Physically-Based Methods for Polygonization of Implicit Surfaces

Luiz Henrique de Figueiredo†
Jonas de Miranda Gomes†
Demetri Terzopoulos‡
Luiz Velho†‡

† IMPA – Instituto de Matemática Pura e Aplicada
Estrada Dona Castorina, 110, Rio de Janeiro, Brazil, 22460

‡ University of Toronto – Department of Computer Science
Toronto, Ontario, M5S-1A4

Abstract

We present discrete physically-based methods for generating polygonal approximations of implicit surfaces. These methods not only generate a combinatorial manifold approximating the surface, but also produce a structure that is well suited to numerical simulations in physically-based modeling and animation systems.

Keywords: implicit models, physically-based models, polygonization, triangulation, domain decomposition.

1 Introduction

Consider a differentiable function $F : R^n \to R$, for which 0 is a regular value. This means that the gradient vector

$$\nabla F(p) = \left[\frac{\partial f}{\partial x_1}(p), \frac{\partial f}{\partial x_2}(p), \cdots, \frac{\partial f}{\partial x_n}(p)\right] \tag{1}$$

is non-zero at all points p in the inverse image $M = F^{-1}(0)$. In this case, the set M is a differentiable manifold of dimension $n-1$ that we shall simply call an *implicit manifold* (Spivak, 1965).

Recently the use of implicit surfaces has attracted the attention of researchers in geometric modeling. Implicit surfaces are suitable for applying visualization techniques based on ray-tracing (see (Hanharan, 1983) (Barr, 1986)), but some difficulties arise when we try to sample or structure points on them in order to gain more information about their topology and geometry (Figueiredo, 1991). One of the important issues in this sampling and structuring problem is the computation of polygonal approximations to the surface. Polygonal approximations enable us to use the fast, special purpose processors of graphic workstations in order to display implicit surface models.

1.1 Polygonization of Implicit Surfaces

To capture the geometry of an implicit manifold, we must sample and structure points on it. In this paper, our objective is to structure the points in order to obtain a combinatorial manifold $\widetilde{M}$ that is close to M in some suitable topology. The manifold $\widetilde{M}$ is called a *polygonal approximation* of the surface M.

Polygonal approximations to implicit manifolds were first described in the classic paper (Allgower & Schmidt, 1985). The method proposed by Allgower and Schmidt consists of the following steps:

1. Compute a triangulation of the ambient space;
2. Replace the function F by its simplicial approximation $\widetilde{F}$ relative to this triangulation;
3. Refine the triangulation so that $\widetilde{F}$ is close to F. The combinatorial manifold is then obtained as the inverse image $\widetilde{F}^{-1}(0)$ of the simplicial approximation.

The Freudenthal triangulation is the simplest triangulation in R^n: the space is subdivided into cubes and the triangulation is obtained by subdividing each n-cube into $n!$ simplices. Figure 1 shows a two dimensional example. For more details the reader should consult (Allgower & Georg, 1990).

Several variations of Allgower's method exist in the graphics literature (Wyvill *et al.*, 1986), (Loreson & Cline, 1987), (Bloomental, 1988), (Velho, 1989), (Hall & Warren, 1990). The correct computation of polygonal approximations to implicit manifolds depends on *a priori* estimates of the variation of the surface geometry (this is the refinement step (3) in the above algorithm). For this reason, some of the aforementioned works involve the computation of adaptive polygonizations in order to get better approximations.

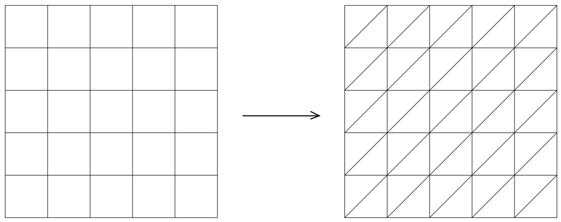


Figure 1: Freudenthal triangulation

1.2 Physically-based Modeling

Modeling is the most labor intensive part in the process of computer graphics. Modeling the motion of objects is often very difficult when the main goal is to generate realistic motion. The best solution to this problem is to model the physical habitat of the object: the motion will be a consequence of the interaction between the object and its environment, according to the laws of physics. A discussion of this physically-based modeling approach is found in several papers in the graphics literature (Barr *et al.*, 1987), (Terzopoulos & Fleischer, 1988).

1.3 Physically-based Polygonization of Implicit Objects

In this paper we use physically-based methods to compute polygonal approximations. These methods yield naturally adapted polygonizations. They also make it possible to construct a model such that the resulting polygonization has a natural physical structure associated with it which can be exploited for physically-based simulations.

Physically based methods in the study of implicit surfaces is a very recent research topic. In (Velho & Gomes, 1991) a spring-mass model is used to construct an adapted shell that approximates the geometry of the manifold. In (Velho & Gomes, 1991a), it is shown how this spring-mass shell can be used to do dynamical simulations with implicit models. In (Figueiredo, 1991), physically-based particle systems are used to sample points on an implicit manifold; algorithms for structuring such samples provide a powerful technique for modeling with implicit surfaces.

The physically-based approach to constructing piecewise linear approximation of implicit manifolds is related to the variational methods used to generate adaptive numerical grids for the numerical solution of partial differential equations (Thompson *et al.*, 1985). However, there are two main differences:

- To our knowledge, the adaptive methods in the numerical grid generation literature are developed for structured grids. The problem of polygonization of implicit surfaces is a non-structured one.
- In numerical grid generation, the physics of the associated problems may drive the adaptation of the grid. In our case, the primary interest is in the geometry and topology of the underlying grid space.

Our methods can certainly be used to generate adaptive numerical grids for problems where the physical domain can be defined implicitly. In fact, polygonization methods for implicit surfaces seem to be a very attractive technique for generating non-structured numerical grids.

In this paper, we are interested in polygonizations that are regular or "quasi-regular" triangulations. A *quasi-regular* triangulation is a 2-dimensional simplicial complex which is constituted by elements that are almost equilateral and equiangular. This type of polygonization is desirable in a number of applications to modeling and numerical simulation.

1.4 Overview

Section 2 describes the two discrete physical systems that we use to construct the polygonal approximation. Section 3 describes the polygonization algorithm using physically-based particle systems. Section 4 describes the polygonization algorithm using a spring-mass physical model. Section 5 gives examples and makes some comparisons between the two approaches described. Section 6 closes with a brief description of our current work in this area.

2 Discrete Physical Systems

A *discrete physical model* abstracts matter as an ensemble of particles related to each other by forces. Several physical phenomena may be naturally modeled using discrete physical systems (Greenspan, 1973). In a discrete physical system the particles interact under the action of internal and external forces. The associated motion equations are easily written as a classical $F = ma$ equation of Newtonian dynamics. Simple numerical integration methods, such as Euler's method generally produce good results.

In this work, we use two discrete physical models: a particle system and a spring-mass system.

2.1 Dynamic Particle Systems

A *particle system* is a finite set of particles which have an initial position in space and whose behavior in time is governed by algorithmic rules. Particle systems were introduced in graphics by Reeves as an algorithmic technique for modeling fire explosions (Reeves, 1983). In a *physical particle system*, the particles have masses and the Newtonian mechanics dictates their dynamical behavior. The motion of a particle depends on its mass, position and velocity, and on the forces acting on it, either by other particles or by the ambient medium. A *physical particle system* is a discrete physical model as defined above.

Physical particle systems have been used to simulate natural phenomena such as waterfalls (Sims, 1990) and fireworks (Weil, 1987). These systems in general require a significant amount of computational effort because of the number of particles involved. In Section 3 we shall use a simple physical particle system to compute a polygonal approximation to an implicit manifold. More recently, (Szeliski & Tonnesen, 1991) have applied physical particle systems to surface modeling.

2.2 Spring-Mass Systems

A *spring-mass system* is a physical particle system structured by connecting pairs of particles with springs. The springs impose *internal forces* that depend on the distance between these particles and govern the global behavior of the system. The resulting structure can be represented as a graph, where each particle is a node, and two nodes are connected when there is a spring joining the corresponding particles. Conversely, each graph linearly embedded in the space is naturally associated to a spring-mass system — a duality that will be exploited in Section 4 for triangulations.

Spring-mass systems are suitable to create physically-based models of deformable objects for dynamical simulation (Haumann, 1987), (Terzopoulos *et al.*, 1989). In the recent paper (Terzopoulos & Vasilescu, 1991), a spring-mass system is applied to adaptive image sampling and surface reconstruction. This approach has several connections with our method.

3 Polygonization using Dynamic Particle Systems

In this section, we describe an algorithm for computing a polygonal approximation of an implicit manifold using a physically-based particle system (Figueiredo, 1991).

3.1 Sampling using Dynamic Particle Systems

To properly sample a geometric object we must compute enough points on it so that its geometry can be reconstructed from the samples within some tolerance. In the case of a manifold given implicitly by a differentiable function $F : R^n \to R$, such a computation requires finding several solutions of the equation $F(x) = 0$. Physically-based methods for the solution of nonlinear equations have been known for some time (Incerti et al., 1979), although it seems that the main interest then was in finding any one solution, and not the many solutions that sampling requires. Consequently, these methods have not been applied to geometric modeling.

The particle systems we use for sampling derive their dynamics from the potential function $|F|$. The particles will seek equilibrium positions on the manifold $F^{-1}(0)$ because these are positions of minimum potential energy. If the gradient of F is non-singular, then these are the only equilibrium positions.

This interpretation of the gradient of $|F|$ as a force field implies the following equation of motion for a unit mass particle:

$$\frac{d^2x}{dt^2} + \gamma\frac{dx}{dt} + \mathrm{sign}(F)\nabla F = 0, \tag{2}$$

where γ is a positive real number representing friction proportional to velocity. (Incerti et al., 1979) have proposed a similar differential equation for finding zeros of functions $R^n \to R^n$.

3.2 Structuring Samples

The samples obtained by simulating the physics of particle systems have no structure other than the equilibrium position of each particle. Moreover,the samples are not evenly distributed across the surface, but rather tend to concentrate around points of high curvature. While this could be exploited for investigations on the geometry of the surface, a polygonal approximation interpolating such samples will rarely be quasi-regular.

In order to obtain a quasi-regular approximation, the sample is subjected to a relaxation process similar to the one used by (Turk, 1991) and (Szeliski & Tonnesen, 1991): particles repel each other with an intensity that rapidly decreases as the distance between the particles increases. Moreover, the movement of each particle is constrained to stay close to the surface by projecting repulsion forces onto the tangent plane.

The result of this relaxation process is a more uniform sampling of the surface. The desired polygonal approximation is then obtained by computing the Delaunay triangulation associated with the points and choosing the triangles that approximate well the tangent planes at each of its vertices.

4 Polygonization using Spring-Mass Systems

In this section, we describe a method to construct a polygonal approximation to an implicit manifold using a spring-mass system.

4.1 Subordinated Triangulation

Initially we define a system of spring-mass elements associated with a Freudenthal triangulation of the space. Like the particle systems described in Section 3.1, this system is subjected to deformation forces derived from the gradient field of the implicit manifold. Its equilibrium position gives a triangulation of a region of the space that contains the manifold M and has the following properties:

- M is transversal to the triangulation;

- The simplices are quasi-regular;

- For each n-simplex σ that intersects M there exists a point $p \in M$ close to the barycenter of σ such that the tangent space of M at p is close to the support hyperplane of one of the faces of σ.

Figure 2 illustrates the properties above in two dimensions. A triangulation with these properties is said to be *subordinated to the surface M* (Velho & Gomes, 1991).

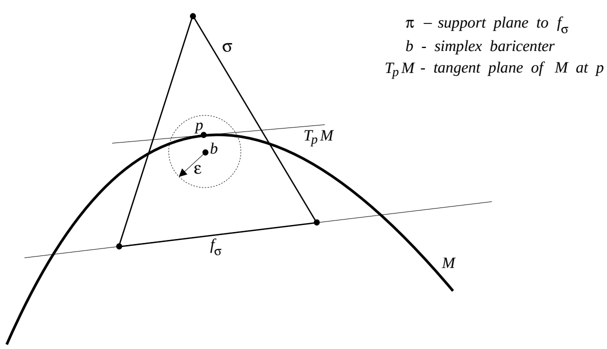


Figure 2: Subordinate triangulation

4.2 Mesh Generation

The spring-mass lattice generation process requires the following steps:

1. A Freudenthal triangulation is created within a volume bounding the implicit manifold;

2. Each simplex of the triangulation that intersects the implicit manifold is identified. Together, they form an intersecting simplicial complex;

3. A spring-mass system is created by associating mass nodes and springs to the vertices and edges of the intersecting complex.

The construction of the Freudenthal triangulation in step 1 is obtained as explained in Section 1.1. The identification of the relevant simplices in step 2 is done through a classification of the simplicial cells by testing the sign of the implicit function at the vertices of each simplex. Assuming that the uniform grid is sufficiently fine, if the signs are the same for all vertices, the simplex must be totally inside or totally outside of the manifold M. If the signs are different, then the simplex must intersect the surface M.

4.3 Mesh Deformation

After generating the mesh we use a physically-based approach in order to obtain the final triangulation that will be used for the polygonization of M. The dynamic simulation submit the spring-mass system to deformation forces with the purpose of conforming it to the shape of the implicit manifold. The process takes into account the internal forces produced by the springs as well as external deformation forces.

The external forces are based on information derived from the geometry of the implicit manifold. More specifically, two opposite attracting and repulsing force fields are generated using the gradient vector field of the implicit manifold. One field defined inside a small neighborhood of the object's boundary generates repelling forces that prevent points from being too close to the surface. The other force field, defined outside this neighborhood, generates attraction forces that pulls points towards the surface.

In order to facilitate the relaxation of the mesh structure into the desirable configuration, the initial rest length of the strings is made smaller than the initial grid spacing. This means that we start the process with a tensioned mesh that moves to a rest position under the action of internal and external forces.

4.4 Polygonization

The polygonization of the implicit manifold M is now obtained using the same technique of Allgower's algorithm described in Section 1.1: since the triangulation obtained is subordinated to M, the manifold intersects each 3-simplex σ in at most 4 distinct points, each one located on a different 1-dimensional face. Therefore, the linear approximation to M inside σ is formed by one or two triangles (2-simplices). The set of all these simplices constitute the combinatorial manifold that approximates M. We shall illustrate the method with some examples in section 5.1.

5 Results

In this section, we show the result of applying the two methods described in Sections 3 and 4 to compute polygonal approximations of implicit surfaces. We also make a comparative analysis of the polygonizations obtained and discuss the differences and similarities between the two methods.

5.1 Examples

Figures 3 and 4 illustrate the polygonization method using the particle systems presented in Section 3. Figure 3-a shows the trajectories of a particle system associated with a two-dimensional curve with 2 connected components described by the implicit equation $y^2 - x^3 + x = 0$. Figure 3-b shows the final equilibrium positions of these particles along the curve. Figure 4-a shows the sample points on the surface of the sphere $x^2 + y^2 + z^2 = 1$. Figure 4-b shows the polygonal approximation for the sphere.

Figures 5 to 7 illustrate the polygonization using the spring-mass system method presented in Section 4. Figure 5 demonstrates the mesh deformation process for the cylinder $x^2 + y^2 = 1$. Figure 5-a depicts the initial mesh created from a Freudenthal triangulation of the ambient space, Figure 5-b shows the final mesh in its equilibrium position. It is apparent that the mesh was constrained to lie in a tubular neighborhood of the implicit surface, conforming to the cylinder's shape. The polygonal approximation is obtained from this deformed mesh.

Figure 6 shows a detail of the polygonization associated with the spring-mass mesh before (a) and after (b) the deformation process. Note how the deformation of the mesh produces a very homogeneous polygon structure, transforming long, thin elements to nearly equilateral ones. This is because the triangulation resulting from the dynamical simulation is subordinate to the surface; as a consequence, the associated polygonization is quasi-regular.

Figure 7 shows the final polygonal approximation for the cylinder.

5.2 Comparisons

The main difference between the two methods presented in this paper is related to the order in which the operations of sampling and structuring of points on the implicit surface are performed.

The dynamical particle systems method in Section 3 first generates samples of the implicit object and subsequently structures these samples in order to create a polygonal approximation of the object.

The spring-mass systems method of Section 4 does the opposite. First the structure is created from a regular tessellation of space and second, this structure is used to sample the implicit object.

It is interesting to note that the physically-based approach is applied only to the sampling process. The structuring operation involves combinatorial methods.

The two methods produce equally good polygonal approximations of implicit surfaces. The combinatorial manifold generated by them is constituted by "almost fat" triangles.

The dynamical systems employed in both methods are very stable. The convergence to an equilibrium state is in general reasonably fast, requiring a small number of time steps (usually less than 100).

6 Conclusions

We have presented a new approach for the polygonization of implicit surfaces based on physically-based methods. The two methods described exploit different strategies to obtain polygonizations that are quasi-regular and faithfully approximate the original implicit objects.

The use of a physically-based approach for the polygonization of implicit objects provides great flexibility and control of the resulting structure.

Although this process is computationally more expensive than traditional methods, due to the numerical simulation of a dynamical system, it produces qualitatively better results.

We are presently incorporating these polygonization methods in a modeling and animation system for implicit objects.

Our current research also includes the development of adaptive physically-based polygonization methods and the application of these methods to numerical grid generation problems for domains defined by implicit surfaces.

In relation to the method of Section 3, we are investigating higher order approximations using intrinsic Voronoi diagrams. This would enable us to do continuous deformations using spline patches.

7 Acknowledgements

This research was partially supported by CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnólogico, the Natural Sciences and Engineering Research Council of Canada and the Information Technology Research Center of Ontario.

8 References

Allgower, E. L. & Schmidt, P. H., (1985): An algorithm for piecewise-linear approximation of an implicitly defined manifold, *SIAM Journal of Numerical Analysis*, **22**, 322–346.

Barr, A., (1986): Ray Tracing Deformed Surfaces *Computer Graphics*, **20**, 4, 287-296, (Proceedings of SIGGRAPH '86).

Barr, A., Barzel, R., Haumann, D., Kass, M., Platt, J., Terzopoulos, D., Witkin, A., (1987): *Topics in Physically-based Modeling*, SIGGRAPH'87 course notes #17.

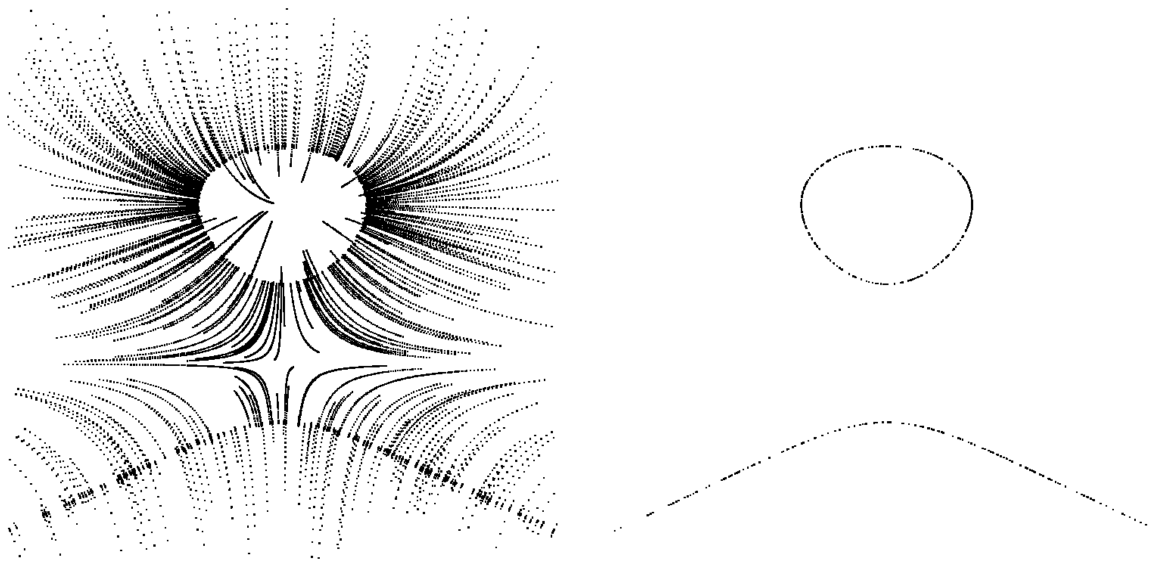
Bloomenthal, J., (1988): Polygonization of implicit surfaces, *Computer Aided Geometric Design*, **5**, 341–355.

Figueiredo, L. H., (1991): *Computational Morphology of Implicit Curves*, Ph.D. thesis in preparation, IMPA.

Greenspan, D., (1973): *Discrete Models*, Addison-Wesley, Reading, MA.

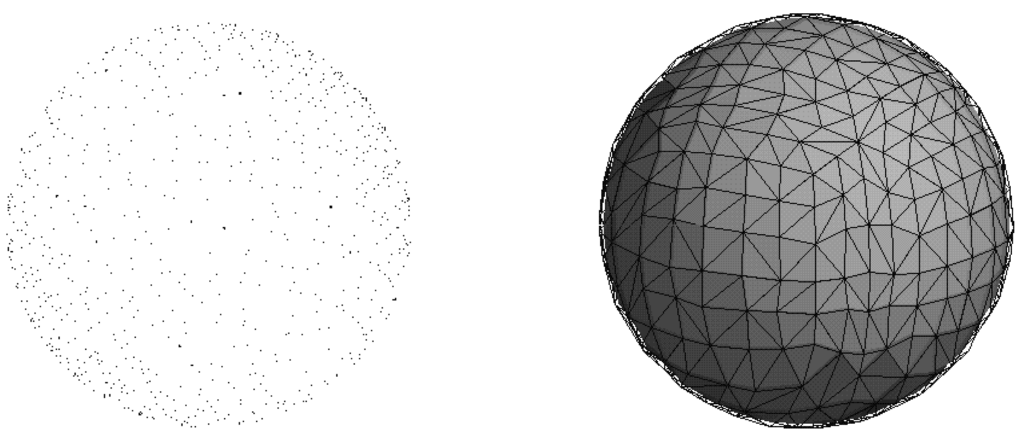
Hall, M. & Warren, J., (1990): Adaptive polygonization of implicitly defined surfaces, *IEEE Computer Graphics & Applications*, **10**, 33–42.

Hanharan, P., (1983): Ray Tracing Algebraic surfaces *Computer Graphics*, **13**, 1, 83-90, Proceedings of SIGGRAPH '83.



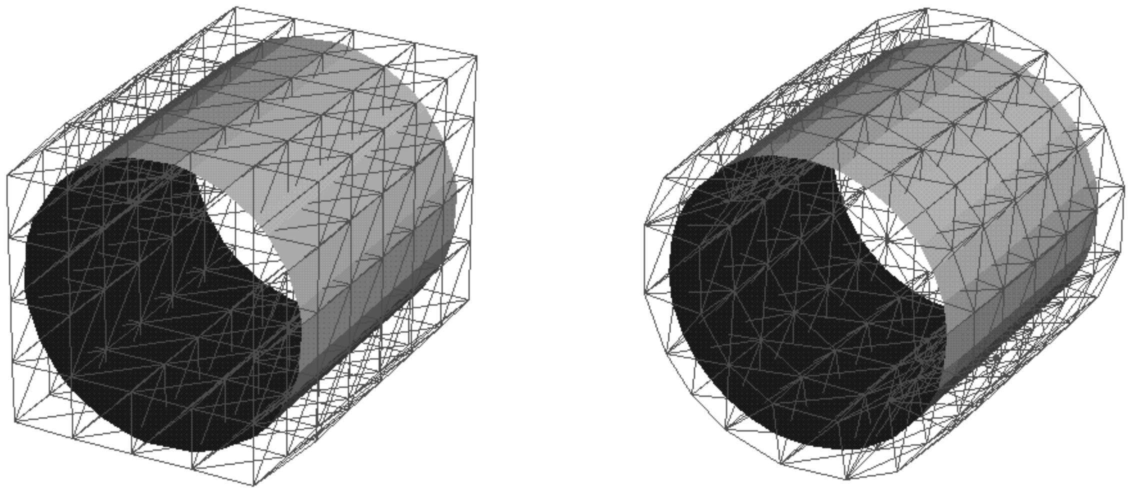
(a) (b)

Figure 3: Trajectories (a) and final positions (b) of particles for 2D curve



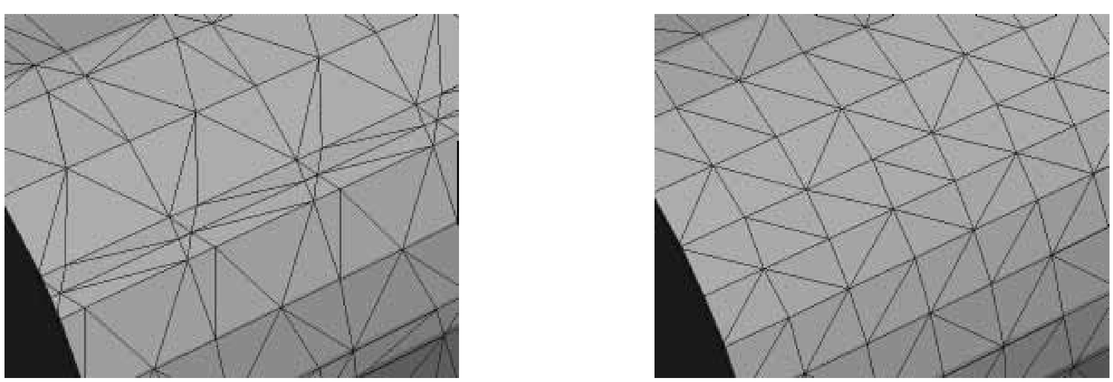
(a) (b)

Figure 4: Sample points on the surface of a sphere (a) and polygonization of the sphere (b)



(a) (b)

Figure 5: 3D mesh before (a) and after (b) deformation



(a) (b)

Figure 6: Detail of the polygonization before (a) and after (b) deformation of the mesh

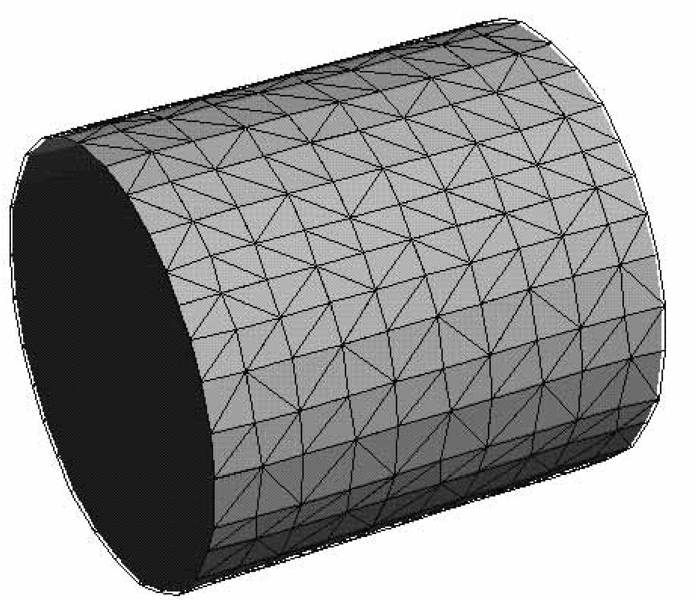


Figure 7: Final polygonization of the cylinder

Haumann, D., (1987): Modeling the physical behavior of flexible objects, in SIGGRAPH'87 course notes #17, 1–13.

Incerti, S., Parisi V., Zirilli, F. (1979): A new method for solving nonlinear simultaneous equations, *SIAM Journal on Numerical Analysis*, **16**, 779–789.

Loreson, W., & Cline, H., (1987): Marching cubes: a high resolution 3d surface construction algorithm, *Computer Graphics*, **21**, 163–169, (Proceedings of SIGGRAPH '87).

Reeves, W. T., (1983): Particle systems—a technique for modeling a class of fuzzy objects, *Computer Graphics*, **17**, 359–376, (Proceedings of SIGGRAPH '83).

Sims, K., (1990): Particle Animation and Rendering using Data Parallel Computation, *Computer Graphics*, **24**, 4, 405–414, (Proceedings of SIGGRAPH '90).

Spivak, M., (1965): *Calculus on Manifolds*, Benjamin.

Szeliski, R,. & Tonnesen, D., (1991): *Surface Modeling with Oriented Particle Systems,* CRL-91/14, Dec. 1991, Cambridge Research Lab–Digital Equipment Corporation, Cambridge, MA.

Thompson, J.F., Waisi, Z. U. Z., Mastin, C. W., (1985): *Numerical Grid Generation, Foundations and Applications*, North Holland, New York.

Terzopoulos, D., & Fleischer, K.; (1988): Deformable models, *The Visual Computer*,**4**, 306–331.

Terzopoulos, D., Platt, J., Fleischer, K., (1989): Heating and Melting Deformable Objects, *The Journal of Visualization and Computer Animation,* **2**, 2, 1991, 68–73. (also in Proceedings of Graphics Interface '89).

Terzopoulos, D. & Vasilescu, M., (1991): Sampling and Reconstruction with Adaptive Meshes, Proceedings of Computer Vision & Pattern Recognition Conference (CVPR-91), Lahaina, HI, 70–75.

Turk, G., (1991): Generating textures on arbitrary surfaces using reaction-diffusion, *Computer Graphics*, **25**, 289–298, (Proceedings of SIGGRAPH '91).

Velho, L. & Gomes, J. deM., (1991): Regular Triangulations of Implicit Manifolds using Dynamics, Proceedings of Compugraphics '91, Sesimbra, PT, 57–71.

Velho, L. & Gomes, J. deM., (1991a): A Dynamical Simulation Environment for Implicit Objects using Discrete Models, Proceedings of 2nd Eurographics Workshop on Animation and Simulation.

Weil, J., (1987): Boom Boom Boom, *(SIGGRAPH video review 1987)*, ACM SIGGRAPH, NY.

Wyvill, G., McPheeters, C., Wyvill, B. (1986): Data Structure for Soft Objects *The Visual Computer*, **2**, 4, 1986, 227–234.