

Control of Tendon-Driven Soft Foam Robot Hands

Cornelia Schlagenhauf^{1*}, Dominik Bauer^{1*}, Kai-Hung Chang^{2*}, Jonathan P. King², Daniele Moro³
Stelian Coros⁴, and Nancy Pollard²

Abstract—This paper presents a series of control strategies for soft compliant manipulators. We provide a novel approach to control multi-fingered tendon-driven foam hands using a *CyberGlove* and a simple ridge regression model. The results achieved include complex posing, dexterous grasping and in-hand manipulations. To enable efficient data sampling and a more intuitive design process of foam robots, we implement and evaluate a finite element based simulation. The accuracy of this model is evaluated using a *Vicon* motion capture system. We then use this simulation to solve inverse kinematics and compare the performance of supervised learning, reinforcement learning, nearest neighbor and linear ridge regression methods in terms of their accuracy and sample efficiency.

Index Terms—Grasping and Manipulation, Teleoperation, Novel mechanism design

I. INTRODUCTION

Over the years, soft robot hands and grippers have been designed from a wide range of materials [1] and actuators [2][3]. The advantages of such soft designs over rigid robot hands range from robust grasping [4][5] to being lightweight and inherently safe [6]. However, in order for soft robots to become more widespread, they need to be adaptable and able to achieve complex dexterous manipulation. Enabling such capabilities requires considerable innovation in hardware, modeling and control [7].

One very promising approach is to design low-cost and easy to fabricate robots which can be customized for specific tasks. For example, King and his colleagues [8] present a novel class of fully-compliant, tendon-actuated soft manipulators made from off-the-shelf components. The primary structure of this type of robot consists of a flexible foam core. The softness and flexibility of the foam robot have been shown to be of great advantage for secure grasping and robust in-hand manipulation [8]. However, working with

*Equal contributions

This work was supported in part by the National Robotics Initiative grant NSF IIS-1637853 and in part by a NASA Space Technology Research Fellowship (award 80NSSC17K0140). Any opinions, figures, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Robotics Initiative or NASA.

¹ Cornelia Schlagenhauf and Dominik Bauer are with the Department of Mechanical Engineering, Karlsruher Institut für Technologie, 76131 Karlsruhe, Germany. {dominik.bauer, cornelia.schlagenhauf}@student.kit.edu

² Jonathan King, Kai-Hung Chang, and Nancy Pollard are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213. nsp@cs.cmu.edu, {jking2, kaihungc}@andrew.cmu.edu

³ Daniele Moro is with the Department of Computer Science, Boise State University, Boise, ID 83725. moro.daniele@gmail.com

⁴ Stelian Coros is with the Department of Computer Science, ETH Zürich, 8092 Zürich, Switzerland. scoros@inf.ethz.ch

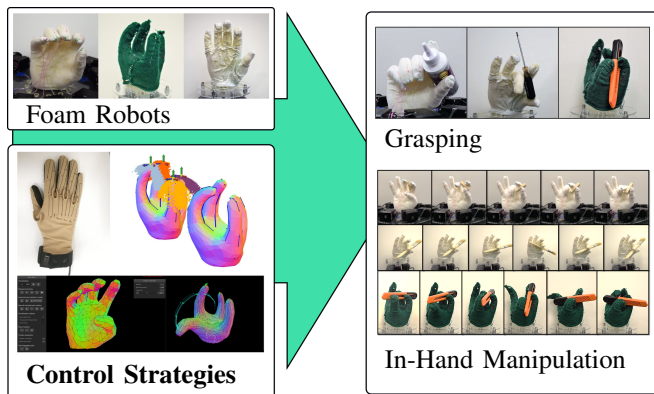


Fig. 1. Intuitive control strategies enable foam robot hands to perform stable grasps and precision in-hand manipulation.

such a hand requires the application of new modeling and control techniques.

The goal of this work is to provide users with tools and strategies to create and control dexterous foam robot hands. The primary contribution of this paper is an evaluation and comparison of different control strategies for solving the inverse kinematics problem of foam robots. To aid in both design and control, we make use of a simulation framework tuned and evaluated for our foam hands.

II. RELATED WORK

The continuously deformable nature of soft robots makes controlling them a hard problem. Their ability to accomplish motions such as buckling, contraction, extension or bending, results in soft robots having virtually infinite degrees of freedom. Additionally, Thuruthel et al. [9] mention non-linear material effects such as compliance and hysteresis, as well as the wide range of design and actuation techniques that account for the non-trivial nature of this problem. Previous works have particularly studied the problem of inverse kinematics (IK) which is concerned with finding a mapping between actuation and desired configuration [10][11][12]. Existing control approaches can be classified into three main categories: model-based controllers, model-free controllers and a combination of both.

Model-based controllers rely on the establishment of a kinematic model from which the actuation can be directly inferred for the desired configuration. Saunders et al. [13] model caterpillar-like soft robots as a series of extensible linkages. For tentacle-shaped soft robots, Marchese et al. [14][15] and Chen et al. [16] use piecewise constant-curvature models to model the robot. For soft robots with

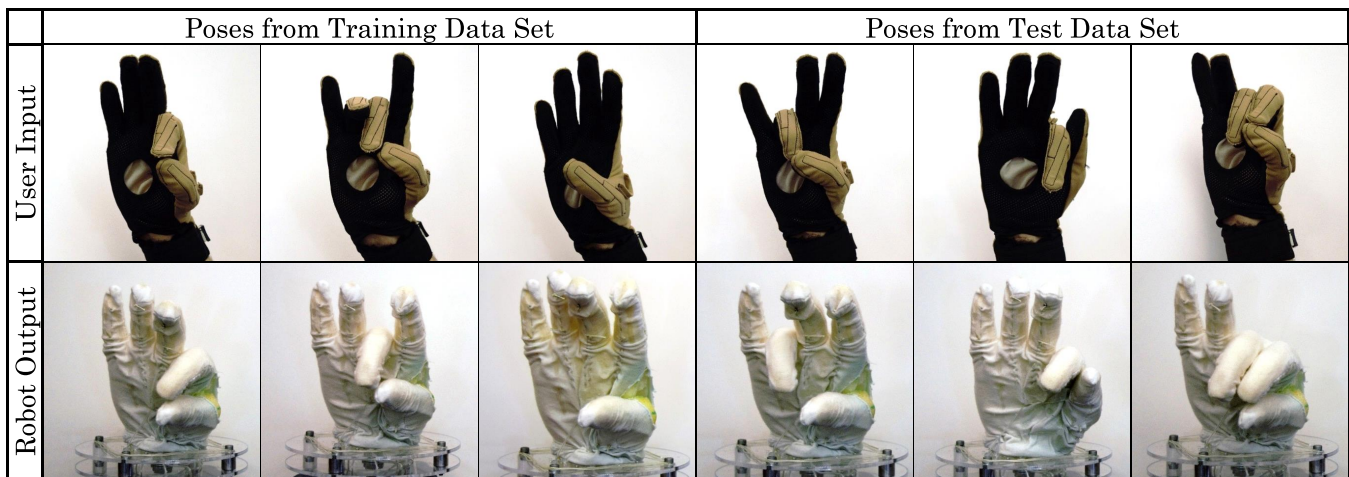


Fig. 2. Top) Input poses from user wearing a *CyberGlove*. Bottom) Output poses from the learned mapping. Left) Poses taken from the training set. Right) Poses not included in the training set.

arbitrary shapes, Duriez [17] presents a real-time solution using a finite element method (FEM).

Model-free approaches offer a wide variety of data driven techniques to control soft robots. Neural networks have successfully been used learn inverse kinematics on a cable-driven soft tentacle manipulator with 2 degrees of freedom [18]. Rolf et al. have proposed an exploration algorithm for creating task space samples for IK learning [10].

Our approach is also model-free in the sense that we collect data – either from the real robot or in simulation – and use that data to build a map from desired pose to actuations. The manipulators we want to control differ significantly from previous designs for the following reasons:

- We allow users to place tendons anywhere on the foam to achieve task specific actuation. This results in complex actuation patterns and infinite amounts of possible routings that need to be controlled.
- There are no restrictions in terms of possible foam geometries. Grippers, anthropomorphic hands and even multi-fingered non-anthropomorphic hands are possible.
- Coupled deformations occur throughout the hand due to the extremely soft compliant nature of the foam core.

The contribution of our paper is to compare and contrast different possible solutions for mapping from pose to actuation, with emphasis on intuitive direct control, data efficiency, ease of learning models for new geometries or tendon layouts, and ability to control the hand to perform practical manipulation tasks.

III. FABRICATION OF SOFT FOAM HANDS

In order to be truly soft, our foam robots are built using only soft materials such as foam casts, textile gloves, fibrous tendons and flexible PTFE tubes for cable routing. We fit a textile skin on a foam core hand model and sew tendons through the skin. Servo motors drive the fingers of our hands by contracting or slackening tendons. All rigid mechanical components (motors and pulleys to drive the tendons) are housed away from the hand.

To obtain the foam core for a robot hand we present a fabrication process which either starts from a physical or a digital hand model:

- Physical hand model: We fabricate a silicone mold of a physical hand, from which we then cast a foam model using a two-part urethane foam compound ¹.
- Digital hand model: We create a digital 3d mesh of the desired hand shape using CAD or 3d sculpting software. From a 3d hand mesh we can either 3d print a positive hand model and create a silicone mold, or directly 3d print a negative model of the hand for use as a mold.

After the foam core is cast, we fit and laminate a textile glove onto it and can sew tendons in arbitrary patterns through the glove. The hand is glued to an acrylic base and we route the tendons through PTFE tubing to minimize friction and connect them to servo motors. Through our fabrication process we can create anthropomorphic hands as well as non-humanlike shapes, as for example planar grippers or three- or four-fingered hands. Textile skins for non-anthropomorphic shapes can be custom knit by automatic processes [19]. In [8] we give a detailed description of the manufacturing process and showcase the grippers and hands we built using this technique.

IV. TELEMANIPULATION: LEARNING ON THE ROBOT

In the most basic scenario, we have only the robot itself, with a given arrangement of tendons and motors, and a device with which the user wishes to control the robot. With this equipment, we must learn a mapping from user gestures or poses to motor actuations that deform the robot in the desired manner. We explore a straightforward mapping, where the user wears a *CyberGlove* and controls an anthropomorphic hand that is similar to their own. However, we wish to allow for flexibility when the geometries of the human and robot hands may differ significantly.

We take inspiration from research on puppeteering in computer graphics. For example, Seol and colleagues [20]

¹Smooth-On FlexFoam-iT! Series

present a method that allows the user to specify how they wish to move in order to create certain character motions. As an example, they might choose to swing their arm to move an elephant’s trunk. In the case of [20], an approach based on feature mapping is used to convert from user motion to character control parameters. In our case, we use linear regression to create a map from *CyberGlove* sensors to tendon activations for the hand.

Our approach works as follows. First, a sampling of tendon activations is used to execute various poses of the foam hand. An operator imitates those poses while wearing the calibrated *CyberGlove*, and the corresponding joint angles of the human hand pose are recorded. Both random tendon activations and tendon activations corresponding to finger-thumb oppositions and grasping postures were used to build a training set of 120 hand poses. For generalization purposes, each pose was recorded 5 times.

A regression model, which takes the 22 joint angles from the *CyberGlove* as input and predicts the corresponding tendon activation levels was trained. The model uses *Kernel Ridge Regression* with a linear kernel. The average RMS error achieved by the model between the measured and the predicted normalized tendon activations was 0.0026, with normalized tendon actuation ranging from zero to one. A normalized actuation value of zero refers to a loose tendon and a value of one specifies the maximum actuation, which was set individually for each tendon by qualitative observation.

Even with a small training set (120 recordings), the learned model was able to reproduce a variety of poses with high accuracy based on qualitative evaluation. Figure 2 shows a comparison of poses supplied by an operator and the poses realized by the foam hand. Both poses taken from the training set and new poses are included.

We note that in order to achieve such results, the careful selection of training poses is crucial. While our first approach was to sample poses with only one finger contracted at a time, we gained the insight that especially for coupled motions such a model does not generalize well. In terms of posing this means that fingertips of opposing fingers do not touch or align for example. Adding specific poses that include coupled tendon contractions, as shown in the trained poses of Figure 2 can significantly increase generalization. Therefore we suggest to use poses that are related to the task that needs to be executed. Using just three additional task-specific poses (shown in Figure 2 on the left) the learned mapping was also precise enough to perform telemanipulation tasks, including grasping objects and in-hand manipulations. Demonstrations are shown in Figure 3 (Left) and in the video attachment for this paper. Since during the described sampling process we rely exclusively on the person wearing the *CyberGlove* to match the robot poses with their hand, this approach may be influenced by subjective impression of how well poses match. A strong advantage of this technique however is the possibility to easily create mappings between the human hand and different hand morphologies. Given that the human operator can create



Fig. 3. Left) Telemanipulation sequence of a small cuboid executed by a human operator using the *CyberGlove* and the trained regression model. Right) Open-loop controlled manipulation sequence created by interpolating between tendon actuation keyframes.

a corresponding hand pose for each robot hand pose, this technique can even be applied to non-anthropomorphic foam hands.

V. DIRECT CONTROL: LEARNING IN SIMULATION

Learning on the robot is straightforward and was successful. However, the amount of test data that can be collected is limited and similarity in poses is only qualitative and depends on the patience, care, and point of view of the user. If we can learn a mapping from poses to actuations in simulation, the comparison between test poses and learned poses can be much more exact, and we can explore how additional data may improve the results. However, for this approach to be effective, the simulation must be a good match to the actual robot hand.

In this section, we first describe our simulation environment, give results from a validation test of this environment, and then compare several strategies for learning a mapping from pose to actuation levels.

A. Simulation Framework

1) *Finite Element Simulation*: We are following the approach of Bern et al. [21] who use a finite element model to capture the deformation behavior of soft plush toys. We transfer their representation of soft plushies consisting of a series of contractile elements (modeled as stiff unilateral springs) to our foam hands. Each foam hand is modeled as a discrete set of nodes denoted as \mathbf{X} for the undeformed robot and \mathbf{x} as the statically stable deformed pose. The total deformation energy of the system is defined as:

$$E = E_{foam} + E_{contractile} + E_{pins}$$

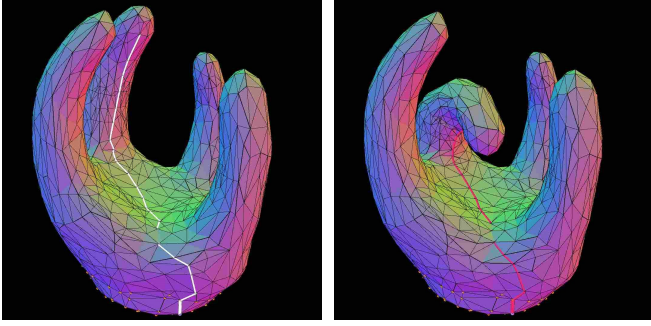


Fig. 4. Following the approach of Bern et al. [22], we use finite element simulation for our soft robots, where tendon contractions result in contraction of the mesh along the tendon routing. Equilibrium poses before and after contraction are shown. We build on this previous research by identifying and evaluating simulation parameters to match foam hands manufactured using our process and by providing an intuitive user interface for interactive tendon design.

where E_{foam} is the energy due to deformations of the simulation mesh, $E_{contractile}$ is the strain energy stored by the contractile elements, and E_{pins} models the behavior of stiff springs that connect a small number of simulation nodes to world anchors in order to eliminate rigid body modes. The elastic behavior of the foam is modeled using linear finite elements with a compressible Neo-Hookean material model. Tendons are modeled as *contractile* elements that abstract the contraction of a tendon as changing the rest length of the underlying unilateral spring model. A contractile element is defined as a piecewise linear curve with two endpoints (x_s, x_t) and n intermediate vertices (x_1, \dots, x_n) . We assume that all points of contractile elements are bound to nodes of the simulation mesh. The initial rest length l_0 of a tendon is defined by the sum of distances between the vertices as

$$l_0 = \|x_s - x_1\| + \sum_{i=1}^{n-1} \|x_i - x_{i+1}\| + \|x_n - x_t\|$$

The contraction level α_c of each tendon describes the contracted length as

$$l_c = l_0 \cdot (1 - \alpha_c)$$

In the following, the word *routing* refers to the choice of endpoints and intermediate vertices of each tendon. The resulting deformation for a tendon routing with the contractions α_c is calculated by minimizing the total energy of the system using Newton’s method. A detailed description of this step and the calculation of deformation energy can be found in [21]. Figure 4 shows a four-fingered hand mesh in equilibrium before and after contracting a tendon.

We tuned simulation parameters through observed visual feedback to match qualitative behavior of the foam hands in simulation with the behavior of physical hands. The obtained values of the material parameters are shown in Table I and are validated with motion capture data in Section V-B.

We created a user interface to enable users to interactively and intuitively create new tendon patterns and evaluate the posing and motion capabilities of their design quickly. Our interface contains the following features:



Fig. 5. Four-fingered hand with Vicon markers.

- users can pick nodes and drag the mesh into desired configurations to create target poses and explore the general workspace of a hand geometry
- quickly pick a set of nodes along which tendons are routed
- add, alter and remove tendons from the design
- create hand motions by contracting tendons, record and play back the created motion sequences

B. Validation of simulation model

In this section we quantify the accuracy of our simulation framework by comparing fingertip trajectories of a simulated and a physical foam hand robot. The deformations of the foam are tracked using a Vicon Motion Capture system. Predicted deformations from simulation, are then compared with the corresponding actual deformations on the physical robot.

1) *Four-Fingered Foam Hand*: The physical foam hand robot used in this experiment is a non-anthropomorphic hand with four fingers and 10 tendons. Each finger is controlled by a pair of antagonistically routed tendons acting as flexor and extensor. In order to introduce abduction and adduction motions, we placed two additional tendons on the left and right side of one finger. As reference in simulation we use the same geometry, with a slightly coarser mesh (980 nodes) compared to the mesh that was used to print the mold of the physical foam hand. This mesh size was chosen to allow interactive simulation in our user interface.

2) *Motion Capture Experiment*: We record fingertip trajectories of our four-fingered foam hand using a Vicon motion capture system with 12 cameras. To get a robust estimate of the position and to prevent occlusions we place four markers around each fingertip as shown in Figure 5. For registration purposes we additionally place markers on the platform on which the hand is mounted and alongside each finger. After the experiment the recorded markers are registered on the 3d mesh. This is done using a standard

ICP algorithm that minimizes the distance of points from the mesh and the markers with respect to each other. Since the markers themselves are not exactly aligned with the surface of the foam we found it difficult to infer the exact position of the fingertip using only the position of the markers. Therefore we define each fingertip position \vec{p}_j with $j = \{1, \dots, 4\}$ as the mean of the corresponding markers k with $k = \{1, \dots, 4\}$, with a distal offset of 5mm normal to the plane spanned by the four markers: $\vec{p}_j = \frac{1}{4} \cdot \sum_{k=1}^4 \vec{p}_{jk} + 0.005 \cdot \vec{n}_j$. The RMS error describing the euclidean distance between the aligned point clouds of our ICP registration was 4.05mm .

In terms of material parameters for the FEM simulation (mass density ρ , Young's modulus E , Poisson's ratio ν) we used the values found in Table I. The goal of this experiment

TABLE I
MATERIAL PROPERTIES USED IN FEM SIMULATION

ρ [kg/m^3]	E [Pa]	ν
160	$3\text{e}6$	0.25

is to give an estimate of how well our simulation can match reality. We ran 5 trials in which each tendon is repeatedly contracted from 0% to 50% of its rest length in steps of 10%. The tendon rest length is distinct for each tendon and is computed in simulation.

3) *Results and Discussion:* A motion sequence of a contracting extensor tendon moving the simulated hand through the waypoints at 10%, 30%, and 50% contraction is shown in Figure 7. The motion is displayed from three different camera views. The fingertip trajectories recorded by the Vicon system are marked as dotted lines, with larger green points at the fingertip positions recorded at the tendon contraction waypoints (10%, 30%, 50%). In each frame, fingertips of the simulated hand are marked as red circles. From this sequence and the video attachment it can be observed that trajectories of the simulated and physical fingertips largely coincide. The resulting error between fingertip positions captured with the Vicon system and from simulation is depicted in Figure 6. The mean position error for all fingers including all activation levels is 0.626cm . For each individual finger median error and the quartile deviations converge to similar values at all contraction levels. This suggests that even large deformations do not significantly decrease the accuracy of our model. Larger initial position errors as observed in finger 1 (Figure 6) can be explained by external disturbances. In general we identify the following sources of position errors:

- small deviations between tendon routings in simulation and reality
- tendon slack
- registration errors in motion capture system
- friction between tendon and glove
- slight relative movements of foam core and glove during actuation

Most of these errors can be mitigated during fabrication of the hand, for example by using teflon-coated tendons or different gluing techniques. The results of this evaluation

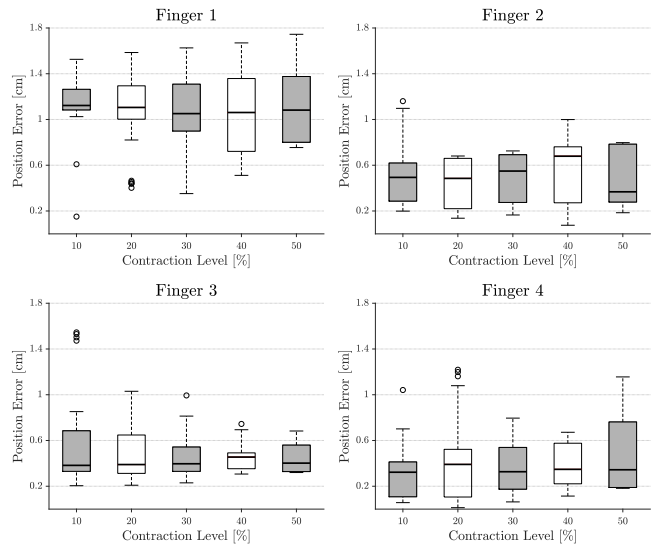


Fig. 6. Position Error between simulated and captured fingertip positions of finger 1-4 at respective contraction level.

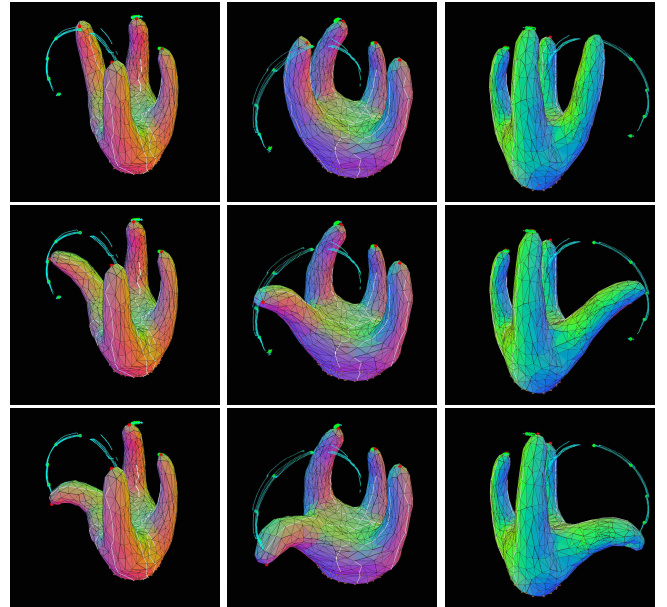


Fig. 7. Simulated hand and motion capture trajectories for an extensor tendon moving from 10%(top) to 50%(bottom) tendon contraction, viewed from three different camera perspectives. Fingertip positions recorded by the Vicon system at each contraction level are averaged over all five trials and marked as large green dots, fingertip positions of the simulation model are marked as red dots.

suggest that our model predicts foam deformations sufficiently well to proceed with simulation based learning of mapping from desired pose to tendon actuation.

C. Learning in Simulation

Collecting data in simulation is faster and easier than collecting data on the physical robot. Making use of the accessibility of large amount of data from the simulation, we are able to apply learning-based methods with complex models. These methods take the concatenated fingertip positions as the input and output the tendon activation that is expected to pose the hand correspondingly. Four different methods are

applied and compared: 1)Nearest neighbor, 2)Linear ridge regression, 3)Neural network using supervised learning, and 4)Deep reinforcement learning.

1) *Learning-based methods*: The **Nearest neighbor** method serves as a straw-man approach. It takes the tendon activation of the pose that is nearest to the desired pose in the pose space based on Euclidean distance and simply returns that tendon activation as the result.

Linear ridge regression is supplied in part because it was used for learning for telemanipulation as described in Section IV. It is perhaps the second simplest sensible approach beyond Nearest Neighbor. We use a linear model with an additional L2 ridge regularizer.

A **neural network using supervised learning** adds additional degrees of freedom and nonlinearity. We include this model to determine whether the additional complexity can improve fit to the data. Our Neural Network model is constructed with four intermediate layers, each of which has 30 units and ReLU non-linear activations. The activation of the output layer is $\tanh(x)$ to match a linear-normalized range $[-1,1]$ of the output activation. The training process runs 300 epochs with a batch size of 20, a learning rate of 0.001 and Adam optimizer with the typical parameter values ($\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$).

Deep reinforcement learning can be considered as an alternative approach to learning a nonlinear model. Based on the success of learning IK on both rigid robot arms and hands [23], deep reinforcement learning is expected to transfer to soft robots. In particular, we apply the deep deterministic policy gradient [24] algorithm combined with hindsight experience replay [25]. The shaped reward function is the negative of the average distance error over all fingers. Hindsight experience replay can be considered as a way to include additional targeted results, as "failed" solutions are reinterpreted during learning as successful solutions to a different problem.

2) *Experiments*: A simulation model of the physical anthropomorphic foam hand shown in Figure 8 (Left) is obtained by using *Autodesk ReMake*[26] to generate a surface mesh from approximately 50 images of the hand taken with a smartphone. We then run *TetGen*[27] to build a 3D finite element mesh of the hand, shown in Figure 8 (Right).

To compare the sample efficiency of all four methods, we use the same datasets for both training and testing. The training dataset collects 100,000 poses while the testing dataset contains 100 poses, all of which are pre-generated in the simulation by drawing randomly from possible tendon activations.

3) *Results and Discussion*: We plot the performance (average distance error in centimeters) with respect to the amount of data used in training. The comparison plot is shown in Figure 9. When training with less than 100,000 samples, the training data is extracted in sequence from the large 100,000 dataset. The plot shows that linear ridge regression is outperformed by all other approaches especially for large datasets, implying that additional model complexity is useful for this test dataset. Overall, and to our surprise, the



Fig. 8. Anthropomorphic foam hand prototype in its rest pose. Left) physical foam hand prototype, Middle) Scan of the prototype, Right) Finite element mesh used in simulation.

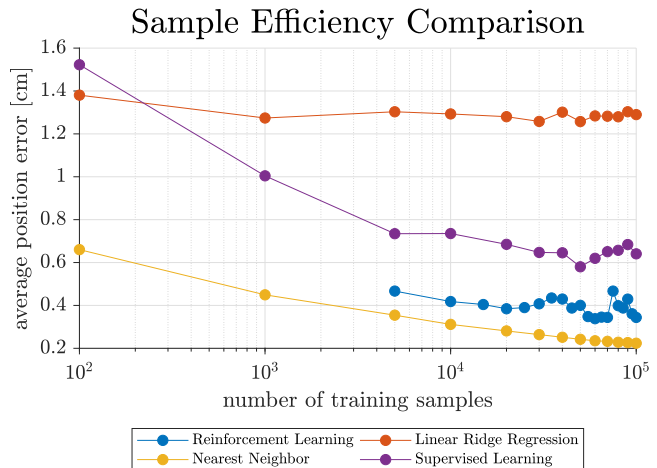


Fig. 9. Performance of four different methods to learn IK in simulation.

nearest neighbor method shows the best performance and the best sample efficiency. However, results from nearest neighbor approaches are typically not smooth for datasets that do not comprehensively cover the space of tendon activations. Lastly, deep reinforcement learning outperforms supervised learning. The main difference between these two approaches is the existence of a loss function. While in supervised learning, the network is trained to fit the tendon activations from the training data, the objective in reinforcement learning is to maximize rewards based on the calculated average distance error, which may be physically more reasonable. Another possible cause is that the reinforcement learning algorithm, DDPG, has an actor-critic mechanism which might help the learning.

VI. DISCUSSION

Linear regression was sensible for learning on the real robot due to the extremely small amount of data available. It is a simple and straightforward model, and performed well in practice, both for poses that were not part of the training set and when teleoperating real-world grasping and manipulation tasks.

When we moved to the simulation environment, we found that more complex models – and even nearest neighbor – appeared to outperform linear regression. We experimented extensively in simulation with direct control using supervised learning, and we found that it behaved smoothly and intuitively in most cases. We believe, however, that linear

regression can still work well in the simulation environment, but that test poses should be selected carefully to cover typical use cases (as was done in the real-robot learning).

Simulation brings advantages of being able to iterate and test in a rapid manner. Based on our validation experiments, we are confident in being able to test tendon routings and iterate on different hand geometries and tendon designs in rapid fashion. Our attached video shows examples of utilizing our simulation system in this manner.

VII. CONCLUSIONS

In this work we presented various control approaches for foam robots and compared them in terms of their performance in simulation. To control the physical foam robot we developed a new approach using a *CyberGlove* and a regression model. To support the design of soft foam robots we created and evaluated a new simulation framework which significantly increases the intuitive design of tendon routings on foam hands. The contributions we made will enable others to quickly design task-specific foam hands that can perform dexterous manipulation tasks.

In our future work we want to focus on optimizing and automating the design of foam hands. Also, we would like to move towards robust autonomous control strategies because we believe that these hands have demonstrated the potential to perform complex dexterous manipulation robustly and with ease.

ACKNOWLEDGEMENTS

The authors thank Justin C. Macey for his assistance running motion capture experiments, as well as Professor James McCann for knitting our gloves.

REFERENCES

- [1] M. Cianchetti, M. Calisti, L. Margheri, M. Kuba, and C. Laschi, "Bioinspired locomotion and grasping in water: the soft eight-arm octopus robot," *Bioinspiration & biomimetics*, vol. 10, no. 3, p. 035003, 2015.
- [2] M. T. Tolley, R. F. Shepherd, B. Mosadegh, K. C. Galloway, M. Wehner, M. Karpelson, R. J. Wood, and G. M. Whitesides, "A resilient, untethered soft robot," *Soft Robotics*, vol. 1, no. 3, pp. 213–223, 2014.
- [3] S. Seok, C. D. Onal, K.-J. Cho, R. J. Wood, D. Rus, and S. Kim, "Meshworm: a peristaltic soft robot with antagonistic nickel titanium coil actuators," *IEEE/ASME Transactions on mechatronics*, vol. 18, no. 5, pp. 1485–1497, 2013.
- [4] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 161–185, 2016.
- [5] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Robust proprioceptive grasping with a soft robot hand," *Autonomous Robots*, Apr 2018. [Online]. Available: <https://doi.org/10.1007/s10514-018-9754-1>
- [6] S. Sanan, J. B. Moidel, and C. G. Atkeson, "Robots with inflatable links," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 4331–4336.
- [7] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, vol. 35, no. 7, pp. 840–869, 2016. [Online]. Available: <https://doi.org/10.1177/0278364915587925>
- [8] J. P. King, D. Bauer, C. Schlagenhaut, K.-H. Chang, D. Moro, N. Pollard, and S. Coros, "Design, fabrication, and evaluation of tendon-driven multi-fingered foam hands," in *2018 IEEE RAS International Conference on Humanoid Robots*. IEEE, 2018.
- [9] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft Robotics*, vol. 5, no. 2, pp. 149–163, 2018, pMID: 29297756. [Online]. Available: <https://doi.org/10.1089/soro.2017.0007>
- [10] M. Rolf and J. J. Steil, "Efficient exploratory learning of inverse kinematics on a bionic elephant trunk," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1147–1160, June 2014.
- [11] T. G. Thuruthel, E. Falotico, M. Cianchetti, and C. Laschi, "Learning global inverse kinematics solutions for a continuum robot," in *RO-MANSY 21 - Robot Design, Dynamics and Control*, V. Parenti-Castelli and W. Schiehlen, Eds. Cham: Springer International Publishing, 2016, pp. 47–54.
- [12] H. Jiang, Z. Wang, X. Liu, X. Chen, Y. Jin, X. You, and X. Chen, "A two-level approach for solving the inverse kinematics of an extensible soft arm considering viscoelastic behavior," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE International Conference, 2017.
- [13] F. Saunders, B. A. Trimmer, and J. Rife, "Modeling locomotion of a soft-bodied arthropod using inverse dynamics," *Bioinspiration & Biomimetics*, vol. 6, no. 1, p. 016001, 2011. [Online]. Available: <http://stacks.iop.org/1748-3190/6/i=1/a=016001>
- [14] A. D. Marchese, K. Komorowski, C. D. Onal, and D. Rus, "Design and control of a soft and continuously deformable 2d robotic manipulation system," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 2189–2196.
- [15] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, vol. 35, no. 7, pp. 840–869, 2016. [Online]. Available: <https://doi.org/10.1177/0278364915587925>
- [16] F. J. Chen, S. Dirven, W. L. Xu, and X. N. Li, "Soft actuator mimicking human esophageal peristalsis for a swallowing robot," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 4, pp. 1300–1308, Aug 2014.
- [17] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 3982–3987.
- [18] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi, "Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 823–834, Aug 2015.
- [19] J. McCann, L. Albaugh, V. Narayanan, A. Grow, W. Matusik, J. Mankoff, and J. Hodgins, "A compiler for 3d machine knitting," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 49, 2016.
- [20] Y. Seol, C. O'Sullivan, and J. Lee, "Creature features: Online motion puppetry for non-human characters," in *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA '13. New York, NY, USA: ACM, 2013, pp. 213–221. [Online]. Available: <http://doi.acm.org/10.1145/2485895.2485903>
- [21] J. M. Bern, K.-H. Chang, and S. Coros, "Interactive design of animated plushies," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 80, 2017.
- [22] J. M. Bern, G. Kumagai, and S. Coros, "Fabrication, modeling, and control of plush robots," in *Proceedings of the International Conference on Intelligent Robots and Systems*, 2017.
- [23] M. Plappert, M. Andrychowicz, A. Ray, B. McGrew, B. Baker, G. Powell, J. Schneider, J. Tobin, M. Chociej, P. Welinder *et al.*, "Multi-goal reinforcement learning: Challenging robotics environments and request for research," *arXiv preprint arXiv:1802.09464*, 2018.
- [24] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.
- [25] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. P. Abbeel, and W. Zaremba, "Hindsight experience replay," in *Advances in Neural Information Processing Systems*, 2017, pp. 5048–5058.
- [26] "Autodesk remake," <https://www.autodesk.com/products/remake/overview>, accessed: 2017-07-19.
- [27] "Tetgen," <http://wias-berlin.de/software/tetgen/>, accessed: 2017-07-19.