paszynsk@agh.edu.pl

Algorithm of hp-adaptation

The automatic hp adaptation algorithm was first described in detail in the books of prof. Leszek Demkowicz, a mathematician of Polish origin, working at the University of Texas in Austin. The analysis of the mathematical properties of the hp-adaptation Algorithm was performed by prof. Ivo Babuška, an American of Czech descent, also working at the University of Texas in Austin [1] [2] [3] [4].

ALGORYTM

Algorytm 1: Algorithm of authomatic hp adaptation

1. Algorithm for automatic hp adaptation (initial mesh, required accuracy, maximum number of iterations)
2. coarse mesh = initial mesh
3. loop i = 1 up to the maximum number of iterations
4. Solve a computational problem on the current sparse mesh (e.g. bitmap projection problem or heat transport problem) by obtaining a solution $u_{h,p}$
5. coarse mesh = sparse mesh
6. Break each sparse mesh element into 4 elements (for rectangular elements in two dimensions) or 8 elements (for cubic elements in three dimensions) (possible also breaking triangular elements in two dimensions and tetrahedral elements, prismatic elements and pyramids in three dimensions) and increase the degree of polynomials by one in each direction on each element (which is natural for rectangular and cubic elements, but requires clarification for elements of a different type)
7. Solve a computational problem on the current dense mesh and obtain a solution $u_{h/2,p+1}$
8. Maximum error $K_{max} = 0$
9. Loop by elements K sparse mesh
10. For each element of the sparse mesh, estimate the relative error (the norm of the difference between the solution on the sparse and dense mesh)

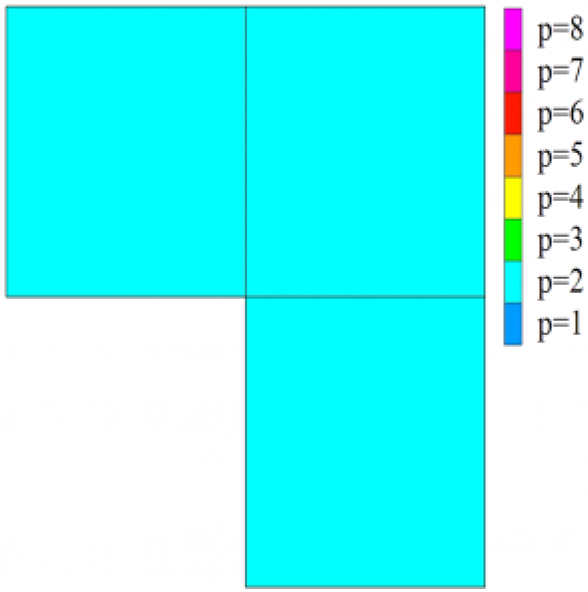
$$K_{rel} = \|u_{h,p} - u_{h/2,p+1}\|_K = \int_K \left(u_{h,p} - u_{h/2,p+1}\right)^2 + \left(\frac{\partial u_{h,p} - u_{h/2,p+1}}{\partial x}\right)^2 + \left(\frac{\partial u_{h,p} - u_{h/2,p+1}}{\partial y}\right)^2$$

1. If $K_{rel} > K_{max}$ then $K_{max} = K_{rel}$

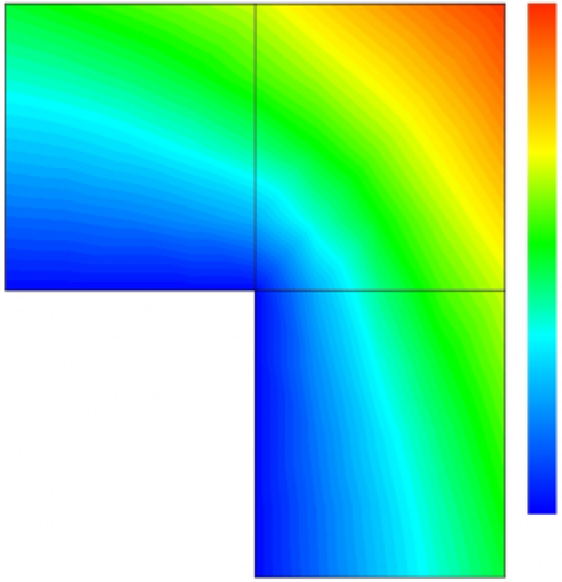
2. End of loop by elements
3. If $K_{max} >$ required accuracy, then it's over.
4. Loop by elements

K sparse mesh

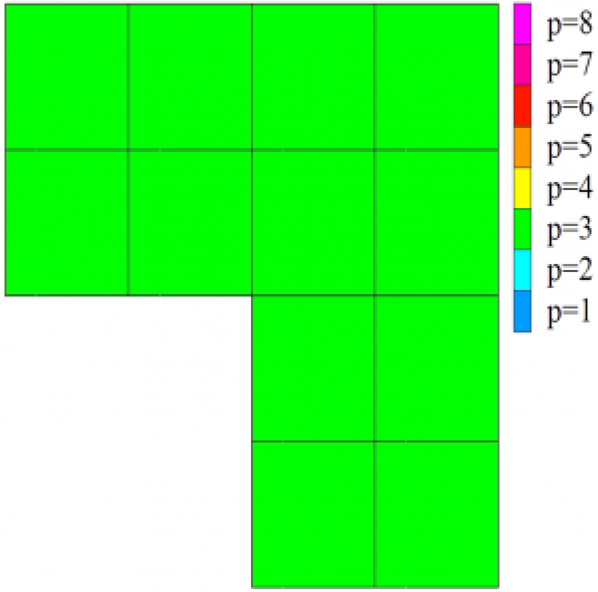
1. If $K_{rel} > 0.33K_{max}$ then select the optimal method of adapting the element from the sparse mesh and apply it to the element K sparse mesh
2. End of loop through items
3. End of iteration



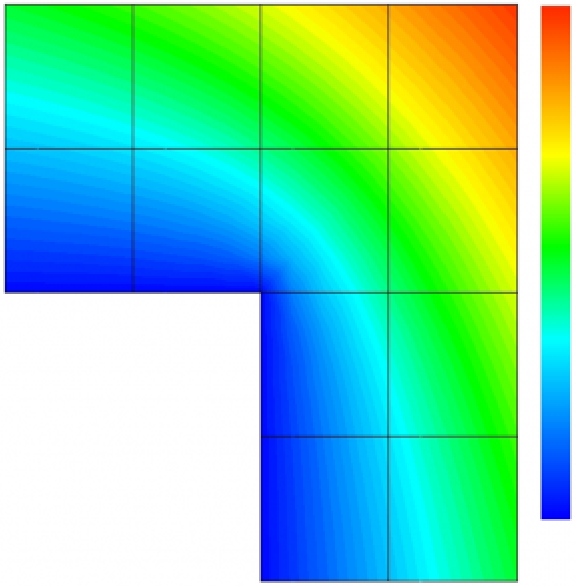
Rysunek 1: Siatka początkowa, wszystkie elementy oddzielone są C0 separatorami, na każdym elemencie siatki rozpiętęto funkcje bazowe powstałe przez iloczyn tensorowy wielomianów kwadratowych w kierunku poziomym oraz pionowym.



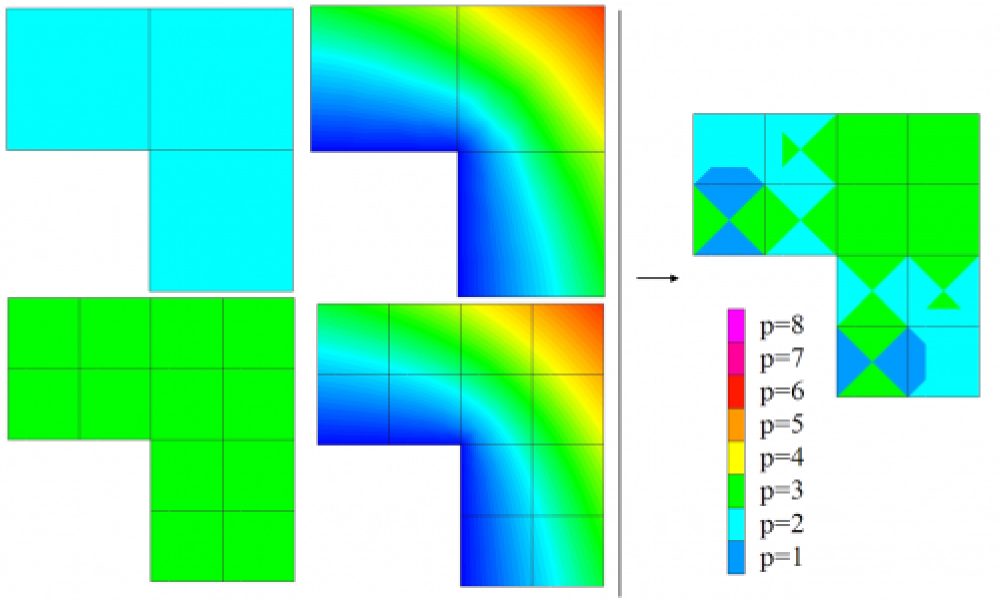
Rysunek 2: Rozwiązanie (pole skalarne temperatury) na siatce rzadkiej.



Rysunek 3: Siatka gęsta powstała przez połamanie każdego elementu siatki rzadkiej na cztery elementy oraz podniesnie stopnia wielomianów w kierunku poziomym i pionowym o jeden, na każdym elemencie.



Rysunek 4: Rozwiązanie (pole skalarne temperatury) na siatce gęstej.



Rysunek 5: Wybór optymalnej strategii adaptacji dla każdego elementu siatki rzadkiej, poprzez porównanie rozwiązań na siatkach żadkiej i gęstej.

The hp-adaptation algorithm is illustrated in the figures Rys. 1 - Rys. 5.

The convention for drawing computational meshes is as follows. Colors in the drawings Rys. 1, Rys. 3 i Rys. 5 denote the degrees of polynomials spanning the edges and interior of the elements. One polynomial degree is provided on each edge, while a horizontal degree and a vertical degree are provided on each interior of the element.

For example, in the picture

Rys. 1 all polynomials on all edges have degree 2, and all polynomials on the inside of all elements have degree 2 in the vertical and horizontal directions. In the picture Rys. 3 all polynomials on all edges have degree 3, and all polynomials on the inside of all elements have degree 3 in the vertical and horizontal directions. See the drawing Rys. 5 When viewing elements by rows from left to right, in the first row on the first element all polynomials on the edges have degree 2, except for the polynomial on the lower edge of the element which has degree 1. On the first element the polynomials in the interior have degree 2 in the vertical and horizontal directions . On the second element, on the left, top and bottom edges, the polynomials have a degree 2, while on the inside, a degree 3 in the vertical direction and a degree 2 in the horizontal direction. On the third and fourth elements, all polynomials on the edges are of degree 3. Likewise, the polynomials in the interior in the vertical and horizontal directions. In the second row of items, the first item (i.e. the fifth global) has degree 3 polynomials on the left and right edges, and a degree 1 polynomial on the top and bottom edges. Inside, the element has degree 1 polynomials in the horizontal direction and degree 3 polynomials in the vertical direction. The second element in the second line has a polynomial degree 2 in the horizontal direction and a polynomial degree 3 in the vertical direction. On the third and fourth elements, all polynomials on all edges have degree 3, and all polynomials on the inside of all elements have degree 3 in the vertical and horizontal directions. The remaining elements in the third and fourth lines have polynomials symmetrically distributed with the polynomials in the first and second lines on the first and second elements.

How is the optimal method of adapting a single element selected? Note that in the case of the adaptation hp algorithm, we have many possibilities to modify a single element:

1. Leaving the item unchanged
2. It is possible to break an element into two new elements vertically
3. It is possible to break an element into two new elements in the horizontal direction
4. It is possible to break an element into four new elements (two in the horizontal direction and two in the vertical direction)

Moreover, for each broken element it is possible to modify the degree of polynomials inside the element

1. Leave element grades unchanged
2. It is possible to raise the step inside the element horizontally

$(p_h, p_v) \rightarrow (p_h + 1, p_v)$

1. It is possible to raise the step inside the element in a vertical direction $(p_h, p_v) \rightarrow (p_h, p_v + 1)$
2. It is possible to raise the step inside the element horizontally and vertically $(p_h, p_v) \rightarrow (p_h + 1, p_v + 1)$

The steps at the edges of the modified elements are determined on the basis of the minimum rule. In other words, the degree of the polynomial at the edge divided by two adjacent elements is set as a minimum of degrees inside the elements in the appropriate direction.
For a vertical edge surrounded by two elements (p^1_h, p^2_v) and (p^2_h, p^2_v) we set the degree $p = \min\{p^1_v, p^2, v\}$.
For a horizontal edge surrounded by two elements (p^1_h, p^2_v) and (p^2_h, p^2_v) ustalamy stopień $p = \min\{p^1_h, p^2, h\}$.
So we have a lot of possibilities to adapt a single element (estimating on the basis of the possible types of adaptation we have 4 (modifications of the degree of the element without breaking the element) + 4 * 4 (breaking the element into two elements in the horizontal direction and four possibilities of modifying the degree of each element) + 4 * 4 (break an element into two elements vertically and four possibilities to modify the degree of each element) + 4 * 4 * 4 * 4 (break an element into four elements and four possibilities to modify the degree of each element). A total of 4 + 4 * 4 + 4 * 4 + 4 * 4 * 4 * 4 = 292 possibilities.
How is the decision about the type of adaptation of a single element made?
The decision is made in accordance with the following Algorithm.

ALGORYTM

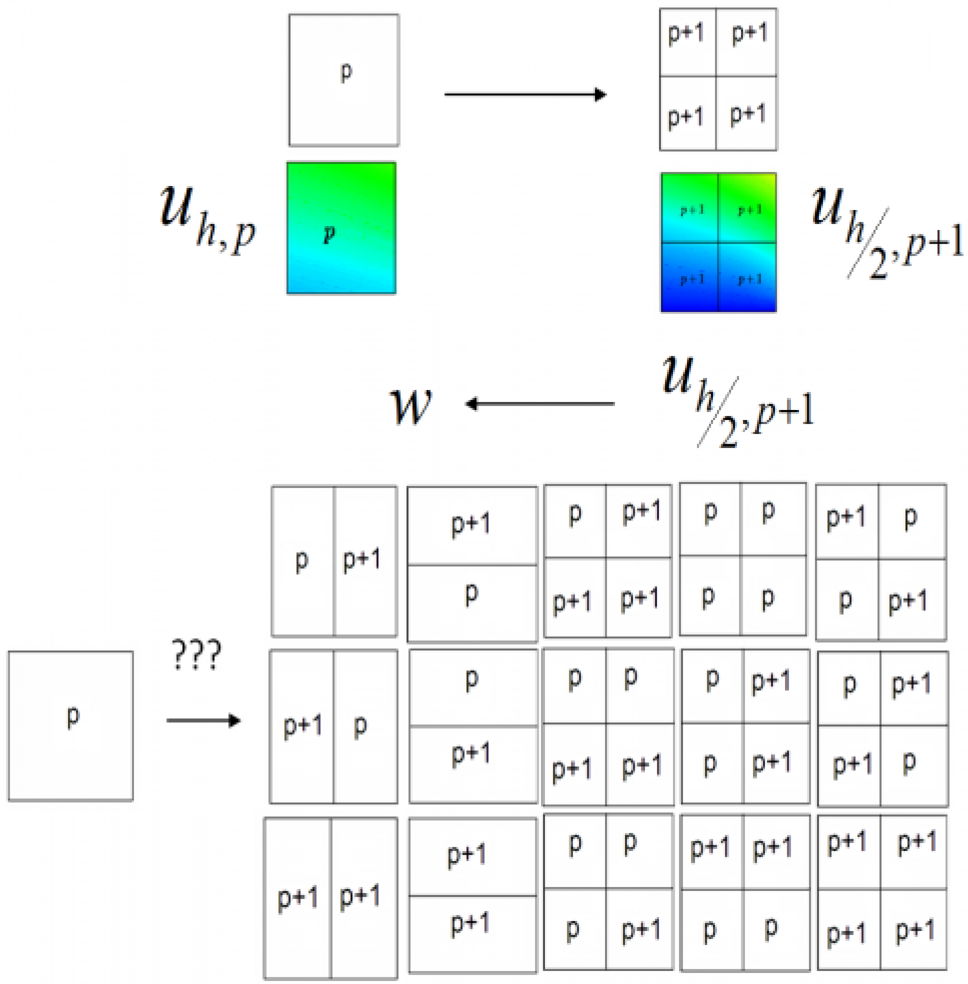
Algorytm 2: Algorithm for selecting the optimal strategy for the adaptation of the element

1. Selecting the optimal adaptation strategy for the element K (a solution on a rare mesh narrowed to an element $u_{h,p}$, dense mesh solution narrowed to the element $u_{h/2,p+1}$)
2. Loop through possible types of element adaptation from 1 to 292
3. The fastest rate of error decrease = 0
4. Optimal adaptation = 0#Copy J, element K, and make the considered adaptation on it
5. Calculate the projection

w solutions on a dense mesh $u_{h/2,p+1}$ on the element being adapted J

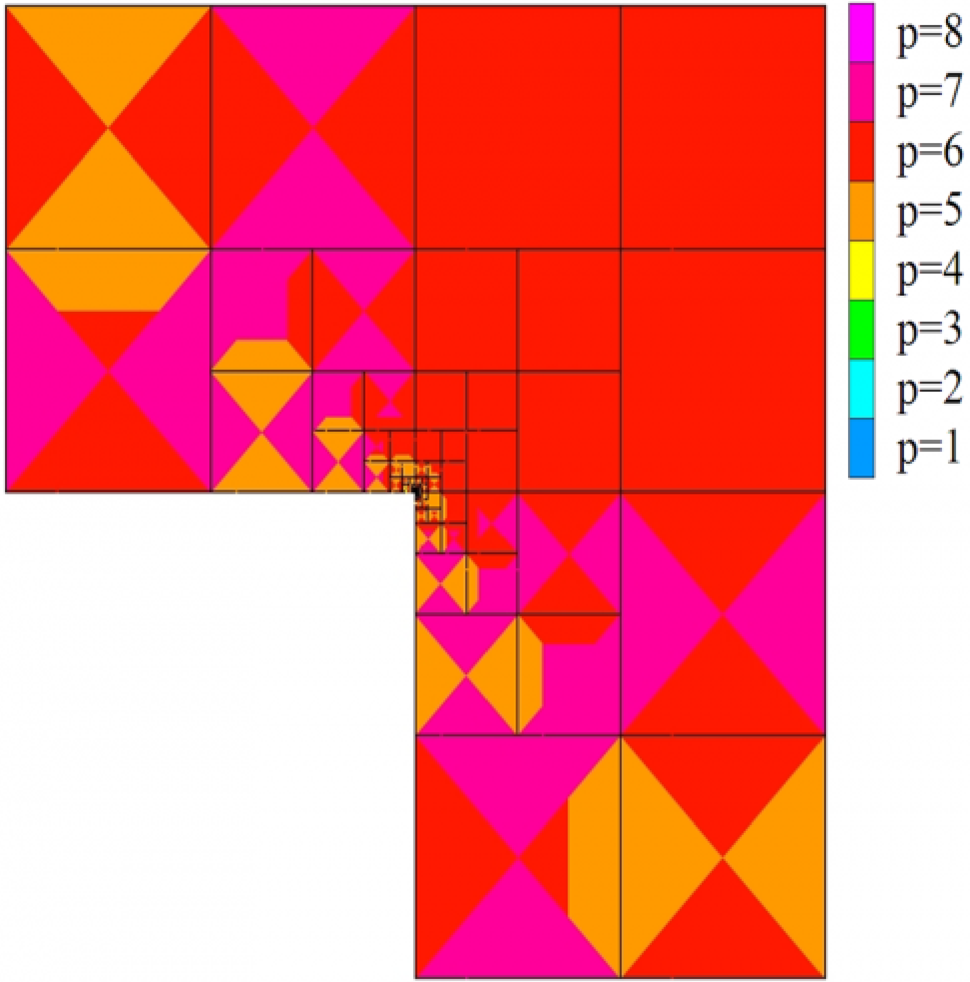
1. Calculate by how much the relative error will drop if we make the adaptation represented by the code, the fall of the error (code) = $\|u_{h/2,p+1} - u_{h,p}\|_K - \|u_{h/2,p+1} - w\|_K$
2. Calculate how many unknowns (how many base functions) we have to add to implement the adaptation represented by the code, adaptation cost (code)
3. Calculate and remember the rate of error decrease (code) = error decrease (code) / adaptation cost (code)
4. If the rate of error decrease (code) is greater than the fastest rate of error decrease, remember the greatest error rate value = error rate rate (code), optimal adaptation = code
5. End of the loop after possible types of adaptation
6. Perform on item K the optimal adaptation found

In other words, we choose the type of adaptation for the element that gives the greatest error reduction at the lowest cost. This quantity is represented by the rate of error decrease, which increases with the decrease in error but decreases with the cost incurred (with the number of functions added to the element because the cost of the calculation on a given element depends on the number of unknowns which are coefficients of the base functions). This algorithm is illustrated in the figure Rys. 6

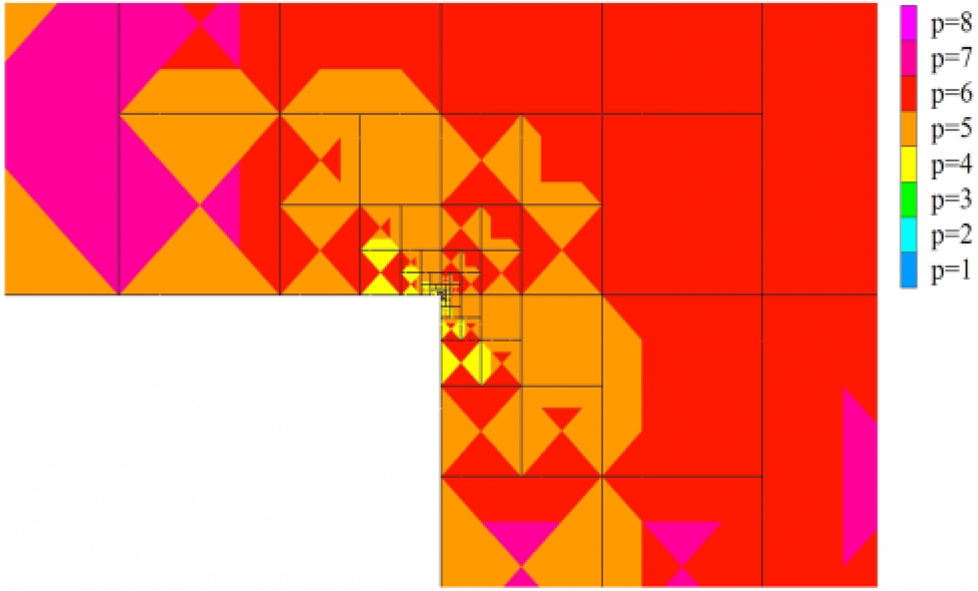


Rysunek 6: Wybór optymalnej strategii adaptacji elementu spośród wielu możliwości, na podstawie obliczonej prędkości spadku błędu.

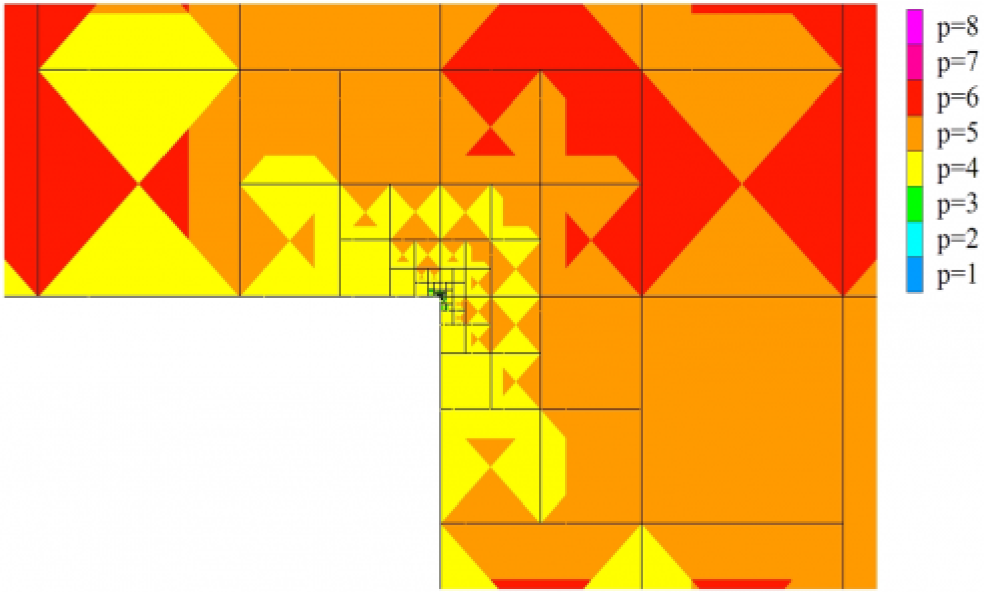
For the exemplary problem of heat transport in the L-shaped area, it is possible to obtain an adaptive hp mesh that allows to solve the problem with the relative error of 0.0001. Such a grid is shown in the figures Rys. 7 - Rys. 13



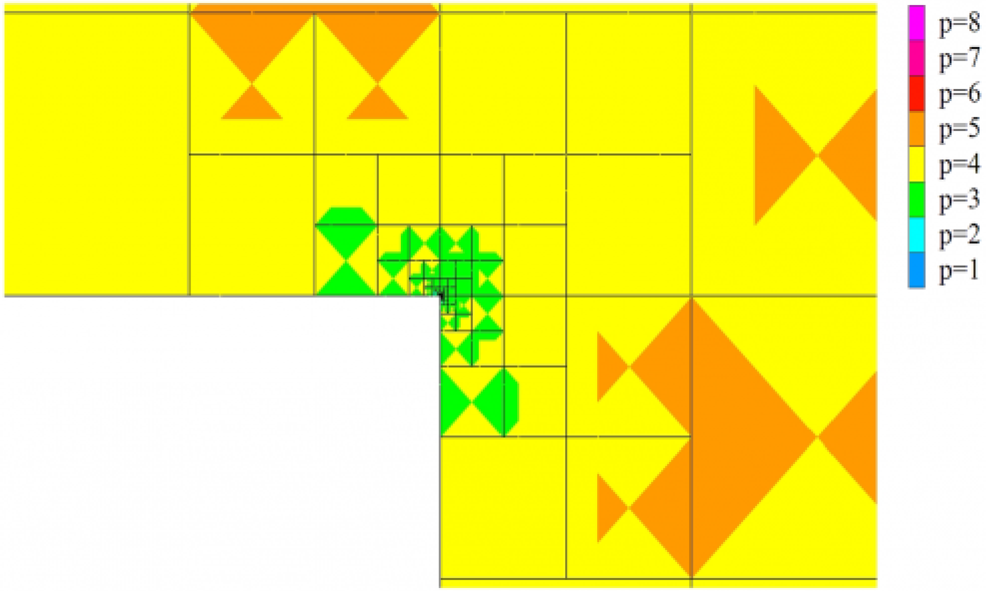
Rysunek 7: Siatka hp adaptacyjna umożliwiająca rozwiązanie problemu transportu ciepła na obszarze w kształcie litery L z dokładnością 0.001 w sensie błędu względnego (różnicy w normie H1 pomiędzy rozwiązaniem na siatce żadkiej a siatce gęstej)



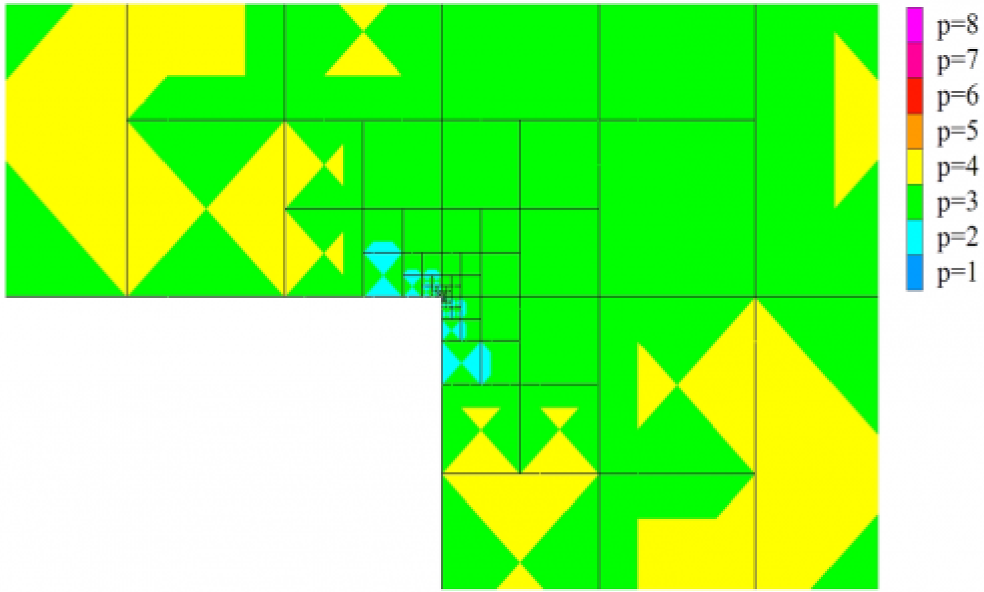
Rysunek 8: Zoom 10 na siatkę optymalną



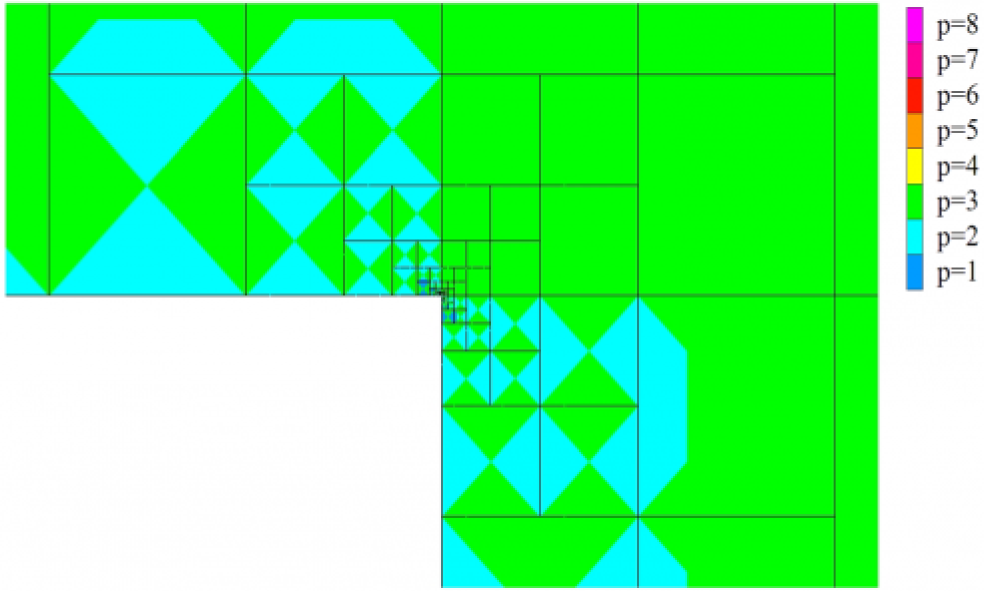
Rysunek 9: Zoom 100 razy na siatkę optymalną.



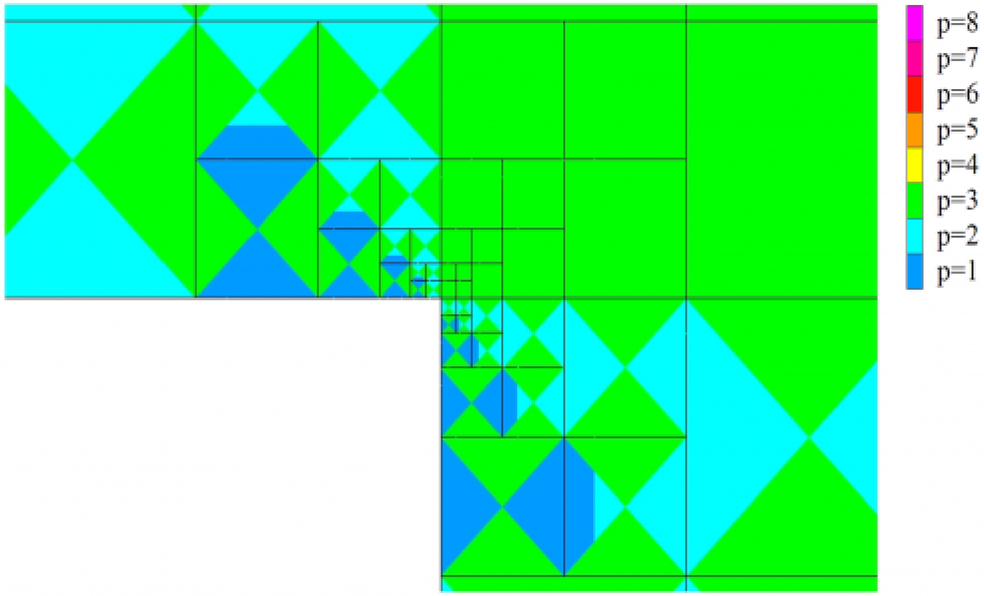
Rysunek 10: Zoom 1000 razy na siatkę optymalną.



Rysunek 11: Zoom 10000 razy na siatkę optymalną.



Rysunek 12: Zoom 100000 razy na siatkę optymalną.



Rysunek 13: Zoom 1000000 razy na siatkę optymalną.

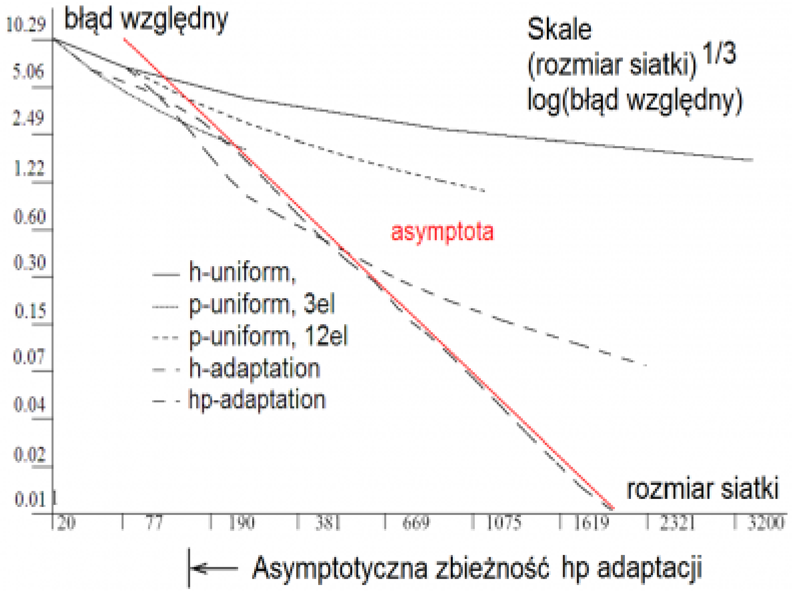
The automatic hp adaptation algorithm is therefore very complicated and difficult to implement. So why is it worth our interest? Consider possible different adaptation algorithms for the exemplary L-shaped heat transport problem, i.e.

1. Homogeneous adaptation algorithm where the number of elements is increased evenly over the entire area
2. A homogeneous p adaptation algorithm, using a mesh made of 3 elements and increasing uniformly the degree of polynomials inside all elements horizontally and vertically, and the degree of polynomials on all edges
3. Homogeneous p adaptation algorithm, using a grid made of 12 elements and increasing uniformly the degree of polynomials inside all elements horizontally and vertically, and the degree of polynomials on all edges
4. Automatic p adaptation algorithm, increasing the degree of polynomials in the interiors of selected elements in selected directions, and modifying the degrees on the edges according to the minimum rule
5. Automatic adaptation algorithm, breaking selected elements in selected directions
6. Automatic hp adaptation algorithm, described in this chapter

If we draw a graph of convergence of these algorithms in such a way that on the horizontal axis we place the size of the grid understood as the number of base functions on the entire grid, which corresponds to the number of unknown coefficients of these base functions, i.e. the size of the matrix of the system of equations to be solved, and on the vertical axis, the relative error calculated in H1 standard between the solution on the coarse mesh and the dense mesh

$$\left\|u_{h,p} - u_{h/2,p+1}\right\|_{H^1} = \frac{\int_\Omega (u_{h,p} - u_{h/2,p+1})^2 + (\frac{\partial(u_{h,p} - u_{h/2,p+1}}{\partial x})^2 + (\frac{\partial(u_{h,p} - u_{h/2,p+1}}{\partial x})^2}{\int_\Omega (u_{h/2,p+1})^2 + (\frac{\partial(u_{h/2,p+1}}{\partial x})^2 + (\frac{\partial(u_{h/2,p+1}}{\partial x})^2}$$

then it turns out that only the automatic hp adaptation algorithm has the feature of exponential convergence. It is definitely the fastest and can solve a given computational problem with an accuracy that is impossible to achieve by other adaptive algorithms.



Rysunek 14: Zbieżność różnych algorytmów adaptacyjnych.

In practice, most engineering problems require accuracy in the range of 5 percent and square polynomials. However, there are computational tasks where high accuracy of the solution is essential. Examples of such computational tasks are, for example, simulations of flows around the wings of aircraft, which require very high accuracy in the boundary layer on the wings of the aircraft, or simulations of the propagation of electromagnetic waves in the rock mass layers to identify oil deposits that require very high accuracy in the vicinity of the antenna receiving.

Bibliografia

1. Leszek Demkowicz: Computing with hp-Adaptive Finite Elements, Vol. I. One and Two Dimensional Elliptic and Maxwell Problems, Taylor & Francis, CRC Press 2006, dostęp:18.10.2019
2. Leszek Demkowicz, Jason Kurtz, David Pardo, Maciej Paszyński, Waldemar Rachowicz, Adam Zdunek A.: Computing with hp-Adaptive Finite Elements, Vol. II.Frontiers: Three Dimensional Elliptic and Maxwell Problems with Applications, Taylor & Francis, CRC Press 2007, dostęp:18.10.2019

3. Ivo Babuška, B. Q. Guo : The hp-version of the finite element method, part 1, Computational mechanics, Maryland Collage Park Campus 1986, dostęp:18.10.2019
4. Ivo Babuška, B. Q. Guo: The hp-version of the finite element method, part 2, Computational mechanics, Maryland Collage Park Campus 1986, dostęp:18.10.2019

Utworzona przez admin. Ostatnia aktualizacja: Środa 28 z Październik, 2020 08:40:32 UTC przez paszynsk@agh.edu.pl. Autor: Maciej Paszynski

STATUS: W opracowaniu Zgłoś do recenzji Edytuj