Mesh Topology Identification for Mass-Spring Models

Gérald Bianchi, Matthias Harders, and Gábor Székely

Swiss Federal Institute of Technology
Computer Vision Laboratory
ETH Zentrum, CH-8092 Zürich, Switzerland
{bianchi,mharders,szekely}@vision.ee.ethz.ch

Abstract. In surgical simulations, the two most popular approaches to model soft tissues are Finite Element Method (FEM) and Mass-Spring System (MSS). Main advantages of FEM are accuracy and realism. Furthermore, the model allows the direct integration of experimentally obtained biomechanical tissue parameters. However, computation times remain high, limiting real-time application of the method. In contrast to this, the main advantage of MSS is low computational complexity and simple implementation. These factors make the latter method highly attractive for virtual reality based surgical simulators. However, the specification of system parameters for a MSS (masses, spring constants, mesh topology) is not straightforward and remains a major difficulty of the approach. In this paper, we propose a solution to this problem based on evolutionary algorithms - our current focus being the determination of mesh topology. We use reference models to obtain the topology of a MSS. First results demonstrate, that the exact recovery of isotropic and anisotropic reference mesh configurations is possible.

1 Introduction

Realistic behavior and real-time capability are two main features required for surgical training simulators. These contradictory requirements pose a major problem to soft tissue modeling. While high accuracy is needed to achieve realism, highly complex models usually lead to increased computation times.

Primarily two approaches are proposed in the literature for modeling soft tissues in medical applications. The first one is the Finite Element Method, which provides a rigorous representation of soft tissue physics based on continuum mechanics. An advantage of the method is the possibility to directly integrate material parameters resulting from soft tissue measurements into the calculations. Different levels of accuracy for deformation simulation have been realized with this method, ranging from elastic linear [8] to nonlinear anisotropic systems [14]. Nevertheless, high computation times still remain an obstacle for real time applicability of the method.

The second approach is Mass-Spring Systems, introduced in [16]. These consist of a mesh of mass points connected by elastic links. System movement is

R.E. Ellis and T.M. Peters (Eds.): MICCAI 2003, LNCS 2878, pp. 50–58, 2003.

evaluated by integrating the fundamental law of dynamics. Due to the simplicity of the motion equations, the MSS is computationally attractive for medical simulators. Various applications have already used nodal systems to simulate soft tissues, for instance [1] simulates the cutting operation of deformable tissue based on mass-spring models and [5] models fat tissue in a craniofacial surgery simulator. However, one has to keep in mind, that a MSS represent a discrete model which only roughly approximates the true physics governing the deformation of an object [7]. Also [3] mentions, that it is delicate to validate deformations of a MSS with experimental biomedical data, since such a system does not rely on continuum mechanics. Nevertheless, the question which degree of realism is needed for achieving training effects when using a simulator still remains unanswered [9]. Assuming that a decision was made for using a MSS, a significant problem remains - the assignment of system parameters. Length, stiffness and transfer function of springs, mass distribution and mesh topology have to be defined. A usual approach to obtain a desired deformation behavior is the manual iterative tuning of a MSS. However, this process is tedious and does not guarantee to find an optimal solution.

In this paper we suggest a different approach to the problem. The main idea is to approximate the behavior of a physically accurate reference model (such as a FEM system) by adjusting the system parameters of the MSS. Since some attempts have already been made to determine mass distribution and spring constants, we direct our initial focus of the parameter adjustment process to the mesh topology. Usually, regular meshes are used in the majority of approaches using a MSS for tissue deformation. However, tissue characteristics like anisotropies or inhomogeneities already require adjustments at the level of the topology. Also in [10] the importance of an appropriate mesh topology has been identified. Meshes with identical mass and spring constants still behave quite differently depending on the overall topology. In the following we will describe the identification of the MSS topology by using a genetic optimization algorithm.

2 Previous Work

Few approaches have already been proposed to automatically identify parameters of phenomenological tissue deformation models.

In [13] and [15], a neuro-fuzzy network is designed to simulate the behavior of soft tissues. Linguistic terms defined by *if-then* rules are used to specify tissue characteristics and so initialise a neural network. The system parameters are then adjusted based on a fuzzy system. Unfortunately, this concept implies that the user manually has to tune the initialization of the parameters.

In [11] an evolutionary strategy is applied to identify spring parameters for a cloth model. The basic idea is to optimize a cost function, which measures the difference between the behavior of a predicted and a reference model.

Other ongoing research compares the deformation behavior of FEM and MSS. Recent work shows that assigning the same stiffness to all springs fails to simulate even a uniform elastic tissue [6]. In the case of triangulated spring meshes, they propose to specify the stiffness values proportional to triangle area and Young's modulus. These concepts were later applied to simulate soft tissue [2]. Biomechanical properties of real rat liver are measured and the experimental data are exported into their model.

The use of simulated annealing for identification of spring parameters from a reference model is suggested in [4]. Moreover, a method is proposed to obtain a homogeneous point and mass distribution.

All the MSSs described in these works have a predefined topology, usually regular, tetrahedral or hexahedral lattices. However, no research regarding the topology design according to the deformation behavior of MSS has been attempted so far.

3 Topology Identification

3.1 Overview

The main idea of our approach for identification of MSS mesh topology is comparison of the deformation behavior of a training model with a known reference system. The ground truth can be based on any deformation approach. However, for the time being we only compare two MSSs, since in this case the exact solution is known. It also has to be mentioned, that since we focus on the topological design, we assume constant mass distribution and spring stiffness for the time being. A cost function will be defined which measures the difference between the training and the reference model and an optimization is carried out based on the genetic algorithm approach.

3.2 Genetic Algorithm

Genetic algorithms [12] are based on the evolution of species. They consist of populations of individuals, each of the latter representing a potential solution to a problem - in our case the mesh topology. The optimization principle consists in evolving the population by means of the three following genetic operators:

Selection. This operator selects within the population a pair of individuals (parents) with a preselected probability. In our case, the selection strategy is based on the roulette wheel approach.

Crossover. This method consists in generating a new pair of individuals (offsprings) from the parents by concatenating the prefix of one parent with the suffix of the other one. The selected parents undergo a crossover with a prefixed probability. In our experiments we obtained the best results with a crossover value of 30%.

Mutation. This operator changes randomly the value of one or more genes of the genome. A genetic algorithm can be quite sensitive to this parameter. For values larger than 5%, the role of mutation becomes predominant and disturbs system convergence. The probability of mutation in our case is 1%.

The evolution of the system by means of these genetic operators converges to a population in which all the individuals are identical. This unique species represents the best solution of the problem. In our approach we used a C++ library for genetic algorithms, which was developed at the Massachusetts Institute of Technology [17]. Before starting the optimization an appropriate cost function measuring the fitness of an individual has to be selected.

3.3 Cost Function

Our cost function is based on the standard formulation of a MSS. We define different load cases on the mesh and compute the static deformations. The equilibrium position of a MSS is determined by Newton's first law of motion.

$$\mathbf{F_{ext}} + \mathbf{F_{int}} = \mathbf{0}$$

where $\mathbf{F_{ext}}$ represents the external forces applied to the system, such as forces exerted by surgical tools or gravity, and $\mathbf{F_{int}}$ corresponds to internal forces resulting from the tension of the springs. We introduce a binary variable α_{ij} defined by

$$\alpha_{ij} = \begin{cases} 1 \text{ if the points } P_i \text{ and } P_j \text{ are linked by a spring} \\ 0 \text{ otherwise} \end{cases}$$

This variable is used to represent the different topologies of the training mesh. If P_i is the i^{th} point of the mesh, the internal force $\mathbf{F_i}$ applied to P_i is now obtained by

$$\mathbf{F_i} = -\textstyle\sum_{\mathbf{j \in N(i)}} \mathbf{k_{ij}} \boldsymbol{\alpha_{ij}} (||\mathbf{X_i} - \mathbf{X_j}|| - \mathbf{l_{ij}} \frac{\mathbf{X_i} - \mathbf{X_j}}{||\mathbf{X_i} - \mathbf{X_j}||})$$

where k_{ij} is the stiffness of the spring between points P_i and P_j with natural length l_{ij}, $\mathbf{X_i}$ and $\mathbf{X_j}$ represent the coordinate vectors of points P_i and P_j, and $N(i)$ is the set of points connected to point P_i. The total internal forces with m points are expressed as

$$\mathbf{F_{int}} = \textstyle\sum_{\mathbf{i=0}}^{\mathbf{m}} \mathbf{F_i}$$

We can now define the cost function as the difference between the deformation of a training MSS M with a topology defined by the values of α_{ij} and a reference model R.

$$f(\alpha_{01}, \alpha_{02}, \alpha_{03}, ...) = \textstyle\sum_{i=1}^{m} \|\mathbf{X_i^R} - \mathbf{X_i^M}\|^\mathbf{2}$$

where $\mathbf{X_i^R}$ is the equilibrium position of the i^{th} point of R and $\mathbf{X_i^M}$ is the equilibrium position of the i^{th} point of M. The best topology of M can now be obtained by minimizing the cost function. Since the function f is not differentiable and the number of variables can become very high depending on the number of connections considered for M, the most appropriate method to optimize f is a genetic algorithm based approach.

3.4 Topology Optimization

After selecting the cost function, the genetic algorithm has to be initialized.

Genome. The genome (or individual) is a string of bits which represents a potential topology of the mass-spring model. The size of the string depends on the number of neighbors we allow for each node. This number defines the search area for optimizing the connections to the node under consideration.

Population. We noticed that population sizes of larger than 20 individuals did not improve the results, but instead increased computation times. Therefore, the population consists of 20 genomes.

Initializing. The population is initialized with random binary values. The points X_i^M of the training model are initialized with the positions of the reference model. We assume the number of nodes of M is equal to the one of R and each point of R has a corresponding point in M. Other topology information of the reference model R is not used. Since our initial experiments are carried out in 2D, we allow at most 8 neighbors for each node.

In order to catch the elastic behavior of the reference model R, we apply representative load cases to it. The row A of the array in Figure 1 shows the different equilibrium positions of an example model while applying different loads. The first column contains the model without loads and the second one shows deformations under stretching loads. Shearing forces are represented in the third column and compression forces in the last one. From this set of load cases, the genetic algorithm has to find a topology of M which approximates all the deformations of R. Our experiments have shown, that incorporation of all load cases into the genetic algorithm from the start of the process gives the best results. Therefore, we extend the cost function over all the deformations (n forces)

$$f(\alpha_{01}, \alpha_{02}, \alpha_{03}, ...) = \sum_{k=1}^{n} \sum_{i=1}^{m} \|\mathbf{X_{i,k}^R} - \mathbf{X_i^M}\|^\mathbf{2}$$

where $\mathbf{X_{i,k}^R}$ is the position of the i^{th} point of R in the k^{th} configuration.

The topology identification is obviously computational expensive. For each individual of the population, we have to compute 7 different equilibrium positions of M and compare them with those of R. The first experiments with a reference and model mesh defined by 5x5 points and 72 springs, provided a solution in about 3 hours on SUN Workstation with eight UltraSparc-III+ CPUs at 900 MHz with 15 GBytes of shared memory. However, we are currently in the process of optimizing this step and initial results show computation times of around 30 mins.

4 Experimental Results

In this section, we show the results obtained by comparing two MSSs in order to verify the ability of the genetic algorithm to recover different known mesh topologies. Experiments were carried out for two different regular models characterizing linear isotropic and anisotropic deformations. The last experiment

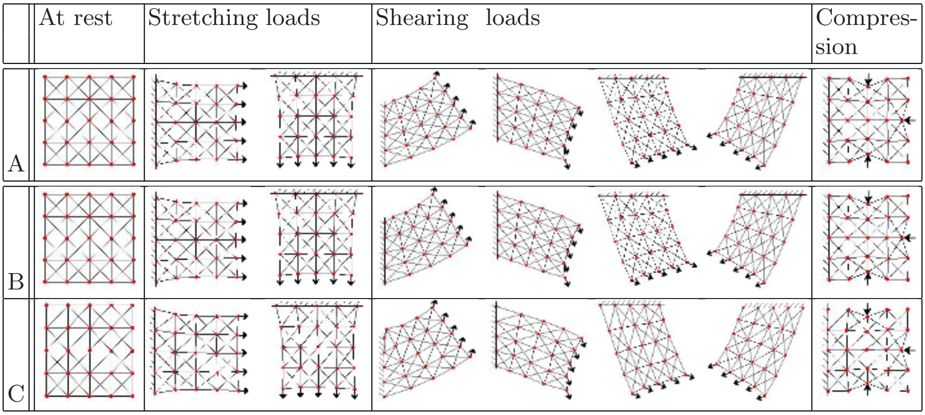

Fig. 1. Row A corresponds to the reference model. Rows B and C are two different results given by the genetic algorithm (20 individuals, mutation rate 1%, crossover rate = 30%)

shows the results obtained for an irregular reference mesh (a cross-section of a kidney).

Four independent trials to obtain the regular isotropic mesh have been carried out. In three cases, the optimization fully recovered the topology of the reference mesh in about 400 iterations as shown in row B of Figure 1. In these cases the cost function reaches zero. However, in one case shown in row C the algorithm did not converge to zero after the maximum number of 1000 iterations. While only a suboptimal solution missing a few links has been found the global behavior remains close to that of *R*. This is also illustrated by the resulting low cost function value reached (0.01). The evolution of the cost function for one optimal and the suboptimal case is depicted by Figure 2.

Fig. 2. Evolution of cost function for optimal and suboptimal case

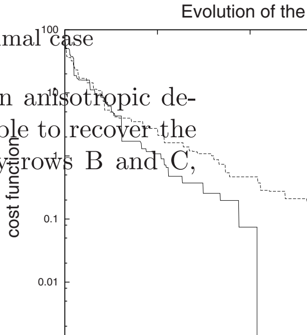

Figure 3 displays the results with a mesh characterizing an anisotropic deformation. Again, out of four experiments the algorithm was able to recover the topology in three cases. The resulting solutions are shown by rows B and C,

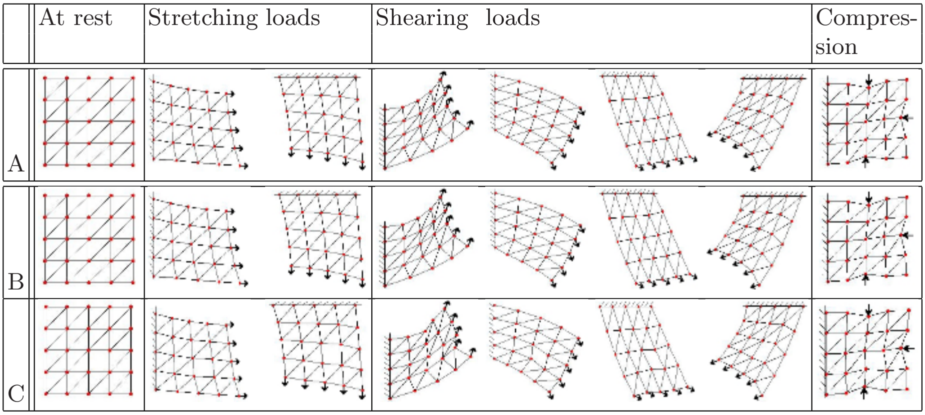

Fig. 3. The first row corresponds to the reference model. Row B and C show the results given by the genetic algorithm (20 individuals, mutation rate 1%, crossover rate = 30%)

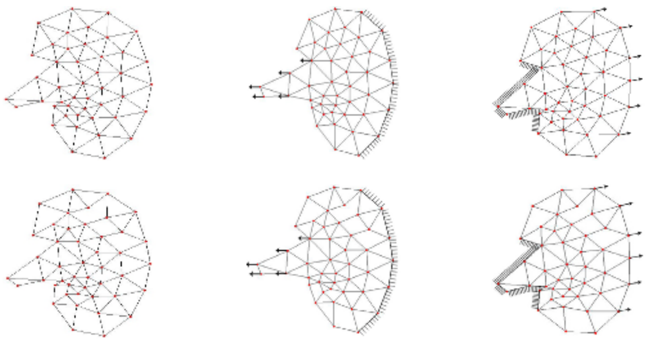

Fig. 4. First row: Reference model (cross-section of kidney). Second row: Result

respectively. Figure 4 depicts the result for an irregular mesh with the same genetic parameters defined previously. In this case not all connections are recovered. However, the global behavior remains closely similar to the observations on the reference.

5 Conclusion and Future Work

We have proposed a method based on a genetic algorithm to identify the topology of MSSs by comparing its behavior with the one of a known reference model. Our algorithm was able to fully or almost perfectly recover the topology of the MSSs. Spring connections characterizing linear isotropic as well as anisotropic deformations were successfully identified.

These initial experiments were realized with a mass spring reference model in order to verify the ability of the genetic algorithm to find an a priori known ground truth. The next step will be the use of a FEM model as the reference model. Moreover, we also plan to integrate the full dynamic behavior of the deformations into the genetic optimization calculations in order to further improve the recovery process. Finally, we will extend the approach to the third dimension.

Acknowledgment. This research has been supported by the NCCR Co-Me of the Swiss National Science Foundation.

References

1. J. Brown, S. Sorkin, J.C. Latombe, K. Montgomery, and M. Stephanides. Algorithmic tools for real-time microsurgery simulation. *Medical Image Analysis*, 6(3): 289–300, September 2002.
2. C. Bruyns and M. Ottensmeyer. Measurements of soft-tissue mechanical properties to support development of a physically based virtual anima model. In *MICCAI 2002*, pages 282–289, 2002.
3. H. Delingette. Towards realistic soft tissue modeling in medical simulation. Technical Report 3506, INRIA Sophia-Antipolis, Septembre 1998.
4. Oliver Deussen, Leif Kobbelt, and Peter Tucke. Using simulated annealing to obtain good nodal approximations of deformable objects. In *Computer Animation and Simulation '95*, pages 30–43. Springer-Verlag, 1995.
5. Keeve E., Girod S., and Girod B. Craniofacial Surgery Simulation. In *4th International Conference on Visualization in Biomedical Computing VBC'96*, pages 541–546, Hamburg, Germany, September 22–25 1996.
6. A. Van Gelder. Approximate simulation of elastic membranes by triangulated spring meshes. *Journal of Graphics Tools*, 3(2): 21–42, 1998.
7. S. F. F. Gibson, G. Mirtich, and B. Mirtich. A survey of deformable modeling in computer graphics. Technical Report 97–19, MERL, November 1997.
8. E. Gladilin, S. Zachow, P. Deuflhard, and H.C. Hege. A biomechanical model for soft tissue simulation in craniofacial surgery. In *Medical Imaging and Augmented Reality (MIAR)*, pages 137–141, 2001. Hong Kong, China.
9. M. Harders, R. Hutter, A. Rutz, P. Niederer, and G. Székely. Comparing a Simplified FEM Approach with the Mass-Spring Model for Surgery Simulation, (MMVR'2003).
10. J.C. Latombe. Modeling human-body soft tissues for surgical applications, February 1999. Research Proposal, Robotics Laboratory Stanford.
11. J. Louchet, X. Provot, and D. Crochemore. Evolutionary identification of cloth animation models. In *Computer Animation and Simulation '95*, pages 44–54. Springer-Verlag, 1995.
12. Z. Michlewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer, 1999.
13. A. Nürnberger, A. Radetzky, and R. Kruse. A Problem Specific Recurrent Neural Network for the Description and Simulation of Dynamic Springs Models. In *IEEE International Joint Conference on Neural Networks 1998 (IJCNN '98)*, pages 468–473, 1998. Anchorage, Alaska.
14. G. Picinbono, H. Delingette, and N. Ayache. Non-linear and anisotropic elastic soft tissue models for medical simulation. In *ICRA2001: IEEE International Conference Robotics and Automation*, pages 1370–1375, Seoul Korea, May 2001.
15. A. Radetzky, A. Nürnberger, and P. Pretschner. Elastodynamic shape modeler: A tool for defining the deformation behavior of virtual tissues. *RadioGraphics 2000*, 20(1): 865–881, January-March 2000.

16. D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. *Computer Graphics (Proc. SIGGRAPH'87)*, 21(4): 205–214, 1987.
17. M. Wall. Galib: A C++ Library of Genetic Algorithm Components, 1999. MIT.