Parallel RBF-FD solution of the Boussinesq's problem

Jure Slak, Gregor Kosec

"Jožef Stefan" Institute, Parallel and Distributed Systems Laboratory

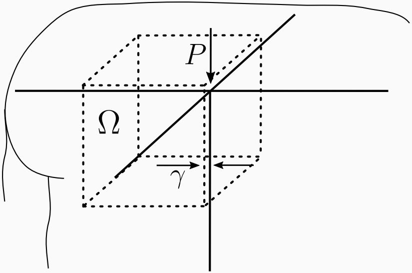
5. 6. 2019, Pareng 2019

Scope

JSI

1. Problem definition
2. Solution procedure
3. Results

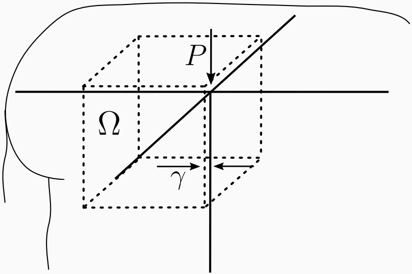
Problem definition



Cauchy-Navier equation

$$(\lambda + \mu)\nabla(\nabla \cdot \vec{u}) + \mu \nabla^2 \vec{u} = \vec{f}$$

in domain $\Omega = [-1, -\gamma]^3$ with Dirichlet boundary conditions.



Closed form solution in cylindrical coordinates (r, ϑ and z):

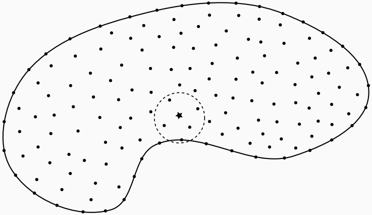
$$u_r = \frac{Pr}{4\pi\mu}\left(\frac{z}{R^3} - \frac{1-2\nu}{R(z+R)}\right), \qquad u_\vartheta = 0, \qquad u_z = \frac{P}{4\pi\mu}\left(\frac{2(1-\nu)}{R} + \frac{z^2}{R^3}\right),$$

$$\sigma_{rr} = \frac{P}{2\pi}\left(\frac{1-2\nu}{R(z+R)} - \frac{3r^2 z}{R^5}\right), \qquad \sigma_{\vartheta\vartheta} = \frac{P(1-2\nu)}{2\pi}\left(\frac{z}{R^3} - \frac{1}{R(z+R)}\right),$$

$$\sigma_{zz} = -\frac{3Pz^3}{2\pi R^5}, \qquad \sigma_{rz} = -\frac{3Prz^2}{2\pi R^5}, \qquad \sigma_{r\vartheta} = 0, \qquad \sigma_{\vartheta z} = 0.$$

Domain discretization:

- Points x_i on the boundary and in the interior
- Point neighborhoods $N(x_i)$



Classical Finite Differences:

$$u''(x_i) \approx \frac{1}{h^2}u(x_{i-1}) - \frac{2}{h^2}u(x_i) + \frac{1}{h^2}u(x_{i+1})$$

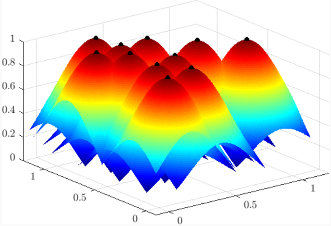
Generalized Finite Differences:

$$(\mathcal{L}u)(x_i) \approx \sum_{x_j \in N(x_i)} w_j^i u(x_j)$$

+ exactness for a certain set of functions (e.g. monomials)

Exactness is imposed for Radial Basis Functions

- Given nodes $X = \{x_1, \ldots, x_n\}$ and a radial function $\varphi = \varphi(r)$
- Generate $\{\varphi_i := \varphi(\| \cdot - x_i \|), x_i \in X\}$



Imposing exactness of

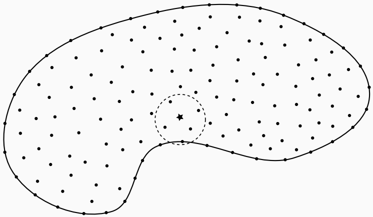
$$(\mathcal{L}u)(x_i) \approx \sum_{x_j \in N(x_i)} w^i_j u(x_j)$$

for each φ_j for $x_j \in N(x_i)$, we get

$$\begin{bmatrix} \varphi(\|x_{j_1} - x_{j_1}\|) & \cdots & \varphi(\|x_{j_{n_i}} - x_{j_1}\|) \\ \vdots & \ddots & \vdots \\ \varphi(\|x_{j_1} - x_{j_{n_i}}\|) & \cdots & \varphi(\|x_{j_{n_i}} - x_{j_{n_i}}\|) \end{bmatrix} \begin{bmatrix} w^i_{j_1} \\ \vdots \\ w^i_{j_{n_i}} \end{bmatrix} = \begin{bmatrix} (\mathcal{L}\varphi_{j_1})(x_i) \\ \vdots \\ (\mathcal{L}\varphi_{j_{n_i}})(x_i) \end{bmatrix}$$

Problem:

$$\mathcal{L}u = f \quad \text{on } \Omega,$$
$$u = u_0 \quad \text{on } \partial\Omega,$$



1. Discretize domain Ω
2. Find neighborhoods $N(x_i)$
3. Compute weights $\boldsymbol{w}^i$ for approximation of $\mathcal{L}$ over $N(x_i)$
4. Assemble weights in a sparse system $Wu = f$
5. Solve the sparse system $Wu = f$
6. Approximate/interpolate the solution

Implementation

JSI

All computations were done using open source Medusa library.

Medusa

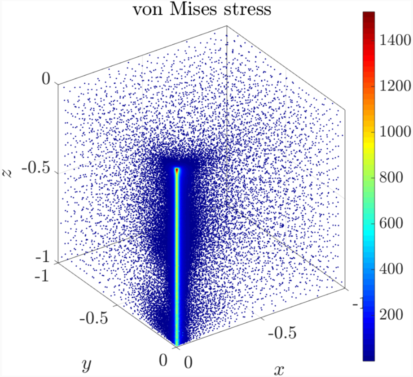
Coordinate Free Meshless Method implementation

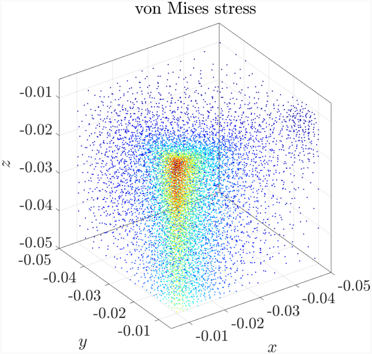
http://e6.ijs.si/medusa/

```
for (int i : domain.interior()) {
    (lam+mu)*op.graddiv(i) + mu*op.lap(i) = 0.0;
}
for (int i : domain.boundary()) {
    op.value(i) = analytical(domain.pos(i));
}
solver.compute(M);
VectorField3d u = solver.solve(rhs);
```

Pardiso sparse solver was used for parallel system solution.

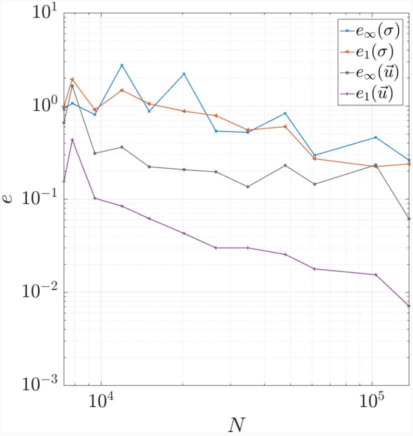
Solution for $\gamma = 0.01$.

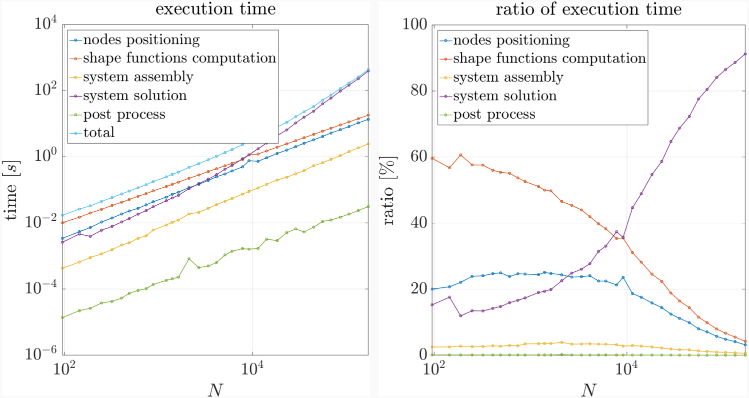




Convergence

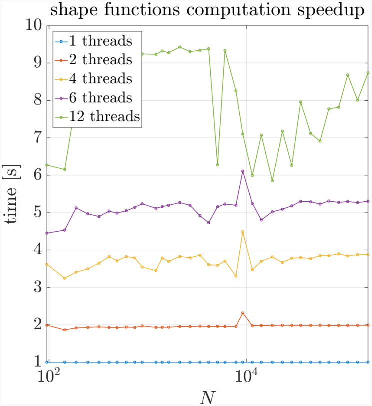
Measuring stress and displacement errors for various N.

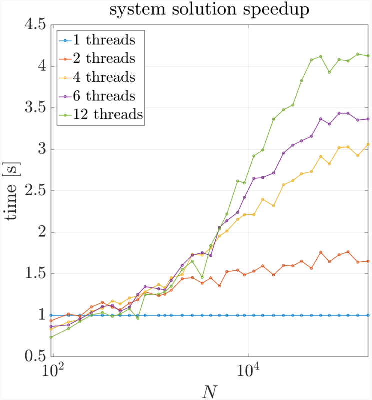




Shared memory parallelization of shape computation and sparse system solution.

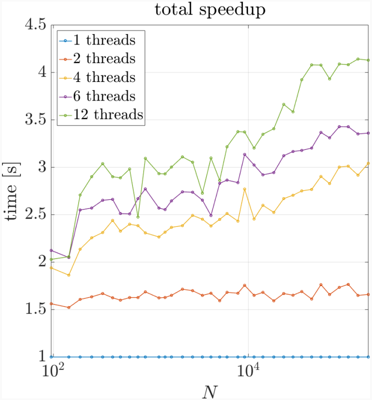
Speedup of shape function computation (left) and speedup of system solution (right)

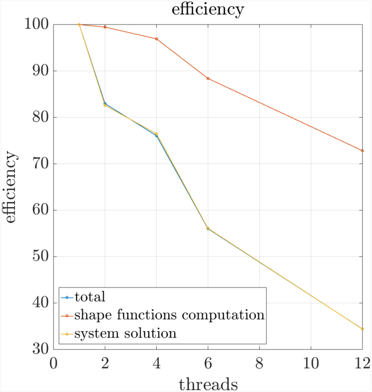




JSI

Total speedup (left) and efficiency (right)





Final remarks

Medusa

Coordinate Free Meshless Method implementation

`http://e6.ijs.si/medusa/`

Thank you for your attention!

Acknowledgments: FWO Lead Agency project: G018916N Multi-analysis of fretting fatigue using physical and virtual experiments and the ARRS research core funding No. P2-0095.