3.36pt

Open source JAVA implementation of the parallel multi-thread alternating direction isogeometric L2 projections solver for material science simulations

Grzegorz Gurgul (AGH)
Maciej Woźniak (AGH)
Marcin Łoś (AGH)
Danuta Szeliga (AGH)
Maciej Paszyński (AGH)

KomPlasTech 2017, January 15-18, 2017, Zakopane, Poland

Agenda

- Backgroud
- Isogeometric L2 projections algorithm (Maciej Paszyński)
- JAVA implementation (Grzegorz Gurgul)
- Conclusions

Background

- Isogeometric L2 projections algorithm

Proposed by prof. Victor Calo: L. Gao, V.M. Calo, *Fast Isogeometric Solvers for Explicit Dynamics*, **Computer Methods in Applied Mechanics and Engineering** (2014).

- Applications to time-dependent problems

Linear elasticity (Fortran sequential): M. Łoś, M. Woźniak, M. Paszyński, L. Dalcin, V.M. Calo, Dynamics with Matrices Possessing Kronecker Product Structure, **Procedia Computer Science** 51 (2015) 286-295

Tumor growth simulations (C++ sequential): M. Łoś, M. Paszyński, A. Kłusek, W. Dzwinel, Application of fast isogeometric L2 projection solver for tumor simulations, **Computer Methods in Applied Mechanics and Engineering** (2017)

Non-linear flow in heterogenous media (Fortran+MPI, parallel): M. Woźniak, M. Łoś, M. Paszyński, L. Dalcin, V. Calo, Parallel fast isogeometric solvers for explicit dynamics, **Computing and Informatics** (2017)

A new open source JAVA code for shared memory machines

Isogeometric L2 projections

In general: non-stationary problem of the form

$$\partial_t u - \mathcal{L}(u) = f(x,t)$$

with some initial state u_0 and boundary conditions

$\mathcal{L}$ – well-posed linear spatial partial differential operator
Weak form: $(\partial_t u + \mathcal{L}u, v)_{L^2} = (f, v)_{L^2}$

Discretization:

- spatial discretization: isogeometric finite element method

$$(\partial_t u_h + \mathcal{L}u_h, v_h)_{L^2} = (f, v_h)_{L^2}$$

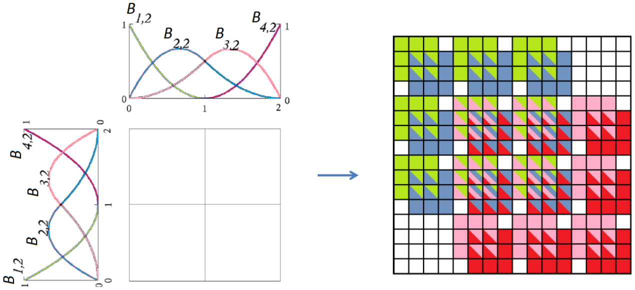
$$u_h = \sum_i \phi_i,\ v_h \in V_h = span\{\phi_1, \ldots, \phi_n\} \text{ (B-splines)}$$

- time discretization with explicit method e.g. forward Euler scheme

$$\mathcal{M}u_h^{(t+1)} = \mathcal{M}u_h^{(t)} + \Delta t \left(\mathcal{L}u_h^{(t)} + \mathcal{F}\right)$$

$$(u_h^{(t+1)}, v_h)_{L^2} = (u_h^{(t)} - \Delta t * \mathcal{L}u_h^{(t)} + \Delta t * \mathcal{F}, v_h)_{L^2}$$

L^2 projections – tensor product basis



Isogeometric basis functions:

- 1D B-splines basis $B_1(x), \ldots, B_n(x)$
- higher dimensions: tensor product basis
  $B_{i_1 \cdots i_d}(x_1, \ldots, x_d) \equiv B_{i_1}^{x_1}(x_1) \cdots B_{i_d}^{x_d}(x_d)$

Gram matrix of B-spline basis on 2D domain $\Omega = \Omega_x \times \Omega_y$:

$$\mathcal{M}_{ijkl} = (B_{ij}, B_{kl})_{L^2} = \int_\Omega B_{ij} B_{kl} \, \mathrm{d}\Omega$$

Standard multi-frontal solver: $O(N^{1.5})$ in 2D, $O(N^2)$ in 3D

L^2 projections – tensor product basis

Isogeometric basis functions:

- 1D B-splines basis $B_1(x), \ldots, B_n(x)$
- higher dimensions: tensor product basis
  $B_{i_1 \cdots i_d}(x_1, \ldots, x_d) \equiv B_{i_1}^{x_1}(x_1) \cdots B_{i_d}^{x_d}(x_d)$

Gram matrix of B-spline basis on 2D domain $\Omega = \Omega_x \times \Omega_y$:

$$\mathcal{M}_{ijkl} = (B_{ij}, B_{kl})_{L^2} = \int_\Omega B_{ij} B_{kl} \, \mathrm{d}\Omega$$

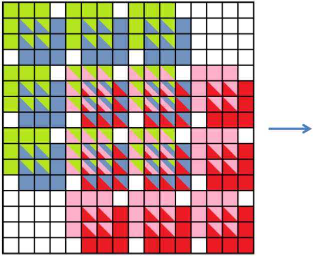
$$= \int_\Omega B_i^x(x) B_j^y(y) B_k^x(x) B_l^y(y) \, \mathrm{d}\Omega$$

$$= \int_\Omega (B_i B_k)(x) \, (B_j B_l)(y) \, \mathrm{d}\Omega$$

$$= \left(\int_{\Omega_x} B_i B_k \, \mathrm{d}x \right) \left(\int_{\Omega_y} B_j B_l \, \mathrm{d}y \right)$$

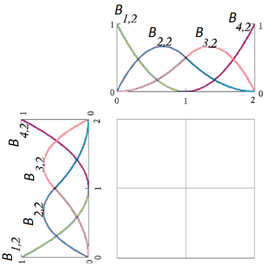
$$= \mathcal{M}_{ik}^x \mathcal{M}_{jl}^y$$

$$\mathcal{M} = \mathcal{M}^x \otimes \mathcal{M}^y \quad \text{(Kronecker product)}$$



$$\begin{bmatrix} A_{11} & A_{12} & \cdots & 0 \\ A_{21} & A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{nn} \end{bmatrix} \begin{bmatrix} y_{11} & y_{21} & \cdots & y_{m1} \\ y_{12} & y_{22} & \cdots & y_{m1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{1n} & y_{2n} & \cdots & y_{mn} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{21} & \cdots & b_{m1} \\ b_{12} & b_{22} & \cdots & b_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1n} & b_{2n} & \cdots & b_{mn} \end{bmatrix}$$

$$\begin{bmatrix} B_{11} & B_{12} & \cdots & 0 \\ B_{21} & B_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_{mm} \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ x_{21} & \cdots & x_{2n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mn} \end{bmatrix}$$



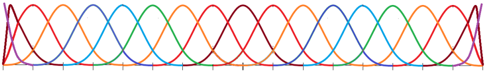
$$(u_h^{(t+1)}, v_h)_{L^2} = (\underbrace{u_h^{(t)} - \Delta t * \mathcal{L} u_h^{(t)} + \Delta t * \mathcal{F}}_{\boldsymbol{F}}, v_h)_{L^2}$$

$$b_{ij} = \int_\Omega B_{ij} \; \boldsymbol{F} \; d\Omega = \int_\Omega B_{i,2} \; B_{j,2} \; \boldsymbol{F} \; d\Omega$$

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & 0 \\ A_{21} & A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{nn} \end{bmatrix} \begin{bmatrix} y_{11} & y_{21} & \cdots & y_{m1} \\ y_{12} & y_{22} & \cdots & y_{m1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{1n} & y_{2n} & \cdots & y_{mn} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{21} & \cdots & b_{m1} \\ b_{12} & b_{22} & \cdots & b_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1n} & b_{2n} & \cdots & b_{mn} \end{bmatrix}$$

$$\begin{bmatrix} B_{11} & B_{12} & \cdots & 0 \\ B_{21} & B_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_{mm} \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ x_{21} & \cdots & x_{2n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mn} \end{bmatrix}$$

Gram matrix of tensor product basis



B-spline basis functions have **local support** (over $p+1$ elements)
$\mathcal{M}^x$, $\mathcal{M}^y$, $\ldots$ – banded structure
$\mathcal{M}^x_{ij} = 0 \iff |i-j| > 2p+1$
Exemplary basis functions and matrix for cubics

$$\begin{bmatrix}
(B_1,B_1)_{L^2} & (B_1,B_2)_{L^2} & (B_1,B_3)_{L^2} & (B_1,B_4)_{L^2} & 0 & 0 & \cdots & 0 \\
(B_2,B_1)_{L^2} & (B_2,B_2)_{L^2} & (B_2,B_3)_{L^2} & (B_2,B_4)_{L^2} & (B_2,B_5)_{L^2} & 0 & \cdots & 0 \\
(B_3,B_1)_{L^2} & (B_3,B_2)_{L^2} & (B_3,B_3)_{L^2} & (B_3,B_4)_{L^2} & (B_3,B_5)_{L^2} & (B_3,B_6)_{L^2} & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\
0 & 0 & \cdots & (B_n,B_{n-3})_{L^2} & (B_n,B_{n-2})_{L^2} & (B_n,B_{n-1})_{L^2} & (B_n,B_n)_{L^2} &
\end{bmatrix}$$

Alternating Direction Solver – 2D

Two steps – solving systems with **A** and **B** in different *directions*

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & 0 \\ A_{21} & A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{nn} \end{bmatrix} \begin{bmatrix} y_{11} & y_{21} & \cdots & y_{m1} \\ y_{12} & y_{22} & \cdots & y_{m1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{1n} & y_{2n} & \cdots & y_{mn} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{21} & \cdots & b_{m1} \\ b_{12} & b_{22} & \cdots & b_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1n} & b_{2n} & \cdots & b_{mn} \end{bmatrix}$$

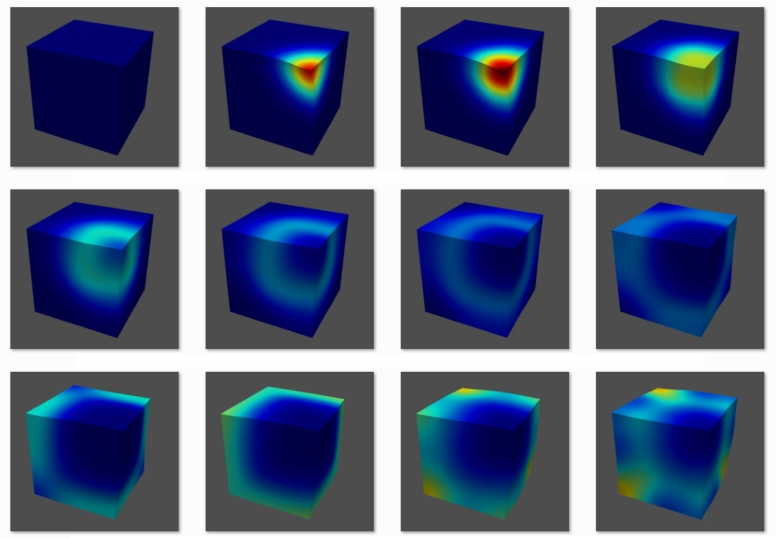
$$\begin{bmatrix} B_{11} & B_{12} & \cdots & 0 \\ B_{21} & B_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_{mm} \end{bmatrix} \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ x_{21} & \cdots & x_{2n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mn} \end{bmatrix}$$

Two one dimensional problems with multiple RHS:

- $n \times n$ with m right hand sides $\rightarrow O(n * m) = O(N)$
- $m \times m$ with n right hand sides $\rightarrow O(m * n) = O(N)$

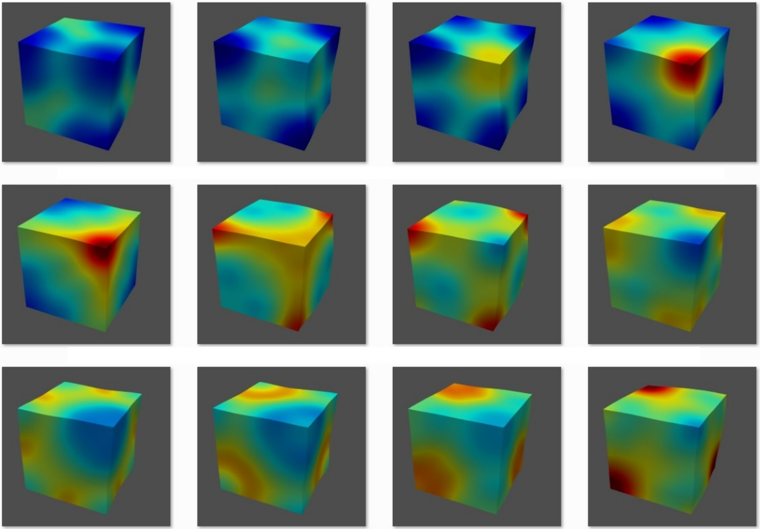
Linear computational cost $O(N)$

Hitting elastic material (1/2)



Snapshoots from the simulation

Hitting elastic material (2/2)



Snapshoots from the simulation

JAVA implementation (1/10)

TO DO

Our research is funded by Polish National Science Centre
grant no. DEC-2015/19/B/ST8/01064