Tendon Optimization for Robotic Hands

Hyojae Park[1], Cornelia Bauer[1], Dominik Bauer[1], Nancy S. Pollard[1]

***Abstract*— Designing tendon-driven anthropomorphic robotic hands is a tedious process, where designers are often faced with the choice of how to configure tendons for the best motion within an actuator budget. Furthermore, the design process can be significantly longer for soft, dexterous hands, which often cannot be tested in simulations. To design robots that model human motion, which may be more trusted by humans, we first present a method of converting motion capture data into approximated tendon actuation. This information is then used to determine the best set of tendons and joint moment arms for the given motion capture, providing researchers with pivotal design information without physical testing. We evaluate our algorithm and find that the average marker position error converges to 2.8 mm as the number of tendons increases, outperforming a baseline greedy algorithm by $2\times$.**

I. INTRODUCTION

To achieve a **human-trusted design**, we want a robot that can replicate human motion, which is seen as trustworthy and intuitive. This is difficult to do with tendon-based robotic hands since tendons are limited by bulky, expensive motors. In comparison to a human hand, whose 27 joints and over 30 muscles give them an immense range of motion, robotic hands are severely constrained [1]. Thus, this work outlines a method for designing tendon-based hands that can nevertheless closely approximate human hand motion.

Specifically, due to the limited number of tendons allowed in robotic hands, researchers have to spend countless hours optimizing their hand designs to maximize performance [2]. To automate this time-exhausting process, this work outlines an algorithm that uses evolutionary learning methods to select the best combination of tendons for performing a desired set of motions. This contribution provides a key step in enabling the rapid generation and use of robotic hands by eliminating significant overheads in tendon optimization.

II. RELATED WORK

Tendons have been largely ignored in robotic design optimization. However, studies show that optimized tendon-based robotic hands can surpass human capabilities [3].

One optimization method is computational hand design, which uses simulations to identify the best hand under constraints, while other methods use heuristics optimized through extensive physical testing [4], [2], [5], [6].

Soft dexterous hands have recently shown promise in the field [2]. Such hands have the ability to deform with infinite degrees of freedom, giving them greater potential in complex tasks. However, the lack of structure makes designing and testing soft hands difficult [7]. For example, the placement of actuators in soft hands can be tedious to optimize.

[1]Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA.

III. METHODS

A. Motion Capture

12 motion capture demonstrations were collected from one human subject using a Vicon Motion Capture System. We incorporated 59 markers on their hand, and these markers were grouped into 23 segments.

B. Robotic Hand Model

We assume a soft robotic hand model similar to that shown in Bauer et al. [2]. We assume a standard hand kinematic model with 18 joints (3 thumb, 3 index, 3 middle, 4 ring, 4 pinky, 1 wrist), actuated by abduction/adduction and flexion tendons wired through tendon channels. As a soft model, it returns to a rest pose when no tendons are actuated.

Note that due to this model, when two tendons actuate one joint in the same direction, the greater actuation will have the sole effect. Contrarily, for two tendons tugging in opposite directions, the joint is actuated by their absolute difference.

C. Tendon Optimization Process

Let the number of motion capture frames be n. Let there be j joints and c tendons to choose t tendons from.

We begin with x, the position of each joint. Ideally, t tendons should be chosen and actuated in such a way that the resulting marker position $\tilde{x}$ is as close to x as possible.

To accomplish this, an IK solver is used on x to determine θ, a $n \times j$ matrix containing the joint angles for each frame.

Next, we create the tendon-joint matrix S_{all} with shape $j \times c$ to encode information about which joints each tendon actuates as well as their relative moment arms. The ikth entry in S_{all} is defined as the relative moment arm of the ith joint with respect to the joints actuated by the kth tendon.

S_{all} can be estimated from joint angles in the motion capture by normalizing each angle along their corresponding joints then calculating the average angle of each joint over the motion capture frames.

After choosing t tendons, we only need information about the chosen tendons. Hence, we only need the columns of S_{all} that correspond to those tendons. Let this matrix be S.

With S, we calculate tendon actuation α by minimizing the MSE of the kth tendon at frame l against θ_{lk}.

We can now calculate $\tilde{x}$ (where $\tilde{x} \approx x$) by using S and α. First, we run algorithm 1 to calculate $\tilde{\theta}$ (where $\tilde{\theta} \approx \theta$). The algorithm operates by examining each joint and determining which tendons rotate that joint the most in the positive and negative directions. These two rotations are summed to get $\tilde{\theta}$. Lastly, $\tilde{x}$ is calculated by using FK on $\tilde{\theta}$. The error between $\tilde{x}$ and x can then be retrieved using MSE.

Algorithm 1: calculate_$\tilde{\theta}$

Input: S, α

for *frame_i in* $\{0,...,n\text{-}1\}$ **do**

  for *joint_i in* $\{0,...,j\text{-}1\}$ **do**

    // matrix β has shape $j \times n \times t$

    $\beta[\text{frame\_i}, \text{joint\_i}, :] \leftarrow S[\text{joint\_i}, :] \cdot \alpha[\text{frame\_i}, :]$

$\beta^{+} \leftarrow \text{clip}(\beta, \min = 0)$ // lower bound by 0

$\beta^{-} \leftarrow \text{clip}(\beta, \max = 0)$ // upper bound by 0

$\tilde{\theta} \leftarrow \max(\beta^{+}, \dim = 2) + \max(\beta^{-}, \dim = 2)$

return $\tilde{\theta}$

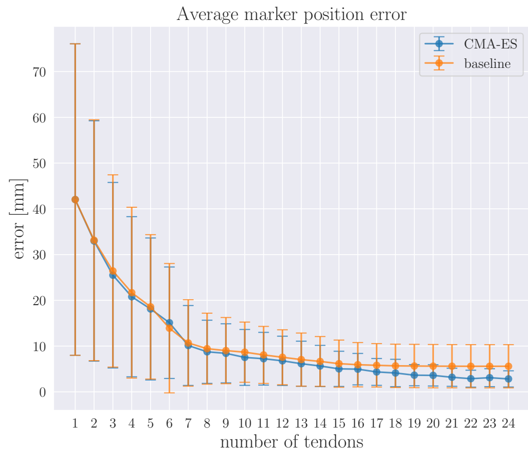


Fig. 1: The average marker position error across motion capture frames. the CMA-ES algorithm converges to 2.8 mm while the baseline converges to 5.6 mm.

We can now select "optimal" t tendons by relying on an optimization algorithm. We utilized CMA-ES, a gradient-free evolutionary algorithm that performs extremely well in optimizing rough, continuous domains [8]. CMA-ES was used to determine the best set of input tendons.

Lastly, S was optimized using CMA-ES. This optimization was performed once the tendons were chosen. While this final step is irrelevant if S is already constant, this step is useful for determining the hypothetical "optimal" moment arms, which can be important in design optimization.

IV. EVALUATIONS

Our algorithm was evaluated from 1 to 24 tendons using the Bridges-2 Supercomputer at the Pittsburgh Supercomputing Center [9]. The evaluation was performed using 12 motion capture movements including grasping small objects.

A greedy algorithm was also evaluated as a baseline. This algorithm looped through all possible tendons and iteratively added the tendon that best decreased the error. S was optimized using CMA-ES at each iteration.

The MSE error was calculated at each frame. Figure 1 highlights the average marker position error for each of the generated hand models. The figure indicates that using CMA-ES for selecting tendons almost consistently outperforms the baseline, especially as the number of tendons increases.

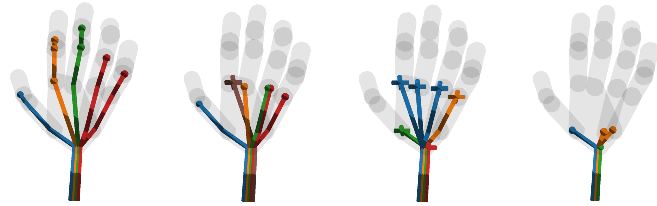
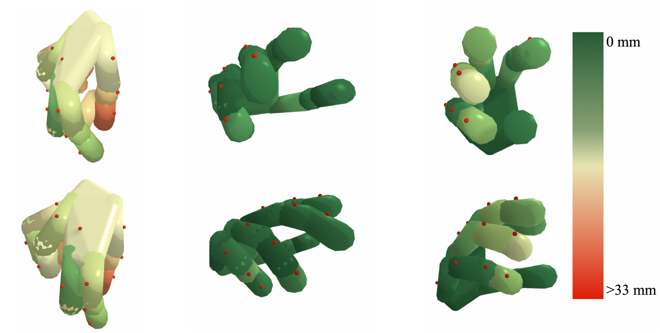


Fig. 2: Tendons selected by the CMA-ES-based algorithm for the 16-tendon hand model (split into 4 hands for visibility). Tendons with sphere ends are flexion tendons and tendons with cross ends are abduction/adduction tendons.



(a) Frame with greatest error

(b) Frame with least error

(c) Frame with median error

Fig. 3: Select frames from the 16-tendon hand model.

Figure 2 depicts the actual tendons selected by our algorithm for the 16-tendon hand model. The algorithm chose 5 abduction/adduction tendons and 11 flexion tendons. Not all joints were connected to a tendon: the ring and pinky DIP joints were left unconnected, indicating that a finer control of lower joints was more important than permitting a wider range of movement for the given motion capture data.

Figure 3 depicts select frames of the 16-model hand approximating the given motion capture. Image 3c illustrates the frame with median error across frames (4.69 mm), and although some error is apparent, the motion still closely resembles the true motion capture. Image 3a shows the worst frame with an average error of 15.03 mm, and image 3b, shows the best frame with an average error of 2.47 mm.

V. CONCLUSION

We presented a method of translating motion capture data into moment arms and tendon actuation. We also highlighted a CMA-ES-based algorithm for selecting optimal tendons and moment arms, which can save substantial overhead in tendon-based robotic hand design optimization. We find that our algorithm translated motion capture into hand movement with relatively low error compared to the baseline.

ACKNOWLEDGMENT

This research was supported by Carnegie Mellon's Summer Undergraduate Research Fellowship awarded to Hyojae Park and NSF Convergence Accelerator award ITE-2344109.

References

[1] Y. Li, J. L. Fu, and N. S. Pollard, "Data-driven grasp synthesis using shape matching and task-based pruning," *IEEE Transactions on visualization and computer graphics*, vol. 13, no. 4, pp. 732–747, 2007.

[2] D. Bauer, C. Bauer, A. Lakshmipathy, R. Shu, and N. S. Pollard, "Towards very low-cost iterative prototyping for fully printable dexterous soft robotic hands," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*, pp. 490–497, IEEE, 2022.

[3] J. M. Inouye and F. J. Valero-Cuevas, "Anthropomorphic tendon-driven robotic hands can exceed human grasping capabilities following optimization," *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 694–705, 2014.

[4] Z. Zhang, Y. Zheng, Z. Hu, L. Liu, X. Zhao, X. Li, and J. Pan, "A computational framework for robot hand design via reinforcement learning," in *2021 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 7216–7222, IEEE, 2021.

[5] S. Puhlmann, J. Harris, and O. Brock, "Rbo hand 3: A platform for soft dexterous manipulation," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3434–3449, 2022.

[6] W. G. Bircher, A. S. Morgan, and A. M. Dollar, "Complex manipulation with a simple robotic hand through contact breaking and caging," *Science Robotics*, vol. 6, no. 54, p. eabd2666, 2021.

[7] C. B. Teeple, R. C. S. Louis, M. A. Graule, and R. J. Wood, "The role of digit arrangement in soft robotic in-hand manipulation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7201–7208, IEEE, 2021.

[8] N. Hansen, "The cma evolution strategy: a comparing review," *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pp. 75–102, 2006.

[9] S. T. Brown, P. Buitrago, E. Hanna, S. Sanielevici, R. Scibek, and N. A. Nystrom, "Bridges-2: A platform for rapidly-evolving and data intensive research," in *Practice and experience in advanced research computing*, pp. 1–4, 2021.