Implicit Manifolds, Triangulations and Dynamics

Luiz Velho [1]
Jonas de Miranda Gomes [1]
Demetri Terzopoulos [2]

[1]IMPA – Instituto de Matemática Pura e Aplicada
Rio de Janeiro, Brazil, 22460-320
[2] University of Toronto – Department of Computer Science
Toronto, Ontario, M5S-1A4

ABSTRACT: This paper presents a method to generate triangulations of the space associated with an implicit manifold. It produces a regular structure that conforms well to the shape of the manifold. The method has applications in geometric modeling, computer animation and simulation.

1 INTRODUCTION

There are two different ways to describe a geometric object in Computer Graphics: parametric or implicit. In the *parametric* form, the points constituting the object are given directly by a local parameterization that defines a map from parameter space to object space. This is the most popular way to describe an object in Computer Graphics. In the *implicit* form, the points belonging to the object are given indirectly through a classification function that defines the relation of spatial points with the object.

In order to manipulate an implicit object more effectively is common to use a spatial subdivision enumeration associated with the object. This auxiliary representation helps in the execution of operations that are difficult to perform on the implicit representation alone.

This paper addresses the problem of constructing spatial subdivision enumerations that are well adapted to the geometry of the associated implicit object. Our goal is to obtain a regular decomposition of space reflecting the shape of the object. We will introduce a method to automatically generate a triangulation of space from an analytic description of the implicit object. The method is based on a spring-mass model and generates a spatial enumeration of regular simplicial cells associated with the manifold.

The organization of the paper is as follows: Section 2 briefly reviews implicitly defined objects. Section 3 discusses space partition schemes. Section 4 presents the spring-mass system construction. Section 5 describes the dynamic simulation algorithm. Section 6 shows some examples. Section 7 discusses applications of the method. Section 8 contains concluding remarks and current research topics.

2 IMPLICITLY DEFINED OBJECTS

In the *implicit* form, the object is defined as the inverse image, $F^{-1}(A)$, of a subset $A \subset \mathbf{R}^m$, where $F\colon U \subset \mathbf{R}^n \to \mathbf{R}^m$, is a function defined in an open subset, U of the euclidean n-space. When $m = 1$ and $A = \{0\}$, F is a point-membership classification function, that returns a value according to the relationship of a given point $p = (x_1, x_2, \dots, x_n)$ with the object $\mathcal{O}$:

$$F(x_1, x_2, \dots, x_n) \begin{cases} < 0, & \text{if } P \text{ is in the interior of the object;} \\ = 0, & \text{if } P \text{ is on the boundary of the object;} \\ > 0, & \text{if } P \text{ is on the exterior of the object.} \end{cases}$$

Intuitively the geometric objects are described implicitly in the following way: a surface in 3-space would be defined taking $n = 3$, $m = 1$ and $A = \{c\}$; a solid in 3-space would be defined taking $n = 3$ and $m = 1$, and $A = (-\infty, c]$; a curve in 3-space is defined by taking $A = \{c\}$, $n = 3$ and $m = 2$ (note that the implicit equation describes the curve as the intersection of two surfaces). In the same way we can describe implicitly geometric objects in dimensions other than 3. In general it is very difficult to study the geometric properties of these objects. By imposing some additional conditions on the function F they are differentiable manifolds. The simplest condition is *transversality*: If A is a manifold (possibly with boundary) and F is transversal to A then the inverse image of A is a manifold (possibly with boundary). When A is a unitary set the transversality condition reduces to the condition that the differential of F has maximal rank. For details about these results the reader should consult (Hirsch, 1976).

We define an *implicit surface* as a codimension 1 implicit manifold, and an *implicit solid* is a codimension 0 implicit manifold defined by $F^{-1}([a, b])$, where b is a real number, and a is either a real number or $-\infty$. The *boundary of the solid*, in the former case is the set $F^{-1}(b) \cup F^{-1}(a)$, and in the latter case, the set $F^{-1}(b)$. From the transversality condition it follows that the boundary is an implicit surface. In what follows, for simplicity, we will deal only with implicit surfaces and solids. We will refer generically to them as *implicit objects*.

If $\mathcal{O}$ is an implicit object the vector field

$$\mathrm{grad}F(p) = (\frac{\partial F}{\partial x_1}(p), \cdots, \frac{\partial F}{\partial x_n}(p))$$

is called the *gradient vector field* of $\mathcal{O}$. This vector field is defined on the domain of the function F, moreover it is orthogonal to the implicit surface and is orthogonal to the boundary of an implicit solid.

If $\mathcal{O} = F^{-1}(0)$ is an implicit surface, then for a small real number $\varepsilon > 0$ the equation $F^{-1}([-\varepsilon, \varepsilon])$ defines an implicit solid containing $\mathcal{O}$. It is called the *closed ε-tubular neighborhood* of $\mathcal{O}$. Geometrically it is the region delimited by the two "parallel" implicit surfaces $F^{-1}(-\varepsilon)$ e $F^{-1}(\varepsilon)$ (see Figure 1(a)).

If $\mathcal{O} = F^{-1}((-\infty, b])$ is an implicit solid, then for a small real number $\varepsilon > 0$ the equation $F^{-1}([b, b+\varepsilon])$ defines an implicit solid. It is called the *closed external ε-collar* of $\mathcal{O}$. Geometrically it is the region delimited by the boundary of the solid $\mathcal{O}$ and by a "parallel" implicit surface $F^{-1}(b+\varepsilon)$ containing the solid (see Figure 1(b)).

We will refer to an ε-tubular neighborhood or to an ε-collar, generically as an ε-neighbourhood of the implicit manifold.

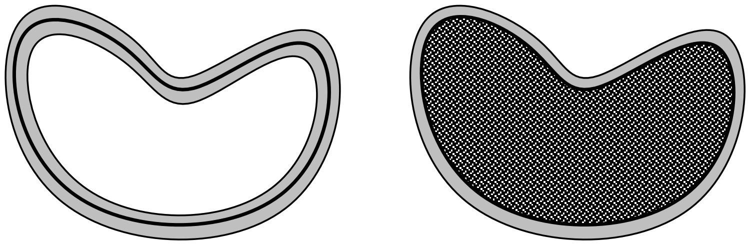


Figure 1: (a) Closed tubular neighborhood (b) Closed external collar.

3 TRIANGULATING THE SPACE

To capture the geometry of an implicit manifold the function F has to be sampled. There are several alternatives to do this, depending on the specific task one wants to perform. For example, in a ray tracing program the intersection of a ray with the surface is computed by searching for the common roots of the ray equation and the implicit function. Another effective way to sample is to triangulate the space and substitute the implicit function F by its affine approximation in the simplicial complex that defines the triangulation. The computational effort involved is small and the results are very effective. The seminal work using PL methods to approximate implicit manifolds in $\mathbf{R}^n$ is (Allgower and Schmidt, 1985).

The *Freudenthal triangulation* is the simplest one to construct in $\mathbf{R}^n$: we subdivide the space $\mathbf{R}^n$ in a uniform cubic grid and the triangulation is obtained by subdividing each n-cube in $n!$ simplices. A two dimensional example of this procedure is illustrated in Figure 2. For the details about the construction of the Freudenthal triangulation and the associated cell decomposition of implicit manifolds, the reader should consult (Allgower, 1990).

The *meshsize* of a triangulation $\mathcal{T}$ is defined by $\delta = \sup_{\sigma \in \mathcal{T}} \text{diam}\, \sigma$, where σ represents a simplex of the triangulation. The meshsize depends on the space norm, and its value for different norms differ by a constant. Using the euclidean norm the meshsize of the Freudenthal triangulation in $\mathbf{R}^n$ is $\sqrt{n}$.

A triangulation is called *regular* if each simplex can be approximated by a regular simplex. In 3-space a regular triangulation has no "bad tetrahedra" in the sense described in (Dey et al, 1991).

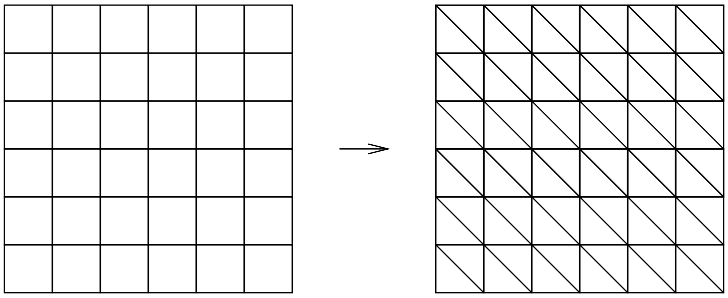


Figure 2: Freudenthal triangulation of the plane.

For a given positive real number $\varepsilon > 0$, we say that a triangulation $\mathcal{T}$ is ε-*subordinated* to an implicit surface $M = F^{-1}(0)$, if the three following conditions are satisfied (see Figure 3(a)):

(1) M is transversal to the triangulation;

(2) each n-simplex of $\mathcal{T}$ is regular.

(3) for each n-simplex σ of $\mathcal{T}$ that intersects M, there exists a k-face, $k < 0$, f_σ of σ, and a point $p \in M \cap \sigma$, at a distance $< \varepsilon$ from the barycenter of σ, such that the tangent plane, T_pM, of M at p is ε-closed to the support plane of f_σ.

The above definition can be extended for implicit solids substituting the third condition by the following one. Figure 3 illustrates condition (3) for the two dimensional case.

(3′) for each simplex σ of $\mathcal{T}$ that intersects M, either $\sigma \subset M$, or there exists a $n-1$-face f_σ of σ, and a point $p \in M \cap \sigma$, at a distance $< \varepsilon$ from the barycenter of σ, such that the tangent plane, T_pM, of M at p is ε-closed to the support plane of the face f_σ.

4 CONSTRUCTION OF SUBORDINATE TRIANGULATIONS

In this section we will describe a method to construct a triangulation subordinated to a given implicit manifold M. The construction process involves a physically-based approach, initially we define a system of spring-mass elements associated with

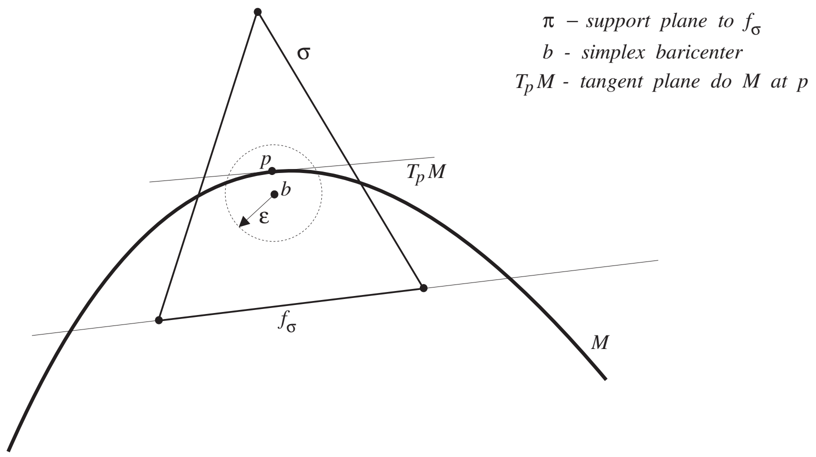


Figure 3: ε-subordinated triangulation.

a Freudenthal triangulation of the space. This system is submitted to forces derived from the gradient field of the implicit manifold and its equilibrium position gives the regular triangulation subordinated to the manifold.

The spring-mass lattice generation process consists of the following steps:

- A Freudenthal triangulation is created for a bounding box of the implicit manifold;
- Each simplex of the triangulation that intersects the implicit manifold is identified;
- To each vertex of the simplex we associate a mass node, and to each 1-dimensional face we associate a spring of the system.

The first step is explained in section 3. The identification of the relevant simplices in the second step is made by testing the sign of the implicit function at the vertices of each simplex. Assuming that the uniform sampling grid is sufficiently fine, then, if the signs are the same for all vertices, the simplex must be totally inside or outside of the object. If the signs are different, the simplex must intersect the boundary of the object.

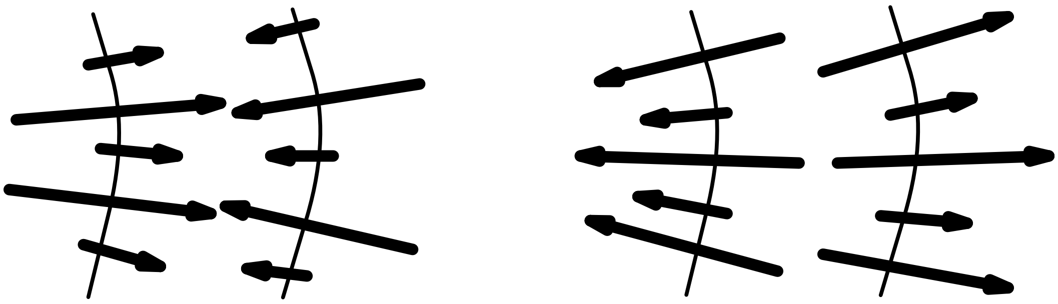
During this process two lists are created: a node list and a spring list. The essential connectivity information is based on the location of grid points. In this way, nodes are labeled according to their spatial locations and springs according to the nodes they connect.

5 DYNAMIC SIMULATION

The dynamic simulation submits 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 the external deformation forces mentioned above. This discrete dynamics system is solved using an explicit Euler time integration procedure.

The strategy adopted here is to enforce the spring-mass mesh to stay within a closed ε-neighborhood of the implicit manifold. Intuitively, the goal is to produce a "thick shell" made of springs and point masses with thickness 2ε around the implicit surface or around the boundary of the implicit solid. In the case of an implicit solid this shell extends to cover the all of the interior. We take ε proportional to the meshsize of the initial triangulation.

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 the ε-neighborhood, generates repelling forces that prevents points from being too close to the surface. The other field, defined outside the ε-neighborhood, generates attraction forces that pulls points towards the surface. This is depicted in Figure 4 for the two-dimensional case.



Attraction field Repulsion field

Figure 4: Force fields.

In order to help the relaxation of the mesh structure into the desirable configuration, the initial rest length of the strings is made equal to the diameter of the ε-neighborhood (e.g. smaller than the grid spacing). This is equivalent to start the simulation with a tensioned mesh that snaps into place moved by internal and external forces.

With the above system the result stated below is true:

Theorem: *The spring-mass system described above gets to an equilibrium position and at this final position it defines a triangulation $\mathcal{T}$ of the space with the following properties:*

(1) *$\mathcal{T}$ is regular;*

(2) *$\mathcal{T}$ is subordinated to the implicit manifold M (in particular it is transversal to M).*

6 RESULTS

This section presents the results using the adaptive mesh algorithm. For clarity, examples are restricted to two dimensions. The implicit object used in the 2D examples is the circle defined by the equation $x^2 + y^2 - r^2 = 0$, $r > 0$.

Figure 5 reveals the evolution of the dynamic simulation. The initial mesh is shown in Figure 5(a) Intermediate configurations can be seen in Figures 5(b) and 5(c). The final solution is shown in 6(d) Note how the geometry of the adaptive mesh converges to the shape of the object. Note also that the final configuration is very regular.

Figure 6 demonstrates the result of different grid sizes. It shows the final meshes respectively for 2×2, 6×6, 12×12 and 60×60 grids. Figure 7 is a detail of the mesh corresponding to a grid of size 80×80.

Figure 8 shows the spring-mass lattice of a disk (a solid object). In the dynamic simulation, the force fields affect only the point masses belonging to the boundary cells. The internal nodes are made passive, and the stiffness of the internal springs is set to 1/10 of that of the boundary springs.

7 APPLICATIONS

This section discusses some applications of the method in computer graphics. It can be used very effectively to solve several problems in geometric modeling and animation.

The spatial subdivision enumeration associated with an implicit object produces two complementary representations: a volumetric decomposition of the domain of the implicit function and a combinatorial manifold approximating the boundary of the object. Both are given as piecewise linear approximations and one is the dual of the other. The former is an affine approximation of F induced by the spatial subdivision enumeration. The latter is a decomposition of the boundary of the object derived from the spatial subdivision enumeration.

Figures 9 and 10 illustrate the application of the method to construct spatial subdivision enumerations in three dimensions. They show the initial and the final structures corresponding to two different implicit objects (respectively a sphere and a torus). Note that the meshsize is increasingly finer.

The polygonization of an implicit surface M is computed from a subordinated triangulation of the domain of M. The surface 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. The set of all these triangles constitute the combinatorial manifold that approximates M.

Figures 11, 12 and 13 illustrate the polygonization of a cylinder defined by the implicit equation $x^2 + y^2 = 1$ Figure 11 shows a sequence corresponding to different phases of the mesh deformation process for a cylinder. Figure 11.a depicts the initial mesh created from a Freudenthal triangulation of the ambient space, Figure 11.b reveals the final mesh in its equilibrium position. The polygonal approximation is derived from the final mesh. It is apparent that the initial mesh was constrained to lie in a tubular neighborhood of the implicit surface, conforming to the cylinder's shape. Figure 12 shows the final polygonal approximations for the cylinder. Figure 13 shows a detail of the polygonization associated with the spatial subdivision enumeration before and after the deformation process (Figures 13.a and 13.b respectively). Note how the deformation of the mesh produces a very homogeneous polygon structure, transforming long and thin elements in almost 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 14 illustrates the use of the spatial subdivision enumeration in physically-based modeling and animation environment. A spring-mass mesh describing the physical properties of the object is automatically generated from this structure. The mesh is incorporated in the physically-based environment as a means of interacting with the implicit object. The visualization of the state of the simulation can be done either using a polygonal approximation of the surface as described previously, or by sampling directly the deformed implicit object. Figure 14 shows a frame of the dynamics simulation of a cylinder falling under a gravitational field. Figure 14.a depicts the spring-mass mesh used in the simulation. Figure 14.b shows the corresponding polygonization of the cylinder.

8 CONCLUSIONS AND CURRENT RESEARCH

In conclusion, we have successfully developed a method that generates optimal triangulation of the space associated with implicit manifolds. The resulting structure is regular and conforms well to the shape of the objects. The mesh resolution can be easily parameterized. The mesh construction process is totally automatic.

The method has multiple advantages. Besides the regular triangulation itself, it constructs in a natural way a spring-mass system associated with the implicit manifold. As we demonstrated above, this can be exploited in different directions: dynamical simulation with implicit objects (Velho and Gomes, 1991) and cell decomposition of implicit manifolds (Figueiredo et al,1991). We are also working on a extension of this method to generate adaptive spatial enumerations subject to determined characteristic of the implicit manifold, such as curvature or some physical property.

9 ACKNOWLEDGEMENTS

This research was partially supported by grants from CNPq, NSERC and ITRC.

10 BIBLIOGRAPHY

1. Allgower, E. L. and Georg K., (1990): "Numerical Continuation Methods, An Introduction". Springer-Verlag, New York.

2. Allgower, E. L. and Schmidt, Ph. H. (1985): *An Algorithm for Piecewise Linear Approximation of an Implicitly Defined Manifold.* SIAM Journal of Numerical Analysis, **24(2)**, 2452–469.

3. Dey, T., Bajaj, C. and Sugihara, K., (1991): *On Good Triangulations in Three Dimensions.* in Proceedings of Symposium on SOlid Modeling Foundations and CAD/CAM Applications. Editors: Jaroslav Rossignac and Joshua Turner. ACM Press.

4. Blinn, J. (1982): *A Generalization of Algebraic Surface Drawing.* ACM Transactions on Graphics, **1(3)**, 235–256.

5. Bloomenthal, J., (1985): *Modeling the Mighty Maple.* Computer Graphics, **19(3)**, 305-311.

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

7. Haumann, D. (1987): *Modeling the Physical Behavior of Flexible Objects.* Siggraph Tutorial Notes in Animation, 1-10.

8. Hirsch, M. W., (1976): "Differential Topology". Springer-Verlag, New York.

9. Rockwood, A. P., (1987): *Blending surfaces in solid geometrical modeling.* Phd dissertation, University of Cambridge.

10. Terzopoulos, D. and Vasilescu, M. (1991): *Sampling and Reconstruction with Adaptive Meshes.* Proc. Computer Vision & Pattern Recognition Conference (CUPR – 91). Lahaina, HI, 70–75.

11. Terzopoulos, D.; Platt J. and K. Fleischer (1989): *Heating and Melting Deformable Objects.* Proceedings of Graphics Interface, 219–226.

12. Terzopoulos, D. and Waters, K. (1990): *Physically-based Facial Modeling, Analysis and Animation.* The Journal of Visualization and Computer Animation, **1**, 73–80.

13. Velho, L. and Gomes, J. de M. (1991): *A Dynamics Simulation Environment for Implicit Objects using Discrete Models.* Second Eurographics Workshop on Animation and Simulation.

14. Figueiredo, L. H., Gomes, J. de M., Velho, L. and Terzopoulos, D.(1992): *Physically-based Polygonization of Implicit Objects.* Graphics Interface '92

15. Wyvill G.; McPheeters C. and Wyvill B. (1986): *Data Structures for Soft Objects.* Visual Computer, **2(4)**, 227–234.

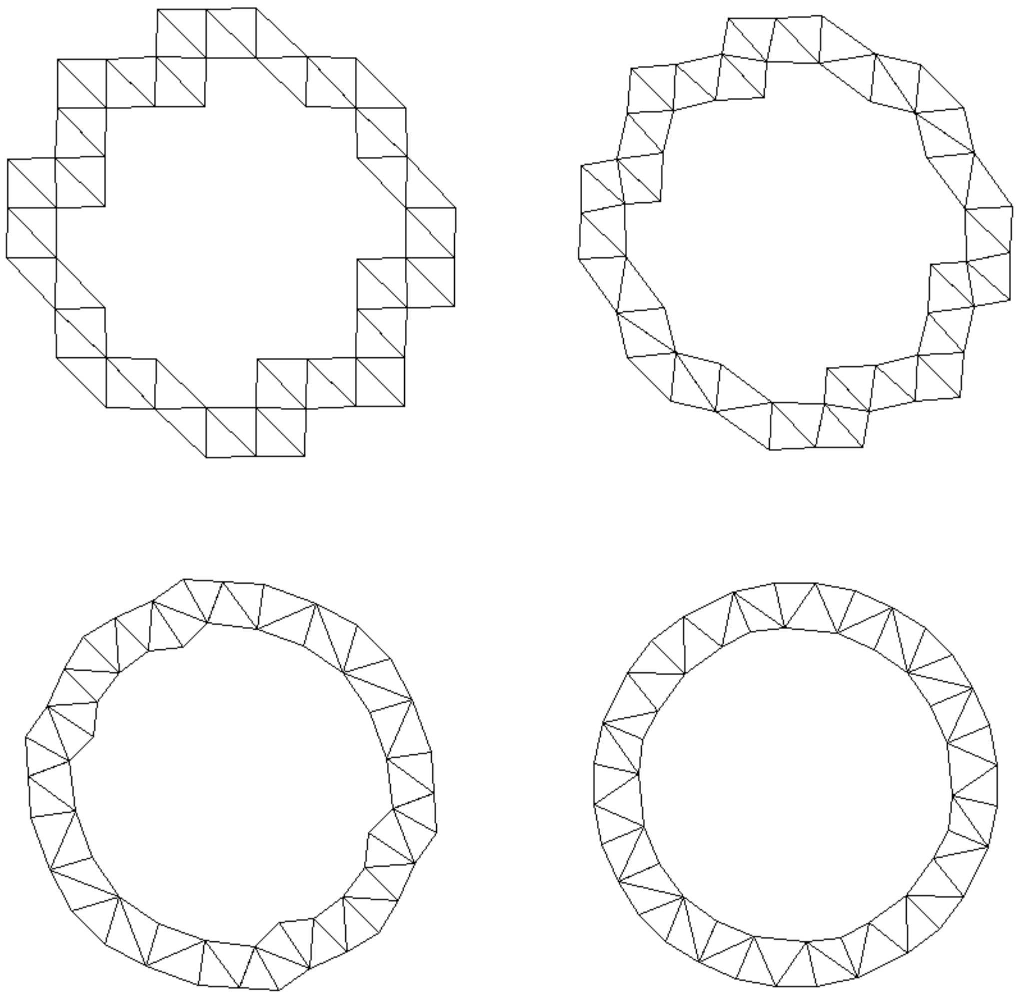


Figure 5: Evolution of the adaptive mesh

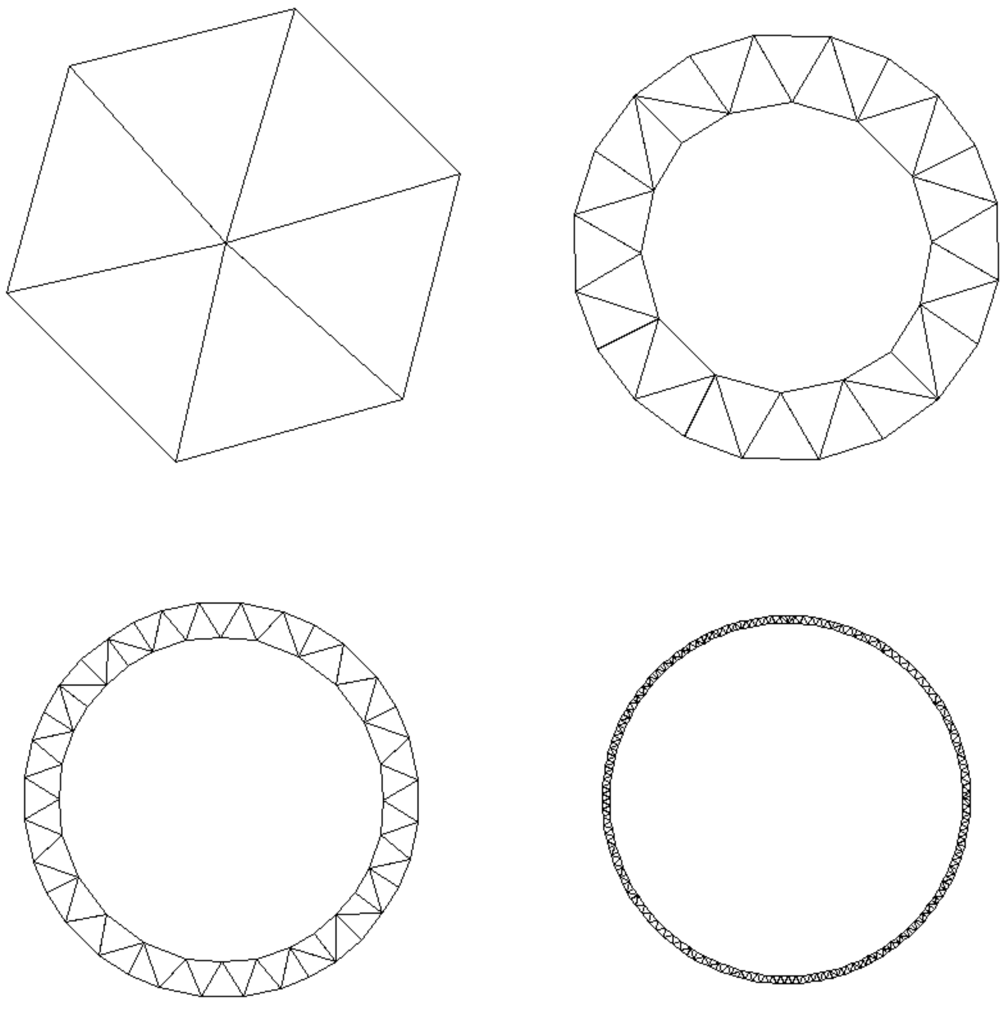


Figure 6: Grid sizes of: 2, 6, 12, 60

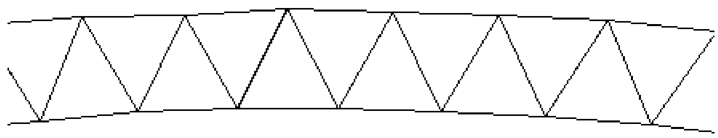


Figure 7: Detail of the shell corresponding to a grid of size 80

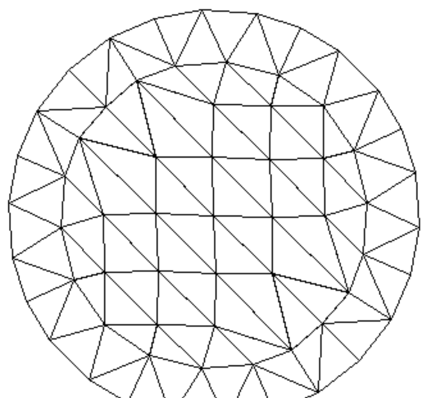


Figure 8: Lattice for a solid object

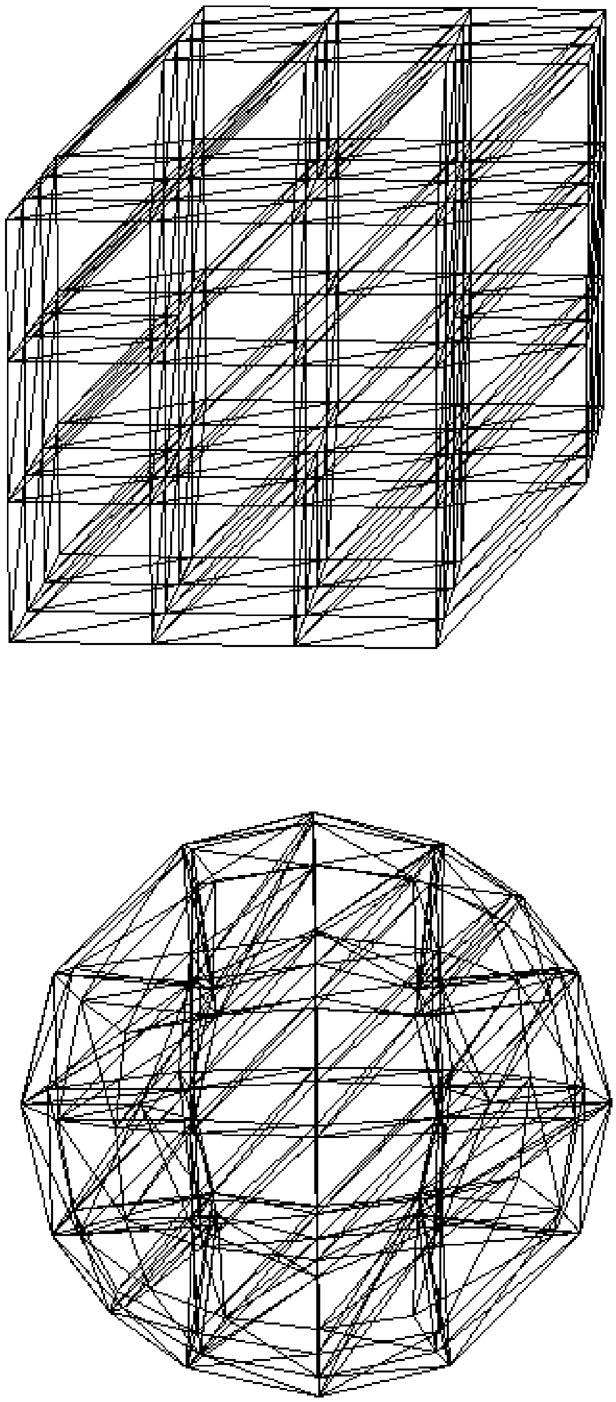


Figure 9: Sphere: Initial and Final Mesh ($4 \times 4 \times 4$)

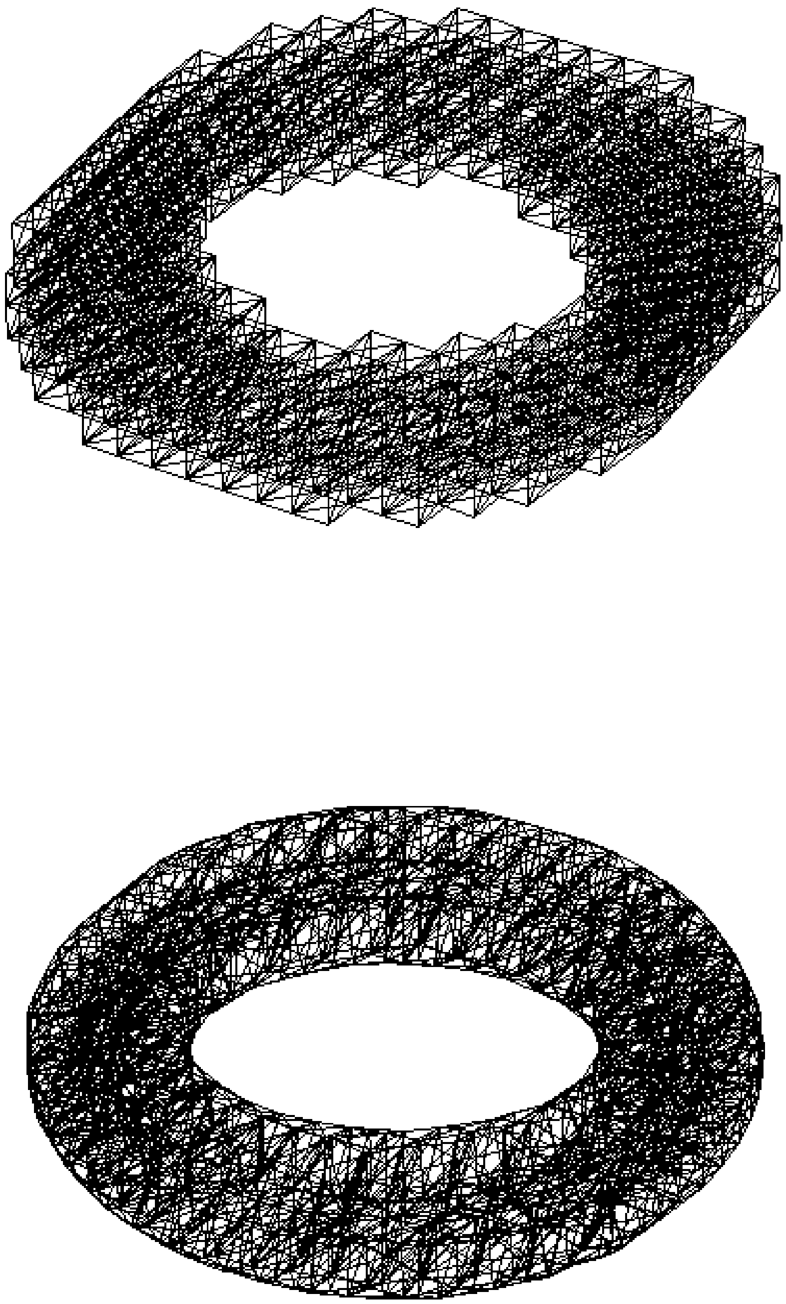


Figure 10: Torus: Initial and Final Mesh ($16 \times 16 \times 4$)

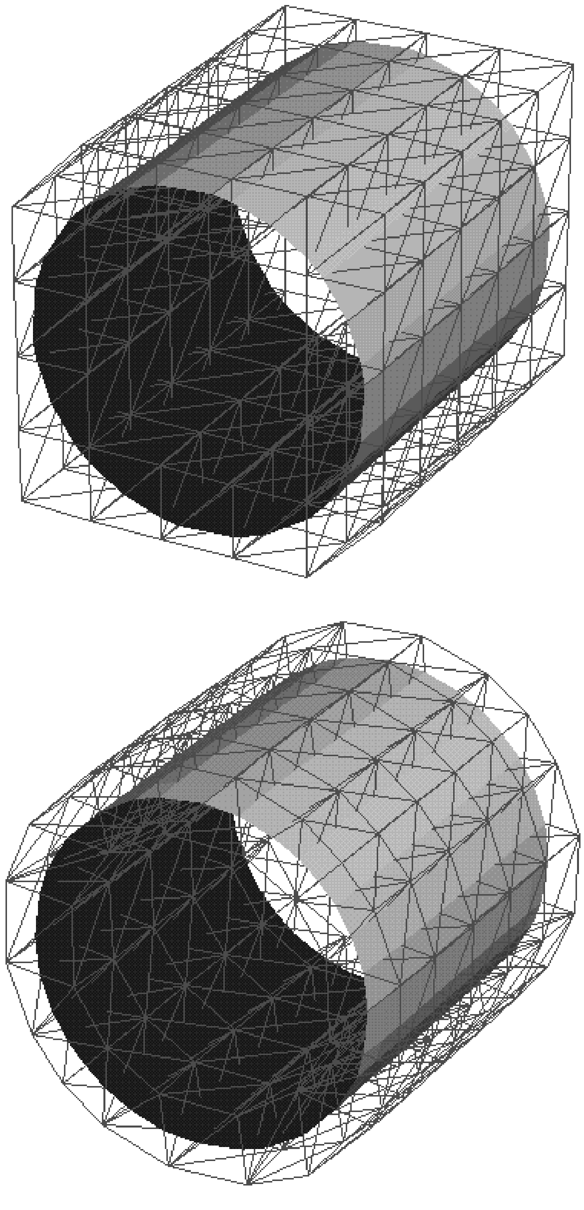


Figure 11: Grid deformation

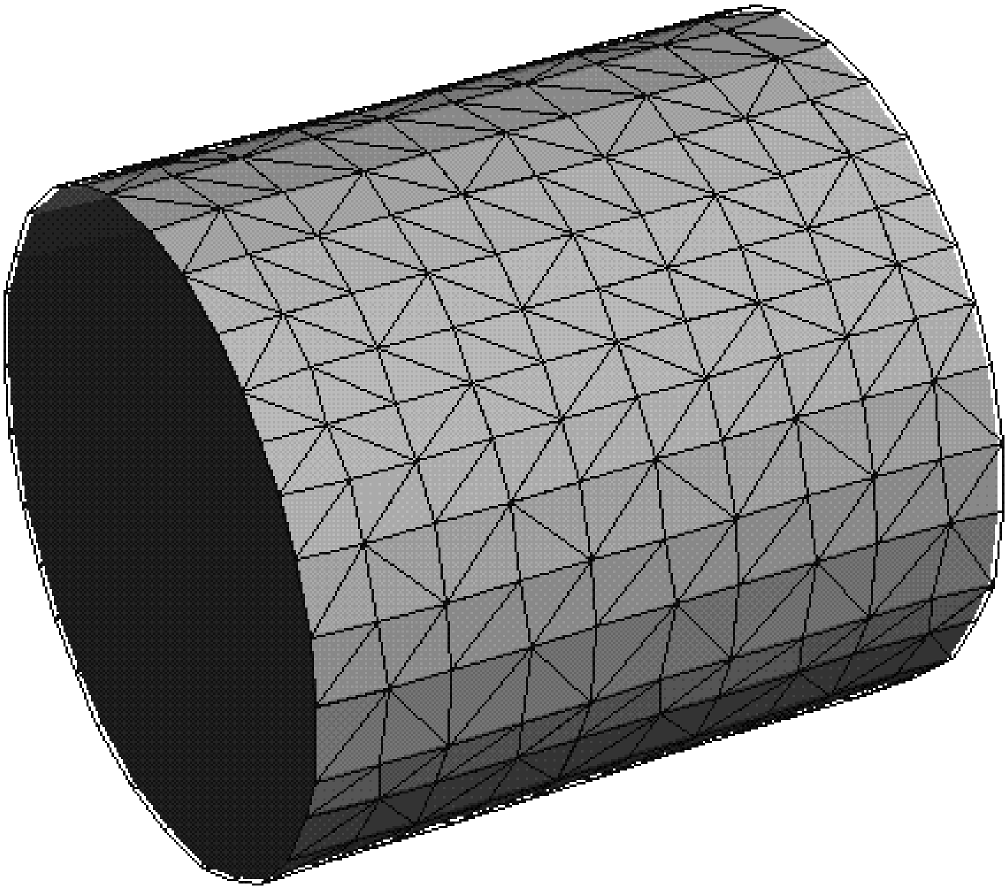


Figure 12: Cylinder

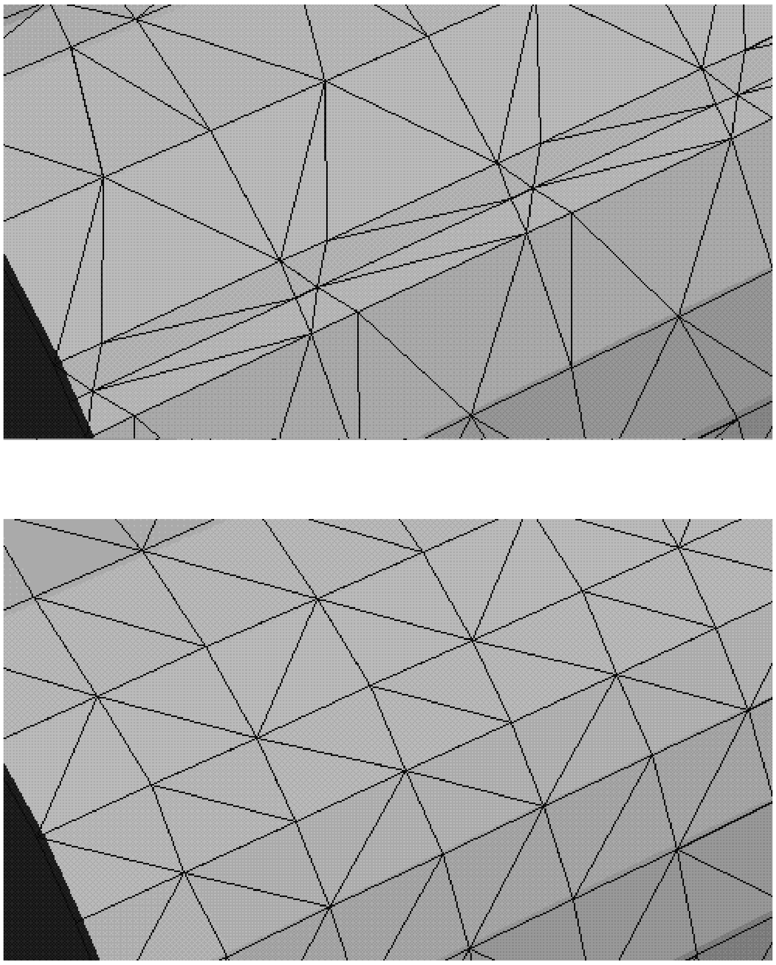


Figure 13: Detail of the polygonization

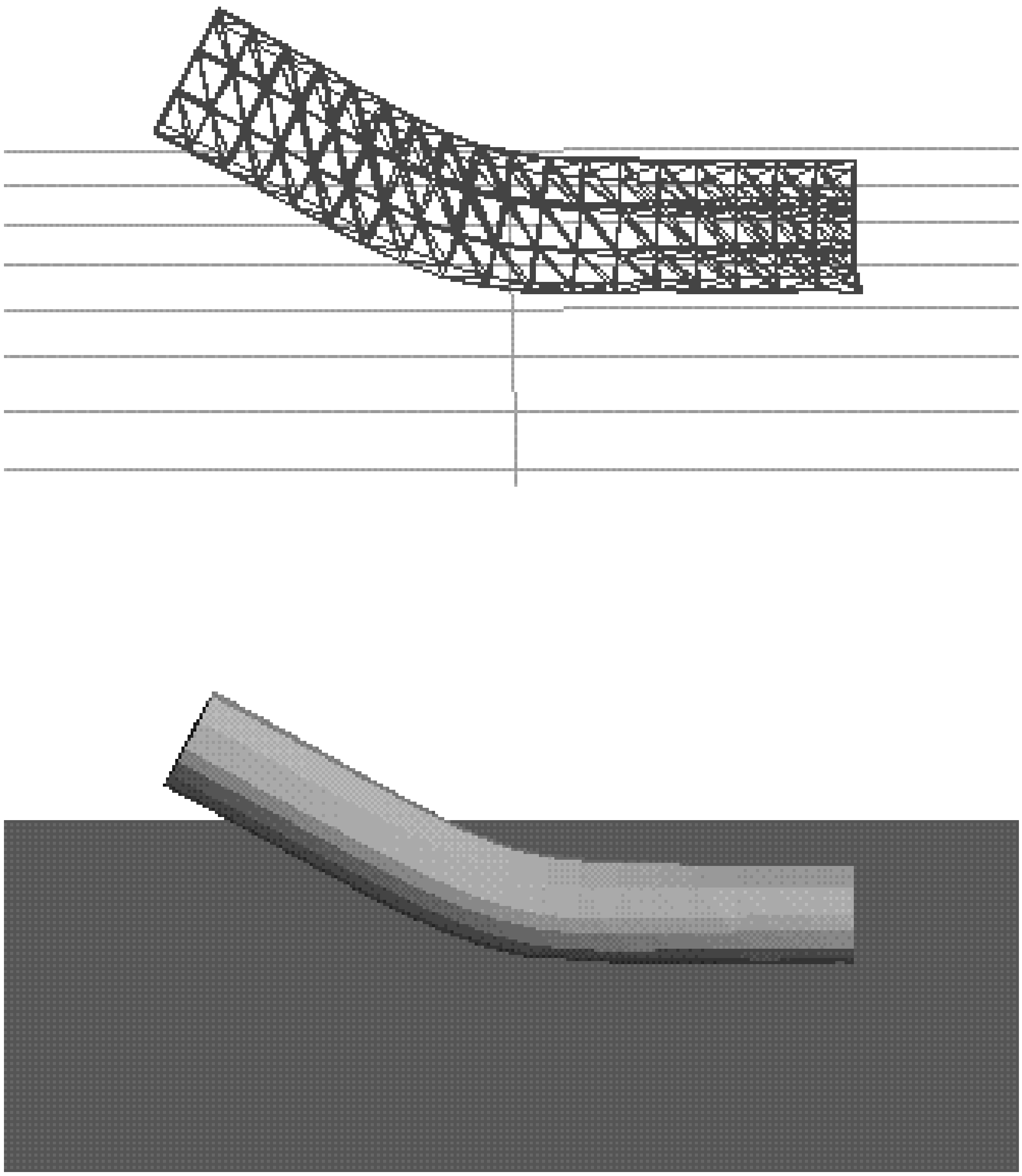


Figure 14: Physically-based animation of the cylinder