XXI Congreso de Ecuaciones Diferenciales y Aplicaciones
XI Congreso de Matemática Aplicada
Ciudad Real, 21-25 septiembre 2009
(pp. 1–8)

Generación automática de mallas de tetraedros con el método del mecano: Descripción y futuro de la nueva técnica

R. Montenegro[1], J.M. Cascón[2], E. Rodríguez[1], J.M. Escobar[1], G. Montero[1]

[1] *Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería, SIANI, Universidad de Las Palmas de Gran Canaria, http://www.dca.iusiani.ulpgc.es/proyecto2008-2011,*
E-mails: {rmontenegro,erodriguez,jmescobar,gmontero}@siani.es.
[2] *Departamento de Matemáticas, Universidad de Salamanca. E-mail:* casbar@usal.es.

Palabras clave: Generación de mallas de tetraedros, refinamiento adaptativo, mallas encajadas, suavizado y desenredo de mallas, método de elementos finitos en 3-D.

Resumen

Mostramos la capacidad del nuevo método del mecano para generar automáticamente mallas de tetraedros de sólidos de geometría compleja cuya superficie tenga género 0. En general, la idea del método es aplicable para mallar dominios tridimensionales cuya frontera puede ser transformada biyectivamente sobre las caras de un mecano que *aproxima* el sólido y que se construye a partir de piezas poliédricas interconectadas. En particular, en este trabajo consideramos un mecano formado por un único cubo. El procedimiento automático de generación de la malla está definido sólamente por una triangulación de la superficie del sólido, un cubo y una tolerancia relativa a la aproximación deseada. Introducimos una técnica automática para asociar las caras del mecano a *patches* de la triangulación superficial del sólido, con el propósito de definir la transformación uno a uno entre dichas superficies. La transformación resultante, entre las mallas de tetraedros del sólido y del mecano, conforma una parametrización discreta de un volumen irregular (sólido) en un simple cubo (mecano). A partir de los resultados mostrados, se puede intuir las posibilidades futuras del método del mecano.

1. Introduction

Many authors have devoted great effort to solving the automatic mesh generation problem in different ways [4, 17, 19, 41], but the 3-D problem is still open [2]. Along the past, the main objective has been to achieve high quality adaptive meshes of complex solids with minimal user intervention and low computational cost. At present, it is well known that most mesh generators are based on Delaunay triangulation and advancing front technique, but problems, related to mesh quality or mesh conformity with the solid boundary, can still appear for complex geometries. In addition, an appropriate definition of element sizes is demanded for obtaining good quality elements and mesh adaption. Particularly, local adaptive refinement strategies have been employed to mainly adapt the mesh to singularities of numerical solution. These adaptive methods usually involve remeshing or nested refinement [5, 20, 23, 26, 37].

We introduced the new meccano technique in [32, 3, 33] for constructing adaptive tetrahedral meshes of solids. We has given this name to the method because the process starts with the construction of a coarse approximation of the solid, i.e. a meccano composed by connected polyhedral pieces. So, the method can be applied with different types of pieces (cuboids, pyramids, prisms, ect.). A simple particular case is when meccano is composed by connected cubes, i.e. a polycube.

The new automatic mesh generation strategy uses no Delaunay triangulation, nor advancing front technique, and it simplifies the geometrical discretization problem for 3-D complex domains, whose surfaces can be mapped to the meccano faces. The main idea of the new mesh generator is to combine a local refinement/derefinement algorithm for 3-D nested triangulations [23], a parameterization of surface triangulations [10] and a simultaneous untangling and smoothing procedure [7]. At present, the meccano technique has been implemented by using the local refinement/derefinement of Kossaczky [23], but the idea could be implemented with other types of local refinement algorithms [20, 26, 37]. The resulting adaptive meshes have good quality for finite element applications.

Our approach is based on the combination of several former procedures (refinement, mapping, untangling and smoothing) which are not in themselves new, but the overall integration is an original contribution. Authors have used them in different ways. Triangulations for convex domains can be constructed from a coarse mesh by using refinement/projection [38]. Adaptive nested meshes have been constructed with refinement and derefinement algorithms for evolution problems [9]. Triangulation maps from physical and paramentric spaces have been analyzed for many author. Significant advances in surface parameterization have been done by several authors [10, 12, 13, 40, 25, 43], but the volume parameterization is still open. Floater et al [14] give a simple counterexample to show that convex combination mappings over tetrahedral meshes are not necessarily one-to-one. Large domain deformations can lead to severe mesh distortions, especially in 3-D. Mesh optimization is thus key for keeping mesh shape regularity and for avoiding a costly remeshing [21, 22]. In traditional mesh optimization, mesh moving is guided by the minimization of certain overall functions, but it is usually done in a local fashion. In general, this procedure involves two steps [15, 16]: the first is for mesh untangling and the second one for mesh smoothing. Each step leads to a different objective function. In this paper, we use the improvement proposed by [7], where a simultaneous untangling and smoothing guided by the same objective function is introduced.

Some advantages of our technique are that: surface triangulation is automatically constructed, the final 3-D triangulation is conforming with the object boundary, inner surfaces are automatically preserved (for example, interface between several materials), node distribution is adapted in accordance with the object geometry, and parallel computations can easily be developed for meshing the meccano pieces. However, our procedure demands an automatic construction of the meccano and an admissible mapping between the meccano boundary and the object surface must be defined.

In this paper we present new ideas and applications of our method. Specifically, we consider a complex genus-zero solid defined by a triangulation of its surface. In this case, it is sufficient to fix a meccano composed by only one cube and a tolerance that fixes the desired approximation of the solid surface. In order to define an admisible mapping between the cube faces and patches of the initial surface triangulation of the solid, we introduce a new automatic method to decompose the surface triangulation into six patches that preserves the same topological connections than the cube faces. Then, a discrete mapping from each surface patch to the corresponding cube face is contructed by using the parameterization of surface triangulations proposed by M. Floater in [10, 11, 12, 13]. The shape-preserving parametrizations, which are planar triangulations on the cube faces, are the solutions of linear systems based on convex combinations. In our case, the solution to several compatibility problems on the cube edges will be discussed.

In the next future, some more effort should be made in an automatic construction of the meccano when the genus of the solid surface is greater than zero. Currently, several authors are working on this aspect in the context of polycube-maps, see for example [40, 25, 43]. They are analyzing how to construct a polycube for a generic solid and, simultaneously, how to define a conformal mapping between the polycube boundary and the solid surface. Although harmonic maps have been extensively studied in the literature of surface parameterization, only a few works are related to volume parametrization, for example a meshless procedure is presented in see [24].

In the following section we present a brief description of the main stages of the method for a generic meccano composed of polyhedral pieces. In section 3 we introduce applications of the algorithm in the case that the meccano is formed by a simple cube. Finally, conclusions and future research are presented in section 4.

2. General Algorithm of the Meccano Technique

The main steps of the general *meccano tetrahedral mesh generation algorithm* are summarized in this section. A more detailed description of this process can be analyzed in [32, 3, 33]. The input data of the algorithm are the definition of the object boundaries (for example by a given boundary triangulation) and a given precision (corresponding to its approximation). The following algorithm describes the whole mesh generation approach.

Meccano tetrahedral mesh generation algorithm

1. Construct a meccano approximation of 3-D solid formed by polyhedral pieces.
2. Define an admissible mapping between the meccano and the object boundary.
3. Construct a coarse tetrahedral mesh of the meccano.
4. Generate a local refined tetrahedral mesh of the meccano, such that the mapping (according step 2) of the meccano boundary triangulation approximates the solid boundary for a given precision.
5. Move the boundary nodes of the meccano to the object surface according to the mapping defined in 2.

6. Relocate the inner nodes of the meccano.
7. Optimize the tetrahedral mesh applying the simultaneous untangling and smoothing procedure.

The first step of the procedure is to construct a meccano approximation by connecting different polyhedral pieces. Once the meccano approximation is fixed, we have to define an *admissible* one-to-one mapping between the boundary faces of the meccano and the boundary of the object. In step 3, the meccano is decomposed into a coarse and valid tetrahedral mesh by an appropriate subdivision of its initial polyhedral pieces. We continue with a local refinement strategy to obtain an adapted mesh which can approximate the boundaries of the domain within a given precision. Then, we construct a mesh of the domain by mapping the boundary nodes from the meccano plane faces to the true boundary surface and by relocating the inner nodes at a reasonable position. After these two steps the resulting mesh is tangled, but it has an admissible topology. Finally, a simultaneous untangling and smoothing procedure is applied and a valid adaptive tetrahedral mesh of the object is obtained.

We note that the general idea of the meccano technique could be understood as the connection of different polyhedral pieces. So, the use of cuboid pieces, or a polycube meccano, are particular cases.

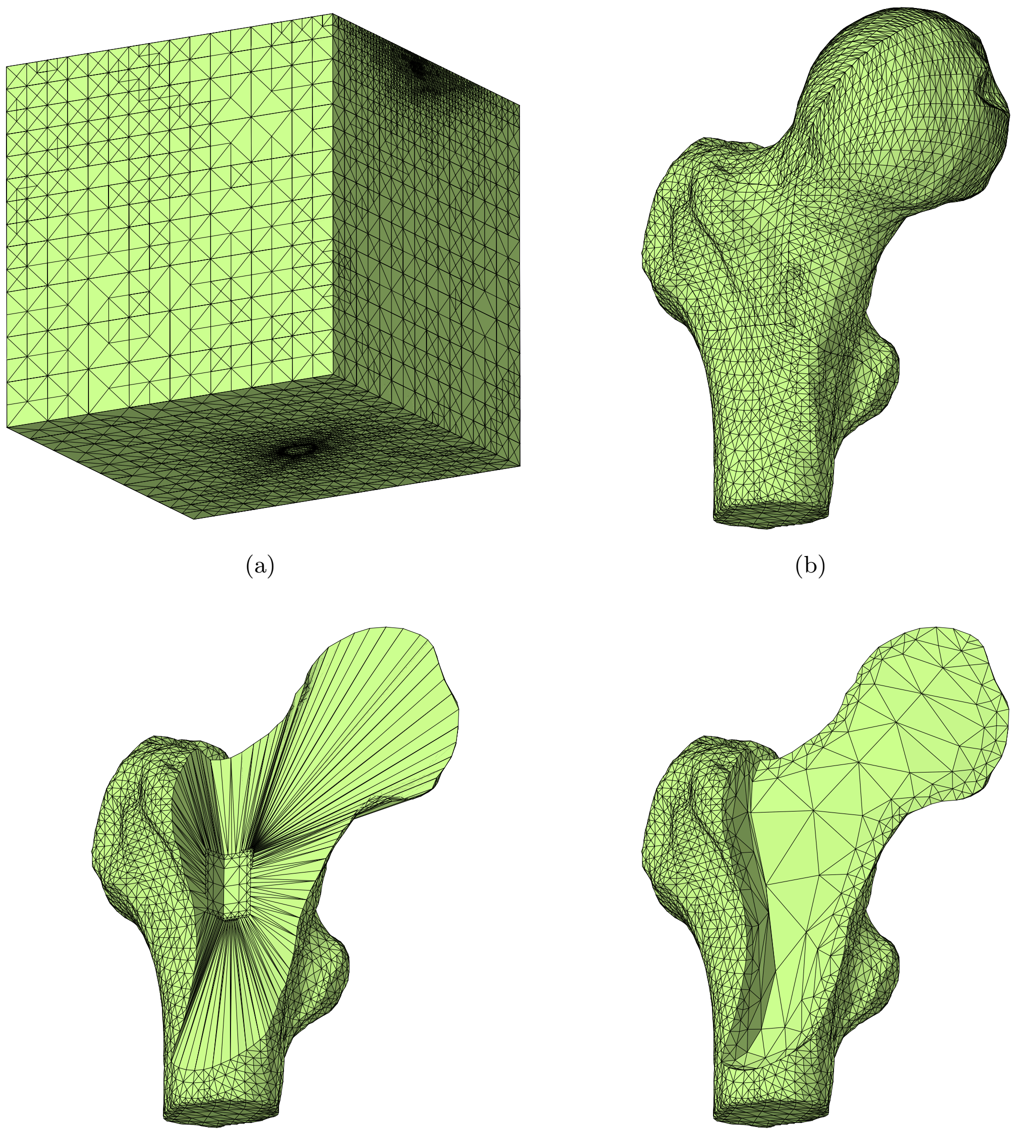
3. Application of the Meccano Technique to a Complex Genus-Zero Solid

In this section, we present the application of the meccano algorithm in the case that the solid surface is genus-zero, and the meccano is formed by one cube. We assume a triangulation of the solid surface as data. The main result of this section is the automatic parametrization between the surface triangulation of the solid and the cube boundary. For this purpouse, we divide the surface triangulation into six patches, with the same topological connection that cube faces, such that each patch is mapped to a cube face.

We note that being even poor the quality of this initial triangulation, the meccano method can reach a high quality surface and volume triangulation.

The parametrization of a surface triangulation patch to a cube face has been done with GoTools core and parametrization modules from SINTEF ICT, available in the website http://www.sintef.no/math_software. This code implements Floater's parametrization in C++. Specifically, in the following application we have used mean value method for the parametrization of the inner nodes of the patch triangulation, and the boundary nodes are fixed with chord length parametrization [10, 12].

We have implemented the meccano method by using the local refinement of ALBERTA. This code is an adaptive multilevel finite element toolbox [39] developing in C. This software can be used for solving several types of 1-D, 2-D or 3-D problems. ALBERTA uses the Kossaczky refinement algorithm [23] and requires an initial mesh topology [38]. The recursive refinement algorithm could not terminate for general meshes. The meccano technique constructs meshes that verify the imposed restrictions of ALBERTA in relation to topology and structure. They can be refined by its recursive algorithm, because they are loop-free, and the degeneration of the resulting triangulations after successive refinements is avoided. The minimum quality of refined meshes is function of the initial mesh quality [27, 42].



(a) (b) (c) (d)

Figura 1: (a) Refined tetrahedral mesh of the cube, (b) resulting bone tetrahedral mesh after inner node relocation and mesh optimization. Cross sections of the bone before (c) and after (d) the application of the mesh optimization process.

The performance of our novel tetrahedral mesh generator, for meshing complex genus-zero solid, is shown in its application to a bone whose surface triangulation has been obtained from the website *http://www-c.inria.fr/gamma/download/affichage.php?dir=-ANATOMY&name=ballJoint*, and it can be found in the CYBERWARE Catalogue. This surface mesh contains 274120 triangles and 137062 nodes. The meccano technique constructs a tetrahedral mesh with 47824 tetrahedra and 11525 nodes. This mesh has 11530 triangles and 5767 nodes on its boundary and it has been reached after 23 Kossaczky refinements from the initial subdivition of the cube into six tetrahedra. A tangled tetrahedra mesh with 1307 inverted elements appears after the mapping of the meccano external nodes to the initial triangulation of the bone surface. A node relocation process reduces the number of inverted tetrahedra to 16. Finally, our mesh optimization algorithm produces a high quality tetrahedra mesh: the minimum mesh quality is 0,15 and the average quality is 0,64. Resulting meshes are shown in Figure 1. We note that the cube of Figure 1 (a) is located inside the bone. In Figure 1 (c) it can be observed its location and the tangled tetrahedral mesh that is obtained after the mapping of external nodes of the cube to the bone surface.

4. Conclusions and Future Research

The meccano technique is a very efficient mesh generation method for creating adaptive tetrahedral meshes of a solid whose boundary is a surface of genus 0. We remark that the method requires minimum user intervention and has a low computational cost. The procedure is fully automatic and it is only defined by a surface triangulation of the solid, a cube and a tolerance that fixes the desired approximation of the solid surface.

We have introduced an automatic partition of the given solid surface triangulation for fixing an addmisible mapping between the cube faces and the solid surface patches, such that each cube face is the parametric space of its corresponding patch.

The mesh generation technique is based on sub-processes (subdivision, mapping, optimization) which are not in themselves new, but the overall integration using a simple shape as starting point is an original contribution of the method and it has some obvious performance advantages. Another interesting property of the new mesh generation strategy is that it automatically achieves a good mesh adaption to the geometrical characteristics of the domain. In addition, the quality of the resulting meshes is high.

The main ideas presented in this paper can be applied for constructing tetrahedral or hexahedral meshes of complex solids. In future works, the meccano technique can be extended for meshing a complex solid whose boundary is a surface of genus greater than zero. In this case, the meccano can be a polycube or constructed by polyhedral pieces with compatible connections. At present, the user has to define the meccano associated to the solid, but we are implemented a special CAD package for more general input solid.

Acknowledgments

This work has been supported by the Spanish Government, “Secretaría de Estado de Universidades e Investigación”, “Ministerio de Ciencia e Innovación”, and FEDER, grant contract: CGL2008-06003-C03. We would also like to thank the authors of ALBERTA [38] for the code availability in internet [39] and for their suggestions. Special thanks also to the geometry group at SINTEF ICT, Department of Applied Mathematics, for their codes availability in internet for the parametrization of a surface triangulation patch.

References

[1] M.S. Bazaraa, H.D. Sherali and C.M. Shetty, *Nonlinear Programing: Theory and Algorithms*, John Wiley and Sons Inc., New York, 1993.

[2] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott and T.W. Sederberg, *Isogeometric analysis: Toward unification of computer aided design and finite element analysis*, Trends in Engineering Computational Technology, Saxe-Coburg Publications, Stirling, 2008, pp. 1-16.

[3] J.M. Cascón, R. Montenegro, J.M. Escobar, E. Rodríguez and G. Montero, *A new meccano technique for adaptive 3-D triangulation*, in: Proc. 16th International Meshing Roundtable, Springer, Berlin, 2007, pp. 103-120.

[4] G.F. Carey, *Computational Grids: Generation, Adaptation, and Solution Strategies*, Taylor & Francis, Washington, 1997.

[5] G.F. Carey, *A Perspective on Adaptive Modeling and Meshing (AM&M)*, Comput. Meth. Appl. Mech. Eng. 195 (2006) 214-235.

[6] J.M. Escobar and R. Montenegro, *Several Aspects of Three-dimensional Delaunay Triangulatio*n, Adv. Eng. Soft. 27 (1996) 27-39.

[7] J.M. Escobar, E. Rodríguez, R. Montenegro, G. Montero and J.M. González-Yuste, *Simultaneous Untangling and Smoothing of Tetrahedral Meshes*, Comput. Meth. Appl. Mech. Eng. 192 (2003) 2775-2787.

[8] J.M. Escobar, G. Montero, R. Montenegro and E. Rodríguez, *An Algebraic Method for Smoothing Surface Triangulations on a Local Parametric Space*, Int. J. Num. Meth. Eng. 66 (2006) 740-760.

[9] L. Ferragut, R. Montenegro and Plaza, *Efficient Refinement/Derefinement Algorithm of Nested Meshes to Solve Evolution Problems*, Comm. Num. Meth. Eng. 10 (1994) 403-412.

[10] M.S. Floater, *Parametrization and Smooth Approximation of Surface Triangulations*, Comput. Aid. Geom. Design, 14 (1997) 231-250.

[11] M.S. Floater, *One-to-one Piece Linear Mappings over Triangulations*, Mathematics of Computation, 72 (2002) 685-696.

[12] M.S. Floater, *Mean Value Coordinates*, Comput. Aid. Geom. Design, 20 (2003) 19-27.

[13] M.S. Floater and K. Hormann, *Surface parameterization: a tutorial and survey*, in: Advances in Multiresolution for Geometric Modelling, Mathematics and Visualization. Springer, Berlin, 2005, pp. 157-186.

[14] M.S. Floater and V. Pham-Trong, *Convex Combination Maps over Triangulations, Tilings, and Tetrahedral Meshes*, Advances in Computational Mathematics, 25 (2006) 347-356.

[15] L.A. Freitag and P. Plassmann, *Local Optimization-based Simplicial Mesh Untangling and Improvement*, Int. J. Num. Meth. Eng. 49 (2000) 109-125.

[16] L.A. Freitag and P.M. Knupp, *Tetrahedral Mesh Improvement Via Optimization of the Element Condition Number*, Int. J. Num. Meth. Eng. 53 (2002) 1377-1391.

[17] P.J. Frey and P.L. George, *Mesh Generation*, Hermes Science Publishing, Oxford, 2000.

[18] P.L. George, F. Hecht and E. Saltel, *Automatic Mesh Generation with Specified Boundary*, Comput. Meth. Appl. Mech. Eng. 92 (1991) 269-288.

[19] P.L. George and H. Borouchaki, *Delaunay Triangulation and Meshing: Application to Finite Elements*, Editions Hermes, Paris, 1998.

[20] J.M. González-Yuste, R. Montenegro, J.M. Escobar, G. Montero and E. Rodríguez, *Local Refinement of 3-D Triangulations Using Object-oriented Methods*, Adv. Eng. Soft. 35 (2004) 693-702.

[21] P.M. Knupp, *Achieving Finite Element Mesh Quality Via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part II-A Frame Work for Volume Mesh Optimization and the Condition Number of the Jacobian Matrix*, Int. J. Num. Meth. Eng. 48 (2000) 1165-1185.

[22] P.M. Knupp, *Algebraic Mesh Quality Metrics*, SIAM J. Sci. Comput. 23 (2001) 193-218.

[23] I. Kossaczky, *A Recursive Approach to Local Mesh Refinement in Two and Three Dimensions*, J. Comput. Appl. Math. 55 (1994) 275-288.

[24] X. Li, X. Guo, H. Wang, Y. He, X. Gu and H. Qin, *Harmonic Volumetric Mapping for Solid Modeling Applications*, Proc. of ACM Solid and Physical Modeling Symposium, Association for Computing Machinery, Inc., 2007, pp. 109-120.

[25] J. Lin, X. Jin, Z. Fan and C.C.L. Wang, *Automatic PolyCube-Maps*, Lecture Notes in Computer Science, 4975, Springer, Berlin, 2008, pp. 3-16.

[26] R. Löhner and J.D. Baum, *Adaptive H-Refinement on 3-D Unstructured Grids for Transient Problems*, Int. J. Num. Meth. Fluids 14 (1992) 1407-1419.

[27] J. Maubach, *Local Bisection Refinement for N-Simplicial Grids Generated by Reflection*, SIAM J. Sci. Comput. 16 (1995) 210-227.

[28] W.F. Mitchell, *A Comparison of Adaptive Refinement Techniques for Elliptic Problems*, ACM Trans. Math. Soft. 15 (1989) 326-347.

[29] R. Montenegro, G. Montero, J.M. Escobar and E. Rodríguez, *Efficient Strategies for Adaptive 3-D Mesh Generation over Complex Orography*, Neural, Parallel & Scientific Computation 10 (2002) 57-76.

[30] R. Montenegro, G. Montero, J.M. Escobar, E. Rodríguez and J.M. González-Yuste, *Tetrahedral mesh generation for environmental problems over complex terrains*, in: Lecture Notes in Computer Science, Vol. 2329, Springer, Berlin, 2002, pp. 335-344.

[31] R. Montenegro, J.M. Escobar, G. Montero and E. Rodríguez, *Quality improvement of surface triangulations*, in: Proc. 14th International Meshing Roundtable, Springer, Berlin, 2005, pp. 469-484.

[32] R. Montenegro, J.M. Cascón, J.M. Escobar, E. Rodríguez and G. Montero, *Implementation in ALBERTA of an automatic tetrahedral mesh generator*, in: Proc. 15th International Meshing Roundtable, Springer, Berlin, 2006, pp. 325-338.

[33] R. Montenegro, J.M. Cascón, J.M. Escobar, E. Rodríguez and G. Montero, *An Automatic Strategy for Adaptive Tetrahedral Mesh Generation*, Appl. Num. Math. (2009) doi:10.1016/j.apnum.2008.12.010.

[34] G. Montero, R. Montenegro, J.M. Escobar, E. Rodríguez and J.M. González-Yuste, *Velocity field modelling for pollutant plume using 3-D adaptive finite element method*, in: Lecture Notes in Computer Science, Vol. 3037, Springer, Berlin, 2004, pp. 642-645.

[35] G. Montero, E. Rodríguez, R. Montenegro, J.M. Escobar and J.M. González-Yuste, *Genetic Algorithms for an Improved Parameter Estimation with Local Refinement of Tetrahedral Meshes in a Wind Model*, Adv. Eng. Soft. 36 (2005) 3-10.

[36] M.C. Rivara, *A Grid Generator Based on 4-Triangles Conforming. Mesh-refinement Algorithms*, Int. J. Num. Meth. Eng. 24 (1987) 1343-1354.

[37] M.C. Rivara and C. Levin, *A 3-D Refinement Algorithm Suitable for Adaptive Multigrid Techniques*, J. Comm. Appl. Numer. Meth. 8 (1992) 281-290.

[38] A. Schmidt and K.G. Siebert, *Design of Adaptive Finite Element Software: The Finite Element Toolbox ALBERTA*, Lecture Notes in Computer Science and Engineering, Vol. 42, Springer, Berlin, 2005.

[39] A. Schmidt and K.G. Siebert, *ALBERTA - An Adaptive Hierarchical Finite Element Toolbox*, http://-www.alberta-fem.de/

[40] M. Tarini, K. Hormann, P. Cignoni and C. Montani, *Polycube-maps*, ACM Trans. Graph., 23 (2004) 853-860.

[41] J.F. Thompson, B. Soni and N. Weatherill, *Handbook of Grid Generation*, CRC Press, London, 1999.

[42] C.T. Traxler, *An Algorithm for Adaptive Mesh Refinement in N Dimensions*, Computing 59 (1997) 115-137.

[43] H. Wang, Y. He, X. Li, X. Gu and H. Qin, *Polycube Splines*, Comput. Aid. Geom. Design, 40 (2008) 721-733.