Modeling and actuation of cable-driven silicone soft robots

M. Frâncu

University of Copenhagen, Denmark

Abstract

In this paper we present a framework for modeling cable-driven soft robots fabricated from silicone rubber - an incompressible material. Our forward simulation model can use either the standard or the mixed formulation of the finite element method (FEM). The latter prevents volumetric locking for incompressible materials and is more accurate for low resolution meshes. Hence, we show that mixed FEM is well suited for estimating elastic parameters and simulator validation. We also introduce a cable actuation model using barycentric coordinates and then use it to solve some simple control problems.

CCS Concepts

• ***Computing methodologies*** → ***Physical simulation;*** • ***Computer systems organization*** → *Robotics;*

1. Introduction

Soft robots are an emerging type of robots usually made of continuous deformable elastic parts. The majority of these robots are fabricated using silicone rubber which is a soft incompressible material. This means it can easily bend, shear or twist without changing its overall volume, similarly to soft organic tissue. In technical terms, this means it has a Poisson ratio of exactly 0.5 or an infinite bulk modulus. It is well known that such materials cause various problems to FEM simulators. For instance, *locking* manifests through abnormally low deformations even for high resolution meshes. This can be addressed by using incompressibility constraints and a special discretization scheme for pressure known as the mixed formulation of FEM [ST91]. In this paper we employ mixed FEM along with standard FEM to simulate silicone rubber. We show how to use this model to solve inverse problems often occurring in robotics, e.g. parameter estimation and control. This work is based on our previous work on locking and mixed FEM [FARE21] and also draws inspiration from [KN19].

For isotropic elastic materials there are usually two parameters. These can be the Lamé parameters μ and λ, or the Young's modulus E and the Poisson ratio ν. Nonlinear material models that use these parameters include Saint Venant-Kirchoff and Neo-Hookean, of which we will use the latter as in [FARE21] (other nonlinear materials have not been implemented yet). We left out the density of the body ρ as it is easier to measure in the real world. In our case we found $\rho = 1080\ \text{kg/m}^3$ for the Ecoflex silicone used for molding.

2. Related work

Recently a lot of attention has been given to modeling and controlling soft robots fabricated from silicone rubber. Researchers from ETH optimized a soft robot locomotion policy using an underlying inverse FEM model [BBPC19]. The Defrost team devised control policies using the SOFA simulator that also take into account contacts [LVC*15]. Also, a lot of effort was put into the forward simulation of soft robots and their environment [MEM*19, GML*19]. In computer graphics, similar problems were solved for controlling virtual soft characters [TTL12].

For parameter estimation there has been significant work on data-driven cloth models [WOR11, MBT*12]. We also mention work on volumetric soft bodies [BBO*09, MMO16], time dependent estimation [WWY*15], and computational design [STC*13]. A more recent work focuses on visco-elastic parameter estimation for cable-driven soft robots made of silicone rubber foam [HBBC19]. We draw inspiration from all of these sources and others, but what differentiates us is the use of mixed FEM for locking avoidance. We also present a cable model based on a barycentric embedding inside the finite elements, which we have not seen described in the cited work.

3. Forward and backward models

In brief, an FEM simulator most often reduces to solving a nonlinear equation $\mathbf{f}(\mathbf{x}) = 0$. This equation is usually (but not always) the optimality condition of a minimization problem with objective $U(\mathbf{x})$. Hence, $\mathbf{f} = -\partial U/\partial \mathbf{x}$ is the total force acting on the system. One case where the equations are not partial derivatives of an energy function is the mixed formulation of FEM. In this case, the equations form a saddle point problem. We will not go into details about mixed FEM here (see [FARE21] or Section 10.2 in [Wri08]), but we will use it in order to avoid locking. The gist of the mixed pressure-displacement formulation is to enforce a volumetric soft constraint. This is done by discretizing both the positions and the pressure field. The most common choices for the con-

DOI: 10.2312/egs.20211011

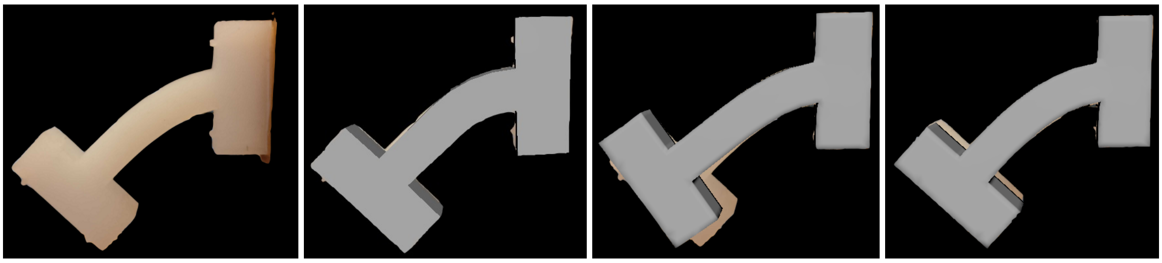

Figure 1: *Parameter estimation for the hammerbot soft robot. From left to right: the hammerbot hanging under gravity, an approximate mesh modeled in Maya, an estimated mesh simulated using standard FEM and finally one using mixed FEM.*

straint are $J - 1 = 0$ or $\log(J) = 0$, where J describes the change of a small volume element (the determinant of the deformation gradient) [BW97]. For the finite element discretization, we use linear shape functions for both the displacement and pressure fields. This is rather non-standard, but it is non-locking nevertheless. After discretization, these constraints can be encoded as a vector function $\phi(\mathbf{x}) = 0$. We wrote the code in C++ and exposed most of the features to Python using pybind11. Our solver of choice was Newton with line search and we implemented it using the Eigen library with MKL and Pardiso solver integration, as described in [FARE21]. In this paper the forward model is used to calculate the current state of the system $\mathbf{x}$, as part of solving an inverse problem.

In general, the inverse problems we are looking to solve try to minimize a loss function $O(\mathbf{x}(\alpha))$ with regard to the model parameters α. As we are usually interested in the gradient of the objective and the positions $\mathbf{x}$ are a function of α we need to apply the chain rule. As $\partial O/\partial \mathbf{x}$ is usually easy to evaluate, the hard part is in evaluating $\partial \mathbf{x}/\partial \alpha$. We exemplify using the following often-encountered least squares scenario:

$$\text{minimize}_\alpha \ \tfrac{1}{2}\|\mathbf{x}(\alpha) - \bar{\mathbf{x}}\|^2, \qquad (1)$$

where $\bar{\mathbf{x}}$ are some target positions. We will modify this problem template to estimate elastic parameters (Section 4) and control a cantilever beam (Section 5). The gradients we are looking for can be evaluated using sensitivity analysis [MMO16]. They can be obtained from solving the set of linear systems

$$\mathbf{H}\frac{\partial \mathbf{x}}{\partial \alpha} = \frac{\partial \mathbf{f}}{\partial \alpha},$$

where $\mathbf{H}$ is the Hessian of $U(\mathbf{x})$. For the mixed formulation, $\mathbf{H}$ can no longer have the same meaning, but it can be calculated as $\mathbf{H} = \frac{\partial^2 U}{\partial \mathbf{x}^2} + \mathbf{G}^T\mathbf{C}^{-1}\mathbf{G}$ through static condensation, where $\mathbf{G} = \nabla\phi(\mathbf{x})$ is the Jacobian matrix of the volumetric constraint function and $\mathbf{C}$ is the so-called *compliance matrix*. We will not go over the construction of the compliance matrix, but in the simplest case you can picture it as $\mathbf{C} = \lambda\mathbf{I}$. In the general case, the identity matrix is replaced by a weighting matrix that depends on the mesh topology, element volumes and discretization order. The only remaining thing to do is to evaluate the gradients of the total force w.r.t. the model parameters, $\partial\mathbf{f}/\partial\alpha$. This is quite easy to do as the dependence on parameters is usually linear. First, we need to be able to split the force into components that are independent functions of each α_i. Then it is simply of matter of dividing by α_i or evaluating the function without the α_i term. The same logic applies to mixed FEM, but this time we need to calculate the volumetric force separately as $\mathbf{G}^T\mathbf{p}$, where $\mathbf{p}$ is a vector of Lagrange multipliers associated with the pressure field.

4. Parameter estimation

The goal of parameter estimation is to estimate the values of some physical parameters so that the simulation gives results that are closest to reality. Considering that we already have a target pose for the body, we want to obtain the unknown parameters. The simplest task we could imagine was to take the output of a known simulation and try to reproduce the parameters by pretending we have forgotten them. This is done by solving the problem from Eq. (1) where the model parameters α are now the material parameters. We simulated a cantilever beam bending under gravity. The model is a box of dimensions $0.1 \times 0.1 \times 1$ meters consisting of 81 nodes and 192 tetrahedra. The Young's modulus is $E = 66$ KPa and the Poisson ratio $\nu = 0.45$. We stored the positions resulting from a quasi-static analysis of the beam and used them later as a target for the parameter estimation problem. We solved the inverse problem using the gradient-free Nelder-Mead algorithm provided by the SciPy library. We were able to reproduce the exact same Lamé parameters that were used in the initial simulation: $\mu = 22$ KPa and $\lambda = 11$ MPa. The initial guesses were $\mu_0 = 20$ KPa and $\lambda_0 = 0.2$ MPa. We managed to reproduce the same results using gradient based optimization for both standard and mixed FEM, but this time we employed a least squares solver fom SciPy.

For a more complex test scenario, we employed the "hammerbot" soft robot with a relatively fine mesh (9011 tetrahedra). We used a photo of the real robot to deform a surface mesh in Autodesk Maya which we then used as a target for our estimation process. This was done by mapping the original surface mesh to the volumetric mesh using barycentric coordinates and deforming it according to the result of the simulation. The mapped mesh and the target one were compared at each iteration of the minimization process resulting in the sought after elastic parameters. We solved the optimization problem using the Nelder-Mead algorithm, as we found the gradient based methods to be struggling. We obtained two sets of results depending on wheter we used standard or mixed FEM. For the former we obtained $\mu = 22.9$ KPa and $\lambda = 0.23$ MPa corresponding to $E = 66.7$ KPa and $\nu = 0.454$. But this results may be skewed due to the initial guess of $E = 66$ KPa and $\nu = 0.45$ and locking artefacts. In other words, the simulation is not able to reproduce the same amount of deformation as the target unless it lowers

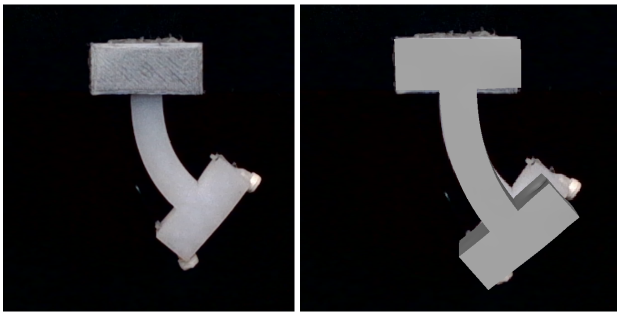

Figure 2: *Validation of our cable actuation model against reality.*

one of the parameters. The minimizer then chooses the closest minimum to the provided initial guess. See Figure 1 for an illustration of the standard FEM simulation using the estimated parameters - as you can see the difference is quite big.

Mixed FEM, on the other hand, is much closer to the target and reality. The parameters obtained were $\mu = 32.3$ KPa and $\lambda = 1.1 \times 10^{10}$ Pa, corresponding to $E = 96.8$ KPa and $\nu = 0.499999$ which are very close to what we expect from the material used (Ecoflex 00-50). This result confirms a trial and error approach we made that put the Young's modulus above 90 KPa, assuming the material was incompressible. But, again, the minimizer is sensitive to initial guesses for the solution. For example if starting from $\lambda_0 = 10^7$ Pa, then the estimation for the Poisson ratio becomes 0.445 and E is close to 100 KPa. Although the mesh shape looks good, this is clearly not correct, as the material is far from being incompressible. This also shows that the cantilever bending test is probably not sufficient. For less accuracy in the forward solver of the quasi-static simulator, the parameters obtained were $\mu = 17.4$ KPa and $\lambda = 5.4 \cdot 10^{10}$ Pa, corresponding to $E = 52.3$ KPa and $\nu = 0.4999999$. This shows that by not allowing the solver to converge, less deformation is achieved and the estimator tries to compensate by lowering the shear modulus. Hence, a non-locking accurate simulator is essential in the parameter estimation process.

In terms of limitations, the non-convexity of the loss function makes starting from different initial guesses quite important, but also transforms the process into more of a manual trial and error routine. This can be hard for a totally unknown material, but for Ecoflex silicone we know that E ranges between 60 and 270 KPa [GML*19, KN19]. Ideally, we would use measurements directly from reality to estimate our parameters, and have multiple scenarios and measurements in order to cope with errors and local minima, as it has been thouroughly done by [MBT*12, HBBC19]. For example, one could use markers on the robot or a point cloud capture with a depth camera; or use multiple camera angles and an extra scenario like extending or twisting the soft robot.

5. Cable actuation model

Cable actuation for soft robots is usually done by running thin wires on the surface of the silicone or through specially created tunnels with one end attached and the other controlled by a motor. We model these cables by dividing them into stiff springs as in [KN19]. The spring nodes are either embedded in the FEM body or are free to move outside. The embedding is done by parametrizing the nodes inside the closest tetrahedron using barycentric coordinates. This means they do not add degrees of freedom to the system. The free nodes are added on top of the FEM nodes and are simulated like any mass-spring system. The free nodes are given masses comparable to the embedded nodes, are affected by gravity and are slightly damped. Although one can use the embedding for outside nodes too, the free nodes have the best appearance when the cable is relaxed (see the accompanying video).

Our cable actuation model relies on an often used approximation for muscles and tendons [TTL12, BBPC19]. Instead of modeling the cable sliding, we consider it attached at every node and the actuation is achieved by reducing the rest length. If all of the spring nodes are embedded, we can write them down as $\mathbf{y} = \mathbf{Wx}$ (i.e. the interpolated points), where $\mathbf{W}$ is the barycentric mapping matrix. The building blocks of this matrix are $\mathbf{W}_{ji} = w_{ji}\mathbf{I}_3$, where w_{ji} is the barycentric coordinate corresponding to the jth spring node and ith global FEM node. The springs are unilateral so that they do not exert any forces when the cable is relaxed. The measure of deformation is given by $\Gamma = L(\mathbf{y}) - \alpha l$, where L is the current length of the spring, l is the rest length and α is a control parameter. The tension along a spring is given by $\tau = -\partial\Psi/\partial\Gamma$, where Ψ is an elastic potential. We use the regularized model from [BKC17] that ensures smoothness for the solver. The force acting on the jth spring node is given by $\mathbf{f}_j^{node} = \mathbf{f}_{j-1}^{spring} - \mathbf{f}_j^{spring}$. The force along the jth spring is obtained by multiplying the normalized spring direction by the corresponding tension τ_j. Finally, for the embedded spring nodes we can apply the inverse barycentric mapping to compute the force contributions to the respective FEM nodes: $\mathbf{f}^{cable} = \mathbf{W}^T\mathbf{f}^{node}$. We chose to integrate this model using the implicit Euler scheme due to its stability. But this lead us to evaluating Hessian matrices and solving linear systems. For this we had to use some finite differences approximations, which made the Newton solver slower.

We were able to simulate the cable actuation for the cantilever beam and a few soft robots, as the figures and the video demonstrate. For the hammerbot presented in the previous section, we validated the cable model against a real-life photo. The values used for the two side cables are $\alpha_1 = 0.82$ and $\alpha_2 = 1.2$ and they were measured from an actual setup using stepper motors. We used a spring constant of $k = 50$ KN/m in our simulation, as using a higher value puts strain on the solver. As you can see from Figure 2 our cable actuation model gets quite close to reality. Using a lower stiffness for the cable results in less deformation. Therefore, we concluded that the accuracy of the model depends a lot on getting the spring constants right (ideally, we would model them as inextensible). In addition, we found that the elastic parameters or type of finite element method had less of an impact on cable-driven deformation.

6. Cable actuation control

In order to test our cable model for controlling a soft robot, we devised two simple tests for the cantilever beam. We put a cable along one of its lower edges and actuated it at $\alpha = 0.5$. We then performed the simulation and obtained a pose bent towards the side with the cable. We then used this pose as a target in the inverse problem in Eq. (1) with α as the only optimization parameter. Again, by us-

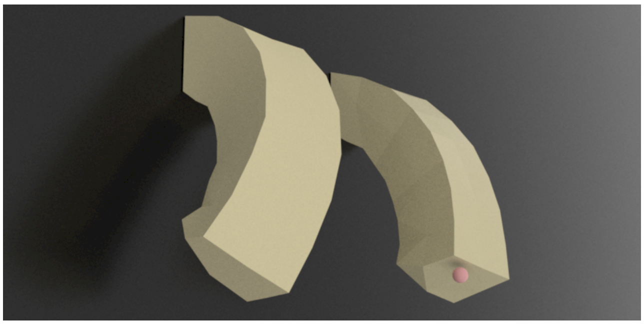

Figure 3: *Cable control experiments. Left: cantilever actuated with $\alpha = 0.5$, right: the same cantilever targeting the red sphere.*

ing the Nelder-Mead algorithm we were able to replicate the initial value of 0.5. We then performed a synthetic "end-effector" control test. We chose the center of the free end-face of the cantilever and took its position from the aforementioned simulation. We then perturbed it to a larger value on the x axis and ran the optimization again. As expected, such an inverse problem often has no solution, but the optimizer can find one that is closest to the point - see the results in Figure 3.

7. Conclusions and future work

In this paper, we have presented a software framework written in C++ and Python (available as open-source: https://github.com/MihaiF/SolidFEM) that can solve inverse problems for cable-driven silicone soft robots. We have demonstrated it on two examples: parameter estimation and cable control. For parameter estimation we were able to deduce parameters that were close to expected values using both gradient-based and gradient-free optimization. Of great importance was the mixed FEM solver that could prevent locking and reproduce the deformation of incompressible silicone better than standard FEM. Our cable model showed a close match to reality, although not perfect due to numerical issues. We then used this model successfully to target a pose and an end-effector position. We would like to use our mixed FEM based framework for more real life parameter estimation. This may also require doing heterogeneous estimation (different parameters for each element) or space-time optimization for estimating dynamic parameters like damping coefficients. As for the cables, we would like to model them as inextensible using hard constraints, similar to the incompressibility constraint in mixed FEM. We hope that using a constraint solver would reduce the current numerical ill-conditioning and allow for more accurate modeling of cable actuation. Just as for parameter estimation, we would like to do more real world validation for cable control and more complex scenarios.

8. Acknowledgements

We would like to thank Max Kragballe Nielsen for providing us with soft robots and materials from his master's thesis. The author has received funding from the European Union's Horizon 2020 research and innovation programme under Marie Sklodowska-Curie grant agreement No. 799001. This report only contains the author's views and the Research Executive Agency and the Commission are not responsible for any use that may be made of the information it contains.

References

[BBO*09] BICKEL B., BÄCHER M., OTADUY M. A., MATUSIK W., PFISTER H., GROSS M.: Capture and modeling of non-linear heterogeneous soft tissue. *ACM Transactions on Graphics 28*, 3 (2009). 1

[BBPC19] BERN J. M., BANZET P., PORANNE R., COROS S.: Trajectory optimization for cable-driven soft robot locomotion. In *Robotics: Science and Systems* (2019). 1, 3

[BKC17] BERN J. M., KUMAGAI G., COROS S.: Fabrication, modeling, and control of plush robots. In *RSJ International Conference on Intelligent Robots and Systems* (2017). 3

[BW97] BONET J., WOOD R. D.: *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press, 1997. 2

[FARE21] FRÂNCU M., ASGEIRSSON A., RØNNOW M., ERLEBEN K.: Locking-proof Tetrahedra. *ACM Transactions on Graphics 40*, 2 (2021). 1, 2

[GML*19] GASOTO R., MACKLIN M., LIU X., SUN Y., ERLEBEN K., ONAL C., FU J.: A validated physical model for real-time simulation of soft robotic snakes. *2019 International Conference on Robotics and Automation (ICRA)* (2019). 1, 3

[HBBC19] HAHN D., BANZET P., BERN J. M., COROS S.: Real2sim: Visco-elastic parameter estimation from dynamic motion. *ACM Transactions on Graphics (TOG) 38*, 6 (2019). 1, 3

[KN19] KRAGBALLE-NIELSEN M.: *Automation of Control and Design Optimization for Soft Robots manufactured with Strong Feasibility Guarantees*. Master's thesis, University of Copenhagen, 2019. 1, 3

[LVC*15] LARGILLIERE F., VERONA V., COEVOET E., SANZ-LOPEZ M., DEQUIDT J., DURIEZ C.: Real-time control of soft-robots using asynchronous finite element modeling. In *2015 IEEE International Conference on Robotics and Automation* (2015). 1

[MBT*12] MIGUEL E., BRADLEY D., THOMASZEWSKI B., BICKEL B., MATUSIK W., OTADUY M. A., MARSCHNER S.: Data-driven estimation of cloth simulation models. In *Computer Graphics Forum* (2012), vol. 31. 1, 3

[MEM*19] MACKLIN M., ERLEBEN K., MÜLLER M., CHENTANEZ N., JESCHKE S., MAKOVIYCHUK V.: Non-smooth newton methods for deformable multi-body dynamics. *ACM Transactions on Graphics 38*, 5 (2019). 1

[MMO16] MIGUEL E., MIRAUT D., OTADUY M. A.: Modeling and estimation of energy-based hyperelastic objects. In *Computer Graphics Forum* (2016), vol. 35. 1, 2

[ST91] SIMO J. C., TAYLOR R. L.: Quasi-incompressible finite elasticity in principal stretches. Continuum basis and numerical algorithms. *Comput. Methods in Appl. Mech. Eng. 85*, 3 (1991). 1

[STC*13] SKOURAS M., THOMASZEWSKI B., COROS S., BICKEL B., GROSS M.: Computational design of actuated deformable characters. *ACM Transactions on Graphics 32*, 4 (2013). 1

[TTL12] TAN J., TURK G., LIU C. K.: Soft body locomotion. *ACM Transactions on Graphics 31*, 4 (2012). 1, 3

[WOR11] WANG H., O'BRIEN J. F., RAMAMOORTHI R.: Data-driven elastic models for cloth: modeling and measurement. *ACM Transactions on Graphics 30*, 4 (2011). 1

[Wri08] WRIGGERS P.: *Nonlinear finite element methods*. Springer, 2008. 1

[WWY*15] WANG B., WU L., YIN K., ASCHER U. M., LIU L., HUANG H.: Deformation capture and modeling of soft objects. *ACM Trans. Graph. 34*, 4 (2015). 1