

Simultaneous Topology and Stiffness Identification for Mass-Spring Models Based on FEM Reference Deformations

Gérald Bianchi, Barbara Solenthaler, Gábor Székely, and Matthias Harders

Swiss Federal Institute of Technology
Computer Vision Laboratory
ETH Zentrum, CH-8092 Zürich, Switzerland

`{bianchi,szekely,mharders}@vision.ee.ethz.ch`

Abstract. Mass-spring systems are of special interest for soft tissue modeling in surgical simulation due to their ease of implementation and real-time behavior. However, the parameter identification (masses, spring constants, mesh topology) still remains a challenge. In previous work, we proposed an approach based on the training of mass-spring systems according to known reference models. Our initial focus was the determination of mesh topology in 2D. In this paper, we extend the method to 3D. Furthermore, we introduce a new approach to simultaneously identify mesh topology and spring stiffness values. Linear elastic FEM deformation computations are used as reference. Additionally, our results show that uniform distributions of spring stiffness constants fails to simulate linear elastic deformations.

1 Introduction

Real-time simulation of soft tissue deformation remains a major obstacle when developing surgical simulator systems. One popular approach is based on the mass-spring model (MSM), which consists of a mesh of mass points connected by elastic links. The method requires the setting of system parameters describing deformation behavior. Parameters, such as mass distribution, coefficients of spring transfer functions and overall connectivity have to be determined. In [1], we have suggested an approach based on genetic optimization for the identification of MSM parameters. Our initial focus has been on mesh topology. The main idea is to compare the deformation behavior of a learning model with that of a known reference system and to utilize genetic algorithms to optimize the parameters of the learning model.

In this paper, we first present the extension of our previously described approach to 3D. Then, we introduce a new approach, which merges topology and spring constant parameter estimation. We show the validity of this method by the successful recovery of the topology of a reference MSM. Next, we introduce Finite Element Models (FEM) as the reference model and obtain MSM parameters describing FEM deformation behavior. Several experiments support the

validity of the acquired parameters. Finally we show, that linear elastic material cannot be approximated with homogeneous MSM parameters, but requires inhomogeneous parameter distributions.

2 Previous Work

The last ten years have seen a growing interest in research on the stiffness value identification of mass-spring systems. Two main approaches have been proposed so far. The first one focuses on the determination of mathematical relationships in the computation of mesh properties of MSMs based on known values. In [3] stiffness values in triangulated spring meshes were computed proportional to triangle area and Young's modulus. Different generic methods for particle-based systems (referred to as generalized mass-spring systems) have been suggested in [7]. One approach obtains stiffness values in rectangular structures according to angles between diagonal springs. Another method computes the spring constants based on the number of connections attached to a mass-point under consideration. Nevertheless, a general, non-heuristic formulation does not yet exist.

The second approach is based on optimization processes, which try to adapt the behavior of a MSM. A few optimization-based approaches have been proposed in the literature, for instance in [2] the use of simulated annealing for spring constant identification is suggested. Neural networks are used for the simulation of dynamic MSMs by [9]. Furthermore, methods based on genetic algorithms for stiffness value determination were discussed in [4] and [6]. However, all the methods described above only work for predefined topologies such as rectangular or tetrahedral structures. In this paper, we suggest a new solution, which simultaneously focuses on connections and elastic constants to be set in MSM systems. Moreover, our method is based on the comparison of deformation behavior of a MSM with a known, possibly more accurate reference system.

3 Topology Identification in 3D

In [1] we proposed an approach based on genetic optimization to identify the topology of mass-spring systems. A MSM (the *learning model*) was trained by means of a known model (the *reference model*) undergoing stretching and shearing induced by external forces. A cost function measured the difference between the behavior of the learning and the reference model, based on the distance between corresponding point positions in both models. In order to test the method, an MSM was used as reference, since in this case the exact solution was known. Our results for two-dimensional test cases have shown, that the method was able to recover the topology of isotropic and anisotropic reference models

Our first extension is the application of the described approach in 3D. Due to the increasing number of springs in this case, the genetic algorithm has to be optimized in order to converge faster towards the optimal solution. Furthermore, a reasonably limited neighborhood for possible spring connections has to be

defined for 3D. Inspired by tightly-packed crystalline structures, we allow at most 26 neighbors (*Moore Neighborhood*) for each node.

3.1 Genetic Algorithm Adaptation

Genetic algorithms (GAs) attempt to mimic natural evolution [8]. They utilize the behaviour of a population of individuals - each one representing a potential solution to a defined problem. The fitness of an individual is determined by a cost function. The optimization principle consists of evolving the population by means of genetic operators such as *mutation* and *crossover*. Mutation applies random changes to the population with a certain probability $pmut$. Crossover creates offsprings by selecting genes from a pair of individuals and combining them into a new one, also with a predefined probability $pcross$. Although GAs do not guarantee a convergence to the global optimum, the reached local optimum may be considered as a good approximation of the exact solution.

Similar to the 2D case, an individual is described by a vector of binary values which represents a potential topology of the mass-spring model. In our experiments we found no significant influence of the population size on the convergence speed of the algorithm as well as on the results. Therefore, a low value was selected for the size to reduce the computation time ($popsiz = 5$). While the

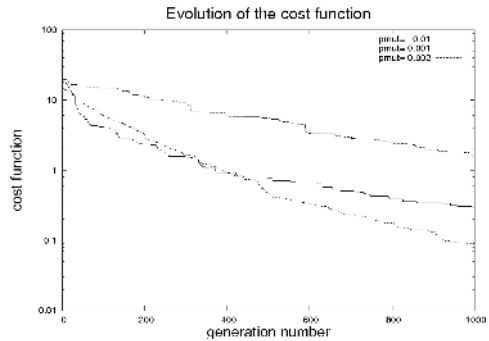


Fig. 1. Comparison of three constant mutation values

crossover operator improves convergence in the early stages of the process, its influence becomes of secondary importance later, when the population is close to the optimum. Therefore, the crossover probability is fixed to a high value ($pcross = 0.8$). In contrast to this, the mutation value has an important effect throughout the whole evolution. Figure 1 depicts the evolution of three trials with different mutation probabilities. While a high value ($pmut = 0.01$) speeds up the convergence at the beginning of the evolution, a lower one ($pmut = 0.002$) gives better results in later stages. Due to this observation, we decided to use an adaptive mutation probability which is inversely proportional to the generation number g : $pmut(g) = \frac{a}{g} + b$. We were able to considerably improve convergence speed with this adaptive mutation strategy. The coefficients a and b are currently determined experimentally - in the case of 3D topology identification, the section $a = 0.2$ and $b = 0.001$ has been found to provide best convergence properties.

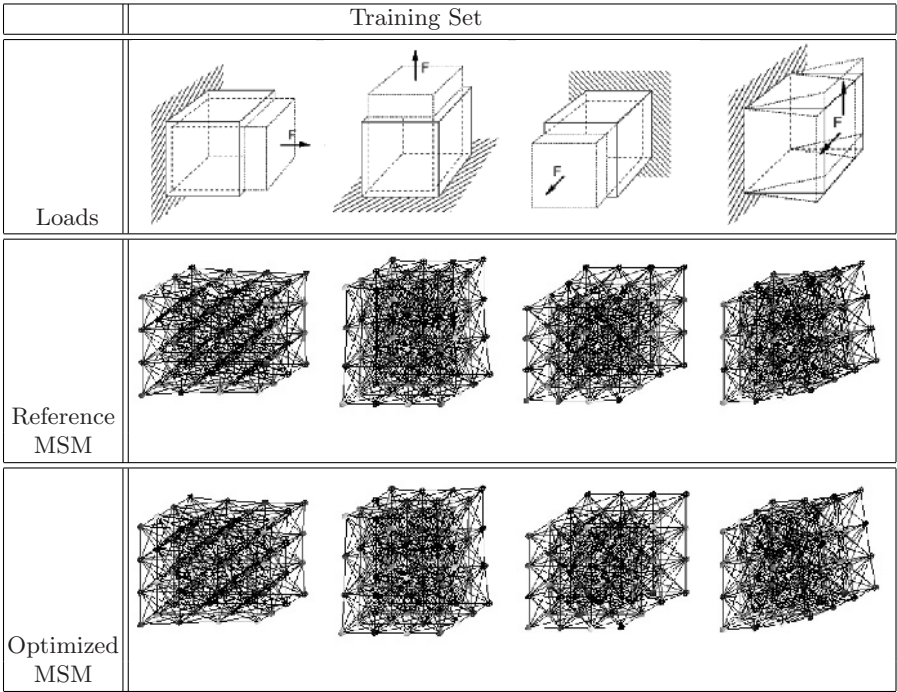


Fig. 2. Row 1: Loads used for training. Row 2: Deformations of reference MSM. Row 3: MSM optimized by genetic algorithm ($generation = 4000$, $popsiz e = 5$, $pmut = \frac{0.2}{g} + 0.001$, $pcross = 0.8$).

3.2 Experimental Results

This section describes the results in the three-dimensional case with a reference mesh consisting of 4x4x4 mass-points and 468 springs. In Figure 2, the first row shows four out of six different load cases used in the training process. The selection of these test cases, including stretching and shearing, was motivated by suggestions made in [2]. The second row depicts the reference MSM deformed by the loads, while row three illustrates the best topology obtained by the genetic algorithm. Here, 5 of 468 springs are missing, however, the normalized difference of point positions between the reference and the learning model is equal to zero. Thus, we were able to very closely approximate the deformation behavior of the reference model and recover almost all connections. Moreover, it proved to be sufficient to use the applied six load cases to obtain the parameters describing the deformation of the reference model.

4 Simultaneous Topology and Spring Constant Identification

In this part we introduce a significant extension of our method. Our focus so far was on the retrieval of mesh topologies. This is now extended by including the

identification of individual spring constants into the approach. We achieve this by representing springs in the optimization population as real-valued constants instead of binary connections. Springs are only present, if their stiffness value is greater than a predefined threshold. This approach allows us to determine topology and stiffness values simultaneously. We again start to develop and validate this method with two-dimensional references. Our first step is to use MSM for comparison, however, we also introduce FEM systems as training references.

4.1 Genetic Algorithm

Binary encoding is no longer appropriate for the current method, since the elasticity of springs is a real value. Therefore, a vector of real constants is used to describe the stiffness values, while an interval is defined to limit the range of possible elasticity values. Furthermore, the mutation operator has to be defined differently. Instead of swapping bits, we add a random value x to the current stiffness, where x is limited to an interval I . The value x is normally distributed with zero mean and standard deviation σ . Since a constant σ will cause the random steps to be too high, once the population is close to the optimum, we propose to decrease σ during the optimization process. This can be achieved by defining the deviation according to $\sigma(g) = pmut(g) * sizeof(I)$. The interval size allows $\sigma(g)$ to take on the largest possible value at the beginning of the optimization. Finally, population size and crossover probability are again fixed throughout the evolution.

4.2 MSM as Reference

We performed a comparison between two MSMs with 5x5 points and 72 springs. The goal of this experiment was to test the capability of the method to recover a single stiffness value of a reference MSM. Therefore, we adjusted all the springs of the reference to the same stiffness value of 5. The interval I of the random mutation value x was assigned to $[0.0, 10.0]$. For the same reasons described in Section 3.1, the crossover probability was set to 0.8. Since these tests were done in 2D, we increased the population size to 10 individuals. Our experiments have shown, that greater values did not improve the results, but only increased computation time. Also, we again used an adaptive mutation probability function. The coefficients $a = 2$ and $b = 0$ provided the best results. The experiment was performed with 4000 generations. We were able to recover the complete topology of the reference model. Furthermore, the mean value of the retrieved stiffness was equal to 5.01 with a standard deviation of 0.65. Thus, we were able to simultaneously recover the connections and elasticity values of a reference MSM.

So far we have trained our learning model based on known MSM configurations. However, we would like to use more accurate references which are capable of physically modeling the elastic behavior of real tissue based on continuum

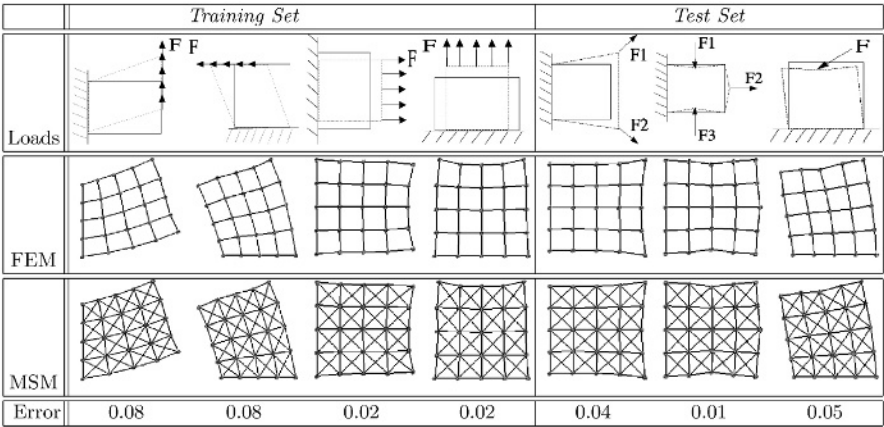


Fig. 3. Column *Training Set*: Input data and result after training process. Column *Test Set*: Comparison between optimized MSM and additional FEM deformations not used during training for evaluation. Row 1: Test loads used for training MSM and comparison. Row 2: Linear elastic FEM deformations. Row 3: Optimized MSM with both topology and stiffness values obtained. Row 4: Comparison between mean error and smallest rest length of MSM.

mechanics. At the same time it is of primary importance, that the related material parameters can be determined systematically based on actual experiments. This extension will be presented in the next section.

4.3 FEM as Reference

Continuum mechanics based Finite Element Models have been used to accurately simulate soft tissue deformation, however, their high computational demands remain an obstacle for real-time applications. Realistic deformation parameters can be obtained by measurements on real organs. For instance in [5], in-vivo experiments have been introduced to determine appropriate material parameters of complex, non-linear constitutive equations. In order to approximate the behavior of these models, our next step focuses on training a learning MSM based on a reference FEM system. The reference mesh consists of quadrilateral elements, where the vertices correspond to the mass-points in the MSM model. The deformations are computed according to a linear elastic model with Young's modulus $E = 1MPa$ and Poisson coefficient $\nu = 0.3$. For the load cases several training sets were examined including shearing and stretching deformations. Sets of four, eight and twelve different load cases were evaluated. The best results were obtained with the latter set. Similar to the previous experiments, the cost function is based on the distance between the FEM vertices and the MSM mass-points. Population size and crossover probability remain identical to those described in the previous section. The coefficients of the mutation probability function were slightly adapted to $a = 5$ and $b = 0$. The first trial indicated, that all spring con-

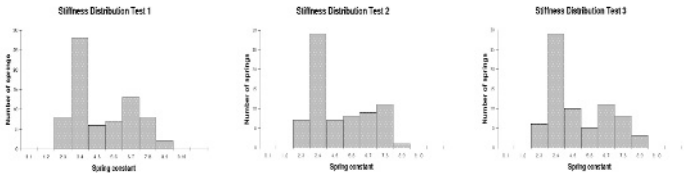


Fig. 4. Comparison of stiffness value distribution obtained by three optimization processes with the same input parameters.

stants were within the interval $[0.0, 10.0]$, therefore, these limits were retained for all experiments. We performed the same optimization steps to obtain the MSM mesh parameters. In order to validate our results, we defined additional load cases and compared the behavior of the previously learned MSM with the corresponding FEM deformations. Figure 3 exemplifies the results of both experiments. The column labelled *Training Set* illustrates the input data as well as the resulting configuration of the training process. The first row depicts four of the twelve test loads applied to the FEM. The second row shows the FEM deformations and the third row contains the learning MSM with the best stiffness values after 3000 generations. In the column *Test Set*, the MSM solution is compared to three elastic linear FEM deformations not included in the training set. The last row shows an error metric for the deformation differences. The mean error distance of the FEM and MSM points is set in relation to the smallest spring rest length of the optimized MSM. As revealed by the error, the stiffness values of the learning MSM seem to be equally well adjusted for both the training and test FEM cases. This experiment shows the capability of our method to finding appropriate stiffness values, allowing us to approximate the deformation behavior of a linear elastic FEM model.

4.4 Stiffness Value Distribution

After obtaining the spring constants from the FEM reference model, we examined the distribution of the stiffness values. Figure 4 shows the results of three trials with the same input data. The plots show that a linear elastic material cannot be approximated with one homogeneous stiffness parameter. Instead, the figures seem to indicate that at least two major classes of stiffness values exist (represented by the two peaks). One possible assumption is that one has to differentiate between diagonal and straight springs. In order to investigate this observation, we performed two additional experiments. The main idea behind these trials is to limit the spring constants to belong to one or two homogeneous classes. In other words, in the first experiment, only one global stiffness value is optimized, while in the second, diagonal and straight springs were optimized as two individual classes. The results of these experiments are summarized in Table 1. It can be observed that the best results are still achieved with our previously performed unlimited optimization experiments with a cost function value

Table 1. Comparison of limited optimizations for stiffness value estimation according to classes.

Limitation	Final cost function	Stiffness values
Single, homogeneous stiffness	0.85	All springs 4.07
Two stiffness classes	0.73	Diagonal springs 3.35, straight springs 4.76
Unrestricted stiffness values	0.58	Spring stiffnesses in interval [0.0, 10.0]. Main classes of stiffness values in [3.0, 4.0] and [6.0, 8.0]

of 0.58. Assigning only one homogeneous stiffness parameter to all springs gives the worst results. This is also in line with the findings described in [3]. Defining two classes of springs with differing constant values slightly improves the results, however, optimal performance cannot be achieved. Furthermore, both retrieved spring constants of the two-classes belong to the same cluster described by the first peak of the distribution graph. This seems to indicate, that such simple rules based on connection topology might not be sufficient to explain the observed pattern. Besides, due to the small size of the meshes, it is not clear how far boundary effect may explain the observed results. Therefore, larger meshes will be investigated to further analyze the non-homogeneous stiffness distribution.

5 Conclusion and Future Work

We have introduced an extension of our previous work based on genetic algorithms to identify the topology of 3D MSMs. Moreover, a successful simultaneous topology and spring constant identification approach has been described. Our method was able to recover topology and stiffness values of reference MSMs. Moreover, we were able to approximate the behavior of FEM deformations with an optimized MSM mesh. Finally we could show, that homogenous stiffness parameters are not appropriate for simulating linear elastic material.

In future work, we will further evaluate adaptive mutation functions in order to improve the convergence behaviour of the genetic algorithm. We will examine the stiffness value distribution more closely and try to find underlying patterns. Finally, neighborhood selection and the approximation of more complex FEM models will be investigated.

Acknowledgment. The authors would like to thank Stefan Weiss for the FEM deformation calculations. This research has been supported by the NCCR Co-Me of the Swiss National Science Foundation.

References

1. G. Bianchi, M. Harders, and G. Székely. Mesh topology identification for mass-spring models. In *MICCAI 2003*, volume 1, pages 50–58, 2003.
2. 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, 1995.
3. A. Van Gelder. Approximate simulation of elastic membranes by triangulated spring meshes. *Journal of Graphics Tools*, 3(2):21–42, 1998.
4. A. Joukhadar, F. Garat, and Ch. Laugier. Parameter Identification for Dynamic Simulation. In *In Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1928–1933, 1997. Albuquerque, US.
5. M. Kauer, V. Vuskovic, J. Dual, G. Szekely, and M. Bajka. Inverse finite element characterization of soft tissues. *Medical Image Analysis*, 6(3):275–287, 2002.
6. J. Louchet, X. Provot, and D. Crochemore. Evolutionary identification of cloth animation models. In *Computer Animation and Simulation '95*, pages 44–54, 1995.
7. A. Maciel, R. Boulic, and D. Thalmann. Deformable tissue parameterized by properties of real biological tissue. In *IS4TM*, pages 74–87.
8. Z. Michlewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1999.
9. A. Nürnberger, A. Radetzky, and R. Kruse. A Problem Specific Recurrent Neural Network for the Description and Simulation of Dynamic Springs Models. In *IEEE IJCNN'98*, pages 468–473, 1998.