Fast Isogeometric Analysis Simulations of a Process of Air Pollution Removal by Artificially Generated Shock Waves.

Krzysztof Misan, Maciej Kozieja
Weronika Ormaniec, Adam Kania,
Marcin Łoś, Jacek Leszczyński, Maciej Paszyński

AGH University of Science and Technology, Kraków, Poland

IGA-ADS Software

- Linear $\mathcal{O}(N)$ cost Alternating-Directions Solver
- Explicit time-integration scheme
- Some implicit time-integration schemes suitable for splitting (Peaceman-Reachford, Douglass-Gunn, generalized α, Adams-Moultons, BDF)
- 2D / 3D
- C++ and GALOIS library for parallelization

Marcin Łoś, Maciej Woźniak, Maciej Paszyński, Andrew Lenharth, Mohammed Amber Hassaan, Keshav Pingali,
IGA-ADS: Isogeometric Analysis Finite Element Method using Alternating-Directions solver,
Computer Physics Communications 217 (2017):99–116
doi.org/10.1016/j.cpc.2017.02.023

github.com/marcinlos/iga-ads

Strong form equation

We employ advection-difusion equation to model the concentration of the water vapor forming a cloud, mixed with the pollution particles.

Strong form equation

$$\frac{\partial u}{\partial t} = f + (b \cdot \nabla)u + \nabla \cdot (K \nabla u) \text{ in } \Omega \times (0, T] \quad (1)$$

$$\nabla u \cdot n = 0 \text{ in } \partial\Omega \times (0, T] \quad (2)$$

$$u = u_0 \text{ in } \Omega \times 0 \quad (3)$$

where u is the concentration scalar field, b is the assumed air velocity vector field, $K = \begin{pmatrix} K_{11}00 \\ 0K_{22}0 \\ 00K_{33} \end{pmatrix}$ is an isotropic diffusion matrix, and f is the source term.

Explicit method and weak form formulations

We formulate the explicit method as:

Explicit method formulation

$$\frac{u^{t+1} - u^t}{dt} = f^t + (b \cdot \nabla)u^t + \nabla \cdot (K \nabla u^t) + cu^t \tag{4}$$

we derive the weak formulation, using test functions v, integrating by parts the diffusion term:

Weak form formulation

$$\left(u^{t+1}, v\right) = (u^t, v) + dt\,(f^t, v) - dt\,(K \nabla u^t, \nabla v) + dt\,(b \cdot \nabla u^t, v)\ \forall v \in V$$

B-spline discretization

We discretize with B-splines over $\Omega = [0,1]^3$:

$$u^{t+1} = \sum_{i=1,\ldots,N_x;j=1,\ldots,N_y;k=1,\ldots,N_z} u^{t+1}_{ijk} B^x_i B^y_j B^z_k;$$

$$u^{t} = \sum_{i=1,\ldots,N_x;j=1,\ldots,N_y;k=1,\ldots,N_z} u^{t}_{ijk} B^x_i B^y_j B^z_k$$

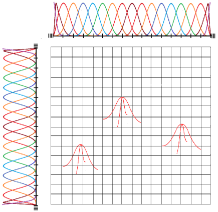


Figure: Two dimensional example of approximation with B-splines

Discrete weak formulation

B-splines for trial and test

$$\sum_{ijk} u^{t+1}_{ijk}\left(B^x_i B^y_j B^z_k, B^x_l B^y_m B^z_n\right) = \sum_{ijk} u^{t}_{ijk}\left(B^x_i B^y_j B^z_k, B^x_l B^y_m B^z_n\right) -$$

$$dt[\sum_{ijk} u^{t}_{ijk}\left(K\frac{\partial B^x_i}{\partial x} B^y_j B^z_k, \frac{\partial B^x_l}{\partial x} B^y_m B^y_n\right) - \sum_{ijk} u^{t}_{ijk}\left(KB^x_i \frac{\partial B^y_j}{\partial y} B^z_k, B^x_l \frac{\partial B^y_m}{\partial y} B^z_n\right) -$$

$$\sum_{ijk} u^{t}_{ijk}\left(KB^x_i B^y_j \frac{\partial B^z_k}{\partial z}, B^x_l B^x_m \frac{\partial B^z_n}{\partial z}\right)] + dt\sum_{ijk} u^{t}_{ijk}\left(b\frac{\partial B^x_i}{\partial x} B^y_j B^z_k, B^x_l B^y_m B^z_n\right) +$$

$$\sum_{ijk} u^{t}_{ijk}\left(bB^x_i \frac{\partial B^y_j}{\partial y} B^z_k, B^x_l B^y_m B^z_n\right) + \sum_{ijk} u^{t}_{ijk}\left(bB^x_i B^y_j \frac{\partial B^z_k}{\partial z}, B^x_l B^y_m B^z_n\right)$$

$$+ \left(f^t, B^x_l B^y_m B^z_n\right)$$

$l = 1, ..., N_x; m = 1, ..., N_y; n = 1, ..., N_z$

where $(u, v) = \int_\Omega u(x, y, z; t)v(x, y, z; t)dxdydz$ for a fixed t.

Kronecker product

In general, Kronecker product matrix $\mathcal{M} = \mathcal{M}^x \otimes \mathcal{M}^y \otimes \mathcal{M}^z$ over 3D domain $\Omega = \Omega_x \times \Omega_y \times \Omega_z$ is defined as:

Kronecker product

$$\mathcal{M}_{ijklmn} = \int B_i^x B_j^y B_k^z B_l^x B_m^y B_n^y dxdydz =$$

$$\int B_i^x B_l^x dx \int B_j^y B_m^y dy \int B_k^z B_n^z dz = \mathcal{M}_{il}^x \mathcal{M}_{jm}^y \mathcal{M}_{kn}^z$$

Due to the fact, that one-dimensional matrices discretized with B-spline functions are banded and they have $2p+1$ diagonals (where p stands for the order of B-splines), since:

$$(\mathcal{M})^{-1} = (\mathcal{M}^x \otimes \mathcal{M}^y \otimes \mathcal{M}^z)^{-1} = (\mathcal{M}^x)^{-1} \otimes (\mathcal{M}^y)^{-1} \otimes (\mathcal{M}^z)^{-1}$$

we can solve our system in a linear computational cost.

Cloud formation and thermal inversion

The scalar field u represents the water vapor forming a cloud, mixed with the pollution particles.

Strong form equation

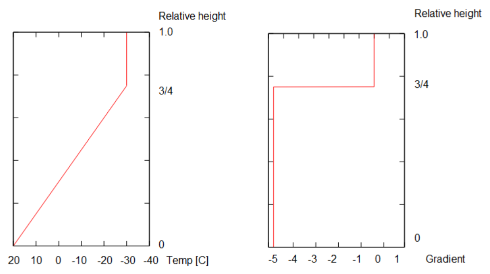
$$\frac{\partial u(x,y,z;t)}{\partial t} + \frac{\partial T(y;t)}{\partial y}\frac{\partial u(x,y,z;t)}{\partial y}$$
$$-K_x \frac{\partial^2 u(x,y,z;t)}{\partial x^2} - K_y \frac{\partial^2 u(x,y,z;t)}{\partial y^2} - K_z \frac{\partial^2 u(x,y,z;t)}{\partial z^2} = f(x,y,z;t)$$
$$(x,y,z;t) \text{ in } \Omega \times (0,T]$$
$$\nabla u(x,y,z;t) \cdot n(x,y,z) = 0, \quad (x,y,z;t) \text{ in } \partial\Omega \times (0,T]$$
$$u(x,y,z;0) = u_0 \text{ in } \Omega \times 0$$

where u is the concentration scalar field, where the advection is driven by the temperature gradient in the vertical direction

Temperature gradient and diffusion

Temperature gradient

$$\frac{\partial T(y;t)}{\partial y} = \begin{cases} 0 \text{ for } y > \frac{3}{4} & -5 \text{ for } y <= \frac{3}{4} \end{cases} \quad (5)$$



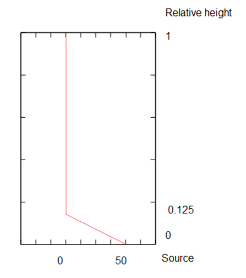
Diffusion

$K_x = K_y = 1.0$ the horizontal diffusion, $K_z = 0.1$ the vertical diffusion

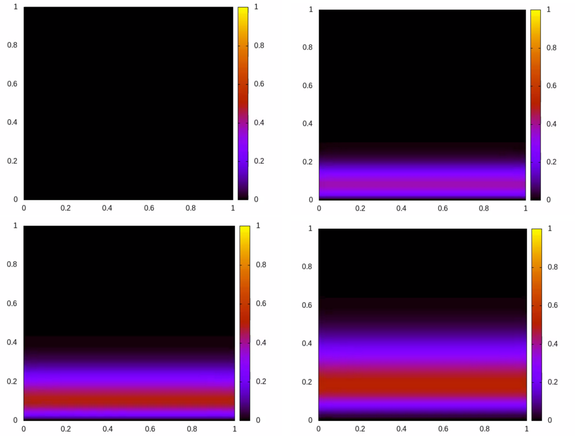
Source term

Source term

$$f(x, y; t) = \begin{cases} 50 - 400y \text{ for } y < 0.125; \ 0 \text{ otherwise} \end{cases} \quad (6)$$



Simulation results



Simulation results

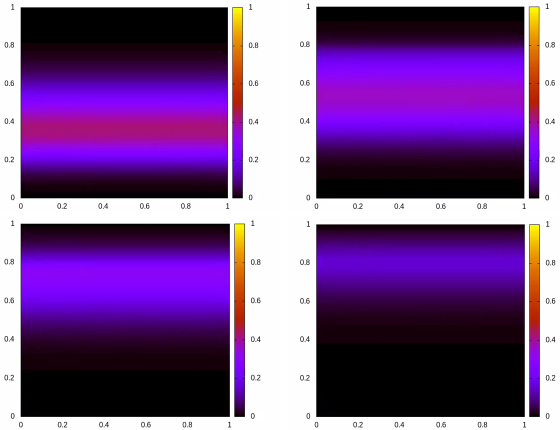


Figure: Formulation of the cloud through thermal inversion

Reducing air pollution by the use of artificial shock wave

- Due to the ground inversion pollutants can be trapped at low altitude causing damage to humans and other living organisms.
- Vertical air movement can break the inversion layer and introduce temporary upward mixing effect which in turn cause decrease in the pollution level at the lower altitudes.
- Shock wave generator [Leszczyński et.al. The method of reducing dust accumulation in the smog layer, which is the inversion layer. European Patent Office EP20217680 (2020)] The explosions of the acetylene-air mixture reaches a pressure of about 1 MPa. One shock wave every 10 seconds.

Figure: The shock wave generator. The drop of the concentration of PM25

Shock wave generation

```
template <int t_begin, int t_end>
double clock(double t) {
if (t < t_begin || t > t_end) return 0;
t = (t - t_begin) / (t_end - t_begin);
return max(sin(2*Π*t) * cos(t), 0); }
```

```
template <int t_begin, int t_end>
double cannon(double x, double y, double z, int
iter){
if ((x > 0.2 && x < 0.8) && (y > 0.2 && y < 0.8)) {
double t = clock<t_begin, t_end>(iter);
if (t == 0) return 0;
return 300 * (1 - z) * max(cos(2*Π*x)* cos(2*Π*y),
0)*t; } }
```

```
cannon<0, 300>(x, y, z, iter)
cannon<200, 500>(x, y, z, iter)
cannon<400, 700>(x, y, z, iter)
cannon<600, 900>(x, y, z, iter)
```

Simulation results

Front view. Cross section OXZ

40x40x40 elements, quadratic B-splines.

Two hours of computation on

i7 8700 8th generation, 12 cores (6 hyperthreading) 16 GB RAM.

Time step $dt = 10^{-5}$, 30,000 time steps.

Top view. Cross section OXY

40x40x40 elements, quadratic B-splines.

Two hours of computation on

i7 8700 8th generation, 12 cores (6 hyperthreading) 16 GB RAM.

Time step $dt = 10^{-5}$, 30,000 time steps.

Krzysztof Misan, Weronika Ormaniec, Adam Kania, Maciej Kozieja, Marcin Łoś, Dominik Gryboś, Jacek Leszczyński, Fast isogeometric analysis simulations of a process of air pollution removal by artificially generated shock waves **Lecture Notes in Computer Science** 13352 (2022) 298-311 *International Conference on Computational Science ICCS 2022*

Krzysztof Misan, Maciej Kozieja, Anna Paszyńska, Maciej Paszyński, Prototype of cooperative computational framework for incorporating air pollution prognosis in urban design, **Lecture Notes in Computer Science** 13492 (2022) 1611-3349 *International Conference on Cooperative Design, Visualization and Engineering, CDVE 2022*