Employing Adaptive Finite Elements to Model Squeezing of a Layered Material in 3D

Marcin Sieniek, Piotr Gurgul, and Maciej Paszyński

***Abstract*—In this paper we employ 3D hp-adaptive finite element method (hp-FEM) to model the behavior of a squeezed layered material. Under a moderate pressure, the linear elasticity model can be used to imitate the process. Thanks to hp-adaptivity, only the regions where the error rate is high are refined, making sure that all peculiarities are automatically localized.**

***Index Terms*—Finite element method, hp adaptivity, linear elasticity.**

I. HP-ADAPTIVE FINITE ELEMENT METHOD

Finite element method (FEM) has long been an important tool for modeling a variety of processes including applications in mechanics [1], material science [2], [3], geology [4]-[6] and nano- engineering [7].

The hp-FEM is the most sophisticated version of the mesh adaptive algorithms [1], where elements are automatically h-refined (broken into smaller elements) or p-refined (polynomial order of approximating base is increased over selected elements). The decisions about finite elements that need to be refined are made iteratively based on the *a posteriori* error decrease estimate.

A. Finite Element And Its Shape Functions

In 1D, approximation base consists of basis functions defined with formulas (1). The example of such functions on element $[0,1]$ is shown in Fig. 1.

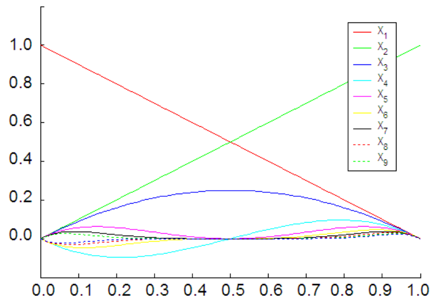

Fig. 1. 1D Shape functions on element [0, 1]

Manuscript received November 14, 2012; revised March 8, 2013. This work was supported in part by the Polish National Science Center funds allocated with the decision DEC-2011/03/N/ST6/01397 and in part by Polish National Science Center grant no. NN519447739. The work of the second author was partly supported by The European Union by means of European Social Fund, No. UDA – POKL.04.01.01-00-367/08-00.

M. Sieniek and P. Gurgul are with the AGH University of Science and Technology (e-mails: msieniek@agh.edu.pl, pgurgul@agh.edu.pl).

M. Paszyński is with the AGH University of Science and Technology (e-mail: paszynsk@agh.edu.pl).

$$\hat{\chi}_1(\xi)=1-\xi$$
$$\hat{\chi}_2(\xi)=\xi$$
$$\hat{\chi}_3(\xi)=(1-\xi)\xi$$
$$\hat{\chi}_l(\xi)=(1-\xi)\xi(2\xi-1)^{l-3} \text{ for l=4,…,p+1} \quad (1)$$

In 3D, basis functions are constructed as tensor products of 1D basis functions.

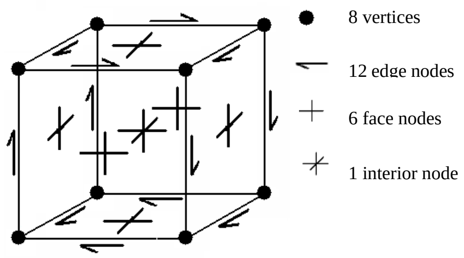

Fig. 2. 3D finite element's nodes

The basis functions are associated with nodes of a 3D finite element as shown in Fig. 2 on vertices we define the trilinear functions according to formulas (2) and as presented in Fig. 3 (order of approximation being equal to 1 in each vertex).

$$\hat{\phi}_1(\xi_1,\xi_2,\xi_3)=\hat{\chi}_1(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_1(\xi_3)$$
$$\hat{\phi}_2(\xi_1,\xi_2,\xi_3)=\hat{\chi}_2(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_1(\xi_3)$$
$$\hat{\phi}_3(\xi_1,\xi_2,\xi_3)=\hat{\chi}_2(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_1(\xi_3)$$
$$\hat{\phi}_4(\xi_1,\xi_2,\xi_3)=\hat{\chi}_1(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_1(\xi_3)$$
$$\hat{\phi}_5(\xi_1,\xi_2,\xi_3)=\hat{\chi}_1(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_2(\xi_3)$$
$$\hat{\phi}_6(\xi_1,\xi_2,\xi_3)=\hat{\chi}_2(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_2(\xi_3)$$
$$\hat{\phi}_7(\xi_1,\xi_2,\xi_3)=\hat{\chi}_2(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_2(\xi_3)$$
$$\hat{\phi}_8(\xi_1,\xi_2,\xi_3)=\hat{\chi}_1(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_2(\xi_3) \quad (2)$$

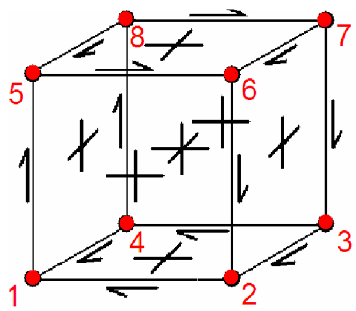

Fig. 3. Numbering scheme for basis functions associated with vertices, as defined in (2)

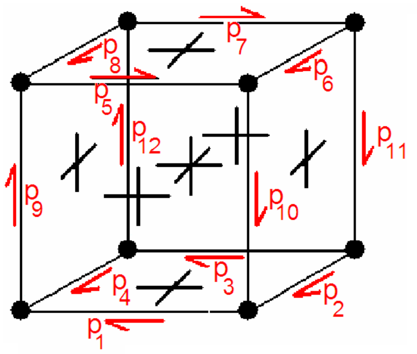

Fig. 4. Numbering scheme for basis functions associated with edges, as defined in (3)

On edges we define the functions according to formulas (3) and as presented in Fig. 4 (order of approximation being equal to $(p_i - 1)$ on i[th] edge).

$$
\begin{aligned}
&\hat{\phi}_{9,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+j}(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_1(\xi_3) && j = 1,\dots,p_1 - 1\\
&\hat{\phi}_{10,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_2(\xi_1)\hat{\chi}_{2+j}(\xi_2)\hat{\chi}_1(\xi_3) && j = 1,\dots,p_2 - 1\\
&\hat{\phi}_{11,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+j}(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_1(\xi_3) && j = 1,\dots,p_3 - 1\\
&\hat{\phi}_{12,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_1(\xi_1)\hat{\chi}_{2+j}(\xi_2)\hat{\chi}_1(\xi_3) && j = 1,\dots,p_4 - 1\\
&\hat{\phi}_{13,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+j}(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_2(\xi_3) && j = 1,\dots,p_5 - 1\\
&\hat{\phi}_{14,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_2(\xi_1)\hat{\chi}_{2+j}(\xi_2)\hat{\chi}_2(\xi_3) && j = 1,\dots,p_6 - 1\\
&\hat{\phi}_{15,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+j}(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_2(\xi_3) && j = 1,\dots,p_7 - 1\\
&\hat{\phi}_{16,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_1(\xi_1)\hat{\chi}_{2+j}(\xi_2)\hat{\chi}_2(\xi_3) && j = 1,\dots,p_8 - 1\\
&\hat{\phi}_{17,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_1(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_{2+j}(\xi_3) && j = 1,\dots,p_9 - 1\\
&\hat{\phi}_{18,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_1(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_{2+j}(\xi_3) && j = 1,\dots,p_{10} - 1\\
&\hat{\phi}_{19,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_2(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_{2+j}(\xi_3) && j = 1,\dots,p_{11} - 1\\
&\hat{\phi}_{20,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_2(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_{2+j}(\xi_3) && j = 1,\dots,p_{12} - 1
\end{aligned}
\tag{3}
$$

On faces there are 6 basis functions (of approximation order equal to $(p_{iv} - 1)(p_{ih} - 1)$) as decapitated on Fig. 5.

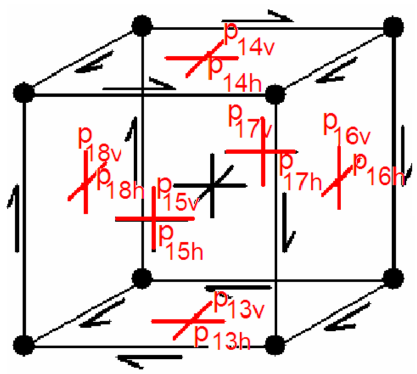

Fig. 5. Numbering scheme for basis functions associated with faces, as defined in (4)

$$
\begin{aligned}
&\hat{\phi}_{21,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+i}(\xi_1)\hat{\chi}_{1+j}(\xi_2)\hat{\chi}_1(\xi_3)\\
&i = 1,\dots,p_{h1} - 1,\ j = 1,\dots,p_{v1} - 1\\
&\hat{\phi}_{22,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+i}(\xi_1)\hat{\chi}_{1+j}(\xi_2)\hat{\chi}_2(\xi_3)\\
&i = 1,\dots,p_{h2} - 1,\ j = 1,\dots,p_{v2} - 1\\
&\hat{\phi}_{23,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+i}(\xi_1)\hat{\chi}_1(\xi_2)\hat{\chi}_{2+j}(\xi_3)\\
&i = 1,\dots,p_{h3} - 1,\ j = 1,\dots,p_{vv} - 1\\
&\hat{\phi}_{24,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+i}(\xi_1)\hat{\chi}_2(\xi_2)\hat{\chi}_{2+j}(\xi_3)\\
&i = 1,\dots,p_{h4} - 1,\ j = 1,\dots,p_{v4} - 1\\
&\hat{\phi}_{25,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_1(\xi_1)\hat{\chi}_{2+i}(\xi_2)\hat{\chi}_{2+j}(\xi_3)\\
&i = 1,\dots,p_{h5} - 1,\ j = 1,\dots,p_{v5} - 1\\
&\hat{\phi}_{26,i,j}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_2(\xi_1)\hat{\chi}_{2+i}(\xi_2)\hat{\chi}_{2+j}(\xi_3)\\
&i = 1,\dots,p_{h6} - 1,\ j = 1,\dots,p_{v6} - 1
\end{aligned}
\tag{4}
$$

Finally, there's a single interior node, which has $(p_{ix} - 1)(p_{iy} - 1)(p_{iz} - 1)$ bubble functions associated with it, with formulas shown in (5) (see Fig. 6).

$$
\begin{aligned}
&\hat{\phi}_{27,i,j,k}(\xi_1,\xi_2,\xi_3) = \hat{\chi}_{2+i}(\xi_1)\hat{\chi}_{2+j}(\xi_2)\hat{\chi}_{2+k}(\xi_3)\\
&i = 1,\dots,p_x - 1,\ j = 1,\dots,p_y - 1,\ k = 1,\dots,p_z - 1
\end{aligned}
\tag{5}
$$

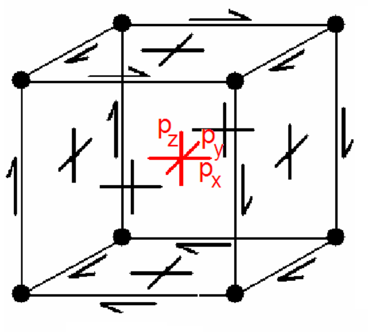

Fig. 6. Numbering scheme for basis functions associated with the interior, as defined in (5)

B. P and H Refinements

The quality of the solution depends on the size of the elements and the p_X parameters referenced above (potentially different on each element). For a given solution, the quality can be improved on element *K* by either dividing it into 8 smaller elements (this is called h-refinement) or increasing parameters (called p-refinement).

C. Adaptive Algorithm

Although we can manually adjust these parameters in certain regions of the domain where we require higher precision (by performing refinements of the mesh *a priori*), such an adjustment often turns out impractical. Instead we apply these refinements around the detected peculiarities of the computational mesh iteratively, based on *a posteriori* error estimates. The iteration is repeated until the maximum error reaches a designated threshold (desired_err according to Alg. 1).

In this paper we are applying the method described above to model 3D displacements of a layered material under pressure.

function adaptive_fem(*initial_mesh* , *desired_err*, *coef*)
coarse_mesh = *initial_mesh*

repeat
coarse_u = solve the problem on *coarse_mesh*
fine_mesh = copy *coarse_mesh*
divide each element *K* of *fine mesh* into 8 new elements (K1 .. K8)
increase polynomial order of shape functions on each element of *fine mesh* by 1
fine_u = solve the problem on *fine_mesh*
max_*err* = 0
for each element K **of** *fine mesh* **do**
K_err = compute relative decrease error rate on K
if *K_err* > max_*err* **then**
max_*err* = *K_err*
end if
end do
adapted_mesh = **new** empty_*mesh*
for each element K **of** *coarse_mesh* **do**
if *K_err* > coef * max_*err* **then**
choose a combination of refinements on element *K* from *fine_mesh* to *adapted_mesh*
else
add K from *coarse_mesh* to *adapted_mesh*
end if
end do
coarse_mesh = *adapted_mesh*
output *fine_u*
until max_*err* < *desired_err*
return (*fine_u*, *fine_mesh*)

Alg. 1 hp-adaptive finite element method algorithm

II. Problem Formulation

The process of elastic squeezing of a material can be modeled with linear elasticity [8], as shown below.

A. Strong Form

In its strong form, the problem can be described as follows: given $g_i : \Gamma_{D_i} \ni x \to g_i(x) = 0 \in R$, θ, α_{kl} and σ^0_{ij}, find $u_i : \overline{\Omega} \to R$ the displacement vector field such that

$$\sigma_{ij,j} = 0 \text{ on } \Omega \tag{6}$$

$$u_i = g_i \text{ on } \Gamma_{D_i} \tag{7}$$

where:

- σ_{ij} is the stress tensor, defined in terms of the generalized Hooke's law

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl} + \sigma^0_{ij} \tag{8}$$

c_{ijkl} are elastic coefficients (known for a given material),

- σ^0_{ij} is the initial stress,
- ε_{ij} is the strain tensor, defined to be $u_{(i,j)}$, the symmetric part of the displacement gradients

$$\varepsilon_{ij} = u_{(i,j)} = \frac{u_{i,j} + u_{j,i}}{2} \tag{9}$$

- u_i is the displacement vector.
- $u_{i,j}$ are displacement gradients.

B. Weak Form

The weak formulation is obtained by multiplying (6) by test functions $w_i \in V_i$ and integrating by parts over Ω.

$$-\int_\Omega w_{i,j}\sigma_{ij}\, d\Omega + \int_\Gamma w_i \sigma_{ij} n_j\, d\Omega = 0 \tag{10}$$

Since σ_{ij} is symmetric tensor, then $w_{i,j}\sigma_{ij} = w_{(i,j)}\sigma_{ij}$, and $w_i = 0$ on Γ, we get

$$\int_\Omega w_{(i,j)}\sigma_{ij}\, d\Omega = 0 \tag{11}$$

Finally, we substitute (8) into (11) and get

$$\int_\Omega w_{(i,j)} c_{ijkl} u_{(k,l)}\, d\Omega = -\int_\Omega w_{(i,j)}\sigma^0_{ij}\, d\Omega \tag{12}$$

since $\varepsilon_{ij} = u_{(i,j)}$.

C. Abstract Index-Free Notation

For implementation purposes, the most convenient is the following form**:**

Find $\vec{u} \in V$ such that

$$\vec{a}(w,u) = -\vec{\Sigma}(w) \text{ for all } w \in V \tag{13}$$

where

$$\vec{a}(w,u) = \int_\Omega \vec{\varepsilon}(w)^T D \vec{\varepsilon}(w) d\Omega \tag{14}$$

$$\vec{\Sigma}(w) = \int_\Omega \vec{\varepsilon}(w)^T \sigma^0 d\Omega \tag{15}$$

Here:

$$\vec{\varepsilon}(z) = \begin{Bmatrix} z_{1,1} \\ z_{2,2} \\ z_{3,3} \\ z_{2,3} + z_{3,2} \\ z_{1,3} + z_{3,1} \\ z_{1,2} + z_{2,1} \end{Bmatrix}, \quad \vec{\alpha} = \begin{Bmatrix} \alpha \\ \alpha \\ \alpha \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{16}$$

$$D = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix}$$

III. Numerical Results

The above problem has been supplied to an implementation of the hp-FEM described by the first author of this work in [8]. Material properties (Young modulus and

Poisson coefficient) have been read form a 3D bitmap, where their values were proportional to the color saturation. The derivatives were estimated using finite differentials.

It took 4 iterations of the adaptive algorithm to locate peculiarities for this problem and as an outcome we obtained the results decapitated in Fig. 7-Fig. 9.

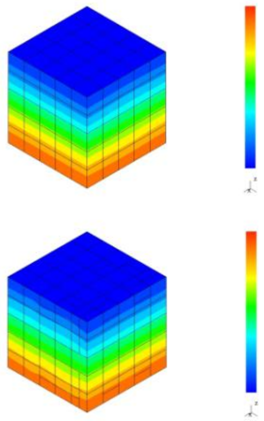

Fig. 7. Strain along Z axis after first and fourth iteration of the adaptive algorithm

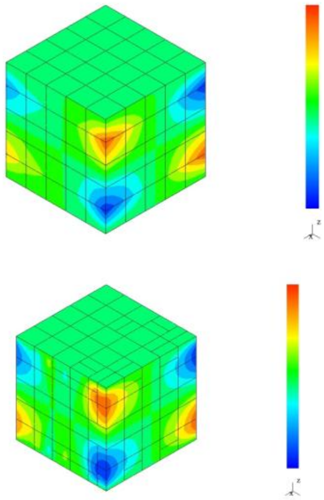

Fig. 8. Strain along X axis after first and fourth iteration of the adaptive algorithm

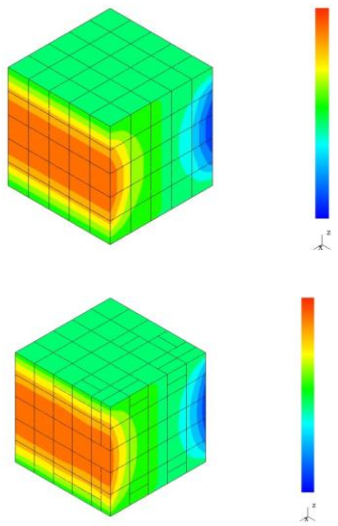

Fig. 9. Strain along Y axis after first and fourth iteration of the adaptive algorithm

As we can see, some strain components which seemed regular and insignificant at first turned out to be quite serious and complex after adaptation. This is especially true for the X axis component, but improvement of solution quality has been observed along the other axes too (see Fig. 10).

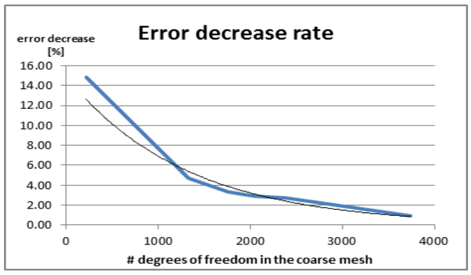

Fig. 10. Error decrease rate [%] with the increase of total number of degrees of freedom (base functions) in the coarse mesh

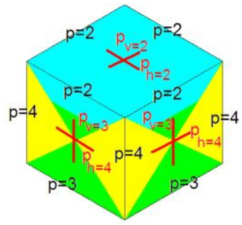

Fig. 11. Graphical notation used in Fig. 12 for various polynomial orders of approximation on element edges and faces

Such a significant error decrease in so few iterations was possible thanks to a careful choice of mesh refinements made by Alg. 1. The refinement of computational mesh in subsequent iterations has been indicated in Fig. 12.

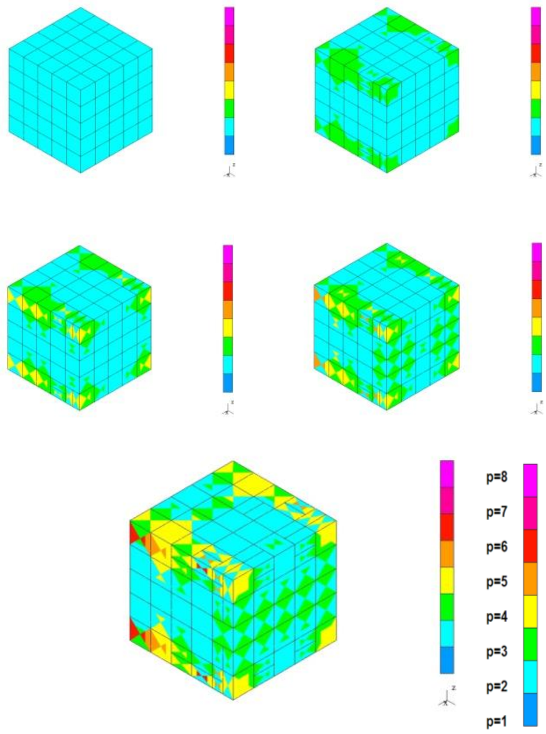

Fig. 12. Coarse computational meshes in subsequent iterations, presented using the convention from Fig. 11. Top left figure decapitates is the initial uniform mesh, whereas the bottom figure presents the final mesh after 4 iterations.

IV. CONCLUSION

In this paper, we have shown how hp-adaptive finite element method can be applied to a practical 3D problem and presented how the adaptivity affects the solution in subsequent iterations. The material coefficients we used came directly from a 3D bitmap of the processed material, where material parameters of a given part were proportional to its color. We have observed an exponential decrease of the error estimate, as expected theoretically.

REFERENCES

[1] L. Demkowicz, J. Kurtz, D. Pardo, M. Paszyński, W. Rachowicz, and A. Zdunek, "*Computing with Hp-Adaptive Finite Elements*," frontiers, *Three Dimensional Elliptic and Maxwell Problems with Applications*, CRC Press, vol. 2, 2008.

[2] M. Paszyński and P. Macioł, "Application of the Fully Automatic 3D hp Adaptive Code to Orthotropic Heat Transfer in Structurally Graded Material Properties," *Journal of Materials Processing Technology*, 2006, vol. 177, no. 1-3, pp. 68-71

[3] J. Gawąd, M. Paszyński, P. Matuszyk, and Ł. Madej, "Cellular Automata Coupled with hp-Adaptive Finite Element Method Applied to Simulation of Austenite-Ferrite Phase Transformation with a Moving Interface," *Steel Research International*, 2008, vol. 79, pp. 579-586.

[4] D. Pardo, C. Torres-Verdin, and M. Paszyński, "Simulations of 3D DC Borehole Resistivity Measurements with a Goal-Oriented hp-Finite-Element Method," Part II: Though Casing Resistivity Instruments, *Computational Geosciences*, Springer, 2008, vol. 12, no. 1, pp 83-89.

[5] D. Pardo, P. Matuszyk, and C. Torres-Verdin, "A. Mora, I. Muga, V. Calo, Simulation of Wireline Sonic Logging Measurements with Borehole-Eccentered Tools Using a High-Order Adaptive Finite Element Method," *Journal of Computational Physic*, vol. 230, no. 16, pp. 6320-6333, 2011.

[6] D. Pardo, V. Calo, C. Torres-Verdin, and M. J. Nam, "Fourier Series Expansions in non-Orthogonal System of Coordinates for the Simulation of 3D-DC Borehole Resistivity Measurements," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 21-24, pp. 1906-1925, 2008.

[7] P. Gurgul, M. Sieniek, and M. Paszyński, "Object-oriented Multiscale HP-Adaptive Finite Element Method," *Computer Methods in Materials Science*, Akapit , vol. 9, pp. 289-295, 2009

[8] T. J. R. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Courier Dover Publications, 2000.

Marcin Sieniek was born on September 3rd 1987 in Krakow, Poland. He graduated from a combined Master of Science and Bachelor of Engineering in Computer Science program at AGH University of Science and Technology in Krakow, Poland in 2011 and from a Bachelor of Business Administration in Management program from Krakow University of Economics in 2012.

Between 2009 and 2012 he worked professionally with the industry leaders such as NVIDIA Corp, Google GmbH and Bloomberg L.P. Currently he is a PhD student in Computer Science at AGH University of Science and Technology and runs a startup project aiming to enhance student collaboration in the educational process. His research interests include finite element method, projection techniques as well as management and transfer of innovations.

Mr. Sieniek has been awarded with multiple stipends and honors, including Polish Ministry of Science and Higher Education stipend (2010), Mayor of Krakow municipal award for scientific impact on the City of Krakow (2008) and a regional Sapere Auso award for special scientific achievements (2008 and 2010)

Piotr Gurgul was born on May 6th 1987 in Krakow, Poland. He graduated from a combined Master of Science and Bachelor of Engineering in Computer Science program at the AGH University of Science and Technology in Krakow, Poland in 2011 and from a Bachelor of Business Administration in Management from Krakow University of Economics in 2012.

Currently he is a Ph.D. candidate and a second year Ph.D. student at the AGH University of Science and Technology. His scientific interests focus on hp-adaptive finite element method, and adaptive solvers.

His work has been rewarded with multiple scholarships, including Polish Ministry of Science and Higher Education stipend in 2009, 2010 and 2011 and European Union Scholarship for Young Scientists. He has also completed a bunch of professional internships overseas, including Google in the US and Australia and Microsoft Corp.

Maciej Paszyński was born on Oct. 12th 1974 in Krakow, Poland. He graduated from a combined Master of Science and Bachelor of Science in computer science program at Jagiellonian University in Krakow, Poland in 1999. He defended Ph.D. in Mathematics with Applications to Computer Science at Institute for Computer Science of the Faculty of Mathematics, Physics and Computer Science of the Jagiellonian University in 2003. In 2010 he has got a habilitation in computer science at Faculty of Electrical Engineering, Automatics, Computer Science and Electronics of the AGH University of Science and Technology, Krakow, Poland.