessing Kronecker Product Structure", *Procedia Comput. Sci*., vol. 51, pp. 286–295, 2015.

[27] M. Woźniak, M. Łoś, M. Paszyński, L. Dalcin, and V. Calo, "Parallel fast isogeometric solvers for explicit dynamics", *Comput. Inform*., vol. 36, no. 2, pp. 423–448, 2017.

[28] M. Łoś, M. Woźniak, M. Paszyński, A. Lenharth, and K. Pingali, "IGA-ADS : Isogeometric Analysis FEM using ADS solver", *Comput. Phys. Commun*., vol. 217, pp. 99–116, 2017.

[29] G. Gurgul, M. Woźniak, M. Łoś, D. Szeliga, and M. Paszyński, "Open source JAVA implementation of the parallel multi-thread alternating direction isogeometric L2 projections solver for material science simulations" *Comput. Methods Mater. Sci*., vol. 17, no.1, pp. 1–11, 2017.

[30] M. Łoś, J. Munoz-Matute, K. Podsiadło, M. Paszyński, and K. Pingali, "Parallel shared-memory isogeometric residual minimization (iGRM) for three-dimensional advection-diffusion problems", *Lect. Notes Comput. Sci*., vol. 12143, pp. 133–148, 2020.

[31] A. Alonso, R. Loredana Trotta, and A. Valli, "Coercive domain decomposition algorithms for advection-diffusion equations and systems", *J. Comput. Appl. Math*., vol. 96, no. 1, pp. 51–76, 1998.

[32] K. Pingali, D. Nguyen, M. Kulkarni, M. Burtscher, M.A. Hassaan, R. Kaleem, T.-H. Lee, A. Lenharth, R. Manevich, M. Mendez-Lojo, D. Prountzos, and X. Sui, "The tao of parallelism in algorithms", *SIGPLAN*, vol. 46, 2011, doi: 10.1145/1993316.1993501.

[33] A. Takhirov, R. Frolov, and P. Minev, "Direction splitting scheme for Navier-Stokes-Boussinesq system in spherical shell geometries", arXiv:1905.02300, 2019.