2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)
COEX, Seoul, Korea, April 14-18, 2019

Structured motor exploration for adaptive learning-based tracking in soft robotic manipulators

Yasmin Ansari, Cecilia Laschi, Senior *Member, IEEE,* Egidio Falotico, *Member, IEEE,*

Abstract— **Kinematic control of soft robotic manipulators is a challenging problem particularly for systems that are both globally and locally redundant. This article presents a learning-based task-space kinematic controller that enables tracking in such soft robotic manipulators. The novelty of the work is a bio-inspired structured sampling mechanism that actively regulates the variance in motor movements during motor exploration. It generates a database that is applied to a direct learning architecture, thereby, formulating an inverse model at the position-level. The controller is validated in simulation on a 12 degrees-of-freedom modular manipulator comprised of elemental modules with three longitudinal actuators and one radial actuator. Experiments demonstrate consistency in performance across multiple unseen trajectories and repeatability of each task. Furthermore, the performance remains uncompromised in altered motor conditions, provided task-relevant motor variance. The results exhibit accurate, repeatable, and adaptive tracking behavior of the system and are promising for the advancement of these systems.**

I. Introduction

Soft robotic manipulators are a new generation of bio-inspired continuum manipulators [1] known to exhibit high dexterity and compliance. These appealing properties facilitate a high-degree of safe human-robot interaction and advanced manipulation of unstructured environments. For these reasons, numerous modular systems have been developed for healthcare, manufacturing, and surgical environments since the early 1990s [2].

Such systems are envisioned to be fundamentally employed for tracking tasks. Their successful application relies on controllers that can guide the motion of the end-effector along desired trajectories while meeting performance criteria. Furthermore, it should adapt to altered actuator and/or task conditions. In this regard, a widely adopted framework is task-space (TS) control. The general idea is that the control objectives are defined in TS, whereas, the manipulator is controlled in actuator-space (AS). Formally, let the coordinates of the end-effector be represented as $\boldsymbol{x}: [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m]^T \in \mathbb{R}^m$ where, $\boldsymbol{x}$ is a vector in continuous TS of dimension m. Additionally, let the input applied to the actuators be represented by $\boldsymbol{q}: \left[(q_1, \ldots, q_\varphi)_1, \ldots (q_1, \ldots, q_\varphi)_\sigma\right]^T \in \mathbb{R}^n$ where, $\boldsymbol{q}$ is a vector in continuous AS of dimension $n = \sigma \times \varphi$; σ are total number of modules and; φ are the number of actuators per module. The key-requirement is to formulate the inverse kinematic (IK) mapping in order to transform co-ordinates from TS to AS. However, IK mappings are rendered non-convex due to redundancy with global manifestations ($m \leq n$) and local manifestations ($m \leq \varphi$) [3]. Consequently, the proper functioning of TS controllers relies on resolving IK mappings for both these phenomena, which is currently an unaddressed problem.

The traditional approach adopted for the development of TS controllers involves numerical methods, and are exhaustively documented in [4]. However, it is non-trivial to estimate/update parameters regarding material properties and/or environmental factors, which indisputably influence the behavior of these systems. Attempts to incorporate such variables combined with the iterative nature of numerical methods is found to be restrictive for real-time control.

Data-driven machine learning algorithms have been demonstrated to be a more effective alternative. Advancements in this respect are rapidly gaining momentum due to the ease in implementation and generalizability. The primary focus in existing literature has been to resolve non-convex IK mappings for globally redundant systems. Initial approaches rely on imposing constraints on the training data through the sampling mechanism [5] or the learner [6]. However, the IK mapping learned from such an approach is able to generate only a single solution-set at run-time, thereby restricting the overall capability of the manipulator for practical scenarios. This limitation was recognized and overcome through the development of mechanisms that capture global properties of the kinematic mapping. In [7], the authors demonstrate this by parameterizing local regions with a bijective function invariant under euclidean geometrical properties and complemented with a distance-based sampling mechanism.

This article tackles the problem at hand by taking inspiration from adaptive capabilities of human inverse models [8]. Such a bio-inspired approach has proven useful in the development of indirect control architectures that exhibit invariance in performance and adaptation in robotic systems [9,10,11]. In particular, these controllers have been developed by translating neuroscientific principles of cerebellar function in various kinds of reflexes. Interestingly, recent investigation [12] on human tracking capabilities in the upper limb suggest the possibility to achieve similar capabilities through a direct learning architecture through active regulation of motor variation during motor exploration. In neurophysiological terms, this process is known as *motor variability* [13]. The variance of actuator movements is generated under the same

The authors acknowledge the support from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2)

Y. Ansari, C. Laschi, E. Falotico are with the Soft Robotics Lab, The BioRobotics Institute, Scuola Superiore Sant'Anna, Pisa, 56025, Italy. (email: y.ansari, c.laschi, e.falotico)@santannapisa.it

978-1-5386-9260-8/19/$31.00 ©2019 IEEE

task conditions, where the task is defined by a discrete starting-point and ending-point. Exploration is directed from the former to the latter. Once the terminating condition is reached, a feedback signal indicates the effectiveness of the executed activation pattern. The key-feature is to exploit the feedback such that that the motor variance is high at the beginning and is actively reduced to gain precision [14]. The study highlights that motor variability serves as an internal gain adaptation mechanism that is useful to improve the confidence of the inverse model. Notably, this principle also lays the computational foundation of skill acquisition through trial-and-error [15]. The significance of motor variability for adaptive motor control is widely acknowledged in the field of sports sciences [16].

The central contribution of this paper is taking cue from motor variability to develop a novel structured sampling mechanism that operates synonymous to a hierarchical reactive controller. It generates a database that has high sampling efficiency and can be applied to a direct learning architecture in order to formulate the controller at the position-level. Consequently, the overall framework is simple and fast to implement. Quantitative analysis in simulation reveals that proposed approach is promising for robust, repeatable, and adaptive tracking in soft robotic manipulators.

Section II presents the formulation of control framework. Section III validates the performance of the controller in simulation followed by a discussion of the obtained results with future outlooks.

II. Control Framework

The objective of this work is the mathematical formulation of a sampling mechanism (ψ) that actively regulates the variation in motor constituents during motor exploration. As highlighted in the previous section, the key ingredients required for its formalization are: (i) defining tasks in TS with a specific starting-point and ending-point; (ii) a feedback mechanism that evaluates the effect of executing an activation pattern in AS in order to fulfill a task; (iii) an action selection mechanism that exploits the feedback to choose the next action from an action-set $\boldsymbol{A}$. The authors exploit the principles of a position-level hierarchical reactive controller presented in [17] to develop a stand-alone structured sampling mechanism, discussed subsequently.

A. Structured Sampling Mechanism

Initially, N points $\boldsymbol{g_i} \in \boldsymbol{G}\, \forall\, i = 1,..,N$ are created in TS by converting the input activation variables into spherical coordinates $(r, \theta, \emptyset)$, where, r is the linear motion capability of the manipulator, θ is 360º for omnidirectional bending, and $\emptyset$ is the bending capability of the manipulator measured with respect to the z-axis. This is to enable a uniform coverage the target workspace.

Let the input applied to the system at time t be $\boldsymbol{q_c}$, and the corresponding end-effector position be $\boldsymbol{x_c}$. The task is to reach point $\boldsymbol{g_i}$ beginning at $i = 1$. The next motor configuration, $\boldsymbol{q_n}$, is obtained at the time-step $t + 1$ after selecting a displacement $\delta\boldsymbol{q}$ (explained in the next paragraph). At the time-step $t + 2$, the motor configuration returns back to $\boldsymbol{q_c}$. This process is repeated until a motor configuration $\boldsymbol{q_n}$ results in an end-effector position $\boldsymbol{x_n}$ closer in euclidean distance with respect to the current point $\boldsymbol{g_i}$. Then $\boldsymbol{q_c}$ is replaced by $\boldsymbol{q_n}$. Once the point $\boldsymbol{g_i}$ is reached, or a timeout is reached, it is replaced with the next consecutive point $\boldsymbol{g_{i+1}}$. The process continues for all the *N* points. Equation (1) highlights this mechanism which efficiently imposes a continuous monotonic sampling in the kinematic domain.

$$\psi = \begin{cases} \boldsymbol{q_n} \leftarrow \boldsymbol{q_c} + \delta\boldsymbol{q} & \boldsymbol{if}\ \|\boldsymbol{x_n} - \boldsymbol{g_i}\| > |\|\boldsymbol{x_c} - \boldsymbol{g_i}\| \\ \boldsymbol{q_c} \leftarrow \boldsymbol{q_n},\ \boldsymbol{g_i} \leftarrow \boldsymbol{g_{i+1}} & \boldsymbol{else} \end{cases} \quad (1)$$

The selection mechanism for δq depends upon the following pre-requisites: (i) $\boldsymbol{A}$ is divided into K subsets $\boldsymbol{a_i} \in \boldsymbol{A}\, \forall\, i = 1,..,K$ of decreasing magnitudes. (Every subset also has the zero element such that it is possible to keep the length of individual actuator unchanged); (ii) the distance d between any two consecutive points $\boldsymbol{g_i}$and $\boldsymbol{g_{i+1}}$ is divided into K sub-regions $\boldsymbol{s_j} \in \|\boldsymbol{g_{i+1}} - \boldsymbol{g_i}\|\, \forall\, j = 1,..,K$. Through pre-defined hard constraints, every sub-region is associated with a specific action subset such that action subsets of larger magnitudes are selected for sub-regions further away from the point and *vice-versa*. Consequently, depending upon d, the manipulator experiences the action range within the reaching of one point, and is our proposed computational interpretation of motor variability. The entire process is repeated for *N* points in order to experience the motor variability throughout the kinematic reachable workspace.

B. Direct Learning Architecture

The mapping $(\boldsymbol{x_n},\ \boldsymbol{x_c}) \rightarrow \boldsymbol{q_n}$ is applied to a universal approximator through the method proposed in [7]. Briefly, it comprises of a single hidden-layer artificial neural network with a tan-sigmoid function applied in the hidden layer for pattern recognition and a linear transfer function applied in the output layer for function fitting [18]. Learning is done using Bayesian regularization. The primary motivation to select this network is its ability to estimate valid mappings even with a randomly selected exaggerated number of neurons in the hidden-layer [19]. The learned function estimates the gradient function between the errors and the motor commands and its displacement. Therefore, the displacement term potentially behaves as an inherent adaptive gain mechanism, and provides the computational principle for improved confidence of the learned model. This is particularly useful to avoid exhaustive sampling of the AS. Furthermore, since the data is bounded by the actuator range, the network will always output a valid actuator configuration. The learnt function is exploited to formulate a closed-loop controller according to Fig. 1.

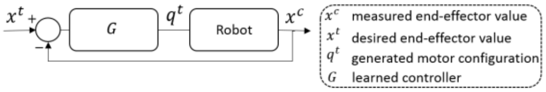
x^t
+
−
G
q^t
Robot
x^c
x^c measured end-effector value
x^t desired end-effector value
q^t generated motor configuration
G learned controller

Fig. 1 Schematic layout of the controller.

III. Simulation

In this section, the controller is verified on an open-source simulator that models the kinematics of modular continuum manipulators through constant curvature approximation. In particular, the mapping is from joint-space (the length of the actuator) to the TS (3D cartesian coordinates) [20]. An elemental module comprises of a triad of longitudinal actuators arranged at a 120° with respect to each other, and are routed along a backbone whose radius can be actively varied. Consequently, $\sigma = 4$. The neutral lengths of the longitudinal and radial actuator(s) are 20cm and 6cm, respectively, and can be independently varied. The overall system is formulated through the serial concatenation of three elemental modules, that is, $\varphi = 3$ (Fig. 2). Consequently, the overall kinematic mapping is redundant both globally ($m = 3 < n_{(=\varphi\times\sigma)} = 12$) and locally ($m = 3 < \varphi = 4$). It is capable of elongation, shortening, omnidirectional bending, and radial variation local to each module.

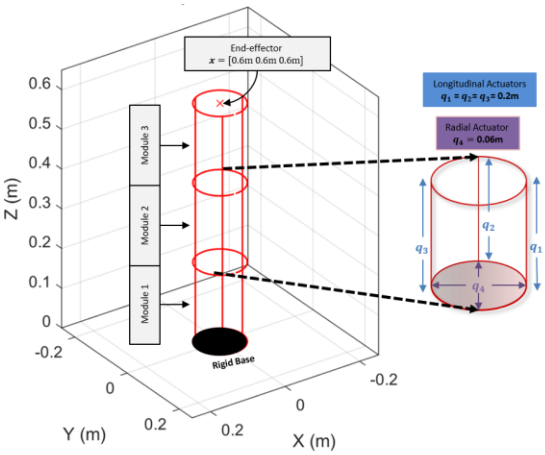

Fig. 2 Overview of simulated manipulator in the neutral position.

A. Algorithmic settings

The target workspace is a subset of the reachable workspace defined in cylindrical coordinates as $R = [50\ 70]$cm, $\varphi = [0°\ 360°]$, $\theta = [0°\ 30°]$. The limits of the action-set for the longitudinal and radial actuators are [10 30]cm and [4 8]cm, respectively. The action-set comprises of a discrete number of primitive actions that correspond to the actuator length. The maximum value of the action-set is 10% of the maximum activation level of the corresponding actuators. A total of 80 targets are created in TS, such that, the distance between two consecutive points is less than 3cm. The task is achieved as soon as the distance with-respect-to the current point is less than 1cm. The timeout is set after 20 trials.

B. Training Data and Learning

A database, *DB1*, is generated by dividing the action-set into four subsets as follows: $\{\{[\pm.06, \pm.08, \pm.1, 0]_r$, $[\pm.2, \pm.4, \pm.6, 0]_l\}$; $[\pm.08, \pm.1, \pm.2, 0]_r$,$[\pm.4, \pm.6, \pm.8, 0]_l\}$; $\{[\pm.2, \pm.4, \pm.6, 0]_r$, $[\pm.6, \pm.8, \pm1, 0]_l\}$;$\{[\pm.4, \pm.6, \pm.8, 0]_r$, $[\pm.8, \pm1, \pm3, 0]_l\}$ cm. The activation is selected from the first subset if the distance to the point is less than 1cm; otherwise it is selected from the second subset if the distance is less than 3cm; otherwise it is selected from the third subset if the distance is less than 7cm; otherwise it is selected from the last subset. These hard constraints are initialized heuristically at the beginning of the algorithm. The structured exploration mechanism, then, generates a dataset of 5234 data points. The variance of the exploration is found to be within 2% of the maximum activation. It is then fed to the learner whose network size is fixed to 35 and is run for 500 epochs. The training time is 35s. The performance of the algorithm is measured through the mean square error of the training data and is found to be 1.3e-4. Note that for all subsequent experiments, the manipulator always begins movement from the neutral position.

C. Tracking

In these set of experiments, the controller is tested for tracking of three unseen trajectories: (i) a vertical line of 10cm; (ii) a horizontal line of 20cm and; (iii) a spiral of height 10cm and radius 10cm. Each trajectory is discretized into way-points, such that, the distance between two consecutive points is 5mm, an unseen criterion for the manipulator. The motion along each trajectory is repeated three times to check the repeatability of the algorithm. The results for the tracking task are depicted in Fig. 3(a,b,c) The average accuracy of the aforementioned tasks are 6mm, 4mm, and 8mm, respectively. These results show the effectiveness of the algorithm to generate valid, bounded, and high-performance IK solution for the desired tracking tasks.

D. Set-Point Movements

In this section, the behavior of the controller is investigated for repeated movements between two set-points separated by a distance of 20cm. The average reaching error, depicted in Fig. 4a (top), is found to be 1.1cm. The discrepancy in the tracking error vs. set-point reaching error raises the question whether a higher variance during motor exploration would improve the behavior for both the tasks. Consequently, a second database, *DB2*, is generated for this purpose using the same algorithmic parameters stated in Section III A. Similar to the method adopted previously, the action-set is divided into three subsets as follows: $\{$ $\{[\pm.06, \pm.08, \pm.2, 0]_r$,$[\pm.2, \pm.4, \pm.6, 0]_l\}$;$\{$ $[\pm.2, \pm.4, \pm.6, 0]_r$, $[\pm.6, \pm.8, \pm1, 0]_l\}$;$\{[\pm.4, \pm.6, \pm.8, 0]_r$, $[\pm.8, \pm1, \pm3, 0]_l\}$ cm. The activation is selected from the first subset if the distance to the point is less than 1cm; otherwise it is selected from the second subset if the distance is less than 2cm; otherwise it is selected from the last subset. The structured sampling mechanism generates a dataset of 7967 data points. The variance of the exploration is found to be within 10%. After learning, the behavior of the second controller for the three tracking tasks is illustrated in Fig. 3(d,e,f) and the set-point task is depicted in Fig. 4a (bottom). Access to a larger range of actions (Fig. 4b) proves useful to improve the average reaching accuracy of the set-point movement, specifically down to 6mm. Interestingly though, the average tracking error for the horizontal line, vertical line, and helix has increased to 1cm, 1.2cm, and 1.5cm, respectively. These results suggest that a higher variance does not necessarily imply better overall behavior, but rather, variance is task relevant.

E. Adaptive Behavior

In these set of experiments, the controller developed using *DB1* is tested for altered motor conditions. In turn, this represents a modification of the kinematics, and therefore, requires the controller to generate redundant solutions for successful tracking. The controller is required to repeat the vertical tracking task stated in Section III C but after restricting the motion of all the radial actuators to the following values: (i)7cm; (ii) 6cm; (ii) 5cm. The results are shown in Fig. 5(a,b,c) where the average tracking error is 9mm, 1cm, and 2.55cm respectively. The performance of the controller is relatively invariant in the former two cases and highly variant in the last case. Since the bending curvature increases with a decreasing radius, the controller would require making actuator adjustments of larger magnitude in order to counter bending effects. However, since the variance of the dataset is only 2%, this can explain the large error. The experiment is repeated again for the following restrictions on the radial actuators: (i) radius of the bottom segment is fixed at 7cm; (ii) radius of the central segment is fixed at 6cm; (iii) radius of the top segment is fixed at 5cm. This results in a conical shape of the manipulator. The tracking error (Fig. 5d) is 1.2cm, which highlights that the bending effects of the topmost segment are significantly countered by the bottom two segments. In general, the results highlight the bounded and adaptive capabilities of the learned solution.

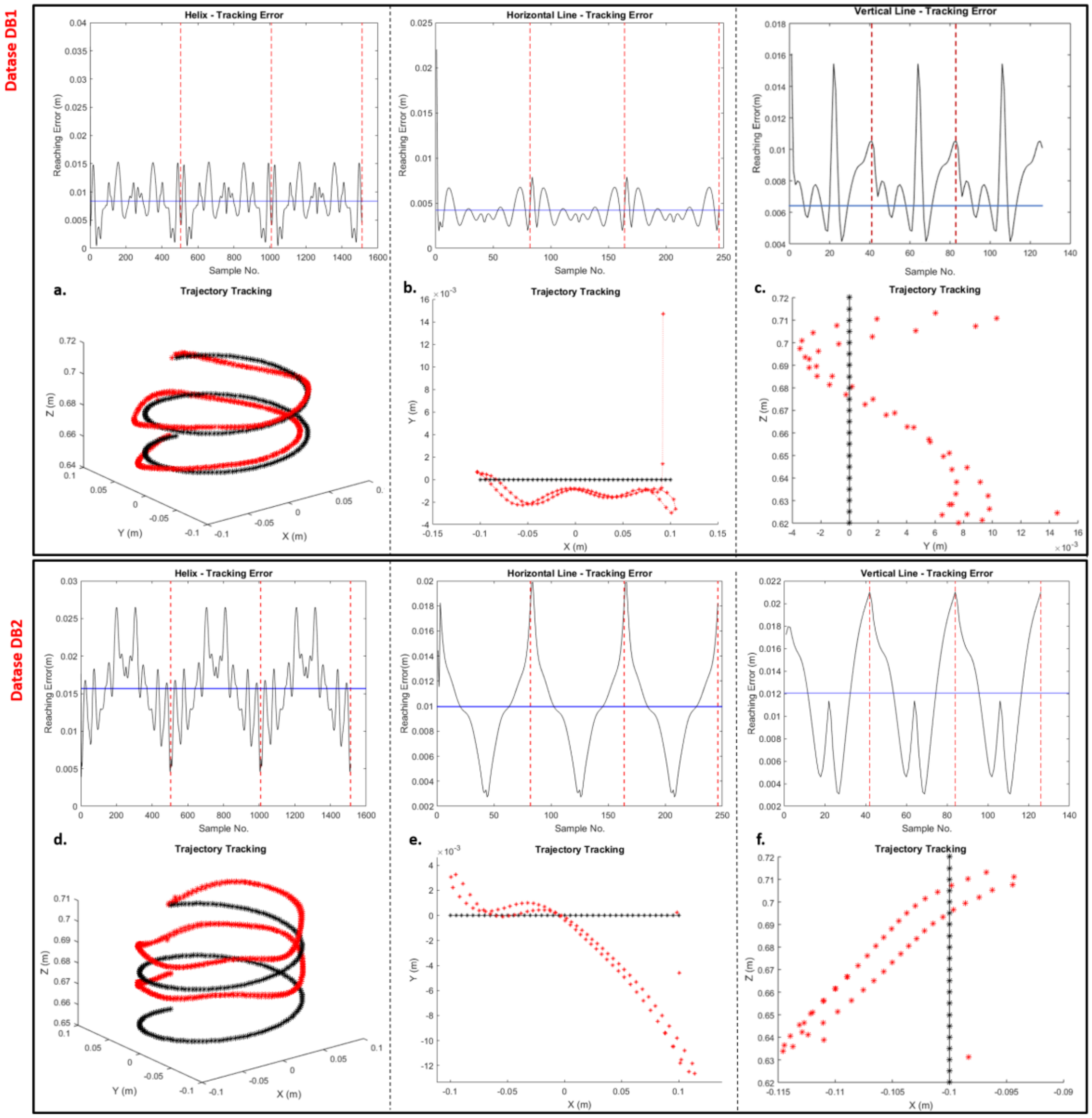

Fig. 3. Performance of proposed controller (all measurements are in m) **(a & d)** Helix; **(b & e)** Horizontal Line; **(c & f)** Vertical Line. The lower two graphs illustrate the motor variance experienced by the longitudinal and radial actuators throughout the motor exploration. Each trajectory is repeated three times to check for robustness. Note that a single trajectory corresponds to the number of samples enclosed within the red-dotted vertical line. The average tracking/reaching error is highlighted in blue in each graph.

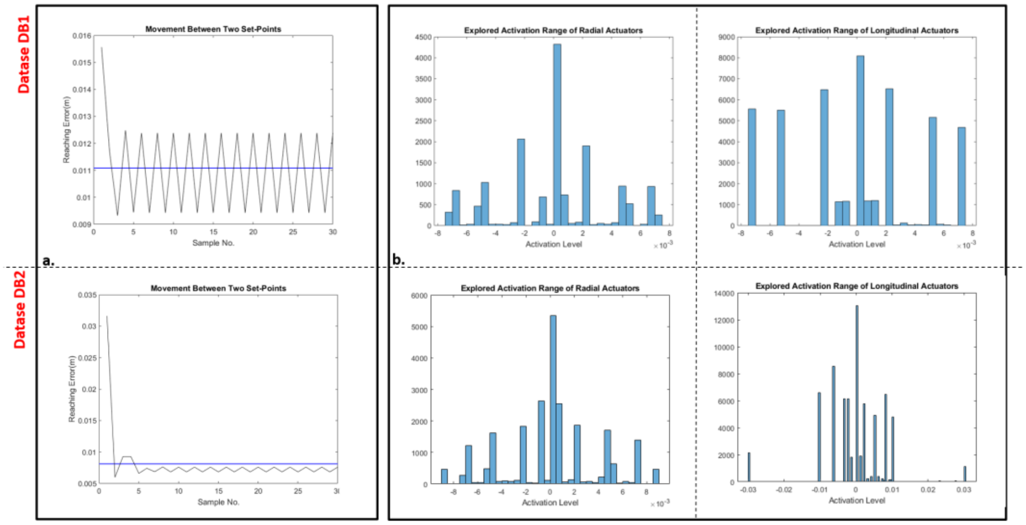

Fig. 4. a. Set-point performance of proposed controller (all measurements are in m). Each trajectory is repeated ten times to check for robustness. The average tracking/reaching error is highlighted in blue in each graph **b.** Motor variance experienced by the longitudinal and radial actuators throughout the motor exploration.

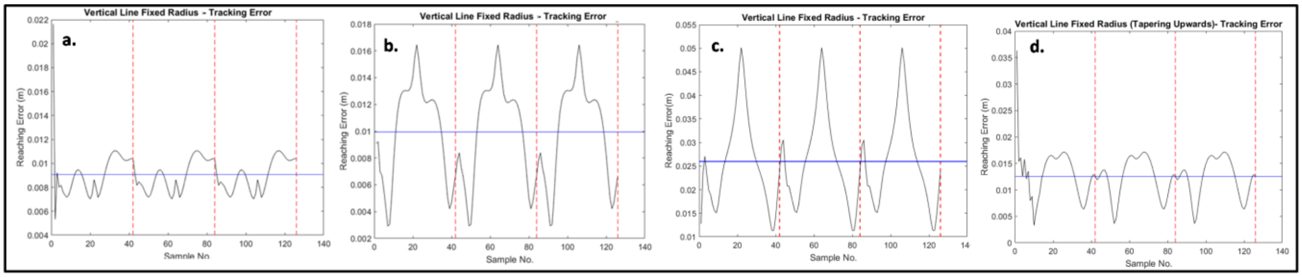

Fig. 5. Performance of proposed controller (all measurements are in m) to altered motor conditions **(a)** Top/Middle/Bottom Radial actuators fixed at 7cm; **(b)** Top/Middle/Bottom Radial actuators fixed at 6cm; **(c)** Top/Middle/Bottom Radial actuators fixed at 5cm; **(d)** Top/Middle/Bottom Radial actuators fixed at 5cm, 6cm, and 7cm, respectively. The graphs demonstrate that ability of the controller to produce redundant solutions with a bounded response.

IV. Conclusion

This paper presents a novel control approach for soft robotic manipulators to enable high performance, repeatable, and robust tracking. It is biologically inspired by the notion of motor variability. Computationally, this has been realized through a structured sampling mechanism that is synonymous to a hierarchical reactive controller. This generates a database that is then applied to a direct learning architecture to develop the inverse model at the position-level. In essence the learning involves estimating the gradient between the errors and the motor commands as well as the applied displacements. The latter component provides the mechanism that improves the confidence of the learned model. The controller has been validated in simulation on a 12DoF modular soft robotic manipulator that is able to elongate, shorten, bend in omnidirections, and vary its radial size local to each segment. Firstly, the repeatability and robustness of the controller is verified through multiple tracking tasks. Our results also demonstrate that high variability does not guarantee optimal behavior across different tasks. Rather it seems, variance is task-relevant. Finally, altered task conditions demonstrate the ability of the controller to generate redundant solutions without compromising on performance criteria, provided motor variability with task relevance. These findings are promising for the advancement of these systems for tracking applications.

References

[1] Laschi, C., Mazzolai, B. and Cianchetti, M., 2016. Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Sci. Robot.*, *1*(1), p.eaah3690.

[2] Walker, I.D., 2013. Continuous backbone "continuum" robot manipulators. *Isrn robotics*, *2013*.

[3] DeMers, D. and Kreutz-Delgado, K., 1994, May. Canonically parameterized families of inverse kinematic functions for redundant manipulators. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation* (pp. 1881-1886). IEEE.

[4] George Thuruthel, T., Ansari, Y., Falotico, E. and Laschi, C., 2018. Control strategies for soft robotic manipulators: A survey. *Soft robotics*, *5*(2), pp.149-163.

[5] Rolf, M. and Steil, J.J., 2014. Efficient exploratory learning of inverse kinematics on a bionic elephant trunk. *IEEE transactions on neural networks and learning systems*, *25*(6), pp.1147-1160.

[6] Melingui, A., Lakhal, O., Daachi, B., Mbede, J.B. and Merzouki, R., 2015. Adaptive neural network control of a compact bionic handling arm. *IEEE/ASME Transactions on Mechatronics*, *20*(6), pp.2862-2875.

[7] George Thuruthel, T., Falotico, E., Manti, M., Pratesi, A., Cianchetti, M. and Laschi, C., 2017. Learning closed loop kinematic controllers for continuum manipulators in unstructured environments. *Soft robotics*, *4*(3), pp.285-296.

[8] Wolpert, D.M., Miall, R.C. and Kawato, M., 1998. Internal models in the cerebellum. *Trends in cognitive sciences*, *2*(9), pp.338-347.

[9] Kawato, M. and Gomi, H., 1992. A computational model of four regions of the cerebellum based on feedback-error learning. *Biological cybernetics*, *68*(2), pp.95-103.

[10] Vannucci, L., Falotico, E., Di Lecce, N., Dario, P. and Laschi, C., 2015, July. Integrating feedback and predictive control in a bio-inspired model of visual pursuit implemented on a humanoid robot. In *Conference on Biomimetic and Biohybrid Systems* (pp. 256-267). Springer, Cham.

[11] Wu, H.G., Miyamoto, Y.R., Castro, L.N.G., Ölveczky, B.P. and Smith, M.A., 2014. Temporal structure of motor variability is dynamically regulated and predicts motor learning ability. *Nature neuroscience*, *17*(2), p.312.

[12] Latash, M.L., Scholz, J.P. and Schöner, G., 2002. Motor control strategies revealed in the structure of motor variability. *Exercise and sport sciences reviews*, *30*(1), pp.26-31.

[13] Dhawale, A.K., Smith, M.A. and Ölveczky, B.P., 2017. The role of variability in motor learning. *Annual review of neuroscience*, *40*, pp.479-498.

[14] Sutton, R.S. and Barto, A.G., 1998. *Introduction to reinforcement learning* (Vol. 135). Cambridge: MIT press.

[15] Bartlett, R., Wheat, J. and Robins, M., 2007. Is movement variability important for sports biomechanists?. *Sports biomechanics*, *6*(2), pp.224-243.

[16] Ansari, Y., Manti, M., Falotico, E., Cianchetti, M. and Laschi, C., 2018. Multiobjective optimization for stiffness and position control in a soft robot arm module. *IEEE Robotics and Automation Letters*, *3*(1), pp.108-115.

[17] Santos, R.B., Rupp, M., Bonzi, S. and Fileti, A.M., 2013. Comparison between multilayer feedforward neural networks and a radial basis function network to detect and locate leaks in pipelines transporting gas. *Chem. Eng. Trans*, *32*(1375), p.e1380.

[18] Foresee, F.D. and Hagan, M.T., 1997, June. Gauss-Newton approximation to Bayesian learning. In *Proceedings of the 1997 international joint conference on neural networks* (Vol. 3, pp. 1930-1935). Piscataway: IEEE.

[19] Rolf, M. and Steil, J.J., 2012, October. Constant curvature continuum kinematics as fast approximate model for the Bionic Handling Assistant. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (pp. 3440-3446). IEEE.