2024 10th IEEE RAS/EMBS International Conference for
Biomedical Robotics and Biomechatronics (BioRob)
1-4 September 2024. Heidelberg, Germany

Model-agnostic Bio-inspired Autonomous Lifelong-learning of Kinematic Control in Tendon-driven Quadruped Robots

Ali Marjaninejad[1,2] , *member, IEEE,*, Francisco J. Valero-Cuevas[1,2,*] , *senior member, IEEE,*

***Abstract*— Robots will achieve true autonomy in controlling their bodies and interacting with their environments only when they are able to learn within the constraints of their designs or environments with minimal to no prior information and can learn from and adapt to new tasks and conditions without experiencing catastrophic forgetting of prior ones. The autonomous learning of control is of an especial important in the case of tendon-driven robots, which offer a wide set of advantages such as flexibility in the placement of actuators, but also introduce further challenges to its modeling and control. Here, we have extended our bio-inspired learning algorithm, General-to-Particular, to a quadruped system and have shown its aptitude in learning to control the system within only a few seconds and performing functional movements with continuous learning. Most importantly, we have successfully utilized simple tactile sensory information to enable the system to distinguish between changes in the amount of load it carries and achieve lifelong-learning without catastrophic forgetting (continually learn without overwriting the previous skills).**

I. INTRODUCTION

Vertebrates can develop motor skills from limited exposure to a task, learn from and adapt to changes when facing a new experience, generalize basic principles across different tasks, and learn how to perform new tasks without overwriting the old ones. This lifelong-learning (L2M) without catastrophic forgetting ability has equipped them with their enviable learning speed, efficiency, and adaptability even without a comprehensive prior about or model of the task, their body, or the environment. The level to which an agent with artificial intelligence can mimic these abilities will be a decisive factor in determining if it can learn, perform, and adapt in real-world applications with limited observability, incomplete or even inaccurate priors, and uncertainties in interacting with the environment or other agents.

,Vertebrate do not have explicit access to their biomechanical models or its parameters (such as maximal muscle forces or the moment-arm values). Although not very often in current commercial robots, it can be the case in some of the robotic applications as well. It is an important challenge when the robot is expected to adapt to the changes to which it is introduced or if it wants to learn in a design-agnostic way rather quickly and using limited experience [1], [2], [3], [4], [5], which opens up new avenues toward brain-body coevolution in robots. By enabling robots to learn by exploring and to create self-awareness of their body dynamics, they will be able to adapt to changes in their body dynamics or even completely new body structures.

Tendon-driven systems are great study cases for design-agnostic autonomous learning of controls since these systems are notorious for their inherent challenges in their controls. Although they can be used to provide great ranges of forces and velocities [6], [7], [8], from the controls perspective, their simultaneously over- and under-determined nature greatly constraints the feasible kinematic state space. Moreover, this makes the control problem even more challenging, as there is no one-to-one relationship between DoFs and actuators, as is the case in joint-driven systems [6], [7].

Even when a successful control strategy is found for a particular task, it is important to be able to distinguish different tasks and utilize the knowledge gathered during learning one to have a head-start in learning the others. Tactile sensory information is an important type of feedback signal that can play an important role in discriminating different tasks or different phases in a single task. It has been shown that tactile sensory information can provide important information such as success in a reaching task [9] or hitting the floor signal (heel strike) during locomotion [10]. Moreover, if placed on the extremities, it can be a good indicator of the weight that the limb is carrying at each moment and therefore it can better guide the control strategy toward adjustments on the muscle forces.

Here we have implemented and expanded our G2P algorithm to a quadruped robot with two DoFs controlled by three tendons on each limb and have studied the effects of different architectures of Artificial Neural Networks (ANNs), incorporating position error feedback, and endpoint tactile sensory information to controls and adaptation (both forward—generalizing to new tasks—, and backward—no catastrophic forgetting of old tasks) across 4 different test cases: performing limb movements in the air, on the floor, on the floor while carrying a light load, and on the floor carrying a heavy load. Moreover, we have tested two different curricula with the forward and reverse order of the test cares mentioned above (light to heavy and vice versa). Our results show that more specialized ANN architecture (one ANN for each limb) outperforms a single large ANN that maps all the input kinematics to the estimated muscle activations. Moreover, our system can achieve reasonable performance (RMSE of joint angles in a 10-second point-to-point and cyclical movement tasks) within only minutes of exploration

[1] Biomedical Engineering Department, University of Southern California
[2] Electrical and Computer Engineering Department, University of Southern California
[2]Division of Biokinesiology and Physical Therapy, University of Southern California
*Corresponding Author's email: valero@usc.edu
This project was supported by award W911NF1820264 by DARPA-L2M program, NIH Grants R01-052345 and R01-050520, and award MR150091 by DoD

and in the absence of any prior model of the system or the environment. Moreover, this proposed model model-agnostic approach will enable robots to learn without being limited to a specific physical design which will also open the doors to the brain-body coevolution idea (lifelong adaptation of both controls and physical structure) in robots.

II. Methods

Here we discuss the design of the quadruped system, tasks it is tested on the learning pipeline, and the variables studied: the effects of position error feedback, tactile sensing, and the architecture of the ANNs used.

A. Quadruped design and the simulation environment

We have designed a tendon-driven quadruped with three tendons and two DoF on each limb (each limb follows a similar design to[5], [11]). Each tendon is actuated using a MuJoCo Muscle actuator with peak active force equal to 120N. Also, this quadruped design is an expansion of the OpenAI's HalfCheetah (which has only 2 limbs) design and inherits most of the other physical parameters from it. The resulting design file (in .xml format; more details and xml design files are available online as a part of the supplementary files) was run by the MuJoCo physics simulator [12]. We have used 2.5 ms time step to reduce the effects of potential integration errors during the simulation.

B. Tasks and Test cases

1) Tasks: We have tested all our cases with two tasks: continuous cyclical movements and point-to-point movements (similar to [5], [11]

a) cyclical movements: During this task, the proximal and distal joints follow sinusoidal trajectories with $\pi/2$ phase difference. In addition, diagonal limbs will be synchronized together (similar to trotting). This task consists of 10 seconds with a frequency of 1.5 Hz (15 cycles during the entire task).

b) point-to-point movements: In point-to-point task, the position of each angle is determined randomly from the range of motion of each limb (Uniformly distributed). In this case, the four limbs are following the same desired positions. This tasks consists of 10 seconds with 10 position commands for each joint (1 second in each position).

2) Test cases: We have tested both tasks in four different cases, as described here.

a) Suspended in Air: In this case, the quadruped is suspended in air and therefore the limbs will not be in contact with the floor. In this case, the only thing the system needs to learn is how to deal with and control dynamics of its own limbs (see Fig. 1a).

b) Walking on the floor: In this case, the system is put on the floor, the limbs can interact with the floor, and therefore can move. In this case, system is going to experience contact dynamics and need to deal with its weight. The density (volumetric mass density) of all parts of the body is set to $1000Kg/m^3$ (except the density of head, neck, and tail, which are mainly added for visual purposes, and are set to $10Kg/m^3$; see Fig. 1b)

c) Walking on the floor with a load (light): This case is similar to walking on the floor with a slight difference that it now has to carry a load (as seen on Fig. 1c). This will have a direct effect on the weight and therefore a robust and successful control strategy would require the system to adopt the changes caused by it. The density of the load in this case is also set to $1000Kg/m^3$.

d) Walking on the floor with a load (heavy): In this case, the load weight is twice that of the previous case; in other words, this case is identical to the previous in all aspects except the load density which is set to $2000Kg/m^3$ (see Fig. 1d).

C. Learning pipeline (G2P algorithm)

To be able to find a mapping between the desired kinematics and the muscle activations that would lead to them on our tendon-driven system, we use the G2P algorithm [5] which is consisted of two main parts:

1) Motor babbling: During this phase, muscle are randomly activated (uniformly distributed from the 0 to 100% activation range) and the resulting sensory input information (Tactile sensory information which is endpoint force values and Kinematics which consist of joint angles, angular velocities, and angular accelerations) are collected. Sensory information and activations are then used to train an ANN as input and desired outputs, respectively. The resulting ANN will then be used to predict muscle activations required to perform the desired tasks during the refinement phases. For all cases in this study, we have performed the babbling phase for 60 seconds (supplementary video, Part I).

2) Refinements: During this phase, the kinematics of the desired task (here, either cyclical or point-to-point) are sent to the ANN to estimate the required muscle activations, and then to perform the move using those activations. Also, for the tactile sensory, we always feed the tactile sensory of the previous time step (except the first simulation step where we feed 0 on the tactile sensory input). The resulting task-specific sensory information and muscle activations are concatenated with the data available so far and used to re-tune the ANN. Please note that the motor babbling provides sparse sampling within a vast volume of the sensory information while refinements enables sampling more specific to the sensory space of a desired task (supplementary video, part II).

a.	Perform motor babbling
b.	Train an ANN with the resulting data
c.	Perform a particular task
d.	Refine the ANN model with all the data collected so far and go to c.

TABLE I
SIMPLIFIED FLOWCHART OF THE G2P ALGORITHM.

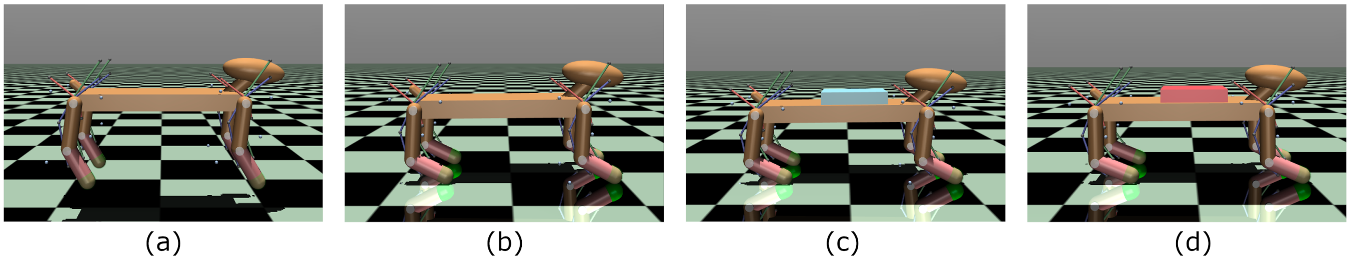

Fig. 1. The simulated tendon-driven quadruped in air (a), on the floor (b), with light load (c), and with heavy load (d).

D. Scaling sensory data

Scaling and normalization of input data can enhance the learning speed and therefore improve on the data-efficiency of a machine learning algorithm. This is even more pronounced when inputs have different units and therefore can have vast differences in their ranges. To address this problem, we have scaled the input data (dividing them by their expected variance). To make sure that we are not depending on prior information, we calculate these scaling factors by running a 60 second babbling (which is done only once for the entire curriculum of tasks and test cases) when the quadruped is on the floor (no load) and use the collected data to calculate scaling factors for all cases. It is obvious that these values do not guarantee unit standard deviation for all sensory data collected across different tasks and test cases, however, it greatly reduces the differences in ranges and therefore helps in faster convergence of the ANN training processes.

E. ANN architectures

We have studied two different ANN architectures to assess which one can lead to more accurate control of movements. A single ANN for the entire system and Multiple ANNs (one for each limb). Regardless of this architecture, all ANNs start with a babbling (in air), and once they got trained, they will concatinate any data coming from new babblings or refinements and re-tune their weights using (train on) this cumulative concatenated data set while warm-starting using the weights from the last case (the ANN weights after the last training and before the addition of the new data). The error defined to train the ANNs is the mean square error (mse) over all muscle activation (fed into the ANN vs. the estimated ones). We use MLP ANNs with one hidden layer (with 24 hidden layer neurons for the single ANN and 6 hidden layer neurons for each ANN in the multiple ANN case), linear activation functions (which we observed to perform better than the case of using sigmoid functions), and the ADAM optimizer [13] implementation in the Keras API from the Tensowflow library.

1) Single ANN: In this case, we use a single ANN that maps all the sensory input into the predicted muscle activations of all 12 muscles. One hypothesis is that since the ANN has access to all sensory inputs (kinematics and tactile sensory of all limbs) at the same time, it might be able to utilize these extra information to better implement the inverse dynamics and therefore enable better control of the limbs.

2) Multiple ANNs: In this case, we use a single ANN for each limb (that is four identical ANNs in total) that maps all the sensory input of that limb into the predicted muscle activation values (3 muscles). An opposing hypothesis to the one brought up in the Single ANN subsection is that a more focused input and output might mean fewer distractive data points for the ANN and therefore lead to better convergence and a superior control of the limbs compared to the previous case.

F. Tactile sensory information utilization

We have studied the potential contribution of tactile sensory in improving the control performance in our G2P framework; especially across different test cases (in terms of both forward and backward generalizability). A MuJoCo touch sensor (sensing the magnitude of the applied force) is used at the end of each limb (see the green areas on Fig. 1a-d). We study the performance of the system across tasks and test cases with and without access to these tactile information.

G. position error feedback

Similar to [11], we have implemented corrective position error feedback on the position error of the joints and here we have studied the amount in which it contributes to the improvement of the accuracy of the limb control in our simulated quadruped across task, test cases, and ANN architectures.

H. Performance metrics

Here, we have assessed the control error on joint positions in the proposed L2M framework. As an essential part of L2M without catastrophic forgetting, the agent should demonstrate both forward learning (being able to learn as it is introduced to new tasks) and backward generalizability (being able to still perform well on the task it has learn earlier even after being trained on the newer tasks) capabilities.

1) Defining of error: Also, to measure how well a system has learned to perform the desired movements, we calculate the round mean square error (rmse) over the joint angles (across all limbs) for the last half of the data (to make sure the effects of transient initial conditions are washed out) and report it. We use rmse as opposed to mse since it preserves the the units of the inputs (radians).

2) Forward learning plots: In these series of plots, we show the error of each task and the progress of error after each babbling or refinement. Please note that for each point in these plots, we use the updated model that was trained with the cumulative data so far.

3) Backward generalization plot: Once babbling and refinement phases across all test cases are over and the ANN parameters are re-tuned using the most updated cumulative data set, we fix the model (ANN parameters) and test all task cases again with this final model. This will show how well the current model can generalize across all tasks it has learned so far without being re-tuned.

III. Results

Here, we have demonstrated the effect of studied configurations on the control accuracy over the joint angles. At each figure, we have only changed the configuration of the interest (position error feedback, tactile sensory information, and the ANN structure) and compared the results while the other configurations are kept the same.

A. Effects of position error feedback

Fig. 2a shows the forward learning plots for the configuration with and without position error feedback in orange and blue, respectively. Also, columns on Fig. 2a-d are corresponding to the test cases of: in air, on the floor, on the floor with the light load, and on the floor with the heavy load (from left to right, respectively). Also, Fig. 3a shows the backward generalization plot for all the test cases mentioned above for the configuration with and without position error feedback (for each test case, right and left box plots, respectively).

As you can see, the plots on Fig. 2a show that the having position error feedback greatly reduces the time needed for the error curves to converge and therefore reduce the number of refinements needed for the error plots to flatten. Also, Fig. 3a (backward generalization results) show that the corrective position error feedback significantly reduces the joint position error across all test cases (One and two stars represent p-valeus equal or smaller than 0.05 and 0.01, respectively, for a one way ANOVA analysis).

In terms of other configurations, these plots represent the results where there is no tactile feedback and the system uses multiple ANNs, however, the general pattern is consistent across other variations (position error feedback always enhances the performance). Please see the supplementary documents for the complete set of result plots for all possible configurations.

B. Effects of ANN architecture

Fig. 2b show the forward learning plots for for all cases for the configuration with multiple, and single ANNs in orange and blue, respectively. Fig. 3b shows the backward generalization plot for all the test cases for the configuration with a multiple and single ANNs (for each test case, right and left box plots, respectively).

As can be seen on both figures, the multiple ANN structure outperforms the single one. Moreover, this difference in performance is more significant in tasks that are experiencing the effects of weight which suggests that multiple ANN structure can better handle (is more robust) the unexpected dynamics of interactions with the floor that are caused by the weight.

One hypothesis would be that since the single ANN has more coefficients (weights and biases) to set, it would need more data to converge. However, we saw a similar pattern (multiple ANNs outperforming the single ANN) even when trained on larger data set. Moreover, the fact that the forward learning curves are almost flat after the second case rejects this hypothesis.

Alternatively, a larger ANN might be more likely to suffer from local minima. In addition, the kinematic input from other limbs seems not to be very helpful and therefore create destructive interference. The differences in the performance are even more statistically significant when we enable position error feedback (see supplementary information). This reinforces the destructive interference in the single ANN hypothesis since there is only a single corrective position error feedback signal for each joint for the multiple ANN structure being fed back to the ANN while all the corrective position error feedback signals are fed back to the same ANN in the single ANN structure.

C. Effects of tactile sensory information

Fig. 2c show the forward learning plots for for all cases for the configuration with and without tactile sensory in orange and blue, respectively. Fig. 3c shows the backward generalization plot for all the test cases for the configuration with and without tactile sensory (for each test case, right and left box plots, respectively).

Here, we have selected the results with configuration without position error feedback to isolate the contributions of tactile sensory. We see that pretense of feedback highly enhances the accuracy and therefore covers most of the contributions of the tactile sensory (please see the supplementary figures).

As can be seen on both figures, overall, the tactile feedback contributes in enhanced performance. It is especially the case for the backward generalization results of the more extreme cases (on air and with the heavy load). We believe that it is because the presence of the tactile sensory enables the mapping to differentiate cases with different weight and therefore select the appropriate muscle activation values for each case. On the contrary, the configuration without the tactile sensory do not have access to these discriminatory information and the ANN tries to minimize the error over the entire data set (across all cases) and therefore would not have major performance drop for the ”on floor” and ”with light load” cases but would suffer more (compare to the configuration with the tactile sensory) on the more extreme cases (”in air” and ”with heavy load”).

Importantly, we see an even more pronounced contribution for the sensory in the single ANN structure (please see the supplementary info) that is persistent for all four test cases. It suggests that having access to sensory signals from other

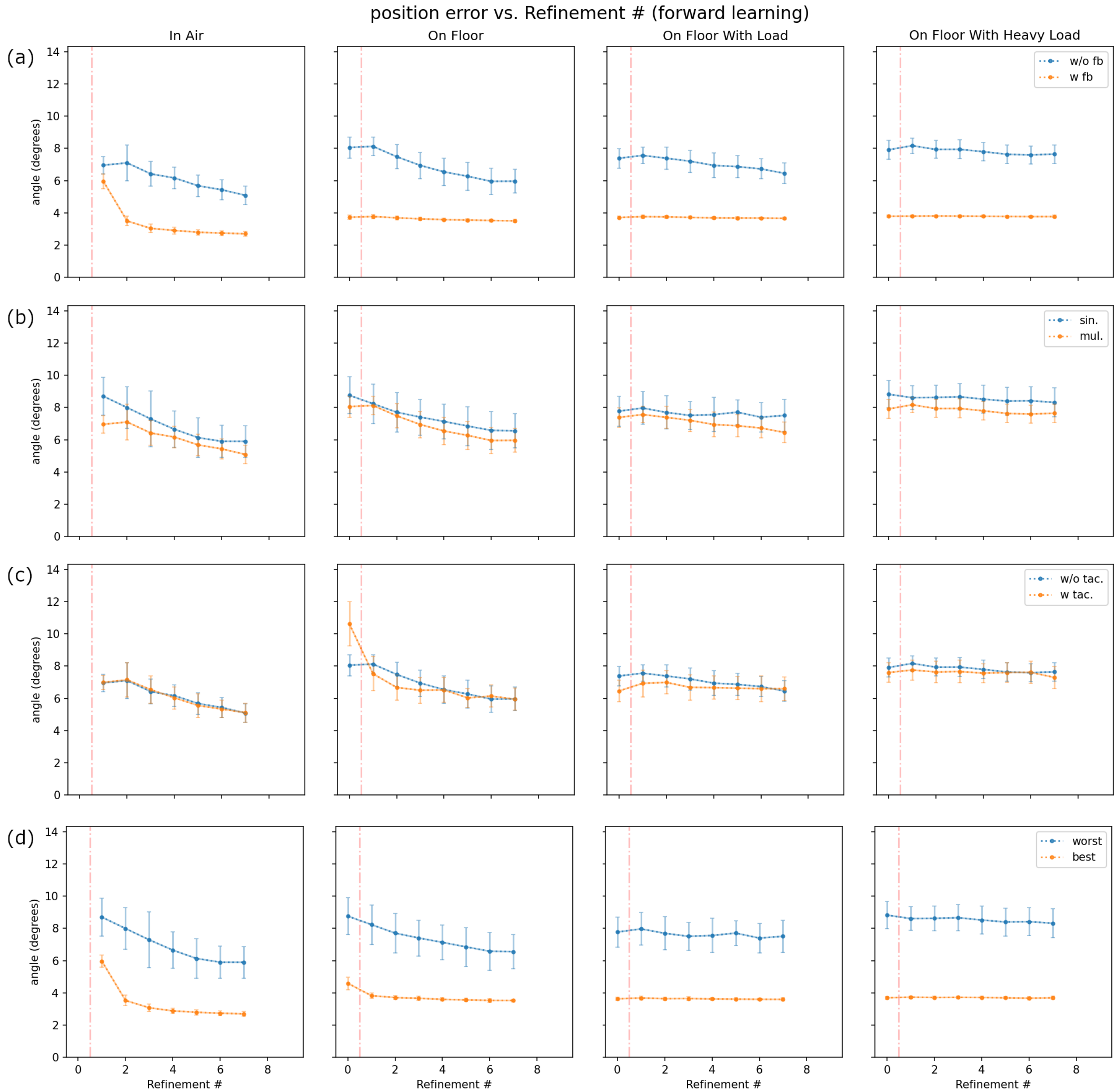

Fig. 2. Forward learning plots across test cases (in air, on floor, on floor with load, on floor with heavy load). Comparing the cases with and without feedback (a), with single or multiple ANNs (b), with or without tactile sensory (c), and best vs. worst combination of features (d).

limbs can further enhance the performance since it provides a more complete set of information regarding the total weight of the system and the part that is being carried by each leg. Therefore, similar to biological systems, it seems like having specialized networks that are interconnected with each other at some level (here, sharing the sensory information of each limb with other limbs) seems to be a promising future design architecture to pursue.

Also, please note that the ”with tactile sensory” case has a relatively higher error on the 0^{th} refinement of the ”on the floor” case (the test results before babbling or being trained on the information collected from the new test case). This is because the input nodes associated with the tactile sensory information so far have always fed with zeros (since there is no touch during the ”in air” case) and therefore, the weights associated with them are not refined. These weights, however, are pruned right after the system is exposed to the new environment and the ANNs are trained with data including tactile sensory information.

D. Stacking all improvements

In this part, in order to assess the overall contribution of these configurations, we have stacked all the configurations that had lead to an improvement in performance (A system with position error feedback, tactile sensory, and multiple ANNs) and have compared the resulting performance with

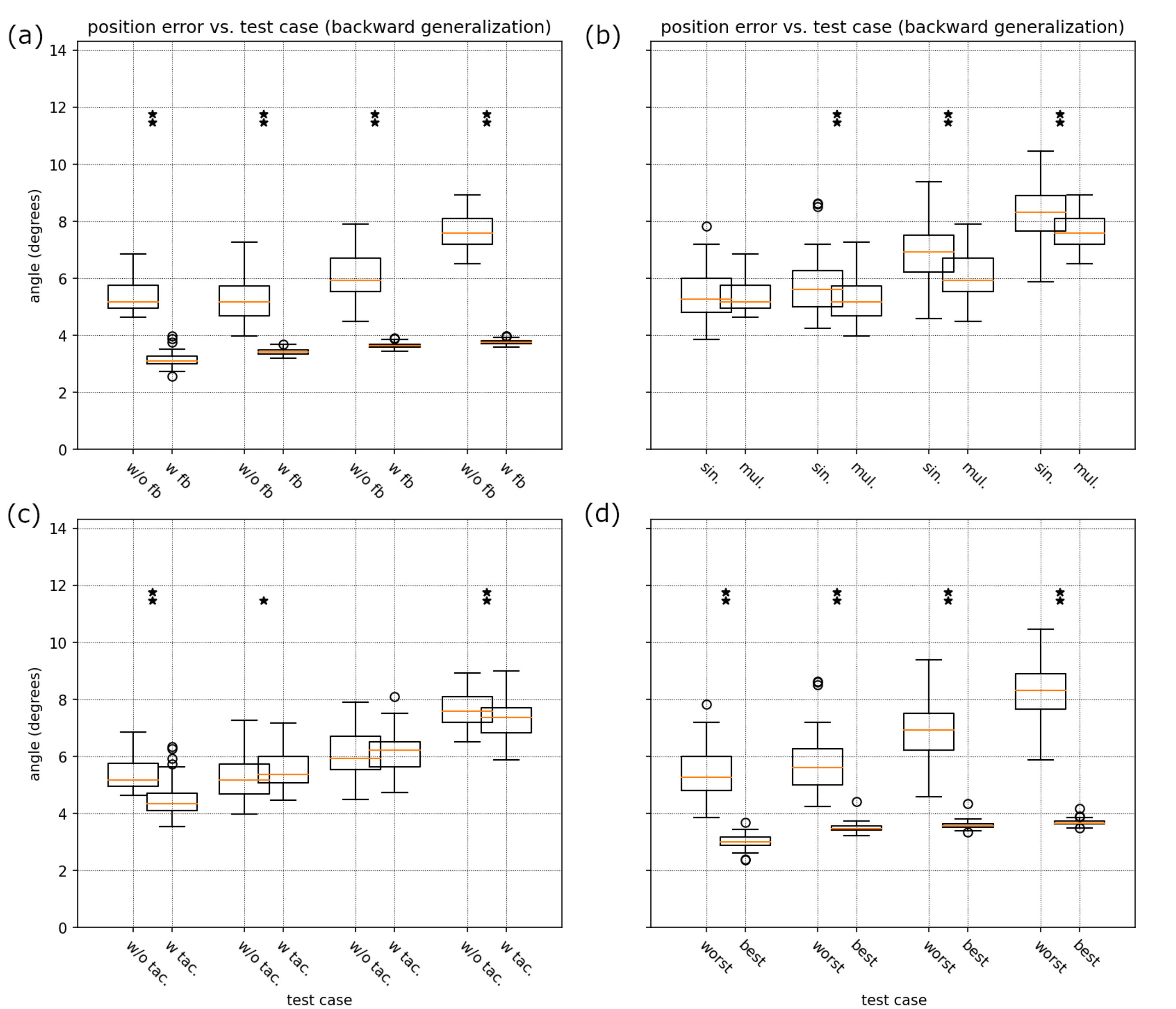

Fig. 3. Backward generalization plots across test cases. Comparing the cases with and without feedback (a), with single or multiple ANNs (b), with or without tactile sensory (c), and best vs. worst combination of features (d).

the system without any of these configurations. We label these two systems as higher performance and lower performance configurations, respectively. Please note that the without enhancement system is similar in configurations to the original G2P algorithm presented in [5].

Fig. 2d and 3d show the plots for the forward learning and backward generalization results, respectively. As it is clear from these figures, there is a significant improvement in the control performance for the higher performance configurations in both forward learning and backward generalization which leads to a better L2M without catastrophic forgetting performance in achieving kinematic control which is the main focus of this paper (also, see supplementary video, parts III and IV).

IV. Conclusion

Here we have implemented task-agnostic autonomous learning for a bio-inspired tendon-driven quadruped and have studied the effects of different configurations on its kinematic control performance. Our results show that the proposed framework is able to learn without any prior knowledge about the design parameters just after a minute of random kinematic exploration (motor babbling) and a few attempts of the task of interest (followed by refinements).

We have shown that the corrective position error feedback that utilized the forward model generated using the G2P algorithm significantly improves the performance. Also, our results show that smaller and more specific neural networks would outperform a single all-in all-out network when only dealt with kinematic information. However, our results suggest that having access to tactile sensory information from other limbs can further enhance the performance of each limbs.

Our proposed framework was able to achieve RMSE of less than 4° in controlling joint positions of a tendon-driven quadruped which satisfied the main goal of this research. Moreover, we have shed some light into the effect of contributions of ANN structure and tactile sensory in kinematic control which can help future work in designing even more accurate and data-efficient control frameworks.

V. Limitations and future work

As brought up in the previous section, our results suggest that although a single ANN for each limb in general outperformed an all-in all-out approach, having access to tactile information of other limbs can favorably affect the kinematic control of a limb. This makes sense since by having access to these information, an ANN would be able to make a better assumption regarding the total weight of the system what fraction of it that the limbs needs to taken care. This is also supported by evidence seen in biology where smaller and densely connected local networks are more sparsely interconnected to other networks at some levels [14], [15]. Implementation of such a hybrid ANN architecture is beyond the scope of the current work but is a very interesting avenue to pursue for future studies.

Here we have achieved to satisfactory levels of kinematic control and it would be interesting to attach it to a higher level controlled (in a hierarchical fashion) to perform functional tasks. Moreover, here we have only shown the effect of studied configurations in kinematic control; however, it would be interesting to also study contributions of the sensory information (either tactile or kinematic) on higher level planing and control task. For example, sensory signals are known to contribute significantly in high level task planning and dexterous manipulation tasks in biological systems [16], [17].

Moreover, although we have used a physically faithful simulator (MuJoCo), it would be an interesting future work to also assess the performance of the system in a real-world, physical implementation. Lastly, here we have studied the L2M capabilities of the proposed framework in adapting and generalizing to different weight loads, however, assessing the adaptiveness of such L2M algorithms to changes to the system design (wear and tear, partial damage or loss of a limb, etc.) would be another interesting research direction that will further pave the way for highly robust, adaptive, and multipurpose L2M robots.

VI. Supplementary Information

Code, design assets, and other information can be found on the project's GitHub repository at: `https://github.com/marjanin/quadruped`

A. Acknowledgement

Funding: NSF CRCNS Japan-US 2113096; DARPA L2M Program W911NF1820264; DoD MR150091; National Institute of Health (NIH) R21-NS113613 to F.J.V.-C. This study; USC Provost Fellowship to A.M. USC Graduate school's Research enhancement fellowship to A.M.

References

[1] D. Kudithipudi, M. Aguilar-Simon, J. Babb, M. Bazhenov, D. Blackiston, J. Bongard, A. P. Brna, S. Chakravarthi Raja, N. Cheney, J. Clune *et al.*, "Biological underpinnings for lifelong learning machines," *Nature Machine Intelligence*, vol. 4, no. 3, pp. 196–210, 2022.

[2] L. Smith, J. C. Kew, X. B. Peng, S. Ha, J. Tan, and S. Levine, "Legged robots that keep on learning: Fine-tuning locomotion policies in the real world," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 1593–1599.

[3] R. Kwiatkowski and H. Lipson, "Task-agnostic self-modeling machines," *Science Robotics*, vol. 4, no. 26, p. eaau9354, 2019.

[4] B. Chen, R. Kwiatkowski, C. Vondrick, and H. Lipson, "Fully body visual self-modeling of robot morphologies," *Science Robotics*, vol. 7, no. 68, p. eabn1944, 2022.

[5] A. Marjaninejad, D. Urbina-Meléndez, B. A. Cohn, and F. J. Valero-Cuevas, "Autonomous functional movements in a tendon-driven limb via limited experience," *Nature machine intelligence*, vol. 1, no. 3, p. 144, 2019.

[6] F. J. Valero-Cuevas, *Fundamentals of neuromechanics*. Springer, 2016.

[7] A. Marjaninejad and F. J. Valero-Cuevas, "Should anthropomorphic systems be "redundant"?" in *Biomechanics of Anthropomorphic Systems*. Springer, 2019, pp. 7–34.

[8] S. Guist, J. Schneider, H. Ma, L. Chen, V. Berenz, J. Martus, H. Ott, F. Grüninger, M. Muehlebach, J. Fiene, B. Schölkopf, and D. Büchler, "Safe & accurate at speed with tendons: A robot arm for exploring dynamic motion," 2024.

[9] M. Schlesinger and D. Parisi, "Multimodal control of reaching-simulating the role of tactile feedback," *IEEE transactions on evolutionary computation*, vol. 5, no. 2, pp. 122–128, 2001.

[10] N. B. Holowka, B. Wynands, T. J. Drechsel, A. K. Yegian, V. A. Tobolsky, P. Okutoyi, R. M. Ojiambo, D. W. Haile, T. K. Sigei, C. Zippenfennig *et al.*, "Foot callus thickness does not trade off protection for tactile sensitivity during walking," *Nature*, vol. 571, no. 7764, pp. 261–264, 2019.

[11] A. Marjaninejad, D. Urbina-Meléndez, and F. J. Valero-Cuevas, "Simple kinematic feedback enhances autonomous learning in bio-inspired tendon-driven systems," *arXiv preprint arXiv:1907.04539*, 2019.

[12] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 5026–5033.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] A. Büschges, "Sensory control and organization of neural networks mediating coordination of multisegmental organs for locomotion," *Journal of neurophysiology*, vol. 93, no. 3, pp. 1127–1135, 2005.

[15] A. Marjaninejad and J. M. Finley, "A model-based exploration of the role of pattern generating circuits during locomotor adaptation," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2016, pp. 21–24.

[16] H. Gertz, D. Voudouris, and K. Fiehler, "Reach-relevant somatosensory signals modulate tactile suppression," *Journal of neurophysiology*, vol. 117, no. 6, pp. 2262–2268, 2017.

[17] F. J. Valero-Cuevas and M. Santello, "On neuromechanical approaches for the study of biological and robotic grasp and manipulation," *Journal of NeuroEngineering and Rehabilitation*, vol. 14, no. 1, p. 101, 2017.