An automated tetrahedral mesh generator for computer simulation in Odontology based on the Delaunay's algorithm

Mauro Massayoshi Sakamoto
Doutorado em Engenharia Elétrica – USP;
Pesquisador do LMAG – PEA – EP-USP;
São Paulo – SP [Brasil]
mauro@pea.usp.br

José Roberto Cardoso
Livre docência em Engenharia Elétrica – EP-USP;
Professor titular da EP-USP;
São Paulo – SP [Brasil]
cardoso@pea.usp.br

José Marcio Machado
Doutorado em Engenharia Elétrica pela USP
Professor adjunto do DCCE – IBILCE – UNESP
São Paulo – SP [Brasil]
jmarcio@ibilce.unesp.br

In this work, a software package based on the Delaunay's algorithm is described. The main feature of this package is the capability in applying discretization in geometric domains of teeth taking into account their complex inner structures and the materials with different hardness. Usually, the mesh generators reported in literature treat molars and other teeth by using simplified geometric models, or even considering the teeth as homogeneous structures.

Key words: Finite Element Method. Tetrahedral mesh generation. Three-dimensional Delaunay algorithm.

1 Introduction

The Finite Element Method (FEM), although possessing a very high potential for applications in computer simulation and analysis of biological tissues and structures, for its operation, it is necessary a good (realistic) discretization of the related geometric domains in finite element meshes. Whenever anisotropic or non-homogeneous properties are present, the different regions of the problem must be conveniently identified and discretized into finite elements with consistency.

Flaws in the mesh generation, also known as degeneracies, such as "overlapping" (superposition of elements) and "sliver" tetrahedra (flat elements of near-zero volume), should be absent. Besides, it is desired that the mesh generation algorithm itself should be robust, and require few manual interactions from the user during the computer simulation.

Although the improvement of three-dimensional mesh generation methods still remains a research topic for the FEM applications community, a remarkable degree of maturation and experience was already reached for some approaches (LIN et al., 1999; KRŠEK; KRUPA, 2003; CLEMENT et al., 2004).

Apart from this, Delaunay's algorithm and its variations are efficient for regular and irregular solid modeling, such as the geometry of the tooth anatomy (HERMELINE, 1982; MARRETO; MACHADO, 1998). This study will describe the implementation of a C++ program, developed with help of object oriented programming concepts, which implements and applies Delaunay's algorithm for the generation of 3D meshes in teeth structures (SAKAMOTO, 2001; MARRETO; MACHADO, 1998).

A special care was taken for the realistic description of internal structures, and meshes with more than 7000 tetrahedra were generated in the test cases. In the next sections, the tooth embedding and image processing is addressed, a short review of the Delaunay's algorithm is presented, and software descriptions and test examples are shown to demonstrate the efficiency code.

2 Tooth embedding and image processing

In order to apply the Delaunay's algorithm in computer simulations in odontology, a set of points describing the tooth surface model must be obtained. For discretization, the tooth is included in resin and this inclusion is sliced in a number of parallel cuts, defining cross sections of the tooth. The different boundary regions (enamel, dentine, pulp) are scanned to generate sets of points.

An inferior molar for instance may be sliced into 12 different cuts with 1 mm of thickness. The union of these point sets reproduces the three-dimensional tooth geometry, and B-spline interpolations are used if it is necessary, to improve some contour interfaces. This is in fact the classical and easiest approach for geometry generation, and is mentioned for example in Lin et al. (1999).

Figure 1 shows a cloud of points for an inferior molar slice and the original boundary regions; Figure 2 shows the dental boundaries detected by the slices, and Figure 3 shows the entire cloud of measured points for all slices. It should be remarked that the Delaunay's algorithm requires only the clouds of points as input, and breakings of prescribed boundaries (if they are present) may be corrected after the discretization by comparison and remeshing.

3 The three-dimensional Delaunay algorithm

Most of the implementations of Delaunay algorithm, a given three-dimensional domain is cha-

Figure 1: The original slice and measure points of the tooth
Source: The authors.

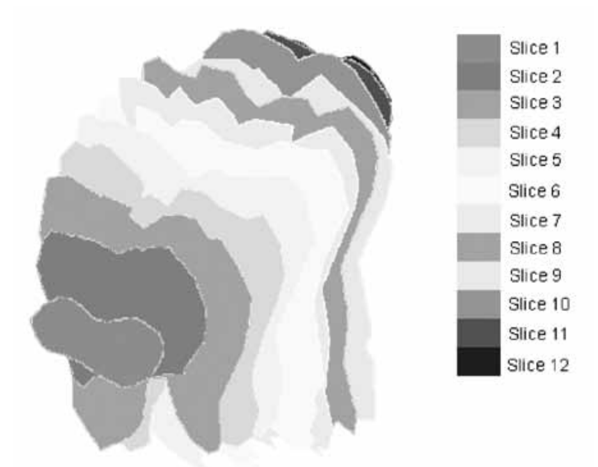

Figure 2: Dental boundaries detected of the slices
Source: The authors.

racterized as a discrete set of points, which will be inserted recursively one at a time as the original mesh grows. The most common implementation of this algorithm is due to Watson (1981), and contains essentially four steps:

An initial mesh of tetrahedra is formed by taking a combination of the 8 vertices of a box which encloses the volume to be discretized into tetrahedra. The box dimensions are selected in such a way that all points in the volume are perfectly included by the box and far from its boundaries;

Figure 3: The entire cloud of measured points for all slices
Source: The authors.

In a typical working stage of Watson technique, a new point from the volume is inserted, and a test is made to select those circumspheres of the existing tetrahedra which will contain this new point. The selected tetrahedral are then compared and their common faces are removed. The mesh is updated by forming new tetrahedra with the inserted point and the remaining faces;

These new tetrahedra in step 2 are combined with those in which the associated circumspheres do not contain the inserted point;

Steps 2 and 3 are repeated recursively for all points forming the volume. Once the mesh is finished, all tetrahedral containing vertices from the enclosing box are deleted.

The basic process may be outlined as follows in Table 1:

Some degeneration may occur in this technique such as: “overlapping” (tetrahedra superposition), “sliver” tetrahedra (flat elements of near-zero volume) and failures in the inclusion test (SUGIHARA; INAGAKI, 1995). This is mainly due to round-off errors and points placed exactly

Table 1: Delaunay tetrahedralization algorithms

```
FOR i=1 TO number of points
    FOR j=1 TO number of tetrahedra
        Compute the circumsphere Cj of the tetrahedron Tj
        IF Cj contains the point i THEN
            Add Tj to the list of faces,
            Add Tj to the list of candidates for deletion,
            Mark Tj for deletion from the list of tetrahedra in
the mesh.
END;
END;
Eliminates the tetrahedra marked for deletion. Eliminate
all common faces in the list of faces. Form new tetrahedra
with point i and the remaining faces according to the
algorithm. Add these tetrahedra to the list of tetrahedra.
END.
```

Source: Watson, 1981.

on the boundaries, but cures and diagnostics were already extensively discussed in several articles (CAVENDISH; FIELD; FREY, 1985; PRIEST, 1991; SHEWCHUK, 1996; 1997) and properly treated in this study.

In relation to treatment of the degeneracies, an alternative developed by the authors to reduce the degenerated Delaunay tetrahedralization is to verify the determinant associated with the new tetrahedra. In this case, no determinant should be null, otherwise the matrix is singular and a new combination should be used. This can be established through a small arbitrary perturbation in the input vertices, or in addition of new points in specific places.

This implementation was executed in two phases:

The programs for mathematical model,

The functions for graphics visualization.

In the phase 1, a version of the algorithm was developed using the C++ oriented objects programming language "for to" of Windows and Linux environment. In the phase 2, the graphic library resources of Mathematics package (version 5.0) were adapted for graphic visualization (WOLFRAM, 1991).

In the next section, it is described the validation tests, performances and some implemented cures for mesh degeneration.

4 Validatiom Tests

Due to the peculiar features of geometry and internal structures of a tooth, some degenerate tetrahedra may occur in the discretization. According to our experiments, the most frequent degeneracies to appear were:

The rupture of internal boundaries between regions;

The generation of "overlapping" elements;

"Sliver" tetrahedral.

The rupture of boundaries (shown in Figure 4) is corrected by the local insertion of a new point, close to the medium point in the affected subdomain boundary. Non-convex subdomains may also disturb the mesh quality, but in this case a simple cure is performed by deleting the tetrahedra in the convex subdomain. Figure 5 shows this kind of correction.

As already mentioned in the previous section, "sliver" tetrahedra and "overlappings" can be miminized by generation of non-singular matrix. The remaining "overlappings" are corrected by slight disturbances of the original point coordinates, followed by insertion, discretization and subsequent reset to the original coordinate values.

However, "overlappings" were very rare since the domain to discretize is not a regular solid (KANAGANATHAN, 1991). The software has detection routines which minimize these degenerations and allow few interventions from the user during the meshing process.

In order to validate the resulting mesh quality, the implementation was tested for a set of discretization points of an inferior molar geometry. Two kinds of meshes were generated: a tooth without inner structures, only enamel, and a tooth with internal structure, i.e., enamel, dentine and pulp. Figure 6 shows a meshed slice from the tooth, and Figure 7 is the entire mesh including all inserted points (without internal structure). The enamel, dentine and pulp meshed structures are shown in Figures 8, 9 and 10 respectively. Figure 11 is a cut in the three-dimensional structure, showing the different meshed regions and their boundaries. It is clear that interfaces between regions and the entire geometry were preserved by the final mesh.

The Table 2 shows the number of elements generated in the mesh of the tooth surface model (only enamel) and the final mesh taking into account the different tissues (enamel, dentine and pulp).

5 Concluding Remarks

In the Finite Element Method (FEM) analysis, solid modeling and discretization are the most time demanding steps, requiring a great amount of user intervention. In this article, a software tool for the meshing of irregular solids was described, which is capable of realistic meshing dental structures.

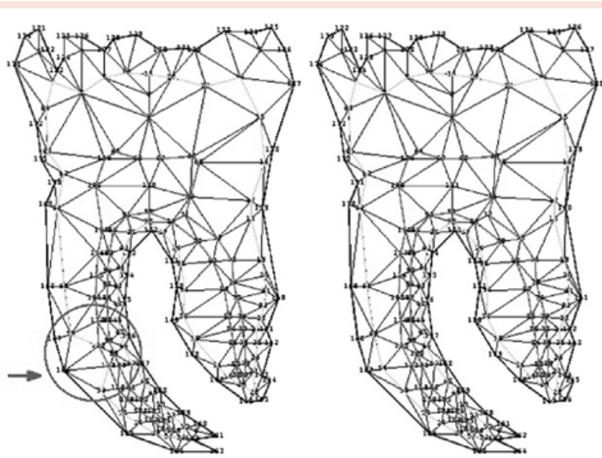

Figure 4: The rupture of boundaries
Source: The authors.

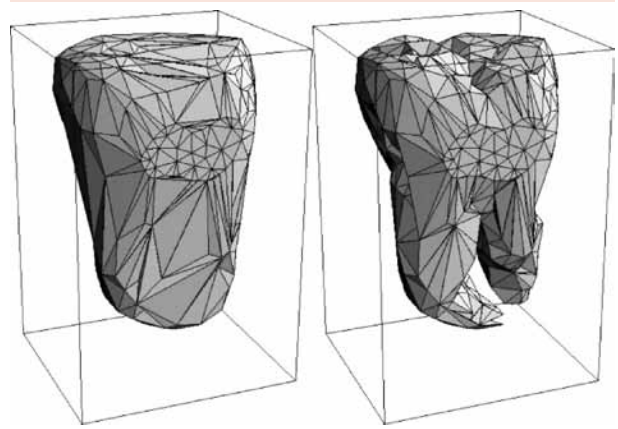

Figure 5: Disturb of the mesh in non-convex subdomains and the correction
Source: The authors.

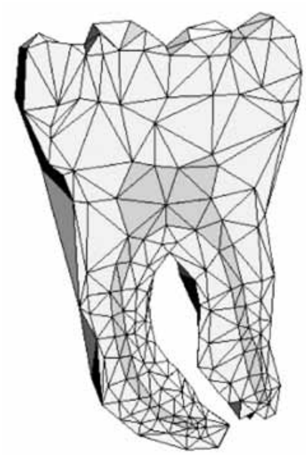

Figure 6: Meshed slice from the inferior molar
Source: The authors.

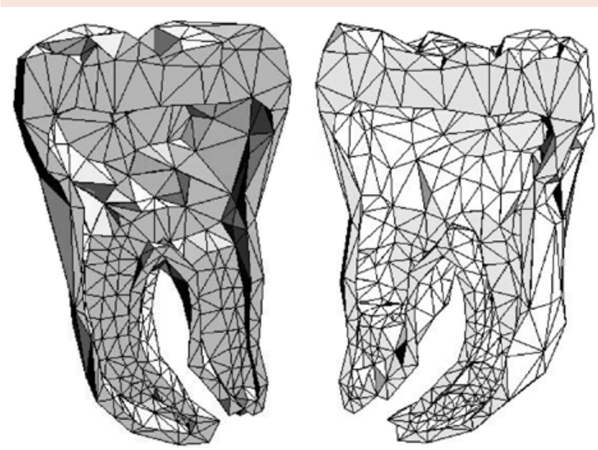

Figure 7: Mesh of the tooth without internal structure (only enamel tissue)
Source: The authors.

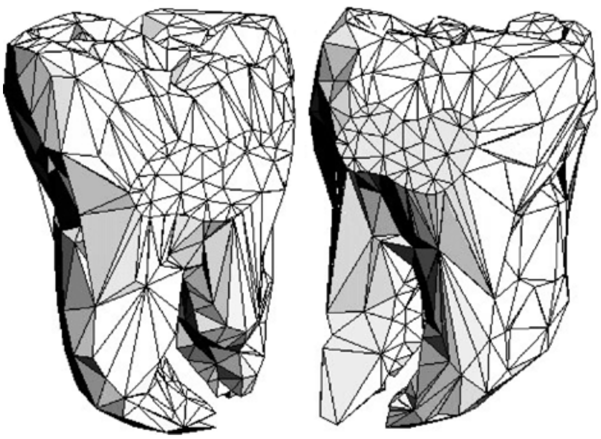

Figure 8: The enamel meshed structure
Source: The authors.

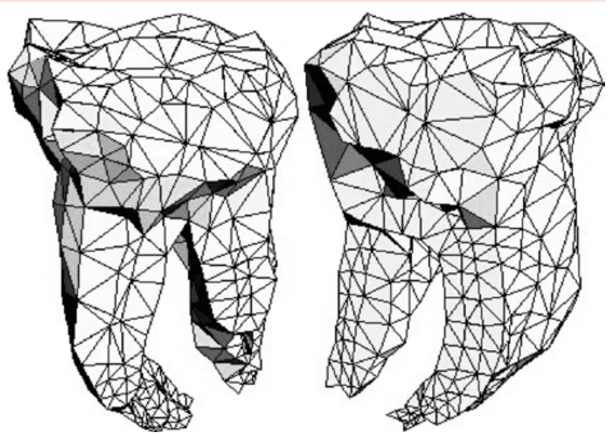

Figure 9: The dentine meshed structure
Source: The authors.

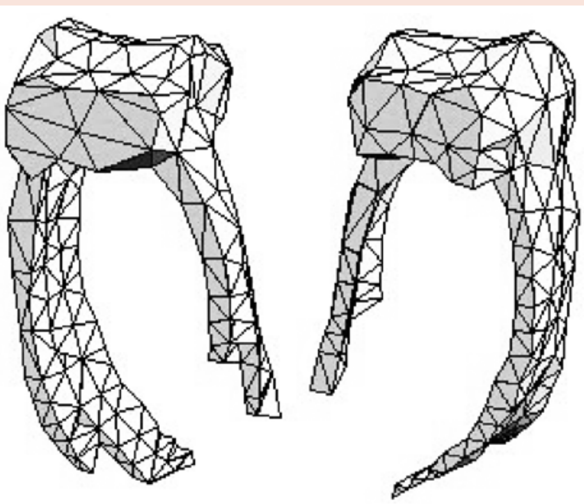

Figure 10: The pulp meshed structure
Source: The authors.

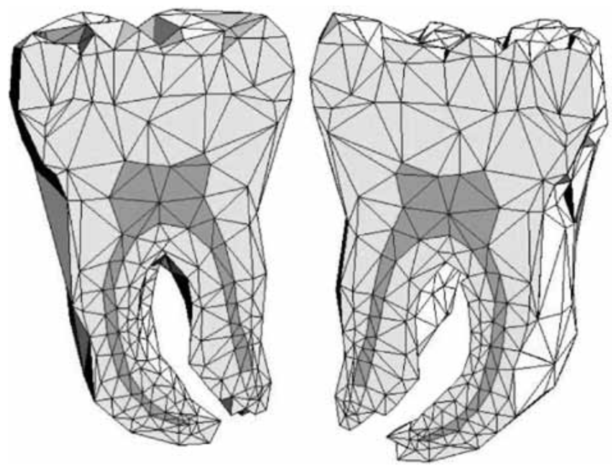

Figure 11: Cut in the three-dimensional structure
Source: The authors.

Table 2: Number of elements for an inferior molar

Materials with different hardness	Tetrahedra number
Only enamel tissue	3251
Enamel, dentine and pulp tissues	7094

Source: The authors.

Results shown were plenty satisfactory, allowing the possibility of accurate stress simulations and other applications in Dentistry.

Future works intend to apply three-dimensional Delaunay refinement algorithms that offer a guarantee on some measure of shape, such as bounded aspect ratio, non-obtuse elements, and tetrahedra with no small angles, except the small angles inherent in the input geometry. Besides, dental biomechanics, determination of stress and strain levels can be investigated.

Acknowledgements

The authors would like to thank the Faculdade de Odontologia (FO) from Araçatuba for their assistance with the supplied material.

References

CAVENDISH, J. C.; FIELD, D. A.; FREY, W. H. An approach to automatic three-dimensional finite element mesh generation. *International Journal for Numerical Methods in Engineering*, USA, v. 21, p. 329-347, 1985.

CLEMENT, R.; SCHNEIDER, J.; BRAMBS, H. J.; WUNDERLICH, A.; SANDER, F. G. Quasi-automatic 3D finite element model generation for individual single-rooted teeth and periodontal ligament. *Computer Methods and Programs in Biomedicine*, Ireland, v. 73, (2), p. 135-144, 2004.

HERMELINE, F. Triangulation automatique dún polyèdre en dimension n. *R.A.I.R.O. Analyse Numérique/Numerical Analysis*, Paris, v. 16, n. 3, p. 211-242, May 1982.

KANAGANATHAN, S.; GOLDSTEIN, N. B. Comparison of four-point adding algorithms for Delaunay-type three-dimensional mesh generators, *IEEE Transactions on Magnetics*, v. 27, n. 3, p. 3444-3451, 1991.

KRŠEK, P.; KRUPA, P., Human tissue geometrical modelling. In: *Applied Simulation Modeling*. Calgary, CA: IASTED, 2003. p. 372-362.

LIN, C. L.; CHANG, C. H.; CHENG, C. S.; LEE, H. E. Automatic finite element mesh generation for maxillary second premolar. *Computer Methods and Programs in Biomedicine*, Ireland, v. 59, p. 187-195, 1999.

MARRETO, C. A. R.; MACHADO, J. M. A compact library for automatic generation of tetrahedral meshes using the mathematica system. *Proc. of 8th International IGTE Symposium on Numerical Field Calculation in Electrical Engineering & European TEAM Workshop*, Austria 1998, p. 116-120.

PRIEST, D. M. Algorithms for arbitrary precision floating point arithmetic. *Proc. 10th Symposium on Computer Arithmetic*, California: IEEE Computer Society Press, 1991. p. 132-143.

SAKAMOTO, M. M. Implementação de um gerador tridimensional de malhas de elementos finitos com aplicações à simulação computacional em odontologia. 2001. Dissertação (Mestrado)- Instituto de Biociências, Letras e Ciências Exatas, Universidade Estadual Paulista, São José do Rio Preto, 2001.

SHEWCHUK, J. R. Robust adaptive floating-point geometric predicates. *Proc. of 12th Annual Symposium on Computational Geometry*, Pennsylvania, 1996, p. 141-150.

SHEWCHUK, J. R. *Delaunay refinement mesh generation*. 1997. PhD. Thesis. School of Computer Science, Carnegie Mellon University, Pennsylvania, 1997.

SUGIHARA, K.; INAGAKI, H. Why is the 3D Delaunay triangulation difficult to construct? *Information Processing Letters*, Ireland, v. 54, p. 275-280, 1995.

WATSON, D. F. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes, *Comput. J.*, London, England, v. 24, p. 167-172, 1981.

WOLFRAM, S. *Mathematica:* A system for doing mathematics by computer. 2nd ed. Redwood City, Canada: Addison-Wesley, 1991.

Recebido em 13 ago. 2008 / aprovado em 11 dez. 2008

Para referenciar este texto

SAKAMOTO, M. M.; CARDOSO, J. R.; MACHADO, J. M. An automated tetrahedral mesh generator for computer simulation in Odontology based on the Delaunay's algorithm. *Exacta*, São Paulo, v. 6, n. 2, p. 237-243, jul./dez. 2008.