# Model-Free Control of Movement in a Tendon-Driven Limb via a Modified Genetic Algorithm

Ali Marjaninejad, *Member, IEEE,* Rohit Annigeri, and Francisco J. Valero-Cuevas, *Senior Member, IEEE*

*Abstract*— **Tendon-driven systems have many benefits over other actuation strategies such as torque-driven systems; however, their over-determined nature and posture-dependent actuation presents strong constraints on their control. Also, parameters or even exact structure of the model in these systems, especially in the biological ones, are normally not clear to the controller. Here, we propose a modified Genetic Algorithm that provides the tendon excursion values for the limb to follow a desired trajectory. Our results show that the proposed algorithm was able to accurately follow the desired trajectory without the model of the system being exposed to it. We believe that this method can enable biologically inspired tendon-driven mechanisms with variable mechanical structures to autonomously control their movements.**

## I. INTRODUCTION

Tendon-driven systems have many advantages over systems with other actuation strategies. Namely, tendon-driven are less noisy, more clean and easier to maintain (since they don't need lubrication), shock absorbent, and less bulky (because they allow remote actuation) [1]. In addition, tendon driven-systems have more flexibility toward providing superior performance in achieving different goals [2]–[5]. For example, these systems are able to provide different end-point forces and velocities using different tendon routings [6]. Moreover, tendon-driven systems are also attractive to study since studying them helps us better understand mechanics of biological systems and their motor control strategies [7], [8].

Due to their nature, tendon-driven systems are harder to control compared to other actuation strategies such as torque-driven systems [6]. Also, the control of the tendon-driven systems is studied very little compared to the control of the other actuation strategies [6], [9]. However, due to its unique properties discussed earlier, there is an increasing interest in their control strategies. In recent years, many scientists tried to address modelling and controlling tendon-driven systems in different ways [1], [3], [4], [10]–[12]. These control strategies vary from simple PID controllers to non-linear and adaptive control strategies to deal with complexities in control of these systems [1], [11]–[15]. However, in many applications, the system is either too complex to precisely model or the system parameters cannot be non-invasively extracted (e.g. person specific moment arm values in human limbs) [10].

When developing motor abilities, biological systems do not have equations for the system that they are controlling. They simply try different strategies and gradually choose the best ones. This is mainly done by trying different activation combinations and evaluating its outputs using different feedback signals (e.g., visual feedback). We believe this is the same strategy that needs to be used to control tendon-driven systems so that the actions of the controller mimic biological systems and it will be able to act autonomously independent of their model. This strategy will also reveal more details on how biological systems develop their motor actions.

Genetic algorithm is a very powerful optimization tool used to solve a vast variety of the problems including control of complex systems [16]–[18]. In this paper, we are presenting a novel controller based on a modified version of GA to control a two Degrees of Freedom (DOF) limb with variable moment arm values using three tendons. We control the limb without the need for any information on model structure or parameter values. We do so by only providing methodically selected inputs and observing their corresponding output. The goal of this paper is to match the joint angles for each DOF with their desired values to follow a predefined trajectory.

The rest of this paper is organized as follows: controller design and simulation model are discussed in section II. Results are provided in section III; and discussions over findings and potential improvements are provided in section IV.

## II. METHODS

We have used a Python (an open source programming language) based simulation for the limb to evaluate the performance of the controller. The controller, however, doesn't have access to the model of the simulator and can only observe the error in each joint.

### A. Simulation Model

We used tendon-driven limb physics to model a two DOF limb controlled with three tendons as described below [6]:

$$\begin{bmatrix} ds_1 \\ ds_2 \\ ds_3 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \end{bmatrix} \begin{bmatrix} dq_1 \\ dq_2 \\ dq_3 \end{bmatrix} \tag{1}$$

where $ds_i$ is the excursion value for the $i^{th}$ muscle, $dq_j$ is the angular displacement for the $j^{th}$ joint, and $R_{ij}$ is the moment

A. Marjaninejad is with the Department of Biomedical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: marjanin@usc.edu).

Rohit Annigeri is with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: annigeri@usc.edu).

F. J. Valero-Cuevas is with the Division of Bio-kinesiology and the Department of Biomedical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: valero@usc.edu).

arm for the $j^{th}$ muscle in the $i^{th}$ joint. The base moment arm values for our model are represented in the moment arm matrix R which is represented in Eq. 2:

$$R = \begin{bmatrix} 2 & 0 & -3 \\ 0 & 1.5 & -4 \end{bmatrix} \quad (2)$$

To make the control of the system more challenging as well as making the model more realistic, we used variable moment arm values for $R_{11}$ and $R_{22}$ by adding sinusoidally changing values to their base values. The final moment arm values for $R_{11}$ and $R_{22}$ as a function of position are shown in Fig. 1. The schematic representation and the 3D simulation environment of this model are also shown in Fig. 2. On Fig. 2 (a), the $i^{th}$ DOF is shown by $D_i$ and the $j^{th}$ tendon is shown by $m_j$.

### B. Genetic Algorithm Controller

We first break the entire movement into discrete postures and then use GA to find the best tendon excursion values in each of these postures to fulfill the task (getting each joints' angle to the predefined desired values). We defined the fitness function as the sum of the square error between the limb angles and the desired angles. We made sure to have the configurations of the algorithm in a way that solutions with better fitness values have a higher chance of reproduction which gives the algorithm a flavor of reinforcement learning.

Moreover, GA is very sensitive to its initial population. A good selection for initial population can reduce the chance of getting stuck in a local minimum as well as the time duration required to find an optimal solution. Since we know that muscle excursion values in a continuous movement are also continuous, we transferred the optimal solution found for each position during the movement to the initial population of the next position. We did so in order to reduce the exploration needed to find the optimal solution for the new position. Since we have to run GA for every each posture in the movement, this adjustment greatly reduces the total time needed to find optimal solutions while lowering the chances of getting stuck in local minima.

The maximum acceptable error value (the threshold for the fitness function) for the GA was set to 0.5 and the population size was 32. All other details on the GA used in this paper are available at https://github.com/marjanin/GA_T.

Since the model has two DOF, in order to control the position of the limb, it is enough to find the excursion values for two tendons and make sure the third one is not slack. Therefore, we found excursion values for two tendons using the proposed GA and plotted (on the results section) the

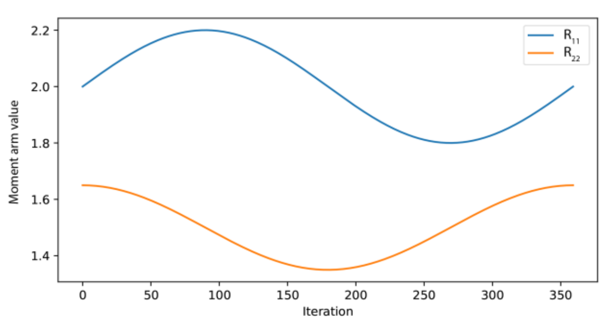


Fig. 1. Moment arm values for $R_{11}$ and $R_{22}$ as a function of position.

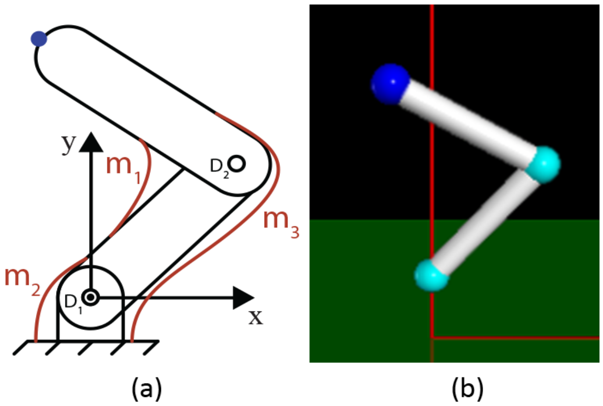


Fig. 2. The schematic (a) and the 3D representation (b) of the simulation model.

values for the third using the model equations. In non-simulation systems, we can just pull the third tendon until it is fighting other motors and is not slack. i.e., this method works without the need of any information on the parameters or the structure of the controlled system.

### C. Task and Fitness Descriptions

The task defined to test our controller was to follow a predefined trajectory. The trajectory consisted of 360 equidistant points selected from a circle. In this paper, we have provided results for a circular trajectory. We also define fitness as the sum of the squared error in each joint (compared to the desired angles).

## III. Results

In this section, we have provided the results for the position tracking task as well as the other performance metrics for the GA. Joint angles, tendon excursion values, and the limb end-point trajectories during the tracking task are represented on Fig. 3 (a), Fig. 3 (b), and Fig. 3 (c), respectively.

Fig. 4 (a) shows the best and the mean fitness values for the GA algorithm for a sample position as a function of the generation number. Fig. 4 (b) represents the time duration for each step to find the optimal solution. Finally, Fig. 4 (c) is the histogram showing the distribution of the data in Fig. 4 (b).

For the sample circular trajectory task discussed here, our modification to GA in transferring the final solution of each step to the initial population for the next step reduced the total run time of the algorithm from 3484.30 seconds (without the modification) to 1832.74 seconds (47.4% faster). This means that on average, the optimization takes 5.09 seconds to run for each simulation step.

## IV. Discussion

Fig. 3 (c) shows that our system was able to find the solution within the allowed maximum error value (see methods) in 100% of cases. This is very encouraging, especially considering that the controller was not exposed to any information on the structure of the model or its parameters.

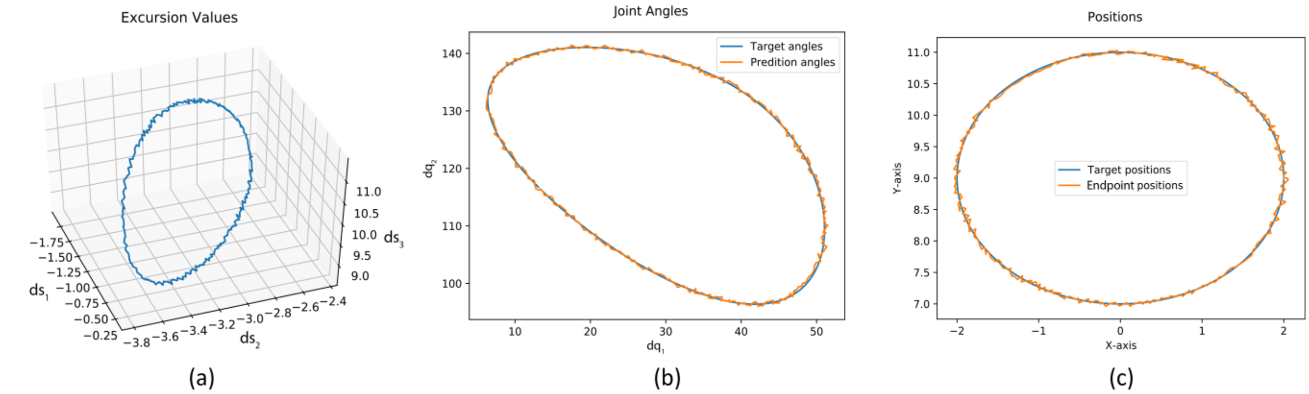


Fig. 3. Tendon excursion values (a), joint angle vs. the desired joint angle trajectories (b), and the limb end-point vs. the desired end-point trajectories (c) during the tracking task.

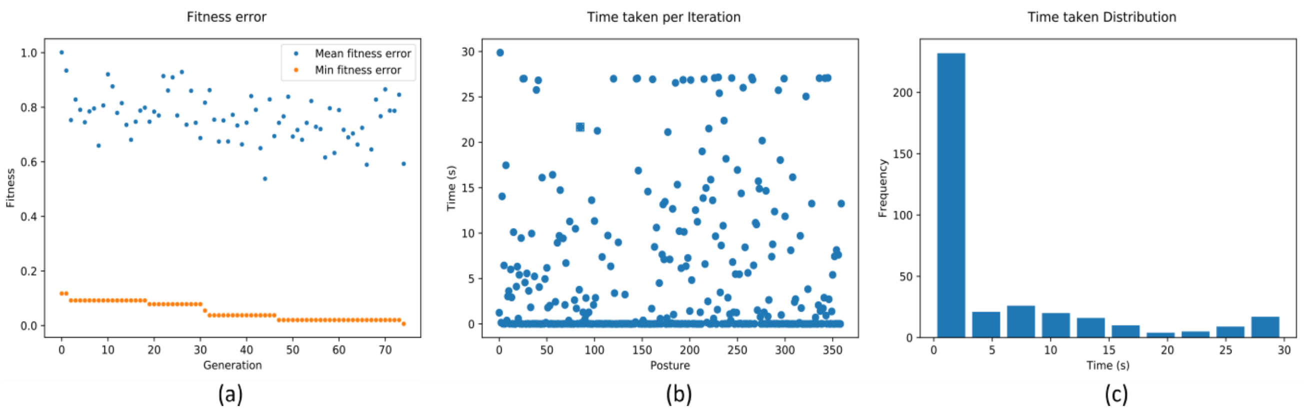


Fig. 4. Performance plots for the GA algorithm. Fitness as a function of generation (a). Elapsed time to find the solution for each posture (b) and the histogram of these time values (c).

Moreover, as can be seen on Fig. 3, all trajectories are limit cycles (closed trajectories that repeat themselves). This, as discussed earlier, shows the continuous transition of variables in our system which is expected in real physical systems. This continuity justifies our modification to the GA to use the final solution from each step as an initial start point for the next step, which considerably reduced the run time of the optimization (see results). In addition, Fig. 4 shows that although our modification was helpful in reducing the run time in most of the steps, the algorithm still needed to explore a wider space for some of the steps. This exploration depends on the maximum allowable error value (threshold for the fitness function) since the maximum allowable error defines if the optimal solution from each step is an acceptable solution for the next step or not (based on the fitness value).

Based on the results provided here, we believe that our method will enable tendon-driven systems to autonomously find corresponding tendon excursion values for every each posture or motion regardless of the differences in their structures or model parameter values.

In terms of the potential future work, using this system in company with biologically inspired limb models that represents elasticity of tendons is very interesting area to investigate since it will provide more information on the effects of physics of the biological systems in their control. Moreover, here we modified and improved the performance of the GA by only utilizing the continuity of the tendon excursion manifolds. We believe that adding up sensory manifold information, similar to what happens in the biological systems [19], [20], can increase the performance of the controller even further. In addition, it would be interesting to evaluate performances of other machine learning algorithms such as neural network controllers in combination with the biological features which are mentioned above and compare and contrast them with the method used here. Lastly, here we used a simplified kinematic model. A more comprehensive approach would be to make the model more realistic by adding inertias and angular acceleration values as well as adding dynamics such as joint impedance values.

## ACKNOWLEDGMENT

This project was supported by NIH Grants R01-052345 and R01-050520, and by the Department of Defense under award number MR150091 to Francisco Valero-Cuevas, and University of Southern California Graduate School's Provost Fellowship to Ali Marjaninejad.

## REFERENCES


[1] B. Okur, O. Aksoy, E. Zergeroglu, and E. Tatlicioglu, “Nonlinear robust control of tendon-driven robot manipulators,” *J. Intell. Robot. Syst.*, vol. 80, no. 1, p. 3, 2015.

[2] A. Marjaninejad and F. J. Valero-Cuevas, “Should anthropomorphic systems be ‘redundant’?,” in *Biomechanics of Anthropomorphic Systems, Springer Tracts in Advanced Robotics (STAR) series*, Springer, 2018.

[3] J. A. Rieffel, F. J. Valero-Cuevas, and H. Lipson, “Morphological communication: exploiting coupled dynamics in a complex mechanical structure to achieve locomotion,” *J. R. Soc. interface*, vol. 7, no. 45, pp. 613–621, 2010.

[4] F. J. J. Valero-Cuevas, V. V. V Anand, A. Saxena, and H. Lipson, “Beyond parameter estimation: extending biomechanical modeling by the explicit exploration of model topology,” *IEEE Trans Biomed Eng*, vol. 54, no. 11, pp. 1951–1964, Nov. 2007.

[5] J. M. Inouye and F. J. Valero-Cuevas, “Anthropomorphic tendon-driven robotic hands can exceed human grasping capabilities following optimization,” *Int. J. Rob. Res.*, vol. 33, no. 5, pp. 694–705, 2014.

[6] F. J. Valero-Cuevas, *Fundamentals of neuromechanics*, vol. 8. Springer, 2015.

[7] D. A. Hagen and F. J. Valero-Cuevas, “Similar movements are associated with drastically different muscle contraction velocities,” *J. Biomech.*, vol. 59, pp. 90–100, Jul. 2017.

[8] D. Urbina-Meléndez, K. Jalaleddini, M. A. Daley, and F. J. Valero-Cuevas, “A Physical Model Suggests That Hip-Localized Balance Sense in Birds Improves State Estimation in Perching: Implications for Bipedal Robots,” *Front. Robot. AI*, vol. 5, p. 38, 2018.

[9] A. Marjaninejad and J. M. Finley, “A model-based exploration of the role of pattern generating circuits during locomotor adaptation,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2016, vol. 2016–Octob.

[10] M. C. Yip and D. B. Camarillo, “Model-less feedback control of continuum manipulators in constrained environments,” *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 880–889, 2014.

[11] M. Li, R. Kang, S. Geng, and E. Guglielmino, “Design and control of a tendon-driven continuum robot,” *Trans. Inst. Meas. Control*, p. 142331216685607, 2017.

[12] M. Meng, X. Wang, and X. Wang, “Adaptive SNN Torque Control for Tendon-Driven Fingers,” in *Advanced Computational Methods in Life System Modeling and Simulation*, Springer, 2017, pp. 231–241.

[13] T. N. Do, T. Tjahjowidodo, M. W. S. Lau, and S. J. Phee, “Performance Control of Tendon-Driven Endoscopic Surgical Robots With Friction and Hysteresis,” *arXiv Prepr. arXiv1702.02063*, 2017.

[14] K. Kawaharazuka, M. Kawamura, S. Makino, Y. Asano, K. Okada, and M. Inaba, “Antagonist inhibition control in redundant tendon-driven structures based on human reciprocal innervation for wide range limb motion of musculoskeletal humanoids,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2119–2126, 2017.

[15] R. Ozawa, K. Hashirii, Y. Yoshimura, M. Moriya, and H. Kobayashi, “Design and control of a three-fingered tendon-driven robotic hand with active and passive tendons,” *Auton. Robots*, vol. 36, no. 1–2, pp. 67–78, 2014.

[16] A. Geramipour, M. Khazaei, A. Marjaninejad, and M. Khazaei, “Design of FPGA-based digital PID controller using Xilinx SysGen® for regulating blood glucose level of type-I diabetic patients,” *Int J Mechatron Electr Comput Technol*, vol. 3, no. 7, pp. 56–69, 2013.

[17] C. Caceres, J. M. Rosario, and D. Amaya, “Approach of Kinematic Control for a Nonholonomic Wheeled Robot using Artificial Neural Networks and Genetic Algorithms,” in *Bioinspired Intelligence (IWOBI), 2017 International Conference and Workshop on*, 2017, pp. 1–6.

[18] K. Jagatheesan, S. Samanta, A. Choudhury, N. Dey, B. Anand, and A. S. Ashour, “Quantum Inspired Evolutionary Algorithm in Load Frequency Control of Multi-area Interconnected Thermal Power System with Non-linearity,” in *Quantum Computing: An Environment for Intelligent Large Scale Real Application*, Springer, 2018, pp. 389–417.

[19] D. Philipona, J. K. O’Regan, and J.-P. Nadal, “Is there something out there? Inferring space from sensorimotor dependencies,” *Neural Comput.*, vol. 15, no. 9, pp. 2029–2049, 2003.

[20] J. A. Berry, R. Ritter, A. Nagamori, and F. J. Valero-Cuevas, “The neural control of movement must contend with trajectory-specific and nonlinearly distorted manifolds of afferent muscle spindle activity,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 1188–1194.