Progressive Volume Models for Rectilinear Data using Tetrahedral Coons Volumes

David J. Holliday and Gregory M. Nielson

Arizona State University, Tempe AZ 85287-5406, USA
holliday|nielson@asu.edu

Abstract. We present a new technique for modeling rectilinear volume data. The algorithm produces a trivariate model, $F(x, y, z)$, which is piecewise defined over tetrahedra that fits the volume data to within a user specified tolerance. The technique is adaptive leading to an efficient model that is more complex where the data demands it. The novelty of the present technique is that a valid tetrahedrization is not required. Tetrahedral cells are subdivided as required by the error condition only. This type of cellular decomposition leads to a continuous model by the use of a tetrahedral Coons volume which has the ability to interpolate to arbitrary boundary data.

1 Introduction

Visualization of a volume data set is used to gain some insight into the data. Most scientific visualization algorithms produce a graphical image (volume rendering [2]) or an entity that can then be rendered (isosurfacing [5]). In the case of volume rendering a useful image is generated but subsequent analysis of the original data is difficult using such output. Geometry, in the form of triangle meshes, is created as output from typical isosurfacing algorithms. This type of output is more amenable to analysis but it does not give a complete picture of the original volume data.

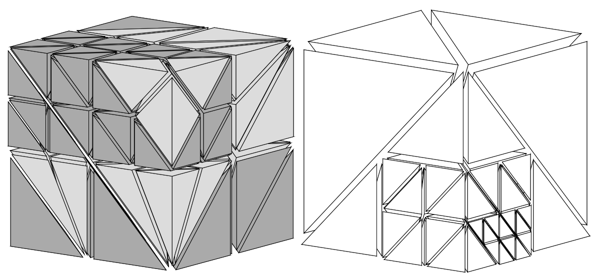
Much can be gained from modeling discrete volume data, i.e., generating an underlying mathematical representation for the data and using that representation for visualization tasks. Benefits of modeling volume data include the compression of large data sets, the application of visualization algorithms, and the ability to perform analyses or simulations.

1.1 Adaptive Approximations

Adaptive approximative models are useful because they can closely approximate portions of data that have large local variations without using too much information to also represent areas where the data is relatively smooth. One way of forming adaptive approximations to volume data is to define an initial, coarse approximation and apply successive refinements until the model closely approximates the data of interest. A model that consists of piecewise linear functions defined over a tetrahedrization is a popular choice for approximating volume data sets. Each vertex in the tetrahedrization has an associated scalar value, or weight, from which a continuous function inside each tetrahedron can be constructed.

There are several methods for performing local refinements on a tetrahedrization [1, 6]. Both algorithms maintain a valid tetrahedrization at all times. (See [9] for definitions and general background material for tetrahedral decompositions.) A tetrahedron is first selected for refinement, generally based on some local error criterion, and then a recursive rule is applied to also refine neighboring tetrahedra. This helps to ensure that two neighboring tetrahedra will always share a common face. Figure 1(a) shows an example of a tetrahedrization that has undergone several refinements using the algorithm presented in [1].

One problem with such a recursive closure scheme is that many tetrahedra that already closely approximate the data may be refined. Many more tetrahedra than necessary may be generated to construct an adequate approximation to the data. It would be advantageous to only subdivide those tetrahedra in which the error is large instead of also having to subdivide neighboring tetrahedra in order to maintain a valid tetrahedrization. Figure 1(b) shows an example of a tetrahedrization where no recursive closure rule is applied to neighbors after a tetrahedron has been refined. Each refined tetrahedron was split into eight sub-tetrahedra.



(a) (b)

Fig. 1. Adaptively refined tetrahedrizations constructed (a) using the red-green algorithm and (b) by not refining neighbors.

If we switch to a refinement strategy which does not maintain a valid tetrahedrization then each tetrahedron cannot have a linear function defined over it and still result in a continuous function across the tetrahedrization. We need to use a different method for defining the volume model over this type of decomposition. In this work we propose the use of a tetrahedral Coons volume defined over each tetrahedron. This will allow us to ignore the incompatibilities between tetrahedra and refine only those tetrahedra that need to be refined based on local error estimates.

This work builds on that presented in [10]. The authors presented a new method for adaptively approximating terrain data and adaptively tessellating parametric surfaces using Coons patches. The idea was to adaptively refine triangles without worrying about also refining neighboring triangles. The approximation to the data then consisted of a collection of Coons patches defined over the triangulation.

1.2 Previous Work

Several different methods of approximating volume data have appeared in the literature. Tensor-product wavelets have been applied to regular volume data to generate multiresolution approximations [7, 8, 4]. Another approach, used by Grosso et. al. [3], is to generate coarse-to-fine approximations using piecewise linear functions over tetrahedrizations. This approach requires a valid tetrahedrization (i.e., one with no T-vertices, see [9]) which is obtained by using the red-green algorithm of [1]. A sequence of approximations of regular volume data can also be generated in a “bottom-up” manner as done in [13]. They start with a tetrahedrization of regular data and merge tetrahedra based on local error estimates. For a different approach to volume models over nonconforming meshes based upon projection operators, we alert the reader to the work of Ohlberger and Rumpf [12].

Our work bears some similarity to that presented in [3]. We generate coarse-to-fine adaptive approximations using functions over tetrahedrizations. The differences are that we use a tetrahedral Coons volume over each tetrahedron (versus linear functions) and so we do not require a valid tetrahedral decomposition.

2 Triangular Coons Patches and Tetrahedral Coons Volumes

2.1 Introduction

Coons patches and volumes have been used in geometric modeling to define patches or volumes that interpolate prescribed boundary curves or functions [9, 11]. The methods generate a smooth surface or volume by blending data from the boundaries in a systematic way. We will first cover triangular Coons patches and then present their extension to tetrahedral Coons volumes. These will then be used in tandem to form approximations to volume data.

2.2 Triangular Coons Patches

The domain of a triangular Coons patch is a triangle so it will be convenient to write points on the surface using barycentric coordinates. The barycentric coordinates of (u_0, u_1, u_2) of a point (x, y) are defined by the relation

$$\begin{bmatrix} x \\ y \end{bmatrix} = u_0\mathbf{P_0} + u_1\mathbf{P_1} + u_2\mathbf{P_2} \tag{1}$$

$$1 = u_0 + u_1 + u_2 \tag{2}$$

where $\mathbf{P_0}$, $\mathbf{P_1}$, and $\mathbf{P_2}$ are vertices of the domain in $\mathbb{E}^2$.

In order to create a triangular Coons patch we require three compatible boundary curves. We make no requirements on the nature of the boundary curves so we will refer to them as functions defined in terms of boundary edges on the domain triangle. The three boundary curves are denoted as $\mathbf{F_0}(s)$, $\mathbf{F_1}(s)$, and $\mathbf{F_2}(s)$ where $\mathbf{F}$ is an underlying function defined on the boundary of a triangular domain and the parameter s

varies from 0 to 1. The boundary curves must be compatible so we require, for example, that $\mathbf{F_0}(0) = \mathbf{F_2}(1)$.

The type of triangular Coons patch used here, the NTW linear/linear patch, was first presented in [11] and more recently appeared in [10]. A surface $\mathbf{S}$ is written as the sum of three components $\mathbf{S_0}(u_1, u_2)$, $\mathbf{S_1}(u_0, u_2)$ and $\mathbf{S_2}(u_0, u_1)$. These components are given by

$$\mathbf{S_0}(u_1, u_2) = \mathbf{F}(u_1\mathbf{P_1} + (1 - u_1)\mathbf{P_0}) + \mathbf{F}(u_2\mathbf{P_2} + (1 - u_2)\mathbf{P_0}) - \mathbf{F}(\mathbf{P_0}) \quad (3)$$

$$\mathbf{S_1}(u_0, u_2) = \mathbf{F}(u_0\mathbf{P_0} + (1 - u_0)\mathbf{P_1}) + \mathbf{F}(u_2\mathbf{P_2} + (1 - u_2)\mathbf{P_1}) - \mathbf{F}(\mathbf{P_1}) \quad (4)$$

$$\mathbf{S_2}(u_0, u_1) = \mathbf{F}(u_0\mathbf{P_0} + (1 - u_0)\mathbf{P_2}) + \mathbf{F}(u_1\mathbf{P_1} + (1 - u_1)\mathbf{P_2}) - \mathbf{F}(\mathbf{P_2}). \quad (5)$$

A point on the patch is written as a barycentric combination of the three components and is given by

$$\mathbf{S}(u_0, u_1, u_2) = u_0\mathbf{S_0} + u_1\mathbf{S_1} + u_2\mathbf{S_2}. \quad (6)$$

Figure 2 illustrates a patch with piecewise linear boundaries.

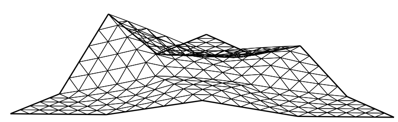


Fig. 2. NTW linear/linear patch that interpolates to piecewise linear boundaries.

2.3 Tetrahedral Coons Volumes

We would like to extend the idea of a triangular Coons patch to define a scalar-valued tetrahedral Coons volume. The volume will be written using barycentric coordinates over a tetrahedral domain and it will be constructed in such a way that it interpolates to four compatible scalar-valued boundary functions.

The tetrahedral Coons volume requires four compatible scalar-valued boundary functions. Each function is defined over a triangular domain and associates a scalar value with a point in its domain. If we denote the vertices of a tetrahedral domain as $\mathbf{P_0}$, $\mathbf{P_1}$, $\mathbf{P_2}$, and $\mathbf{P_3}$ then one compatibility requirement necessitates that $F(\mathbf{P_1}, \mathbf{P_2}\mathbf{P_3})$ and $F(\mathbf{P_0}, \mathbf{P_2}\mathbf{P_3})$ must have the same values along the edge $\mathbf{P_2}$ to $\mathbf{P_3}$ in the domain.

The tetrahedral Coons volume used here is an extension of the NTW linear/linear patch. A volume $V(u_0, u_1, u_2, u_3)$ is written as the sum of four components $V_0(u_1, u_2, u_3)$, $V_1(u_0, u_2, u_3)$, $V_2(u_0, u_1, u_3)$, and $V_3(u_0, u_1, u_2)$. The expression for $V_0(u_1, u_2, u_3)$ is given as follows

$$
\begin{aligned}
V_0(u_1,u_2,u_3) = {} & F(u_1\mathbf{P_1} + u_2\mathbf{P_2} + (1-u_1-u_2)\mathbf{P_0}) + \\
& F(u_2\mathbf{P_2} + u_3\mathbf{P_3} + (1-u_2-u_3)\mathbf{P_0}) + \\
& F(u_3\mathbf{P_3} + u_1\mathbf{P_1} + (1-u_3-u_1)\mathbf{P_0}) - \\
& F(u_1\mathbf{P_1} + (1-u_1)\mathbf{P_0}) - \\
& F(u_2\mathbf{P_2} + (1-u_2)\mathbf{P_0}) - \\
& F(u_3\mathbf{P_3} + (1-u_3)\mathbf{P_0}) + \\
& F(\mathbf{P_0}). \qquad (7)
\end{aligned}
$$

In a manner similar to the triangular Coons patch, the value of the volume corresponding to barycentric coordinates (u_0, u_1, u_2, u_3) is written as a convex combination of the four components and is given by

$$V(u_0, u_1, u_2, u_3) = u_0V_0 + u_1V_1 + u_2V_2 + u_3V_3. \qquad (8)$$

There are other types of triangular Coons surfaces which differ in the way in which the components of a surface are defined and combined [9, 11]. The transfinite scheme used here was chosen primarily because of its simplicity and the ease with which it generalizes to tetrahedral volumes.

We will next describe the way in which triangular Coons patches and tetrahedral Coons volumes will be used to approximate volume data. Using the ideas presented in this section we will be able to build models of volume data in which the need for maintaining a valid tetrahedrization is eliminated.

3 Adaptive Approximations using Tetrahedral Coons Volumes

3.1 Algorithm

The algorithm that we present will be used to adaptively approximate regular volume data. We require that the data set is of size $(2^n+1)\times(2^n+1)\times(2^n+1)$. The reason for this is that vertices in the tetrahedrization will be associated with points in the input. As tetrahedra are refined, new vertices will continue to correspond to data points. In particular, a tetrahedron will be refined by splitting each edge at its midpoint and joining those to form new tetrahedra. Because of the special size of the data set, the midpoint of each edge will also correspond to a data point.

We will now describe the face functions that will be used in order to define tetrahedral Coons volumes over a tetrahedrization like that shown in Figure 1(b). Each face function is scalar-valued and is defined over a triangular domain. Because a tetrahedron may have one of its faces shared by many other tetrahedra, the domain of a face function can be thought of as an adaptive triangular decomposition.

In order to define a continuous function across an adaptively refined domain we define a triangular Coons patch over each triangle in the domain of a face function. The portions of the function that correspond to triangles in the domain without T-vertices are simply planar triangles. If all the face functions for a tetrahedron are single triangles without adaptive refinements then the Coons volume evaluates to a linear function over the tetrahedron.

The algorithm to adaptively approximate volume data is shown in Figure 3. Since the original data is being approximated, we require a user-specified error tolerance, ϵ, for the fitting process. The initial tetrahedrization for the approximation consists of the unit cube tetrahedrized into six tetrahedra where each tetrahedron shares the main diagonal of the cube. We also define a maximum level that prevents the tetrahedra from being subdivided too many times.

There are several strategies for determining if a tetrahedron requires refinement. We refine a tetrahedron if the difference between the weight associated with any data point inside it and the value of the Coons volume evaluated at the data point's location exceeds a tolerance. Different strategies might include comparing the average or the median of the differences to a threshold.

```
repeat
{
    for each tet T_i ∈ 𝒯
    {
        for each data point P = (x, y, z; w) inside T_i
        {
            // convert (x,y,z) to barycentric coordinates with respect to T_i
            // evaluate point using Coons volume associated with tet T_i
            // see equation (8)
            ŵ := V_{T_i}(u_0, u_1, u_2, u_3)
            if (|w − ŵ| > ε)
            {
                mark T_i for refinement
            }
        }
    }
    for each tet T marked for refinement
    {
        if (T not at maximum level)
        {
            split T into eight sub-tetrahedra
        }
    }
} until no tets refined
```

Fig. 3. Algorithm for adaptively approximating regular volume data.

3.2 Results

We demonstrate our adaptive approximation method on two data sets. The first is a synthetic data set of size 33^3 where the dependent values, w, have been computed by

$$\begin{aligned} w = f(x,y,z) = & \frac{1}{2}e^{(-10((x-0.25)^2+(y-0.25)^2))} \\ & +\frac{3}{4}e^{(-16((x-0.25)^2+(y-0.25)^2+(z-0.25)^2))} \\ & +\frac{1}{2}e^{(-10((x-0.75)^2+(y-0.125)^2+(z-0.5)^2))} \\ & -\frac{1}{4}e^{(-20((x-0.75)^2+(y-0.75)^2))}. \end{aligned} \quad (9)$$

The following table summarizes the fitting process for this data set. The user-specified tolerance is given followed by the number of vertices and tetrahedra in the resulting tetrahedrization. The number of volumes indicates the number of tetrahedra that have non-trivial face functions and must be evaluated as tetrahedral Coons volumes.

tolerance	vertices	tetrahedra	volumes	rms error
0.05	756	1455	558	0.0283
0.02	3110	6173	2220	0.00914
0.01	7659	16141	4934	0.00421

Table 1. Statistics for the approximations shown in Figure 4.

In order to visualize the approximations we performed a regular sampling of the Coons volumes and then use marching cubes to generate isosurfaces and normals for shading. The sampled data was also used to compute the rms errors. They were computed by taking the sum of the squares of the differences between the weight associated with a data point and the value of the Coons volume evaluated at that location.

Figure 4 shows the results of the fitting process for this data set. The top row of the figure shows isosurfaces from the original data set (the left is threshold 0.21 and the right is 0.50). The next three rows are the approximations (tetrahedrizations and isosurfaces) using tolerances 0.05, 0.02, and 0.01, respectively.

The second data set is a 33^3 data set from a MRI scan. The top row of Figure 5 shows an isosurface computed from the data set (isosurface threshold 0.095). The following table summarizes the statistics for several approximations of this data set.

Figure 5 shows results of applying our adaptive approximation algorithm to this data set. The top row is an isosurface computed from the original data. Each row thereafter shows an approximation in the form of a tetrahedrization and an isosurface for tolerances 0.07 and 0.03. The isosurface threshold is the same as that used on the original data set.

tolerance	vertices	tetrahedra	volumes	rms error
0.07	18493	35452	11181	0.0232
0.03	26081	63712	10432	0.00398

Table 2. Statistics for the approximations shown in Figure 5.

4 Summary

We have presented a method of performing adaptive approximations of regular volume data using tetrahedral Coons volumes. The advantage of using Coons volumes over existing approaches is that a valid tetrahedrization does not need to be maintained. Only those tetrahedra in which the error is large (i.e., areas where the model does not adequately approximate the data) need to be refined instead of also needing to refine neighboring tetrahedra like existing local refinement algorithms.

Future work includes applying a least squares approach to the fitting process to approximate data that does not meet the size requirements as given in Section 3.1.

5 Acknowledgments

We wish to acknowledge the support of the Office of Naval Research under grant N00014-97-1-0243, the support of the National Aeronautical and Space Administration under NASA-Ames Grant NAG 2-990, and the support of the National Science Foundation under grant IIS 9980166.

References

1. Jürgen Bey. Tetrahedral mesh refinement. *Computing*, 55(4):355–378, 1995.
2. Robert A. Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In John Dill, editor, *SIGGRAPH 88 Conference Proceedings*, Annual Conference Series, pages 65–74. ACM SIGGRAPH, Addison Wesley, August 1988.
3. Roberto Grosso, Christoph Lürig, and Thomas Ertl. The multilevel finite element method for adaptive mesh optimization and visualization of volume data. In Roni Yagel and Hans Hagen, editors, *Proceedings of Visualization '97*, pages 387–394. IEEE Computer Society Press, Novemeber 1997.
4. Insung Ihm and Sanghun Park. Wavelet-based 3D compression scheme for interactive visualization of very large volume data. *Computer Graphics Forum*, 18(1):3–15, March 1999.
5. William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In Maureen C. Stone, editor, *SIGGRAPH 87 Conference Proceedings*, Annual Conference Series, pages 163–169. ACM SIGGRAPH, Addison Wesley, July 1987.
6. Joseph M. Maubach. Local bisection refinement for N-simplicial grids generated by reflection. *SIAM Journal on Scientific Computing*, 16(1):210–227, January 1995.
7. Shigeru Muraki. Approximation and rendering of volume data using wavelet transforms. In Arie E. Kaufman and Gregory M. Nielson, editors, *Proceedings of Visualization '92*, pages 21–28. IEEE Computer Society Press, Oct 1992.

8. Shigeru Muraki. Volume data and wavelet transforms. *IEEE Computer Graphics and Applications*, 13(4):50–56, July 1993.
9. Gregory M. Nielson, Hans Hagen, and Heinrich Müller. *Scientific Visualization: Overview, Methodologies, and Techniques*, chapter Tools for Triangulations and Tetrahedrizations, pages 429–525. IEEE Computer Society, Los Alamitos, CA, 1997.
10. Gregory M. Nielson, David J. Holliday, and Tom Roxborough. Cracking the cracking problem with Coons patches. In David Ebert, Markus Gross, and Bernd Hamann, editors, *Proceedings of Visualization '99*. IEEE Computer Society Press, Novemeber 1999.
11. Gregory M. Nielson, Donald H. Thomas, and James A. Wixom. Boundary data interpolation on triangular domains. Technical Report GMR-2834, General Motors Research Laboratories, 1978.
12. Mario Ohlberger and Martin Rumpf. Adaptive projection operators in multiresolution scientific visualization. *IEEE Transaction on Visualization and Graphics*, 4(4):344–364, 1998.
13. Yong Zhou, Baoquan Chen, and Arie Kaufman. Multiresolution tetrahedral framework for visualizing regular volume data. In Roni Yagel and Hans Hagen, editors, *Proceedings of Visualization '97*, pages 135–142, October 1997.

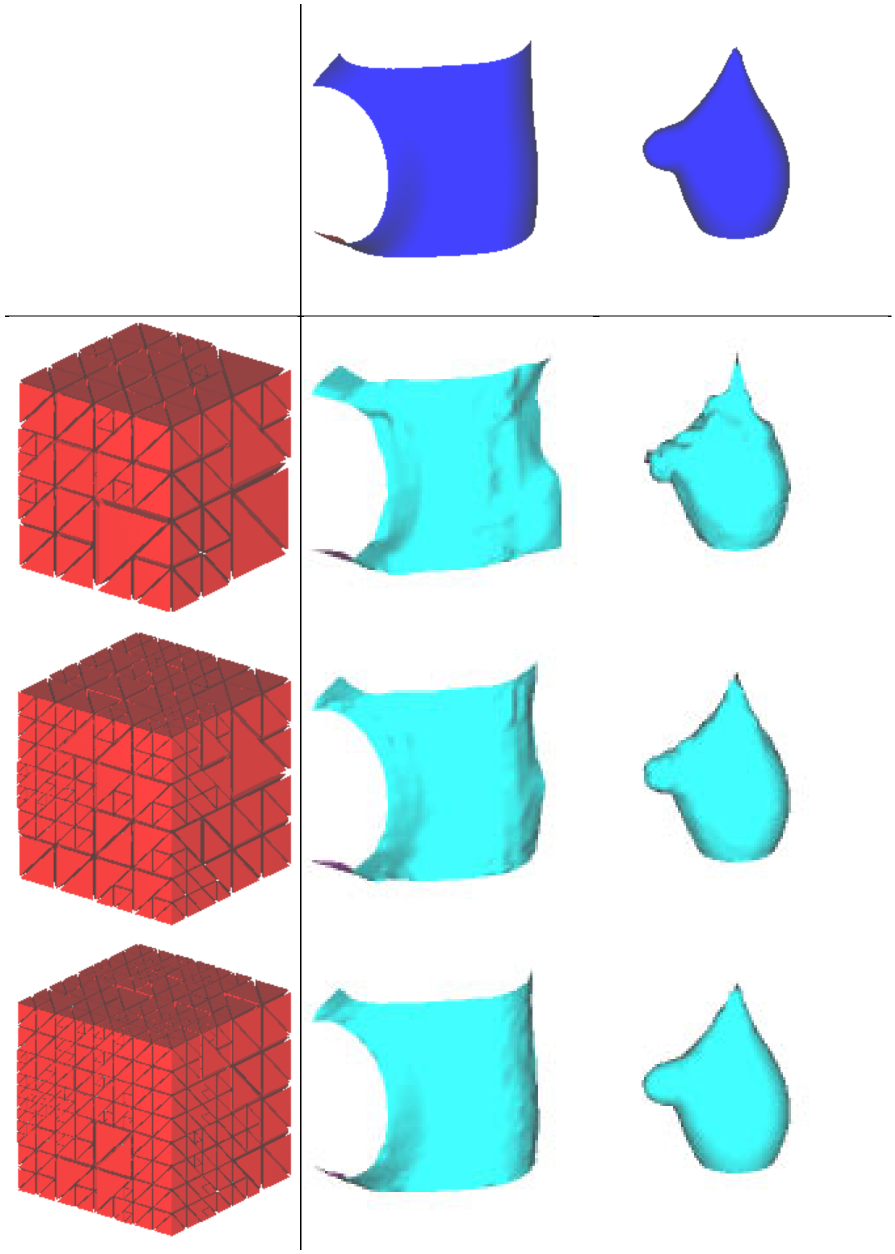


Fig. 4. An adaptively approximated approximated synthetic data set. The top row shows isosurfaces of the original data set. The next rows show the tetrahedrization resulting from a fit and the isosurfaces computed from the model.

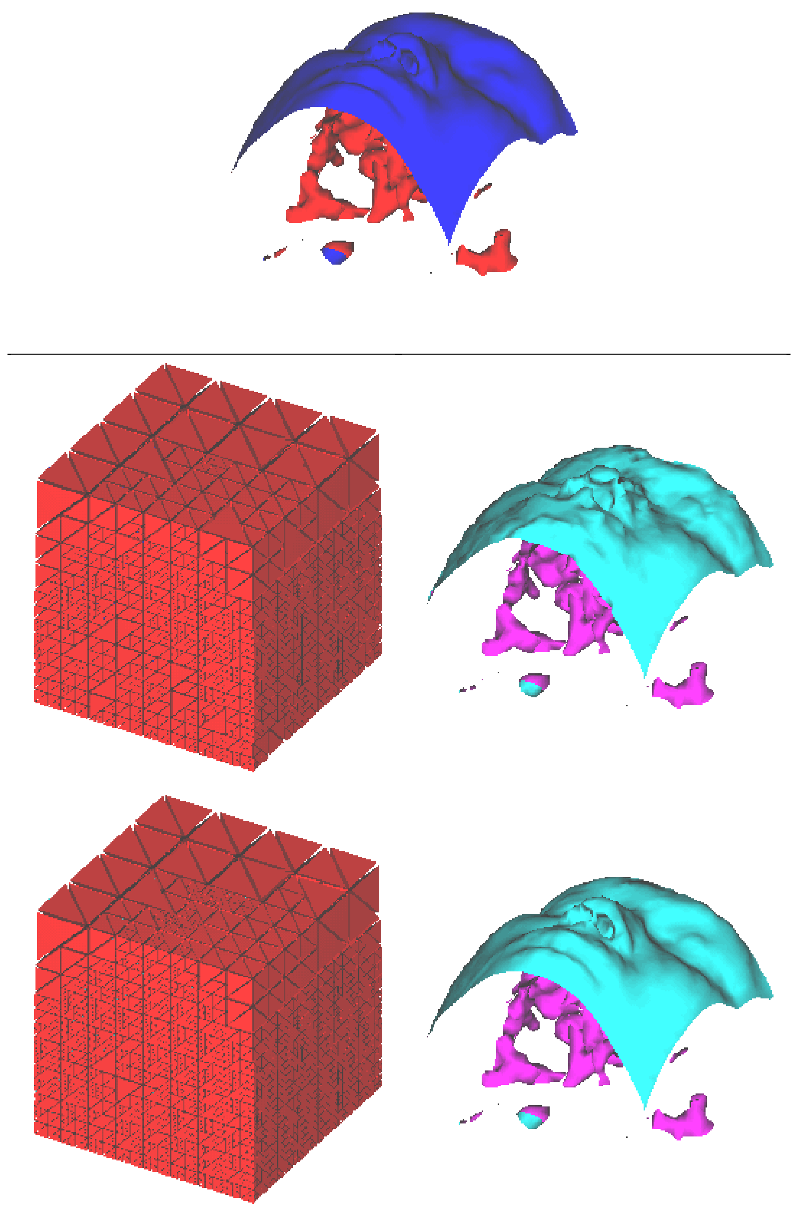


Fig. 5. An adaptively approximated approximated MRI data set. The tetrahedrization resulting from a fit and the isosurfaces computed from the model are shown for various approximations.