

Controlling Anisotropy in Mass-Spring Systems

David Bourguignon and Marie-Paule Cani

iMAGIS-GRAVIR/IMAG-INRIA

iMAGIS is a joint research project of CNRS/INRIA/UJF/INPG

E-mail:{David.Bourguignon|Marie-Paule.Cani}@imag.fr

<http://www-imagis.imag.fr/>

Abstract. This paper presents a deformable model that offers control of the isotropy or anisotropy of elastic material, independently of the way the object is tiled into volume elements. The new model is as easy to implement and almost as efficient as mass-spring systems, from which it is derived. In addition to controlled anisotropy, it contrasts with those systems in its ability to model constant volume deformations. We illustrate the new model by animating objects tiled with tetrahedral and hexahedral meshes.

1 Introduction

Mass-spring systems have been extensively used in Computer Graphics over the last fifteen years, and are still very popular. Easier to implement and faster than finite element methods, these systems allow animation of dynamic behaviors. They have been applied to the animation of inanimate bodies such as cloth or soft material [17, 10, 15, 5] and to the animation of organic active bodies such as muscles in character animation [11, 1, 12].

Both isotropic and anisotropic elastic materials can be found among the objects to animate. For instance, a rubber toy is isotropic, while most natural objects (animal organs, plants) are strongly anisotropic, due to their fiber structure and/or the composite materials they are made of. One of the main drawbacks of mass-spring systems is that neither isotropic nor anisotropic materials can be generated and controlled easily. Another problem is that most of the materials found in nature maintain a constant or quasi-constant volume during deformations (this is well known for muscles, but also holds for inanimate materials). Mass-spring models do not have this property.

1.1 Background

Animating an elastic object using a mass-spring system usually consists of discretizing the object with a given 3D mesh, setting point masses on the mesh nodes and damped springs on the mesh edges. Then, most implementations simply integrate point dynamics equations for each mass from the set of applied forces due to the mesh deformation at the previous time step [10].

Well-known advantages of mass-spring systems include their ability to generate dynamic behaviors, while finite elements methods are generally used in the static case, and their ability to handle both large displacements and large deformations.

Among the intrinsic limitations of mass-spring systems, one of the main problems is parameter setting. Computing the masses in order to set up a homogeneous material can be done by computing each mass according to the volume of the Voronoi region around

it [4]. However, there is no easy solution for spring parameters. Since damped springs are positioned along the edges of a given volume mesh, the geometrical and topological structure of this mesh strongly influences the material behavior. A consequence of this problem is that changing the mesh density during the simulation while maintaining the same global mechanical properties is very difficult [7].

If all springs are set to the same stiffness, the mesh geometry may generate *undesired anisotropy*, as shown in Fig. 1.a. The undesired behavior disappears when hexahedral elements aligned with the forces directions are used (Fig. 1.b). Of course, if the tiling of the object volume was computed from the triangulation of random uniformly-distributed sample points, the unwanted anisotropy problem would tend to disappear when the density of the mesh increases. However, using an extremely dense mesh would reduce efficiency.

Approximating a desired behavior using a given mesh can be achieved, as in [4, 9], by using optimization to tune individual spring stiffnesses. This technique could be used, in theory, for generating both isotropic and anisotropic behaviors. However, due to the large computational cost, this method has only been tested in the 2D case [4].

The most common approach to control the behavior of a mass-spring system, at least along a few “directions of interest”, is to specifically design the mesh in order to align springs on these specific directions, such as in Fig. 1.b. This was done for instance in Miller’s “snakes and worms” models [11] and in the muscle model of Ng and Fiume [12], where some of the springs were aligned with the muscle fibers and the rest were set perpendicular to them. Unfortunately, manually creating such meshes would be time consuming in the general case, where fiber directions generating anisotropy vary in an arbitrary way inside the object. We are rather looking for an approach that uses a 3D mesh obtained, for example, with a commercial meshing package (such as GHS3D [16]) fed with a 3D surface mesh, and still displays the deformable model behavior, with specified properties in specific directions.

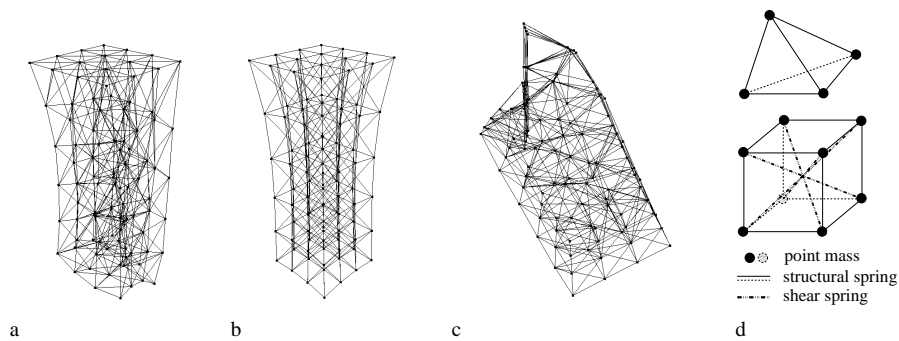


Fig. 1. Mass-spring systems drawbacks. At left, comparison between two meshes undergoing a downward pull at their bottom end while their top end is fixed. We observe undesired anisotropy in the tetrahedral mass-spring system (a), but not in the hexahedral mesh with springs aligned in the gravity and pull force directions (b). At right, equilibrium state of a cantilever beam, which left end is fixed, under force of gravity (c). All things being equal, the mass-spring system considered (tetrahedral mesh) is unable to sustain flexion, as opposed to our model (Fig. 5.c). The spring configurations used for tetrahedral and hexahedral meshes are given in (d).

1.2 Overview

This paper presents an alternative model to classical mass-spring systems that enables one to specify isotropic or anisotropic properties of an elastic material, independently from the 3D mesh used for sampling the object. The approach we use is still related to mass-spring systems, in the sense that we animate point masses subject to applied forces. However, the forces acting on each mass are derived from the anisotropic behavior specified for each of the volume elements that are adjacent to it.

Since there are no springs along the mesh edges, the geometry and topology of the mesh do not restrict the simulated behavior. Moreover, constant volume deformations can be obtained easily, by adding extra forces. We illustrate this on both tetrahedral and hexahedral meshes. Our results show that computation time remains low, while more controllable behaviors are achieved.

2 Modeling Anisotropy

Our aim is to specify the mechanical properties of the material independently from the mesh geometry and topology. In usual mass-spring systems, internal forces acting inside the material are approximated exclusively by forces acting along the edges of the mesh (i.e. along the springs). This is the reason for the undesired anisotropy problem described earlier, and for the difficulty in specifying desired anisotropic properties.

The basic idea of our method is to let the user define, everywhere in the object, mechanical characteristics of the material along a given number of axes corresponding to orientations of interest at each current location. All internal forces will be acting along these axes instead of acting along the mesh edges. For instance, in the case of organic materials such as muscles, one of the axes of interest should always correspond to the local fiber orientation.

Since the object is tiled using a mesh, axes of interest and the associated mechanical properties are specified at the barycenter of each volume element inside the mesh. We currently use three orthogonal axes of interest. The possible use of a larger number of axes will be discussed in Section 6.

2.1 General Scheme

During deformations of the material, the three axes of interest, of given initial orientation, evolve with the volume element to which they belong. In order to be able to know their position at each instant, we express the position of the intersection point of one axis with one of the element faces as a linear combination of the positions of the vertices defining the face. The corresponding interpolation coefficients are computed for each face in the rest position (see Figures 2 and 4).

Given the position of the point masses of a volume element, we are thus able to determine the coordinates of the six intersection points and consequently the three axes that constitutes the local frame, up to the precision of our linear interpolation.

From the deformation of the local frame, we can deduce the resulting forces on each intersection point. Then, for a given face, we can compute the force value on each point mass belonging to this face by “inverse” interpolation of the force value at the intersection point. The interpolation coefficients previously defined are therefore also considered as weighting coefficients of the force on each point mass.

2.2 Forces Calculations

Damped springs with associated stiffness and damping coefficients are used to model stretching characteristics along each axis of interest. In order to specify shearing properties, angular springs are added between each pair of axes. Rest lengths and rest angles are pre-computed from the initial position of the object that defines its rest shape. The equations we use for these springs are detailed below.

Axial damped spring. The spring forces \mathbf{f}_1 and \mathbf{f}_2 between a pair of intersection points 1 and 2 at positions \mathbf{x}_1 and \mathbf{x}_2 with velocities \mathbf{v}_1 and \mathbf{v}_2 are

$$\mathbf{f}_1 = - \left[k_s (\|\mathbf{l}_{21}\| - r) + k_d \frac{\dot{\mathbf{l}}_{21} \cdot \mathbf{l}_{21}}{\|\mathbf{l}_{21}\|} \right] \frac{\mathbf{l}_{21}}{\|\mathbf{l}_{21}\|}, \quad \mathbf{f}_2 = -\mathbf{f}_1,$$

where $\mathbf{l}_{21} = \mathbf{x}_1 - \mathbf{x}_2$, r is the rest length, $\dot{\mathbf{l}}_{21} = \mathbf{v}_1 - \mathbf{v}_2$ is the time derivative of \mathbf{l}_{21} , k_s and k_d are respectively the stiffness and damping constants.

Angular spring. The spring forces $(\mathbf{f}_1, \mathbf{f}_2)$ and $(\mathbf{f}_3, \mathbf{f}_4)$ between two pairs of intersection points (1, 2) and (3, 4) are

$$\begin{aligned} \mathbf{f}_1 &= -k_s \left[\frac{\mathbf{l}_{21} \cdot \mathbf{l}_{43}}{\|\mathbf{l}_{21}\| \|\mathbf{l}_{43}\|} - c \right] \frac{\mathbf{l}_{43}}{\|\mathbf{l}_{43}\|}, & \mathbf{f}_2 &= -\mathbf{f}_1, \\ \mathbf{f}_3 &= -k_s \left[\frac{\mathbf{l}_{21} \cdot \mathbf{l}_{43}}{\|\mathbf{l}_{21}\| \|\mathbf{l}_{43}\|} - c \right] \frac{\mathbf{l}_{21}}{\|\mathbf{l}_{21}\|}, & \mathbf{f}_4 &= -\mathbf{f}_3, \end{aligned}$$

where $\mathbf{l}_{21} = \mathbf{x}_1 - \mathbf{x}_2$ and $\mathbf{l}_{43} = \mathbf{x}_3 - \mathbf{x}_4$, c is the cosine of the rest angle between \mathbf{l}_{21} and \mathbf{l}_{43} , k_s is the stiffness constant.

Here, two approximations are made: first, we assume a small variation of the angle and take the variation of the angle's cosine instead; second, we consider it sufficient to use as unit vector the other vector of the pair, instead of a vector normal to the one considered, in the plane where the angle is measured. These two approximations gave good results in practice. Furthermore, we found no necessity to use damped angular springs.

3 Application to Tetrahedral Meshes

Many objects in Computer Graphics are modeled using triangular surface meshes. Generating a 3D mesh from such a description, using tools like GHS3D [16] yields to tetrahedral volume meshes. This section details our method in this case.

Fig. 2 depicts a tetrahedral element, with the associated frame defining the three axes of interest. We express the position \mathbf{x}_P of point P as a function of the positions of vertices A , B and C of the given face, using barycentric coordinates:

$$\mathbf{x}_P = \alpha \mathbf{x}_A + \beta \mathbf{x}_B + \gamma \mathbf{x}_C$$

(e.g. if $\alpha = 1$ and $\beta = \gamma = 0$, we get $\mathbf{x}_P = \mathbf{x}_A$). Therefore, a force \mathbf{f}_P applied to point P is split into forces $\alpha \mathbf{f}_P$, $\beta \mathbf{f}_P$ and $\gamma \mathbf{f}_P$, respectively applied on points A , B and C .

We can note that since the elementary volume has four faces, and since there are three axes of interest defining six intersection points, two such points may lie on the same face of the volume. This has not been problematic in practice, since forces applied on mesh nodes are correctly weighted.

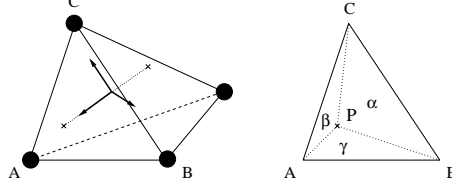


Fig. 2. Tetrahedral element. A point mass is located at each vertex. A local frame is defined at the barycenter of the element (left). Each axis is characterized by the barycentric coordinates α , β and γ (with $\alpha + \beta + \gamma = 1$) of its two intersection points (right, for a given face). These coordinates are easily obtained using an area ratio.

3.1 Volume Preservation

Animating constant volume deformations with a classical mass-spring system is not straightforward. For these systems, forces are only applied along the edges of each volume element, while maintaining a constant volume basically requires adding radial forces or displacements, as shown by Promayon *et al.* [14].

To simply ensure volume preservation, we propose a volume force formulation adapted to tetrahedral volume element. It is loosely related to soft volume-preservation constraint of Lee *et al.* [8].

Let us define \mathbf{x}_B the position of the barycenter of the tetrahedral element, with

$$\mathbf{x}_B = \frac{1}{4} \sum_{i=0}^3 \mathbf{x}_i$$

where \mathbf{x}_i is the position of the i th vertex. Then, we define the force applied on the j th vertex as

$$\mathbf{f}_j = -k_s \left[\sum_{i=0}^3 \|\mathbf{x}_i - \mathbf{x}_B\| - \sum_{i=0}^3 \|\mathbf{x}_i - \mathbf{x}_B\|_{t=0} \right] \frac{\mathbf{x}_j - \mathbf{x}_B}{\|\mathbf{x}_j - \mathbf{x}_B\|}$$

where k_s is the constraint stiffness and $\sum_{i=0}^3 \|\mathbf{x}_i - \mathbf{x}_B\|_{t=0}$ is the rest length of this “volume spring”. It was not necessary to add damping forces with this constraint.

This method gave satisfactory results in practice, since we get less than 1.5% volume variation in our experiment (see Fig. 3), but results depend on the material parameters chosen and the type of experiment conducted. In applications where these volume variations are considered too high, volume preservation could be enforced directly as a hard constraint like in Witkin’s work [20, 19].

4 Application to Hexahedral Meshes

The use of hexahedral meshes is not as common as tetrahedral ones, since the geometry they can define is more limited. However, these meshes may be useful for animating objects modeled using voxels [2]. This kind of data, with information about material characteristics specified in each voxel (possibly including anisotropy), may be provided by medical imaging applications.

Applying the general method presented in Section 2.1 to hexahedral meshes is straightforward. Fig. 4 depicts an hexahedral element, with the associated frame defining the three axes of interest. We express the position \mathbf{x}_P of point P as a function of

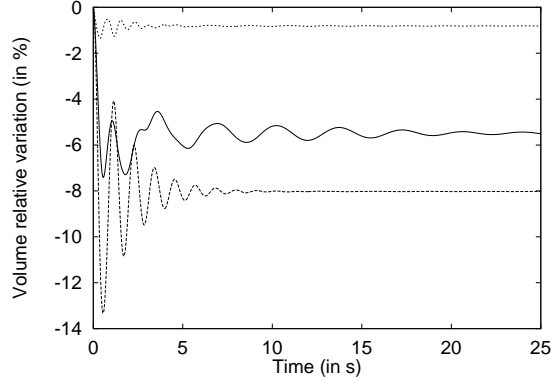


Fig. 3. Volume preservation experiments using the same tetrahedral mesh lying on a table under force of gravity. In our model, one axis of interest is set to the vertical direction (the direction of application of gravity) and the two others in horizontal directions. Parameters are chosen identical along the 3 axes. The same stiffness and damping values are used in all experiments. Bottom graph: our model without volume preservation forces. Middle graph: equivalent mass-spring system. Top graph: our model with volume preservation forces.

the positions of vertices A , B , C and D of the given face, using bilinear interpolation coordinates:

$$\mathbf{x}_P = \zeta \eta \mathbf{x}_A + (1 - \zeta) \eta \mathbf{x}_B + (1 - \zeta)(1 - \eta) \mathbf{x}_C + \zeta(1 - \eta) \mathbf{x}_D$$

(e.g. if $\zeta = 1$ and $\eta = 1$, we get $\mathbf{x}_P = \mathbf{x}_A$). Therefore, a force \mathbf{f}_P applied to point P is split into forces $\zeta \eta \mathbf{f}_P$, $(1 - \zeta) \eta \mathbf{f}_P$, $(1 - \zeta)(1 - \eta) \mathbf{f}_P$ and $\zeta(1 - \eta) \mathbf{f}_P$, respectively applied on points A , B , C and D .

Here, there is only one intersection point per face of the volume element. Since the element has eight vertices, the system is under-constrained instead of being over-constrained, as in the tetrahedral case. As a consequence, each elementary volume may have several equilibrium states, corresponding to the same rest position of the three axes of interest but to different positions of the vertices, if volume preservation forces are not applied.

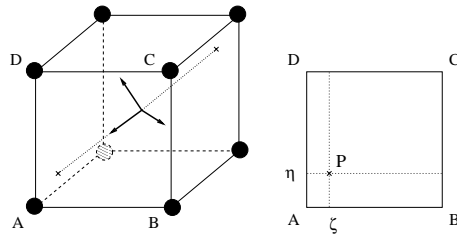


Fig. 4. Hexahedral element. A point mass is located at each vertex. A local frame is defined at the barycenter of the element (left). Each axis is characterized by the bilinear interpolation coordinates ζ and η (with $0 \leq \zeta \leq 1$ and $0 \leq \eta \leq 1$) of its two intersection points (right, for a given face).

4.1 Volume Preservation

Given the characteristics of hexahedron geometry, we use a slightly different expression for volume preservation forces, while keeping the idea of employing a set of forces that act in radial directions with respect to the volume element. This formulation is also loosely related to soft volume-preservation constraint of Lee *et al.* [8].

Let us define \mathbf{x}_B as the position of the barycenter of the hexaedral element, with

$$\mathbf{x}_B = \frac{1}{8} \sum_{i=0}^7 \mathbf{x}_i$$

where \mathbf{x}_i is the position of the i th vertex.

Then, we define the force applied on the j th vertex as

$$\mathbf{f}_j = - \left[k_s (\|\mathbf{l}\| - \|\mathbf{l}\|_{t=0}) + k_d \frac{\dot{\mathbf{l}} \cdot \mathbf{l}}{\|\mathbf{l}\|} \right] \frac{\mathbf{l}}{\|\mathbf{l}\|}, \quad \mathbf{l} = \mathbf{x}_j - \mathbf{x}_B, \quad \dot{\mathbf{l}} = \mathbf{v}_j - \mathbf{v}_B,$$

where \mathbf{v}_j and \mathbf{v}_B are respectively velocities of the j th vertex and barycenter, $\dot{\mathbf{l}}$ is the time derivative of \mathbf{l} , k_s and k_d are respectively the stiffness and damping constants. This is the classical formulation for a damped spring tying the j th vertex to the barycenter (see Section 2.2).

5 Results

All the experiments presented in this section have been computed by setting point masses to the same value. Thus, objects sampled using tetrahedral meshes are generally heavier than those sampled using hexahedral meshes. Moreover, objects are slightly inhomogeneous in the former case, since mesh nodes are not evenly distributed. Better results would be obtained by computing the mass values according to the density of the simulated material and to the volume of the Voronoi region associated with each point mass, as was done by Deussen *et al.* [4]. However, we found the results quite demonstrative as they are.

Numerical simulation of all experiments was achieved using Stoermer's explicit integration method [13] with no adaptive time step, and therefore might be improved.

Each figure depicts outer mesh edges and one of the three axes of interest inside each elementary volume. In Fig. 6 this axis represents the orientation along which the material is the stiffest.

5.1 Comparison with Mass-Spring Systems

The same experiments as in Fig. 1 are performed using our model instead of a classic mass-spring system (see Fig. 5). Here, one axis of interest is set to the vertical direction (the direction of application of gravity and pull forces) and the two others in horizontal directions. The same stiffness and damping values are used in each direction.

5.2 Controlling Anisotropy

A set of experiments with different anisotropic behaviors is presented in Fig. 6. It is interesting to notice that isotropic material can be modelled using a random orientation for the stiffest axis in each volume element.

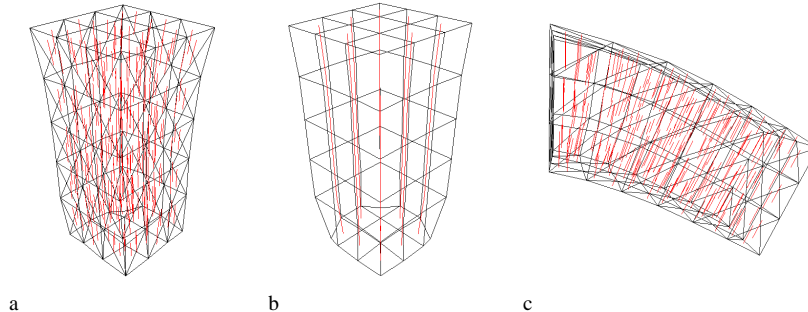


Fig. 5. Experiments similar to those of Fig. 1, but computed with our model. As expected, we do not observe undesired anisotropy in both the tetrahedral (a), and the hexahedral (b) meshes. With the same mesh and material parameters as in Fig. 1, our tetrahedral model is perfectly able to sustain flexion, as shown by its equilibrium state (c).

5.3 Performance Issues

Our benchmarks are on an *SGI O2* workstation with a MIPS R5000 CPU at 300 MHz with 512 Mbytes of main memory. Experiments use tetrahedral and hexahedral meshes lying on a table under force of gravity. Other conditions are similar to those of volume preservation experiments (see caption of Fig. 3). Note that material stiffness strongly influences computation time since we use an explicit integration method.

Maximum number of springs per element. For a classical mass-spring system, a tetrahedral element has 6 structural springs along its edges, and an hexahedral element has 12 structural springs along its edges plus 4 shear springs along its main diagonals. We do not use bending springs between hexahedral elements, as in Chen’s work [2]. This has to be compared with 3 axial springs, 3 angular springs and 4 volume springs (undamped), that gives approximately 10 springs for our tetrahedral element, and 3 axial springs, 3 angular springs and 8 volume springs, that gives 14 springs for our hexahedral element.

We can conclude from the results displayed in Table 1 that simulating anisotropic behavior and ensuring volume preservation are not very expensive in our model. These properties make it suitable for interactive applications. However, the cost of our method is directly related to the number of elements. Thus, unlike mass-spring systems, our benchmark experiment using the tetrahedral mesh is slower than the one using the hexahedral mesh.

		Masses	Elements	Springs	Sp./Elt.	Time (in s)
Ms.-Sp. Sys.	Tetra	222	804	1175	1.461	0.129
	Hexa	216	125	1040	8.320	0.117
Our Model	Tetra	222	804	≈ 8040	≈ 10	1.867
	Hexa	216	125	1750	14	0.427

Table 1. Benchmarks results for classical mass-spring system and our model with tetrahedral and hexahedral meshes. See explanations in the text concerning the estimated number of springs per element in our model. *Legend:* Ms.: mass, Sp.: spring, Etl.: element, Time: time spent to compute one second of animation, with a time step of 0.01 s.

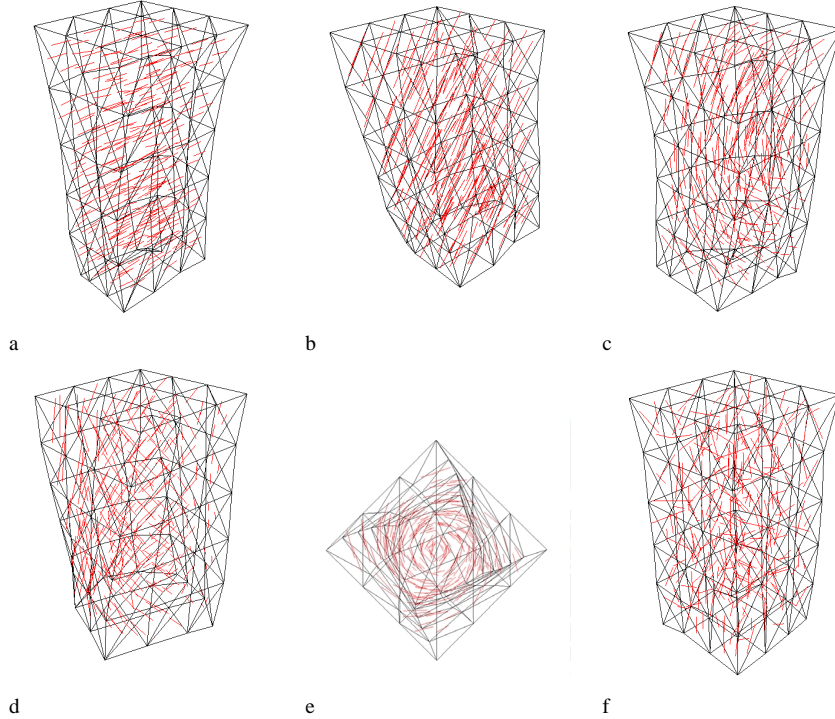


Fig. 6. Different anisotropic behaviors were obtained using the same tetrahedral mesh undergoing a downward pull at its bottom end while its top end is fixed. Anisotropy is tuned by changing the stiffest direction in the material. This direction is: (a) horizontal (as a result, the material tends to get thinner and longer), (b) diagonal (with angle of $\frac{\pi}{4}$, which constrains the material to bend in this manner), (c) hemicircular (as a C shape, which causes a snake-like undulation of the material), side (d) and top view (e), concentric helicoidal (the material successively twists and untwists on itself) and finally (f) random (the material exhibits an isotropic behavior).

6 Conclusion and Future Work

We have presented an alternative formulation for mass-spring systems, where anisotropy of a deformable volume is specified independently from the geometry of the underlying mesh. There are no requirements for the mesh, that may be built from either tetrahedral or hexahedral elements. Moreover, a method for generating constant volume deformations is provided.

The new model stays very close to mass-springs systems, since it is as easy to implement and almost as efficient in computation time. It also benefits from the ability of mass-spring systems to animate large deformations and large displacements.

Further investigations are needed in order to validate our model. In particular, we are planning to study the equivalent stiffness along orientations that do not correspond to axes of interest. Once this is done, we may be able to generalize the method to anisotropic material where more than three axes of interest are defined.

Other interesting possibilities arise by combining different volume element types to obtain an hybrid mesh which better approximates the shape of the object; or by using

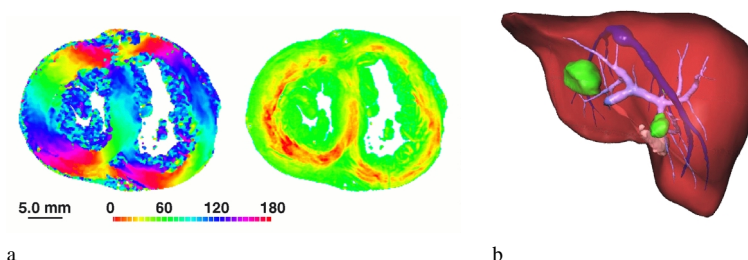


Fig. 7. Two examples of complex anisotropic materials. In (a), angular cartographies of the muscle fiber direction obtained on a human heart (at left, map of the azimuth angle, at right, map of the elevation angle). In (b), a human liver with the main veinous system superimposed.

elements of different orders (linear vs quadratic interpolation, etc.) in the same mesh.

On the application side, we are currently working on human heart motion simulation. This is a challenging problem since the heart is an active muscle of complex geometry, where anisotropy (caused by muscle fibers varying directions, see Fig. 7.a) plays an important role [6]. Important work has already been done to measure fiber direction inside a human heart [18]. We plan to use this data for animating a full scale organ. To do so, we will have to change our linear axial springs to non-linear active axial springs, whose stiffness and rest length vary over time.

The human liver is also a good example of anisotropic material, although it has been previously animated using isotropic elastic models [3]. In fact, it can be seen as a composite material: the root-like structures of rather rigid vessels are embedded in the liver tissue itself, which is a soft material (see Fig. 7.b).

Future work finally includes possible generalization to surface materials, such as cloth. To do so, extra parameters controlling bending will have to be added to the current volume model.

Acknowledgments

This work was performed within the framework of the joint incentive action “Beating Heart” of the research groups ISIS, ALP and MSPC of the French National Center for Scientific Research (CNRS). It is partly supported by the Liphia Santé Company, a subsidiary of the group MERCK KGaA. The human liver data were kindly provided by the Epidaure project and IRCAD. We thank François Faure and Jacques Ohayon for fruitful discussions; James Stewart and George Drettakis for rereading the paper. Finally, thanks to our first reviewer for her/his helpful comments.

References

- [1] J. E. Chadwick, D. R. Haumann, and R. E. Parent. Layered construction for deformable animated characters. In *SIGGRAPH '89 Conference Proceedings*, pages 243–252. Addison Wesley, July 1989.
- [2] Y. Chen, Q. Zhu, and A. Kaufman. Physically-based animation of volumetric objects. In *Proceedings of IEEE Computer Animation '98*, pages 154–160, 1998.
- [3] S. Cotin, H. Delingette, and N. Ayache. Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions On Visualization and Computer Graphics*, 5(1):62–73, January 1999.
- [4] O. Deussen, L. Kobbelt, and P. Tucke. Using simulated annealing to obtain good nodal approximations of deformable objects. In *Proceedings of the 6th Eurographics Workshop on Animation and Simulation*, pages 30–43. Springer-Verlag, Sept. 1995.

- [5] P. Howlett and W. T. Hewitt. Mass-spring simulation using adaptive non-active points. In *Proceedings of Eurographics '98*, pages 345–354. Blackwell Publishers, 1998.
- [6] P. Hunter. Myocardial constitutive laws for continuum mechanics models of the heart. *Adv. Exp. Med. Biol.*, 382:303–318, 1995.
- [7] D. Hutchinson, M. Preston, and T. Hewitt. Adaptive refinement for mass/spring simulation. In *Proceedings of the 7th Eurographics Workshop on Animation and Simulation*, pages 31–45. Springer-Verlag, Sept. 1996.
- [8] Y. Lee, D. Terzopoulos, and K. Waters. Realistic face modeling for animation. In *SIGGRAPH '95 Conference Proceedings*, pages 55–62. Addison Wesley, Aug. 1995.
- [9] J. Louchet, X. Provot, and D. Crochemore. Evolutionary identification of cloth animation models. In *Proceedings of the 6th Eurographics Workshop on Animation and Simulation*, pages 44–54. Springer-Verlag, Sept. 1995.
- [10] A. Luciani, S. Jimenez, J. L. Florens, C. Cadoz, and O. Raoult. Computational physics: A modeler-simulator for animated physical objects. In *Proceedings of Eurographics '91*, pages 425–436. North-Holland, Sept. 1991.
- [11] G. S. P. Miller. The motion dynamics of snakes and worms. In *SIGGRAPH '88 Conference Proceedings*, pages 169–178. Addison Wesley, Aug. 1988.
- [12] V. Ng-Thow-Hing and E. Fiume. Interactive display and animation of b-spline solids as muscle shape primitives. In *Proceedings of the 8th Eurographics Workshop on Computer Animation and Simulation*. Springer-Verlag, Sept. 1997.
- [13] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C, Second Edition*, chapter Integration of ODE: Second-Order Conservative Equations, pages 732–734. Cambridge University Press, 1992.
- [14] E. Promayon, P. Baconnier, and C. Puech. Physically-based deformations constrained in displacements and volume. In *Proceedings of Eurographics '96*, pages 155–164. Blackwell Publishers, Sept. 1996.
- [15] X. Provot. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Proceedings of Graphics Interface '95*, pages 147–154. Canadian Human-Computer Communications Society, May 1995.
- [16] Simulog. Tet-mesh - ghs3d. <http://www.simulog.fr/tetmeshf.htm>. Tetrahedral mesh generator and optimizer.
- [17] D. Terzopoulos, J. Platt, and K. Fleischer. Heating and melting deformable models (from goop to glop). In *Proceedings of Graphics Interface '89*, pages 219–226. Canadian Human-Computer Communications Society, June 1989.
- [18] Y. Usson, F. Parazza, P. Jouk, and G. Michalowicz. Method for the study of the three-dimensional orientation of the nuclei of myocardial cells in fetal human heart by means of confocal scanning laser microscopy. *J. Microsc.*, 174(2):101–110, 1994.
- [19] A. Witkin. *SIGGRAPH Course Notes 60*, chapter Constrained Dynamics. ACM, 1993.
- [20] A. Witkin and W. Welch. Fast animation and control of nonrigid structures. In *SIGGRAPH '90 Conference Proceedings*, pages 243–252. Addison Wesley, Aug. 1990.