Stable Open Loop Control of Soft Robotic Manipulators

Thomas George Thuruthel, Egidio Falotico, Mariangela Manti, and Cecilia Laschi

***Abstract*—Dynamic control of soft robotic manipulators is a challenging field still in its nascent stages. Modeling is still a major hurdle due to its high dimensional nonlinear dynamic properties. Even if accurate models of these high dimensional nonlinear systems are available, the computational burden of the models or sensory requirements poses problems for robust and stable control. This letter addresses these two problems by using a data-driven model that uses only mechanical feedback for stable control of the soft robotic manipulator. Using a learning-based open loop dynamic controller, we investigate the self-stabilizing behavior that can be obtained from the complex dynamics of a soft manipulator. Experimental findings illustrate the advantage of using open-loop dynamic controllers for accurate self-stabilizing control of soft robotic manipulators without any sensory feedback.**

***Index Terms*—Soft material robotics, AI-based methods, robust/adaptive control of robotic systems.**

I. Introduction

EXPLOITING the morphological properties of a dynamic system as a control strategy is not an unfamiliar concept for locomotion. Various demonstrations for passive walking [2], open loop dynamic locomotion [3], [4] and for energy efficiency [5] have been successful implemented. Similar examples for manipulation tasks are scarce mainly because of the relative ease of controlling traditional manipulation tasks.

With the expanding interests and emergent technologies in soft robotics field, specifically in soft robotic manipulators, the scope of manipulation tasks have expanded to more complicated tasks. Their intrinsic "softness" which makes them inherently safe correspondingly complicates their modelling and control process. However, this is not necessarily a downside as this would lead to a proportional increase in the 'richness' of their internal mechanical feedback. Appropriate examples of this phenomenon would be the universal grippers based on granular jamming [1] and the simple under-actuated grippers made by soft lithography [6]. However, most of these works have been limited to static grasping tasks with minimal control.

Manuscript received September 9, 2017; accepted December 28, 2017. Date of publication January 24, 2018; date of current version February 8, 2018. This letter was recommended for publication by Associate Editor X. Liu and Editor Y. Sun upon evaluation of the reviewers' comments. This work was supported by the European Commission through the I-SUPPORT Project (HORIZON 2020 PHC-19, #643666).*(Corresponding author: Thomas George Thuruthel.)*

The authors are with the Soft Robotics laboratory, Scuola Superiore Sant'Anna, Pisa 56127, Italy (e-mail: t.thuruthel@santannapisa.it; egidio.falotico@santannapisa.it; m.manti@santannapisa.it; cecilia.laschi@santannapisa.it).

Digital Object Identifier 10.1109/LRA.2018.2797241

This letter is an investigation into the capabilities of a soft manipulator in a dynamic motion task. More specifically we are concerned with the existence of the stable trajectories that can be used for stable open loop control, similar to finding limit cycles in locomotion tasks. Plooij *et al.* investigated this possibility on a two DoF planar rigid manipulator [7]. Their preliminary results showed that it was possible to obtain stable open loop trajectories for repetitive manipulation tasks. Further they were able to extend the approach to be more robust to model uncertainties in [10]. However, being a model based approach and being heavily sensitive to the dynamic model, it is not directly applicable for soft robotic manipulators. This is because of the difficulties involved with obtaining analytical dynamic models of these systems. More importantly, models that relate the mapping from the actuator space to task space without significant simplifications are still under investigation. Significant works are still restricted to simulations [16] or to planar cases [15]. Our previous work based on a learning based model and trajectory optimization was tested on a single module manipulator [8]. Furthermore, these approaches were validated only in open loop because of the computational complexity of model and the optimization part. Even if computationally tractable models are available, it is not straightforward to obtain reliable state information without affecting the manipulator dynamics and in unstructured environments. Therefore, our aim is to exploit the natural attractor dynamics observed in complex nonlinear systems and study the stability of these mechanically stabilized systems.

Our approach is an extension of the previous work in [8] to a two section manipulator while improving the controller accuracy. An experimental approach is adopted to study the self-stabilizing behavior of some particular trajectories. The region of attraction for each stable trajectory is further estimated using the learned forward dynamic model. This work shows that it is much easier to generate highly stable open loop trajectories with reasonable accuracy to the desired path. Since the manipulator motion does not depend on any sensory feedback, under unobstructed execution, the motion is highly repeatable. Also the stability of the motion is independent from the accuracy of the dynamic model. Such an approach is ideal for repetitive industrial tasks where sensory information is scarce or expensive; for instance when the environment is cluttered. Due to the inherent compliance of the system, perfectly safe interactions between the robot and the user/environment can be performed without compromising on the accuracy.

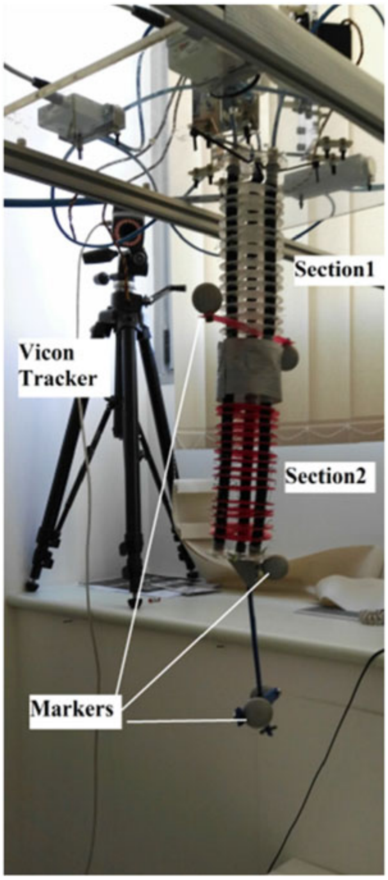

Fig. 1. The pneumatically actuated soft manipulator used for the experiments, Section I is fully actuated while Section II is completely passive. Three markers are attached to the distal section and two to the proximal one.

II. Experimental Setup

The experiments are conducted on a two section pneumatically actuated soft manipulator [9], [14]. The manipulator is cylindrical with each section having three radially arranged pneumatic chambers (see Fig. 1). The distal section is kept underactuated. The passive section increases the dynamic workspace considerably yet not much the static workspace. This would allow us to formulate more complex trajectories to execute and also provide a comparison to our previous work on a static controller for the same manipulator. Therefore the setup has only three control inputs. An electronic proportional micro regulator Series K8P pressure regulator is used for the closed loop control of the chamber pressures. The Vicon tracking system is used to track the five markers attached along the manipulator (Fig. 1). All the five markers are not necessary for the learning process as will be described later. The manipulator is mounted vertically. This is important to obtain the same manipulator configuration when the manipulator is fully passive.

III. Controller Formulation

In this section we describe the formulation of the model-free controller, which will be used for calculating the open loop control policies. The controller is based on a learned forward dynamic model and a trajectory optimization component. It is based on the author's previous work in [8], but with minor modifications that allow for more accurate tracking.

A. Forward Model

Since analytical models of the dynamics of a soft robotic manipulator are difficult to frame we have developed an offline learning based approach for obtaining the forward dynamic model. This allows us to formulate a direct mapping from the input space to the task space. Defining the task space variable as, with a dimension related to the number of markers, the forward dynamic mapping can be represented as: $(\tau, x, \dot{x}) \rightarrow \ddot{x}$. Where $\boldsymbol{\tau}$ are the force inputs. In our case this will be pressure values inside the chambers. The actual values of the pressure values are not accessible as we do not have external sensors for this purpose. Also, even if the actual values are accessible, it is not directly controllable due to the low level pressure regulator. However, since we rely on a data driven method, the actual pressure can be assumed to be directly proportional to the commanded signal (for a constant sampling period). Then the current pressures (P_i) can be written as a function of the previous pressure values (P_{i-1}) and the commanded signal (u_i):

$$P_i = P_{i-1} + f(u_i) \tag{1}$$

This can further be simplified as a function of just u_i and u_{i-1}, since we know the initial pressure values. In our previous work [8] the actual pressure was represented only as a function of the current command signal u_i, which significantly affected the accuracy of the forward model. By keeping the time period of the forward model fixed, we can discretize our forward model only in terms of zero-order variables and obtain a new mapping as shown in equation (2). This helps us in representing the forward model using a recurrent neural network called nonlinear autoregressive network with exogenous inputs (NARX) (Refer to [8] for details on this part).

$$x_{i+1} = f(x_i, x_{i-1}, u_i, u_{i-1}) \tag{2}$$

Here, (x_{i-1}, x_i, x_{i+1}) are the absolute position values of the previous, current and future marker states respectively. The samples for training the forward model is obtained by motor babbling for duration of 240 seconds. The samples are collected at 50 Hz, summing to a sample size of 12000. The pneumatic pressures are limited to a relative pressure of 1.08 bars for the sampling. It was observed that information from the last three markers on the manipulator was sufficient to learn a good model of the forward dynamics. Hence a six dimensional task space/configuration space representation was enough to learn the model of the two section manipulator (the three markers are rigidly attached). A higher dimensional representation would surely be required if the distal section is actuated. The workspace tracked by the tip of the manipulator during the sampling process is shown in Fig. 2.

The learning is done by the Bayesian regularization backpropagation algorithm of MATLAB and an early stopping method is employed. This initial training is required since direct training of the NARX network is high susceptible to the gradient

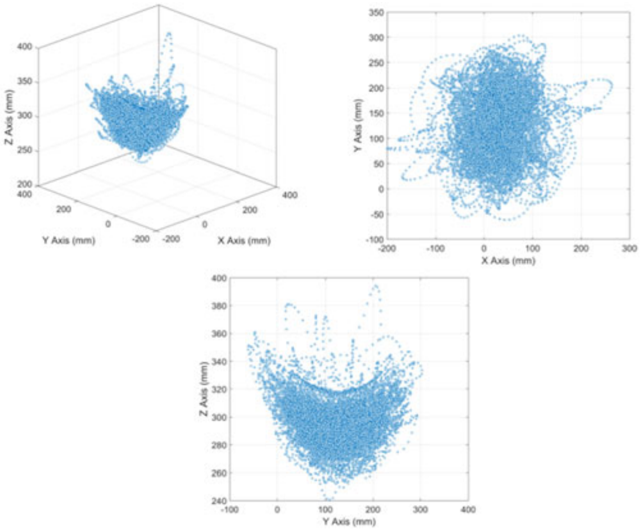

Fig. 2. Dynamic workspace of the manipulator.

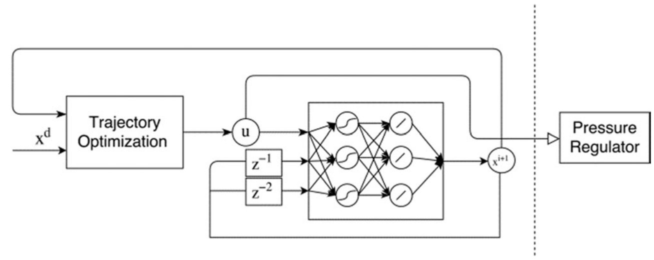

Fig. 3. Schematic of the open loop controller. Here $\boldsymbol{u}$ will contain the whole control policy as an array.

TABLE I
TRAJECTORIES USED FOR EXPERIMENTS

Shape	X	Y
Line	Constant	$150 * \text{Sin}\ (\omega t)$
Circle	$150 * \text{Cos}\ (\omega t)$	$150 * \text{Sin}\ (\omega t)$
Fig. 8	$80 * \text{Cos}\ (\omega t/2)$	$80 * \text{Sin}\ (\omega t)$
Hypotrochoid	$40 * \sin\ (\omega t)$ $+\ 100 * \sin\ (2/3\omega t)$	$40 * \sin\ (\omega t)$ $-\ 100 * \sin\ (2/3\omega t)$

exploding problem. For the second step, the open network is closed and trained with the objective to reduce the multistep prediction error. Tan-sigmoid transfer function is used in the input layer and a linear transfer function is used in the output layer. A single layer network composing of 40 neurons was sufficient to learn complete dynamic model of the two section soft robotic manipulator. The samples are divided in the ratio 70:15:15 for training, validation and testing respectively. With the learned forward model a single multi-step ahead for whole 240 seconds sample has an average error of 1.6 ± 0.88 mm.

B. Trajectory Optimization

To obtain the open loop control policies we use a single shooting method for solving the trajectory optimization problem. It is a common process adopted in cases where closed loop solutions are difficult to formulate analytically or numerically. Given the learned dynamic model, the future states of the system can be predicted with the current control policy:

$$x_{i+1} = f\left(x_i, x_{i-1}, u_i, u_{i-1}\right) \forall i = 0 \,..\, \tfrac{t_f}{dt} \tag{3}$$

Where, u_i is the current control input, dt is the step size of the dynamic model (20 ms), t_f is the control horizon and the function f represents the NARX network. The control policy is given by:

$$\begin{aligned} \Pi\left(t\right) &= u_i^m \quad \forall\, m = 1 \,..M \\ i &= \lfloor \tfrac{t}{dt} \rfloor \qquad \forall\, t = 0 \,..\, t_f \end{aligned} \tag{4}$$

Where M is the number of actuators (three in this case). For dynamic reaching tasks, the objective function tries to reduce the end effector tracking error at the end of the control horizon. Additionally, a term to minimize the control effort is also added. The optimal policy then can be obtained by optimizing the following nonlinear optimization problem:

$$\Pi(t)^* = \min_\tau \left(\left\| x^{task}_{\frac{t_f}{dt}} - x^{des} \right\|^2 + \sum\nolimits_i u_i^T R\, u_i \right)$$

$$\text{subject to } 0 \leq u_i^m \leq u_{\max}^m \;\; \forall \;\; m = 1 \,..\, M \text{ and } i = \lfloor \frac{t}{dt} \rfloor \tag{5}$$

We use the iterative sequential quadratic programming (SQP) algorithm for solving the optimization problem. In order to reduce the computational time of the optimization problem a dimensionality reduction technique is employed (See [8] for more details). The whole control scheme with the forward model is schematized in Fig. 3. The obtained optimal control policy, however, works only in open loop and depends on the initial conditions. Therefore, it cannot accommodate modelling errors, external disturbances during execution of the policy or for different initial conditions. Yet, we show numerically that there are closed trajectories that would exhibit highly stable motion.

C. Trajectory Generation

The trajectories for manipulator are formulated based on observations that the dynamics of the soft manipulator have similarities to the dynamics of spherical pendula. Forced oscillation experiments on a spherical pendulum showed that planar harmonic oscillations are highly unstable over a major portion of the resonant peak and nonplanar harmonic motions were observed to be stable in a spectral neighborhood above resonance [11]. Hence, we try to derive the control policies for various planar and nonplanar trajectories (See Table I). The trajectories are defined only in the XY plane since it is difficult to define the Z coordinates without an analytical model. Therefore the Z coordinates and orientation of the manipulator are free variables. All the trajectories are roughly centered at the manipulator zero position.

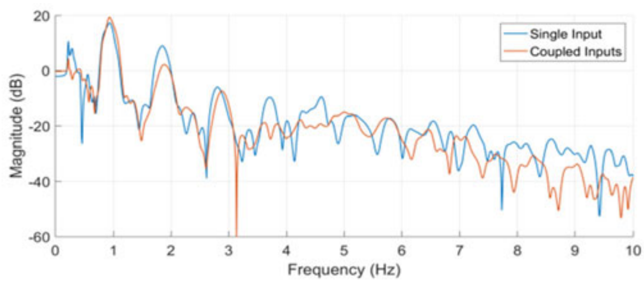

Fig. 4. Frequency response of the manipulator.

TABLE II
TRACKING AND CONTROLLER ACCURACY

	Tracking Error [mm]	Prediction Error [mm]
200 Static Point	21.9 ± 8.5	22 ± 8.2
20 Static Points [8]	46	NA
Line	20.1 ± 25	11.8 ± 6.2
Line with [12]	23	NA
Circle	51 ± 32	35.1 ± 14.8
Fig. 8	21.7 ± 15.3	11 ± 4.9
Hypotrochoid	49.3 ± 23.2	28.5 ± 13.6

For deciding the period of the trajectories, a frequency response analysis of the soft manipulator is performed similar to [12]. Being a highly nonlinear system, the frequency response analysis is only performed to decide the angular frequency ω. Using the MATLAB *tfestimate* function the response of the manipulator to a chirp signal sweeping from 0 to 10 Hz is analyzed using one and two pneumatic chambers (Fig. 4). The amplitude of oscillation of the input signal varies from 0 to 0.9 bars. The system output is defined as the displacement of the end effector. For the multi actuation case, both the chambers are actuated in the same way making them coupled. For both the cases resonance occurred just below 1 Hz. Higher modes are also observable like in linear systems. The period of the trajectories generated are defined just above this resonance frequency by setting ω to 6.25 rad/sec.

For the optimization problem we do not set any additional constraints that help in stabilizing the trajectory like in [7]. It was not even needed to specify a long control horizon (t_f). For all the given trajectories the control horizon was set at 8 seconds. All the control policies are derived with the manipulator at the zero position. Therefore to obtain a periodic behavior, the distal portion of the control policy was repeated to sustain the periodic motion. The period of this repeating signal is 4 seconds for the Fig. 8 trajectory and 3 seconds for all the other trajectories.

IV. Experimental Results

The experimental results are divided into two parts. The first part gives the reader an idea about the advantages and drawbacks of the controller. The second section focuses on the stability of the above mentioned trajectories for the soft manipulator.

A. Controller Accuracy

Since we use an open loop controller, the accuracy of the forward model is vital for accurate motion tracking. Table II evaluates the accuracy of the forward model (as prediction error) and the accuracy of the task execution (as tracking error). The tracking error is heavily reliant on the dynamic constraints of the manipulator for the particular task and can therefore be reduced by careful selection of the target path. The prediction error is also partially dependent on the given task as apparent from the table. Tasks which have higher velocities and higher acceleration values tend to be more inaccurate. This could be due to learning biases incurred due to the low probability of visiting such state spaces during the random exploration process. Another source of error could be attributed to the errors in the tracking and discretization process. At high velocities this could contribute a lot to observed errors. For instance at equilibrium, the manipulator moves at an average speed of 1.3 m/s. At a sampling frequency of 50 Hz, small delays and interpolation errors can therefore create large prediction errors. The tracking error at the end of the control horizon for static points is also mentioned for comparison to our previous works on the same manipulator using a kinematic controller [12] and a dynamic controller [8]. Fig. 5(a) shows the trajectory of the end effector during the Fig. 8 task along with the desired and predicted

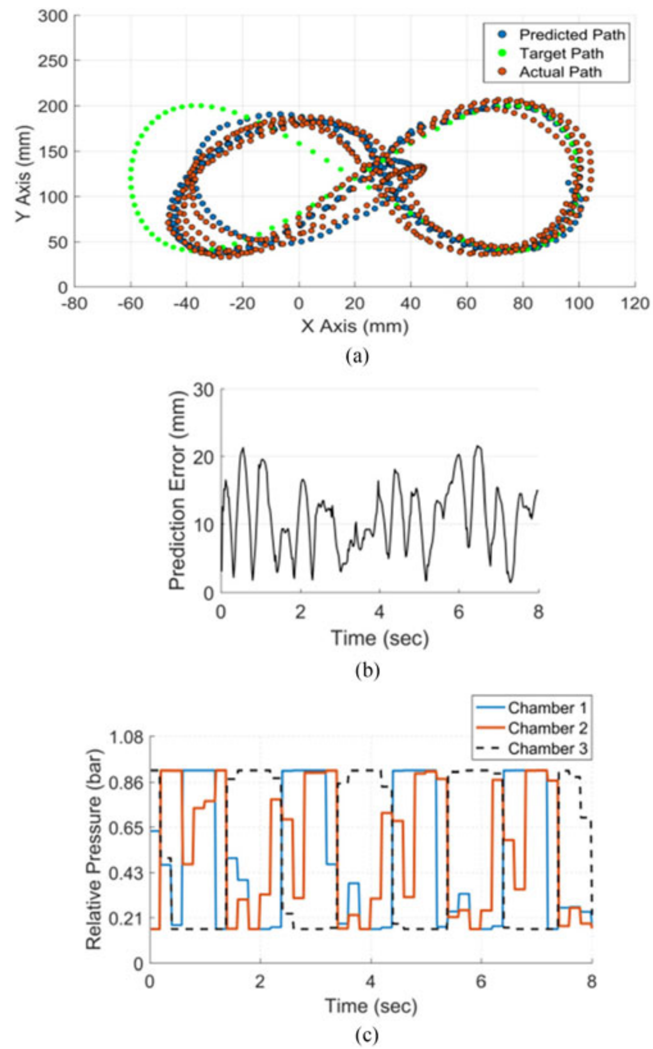

Fig. 5. (a) Trajectory of the end effector during the Fig. 8 task. (b) Prediction error for the complete cycle. (c) Commanded pressure inputs.

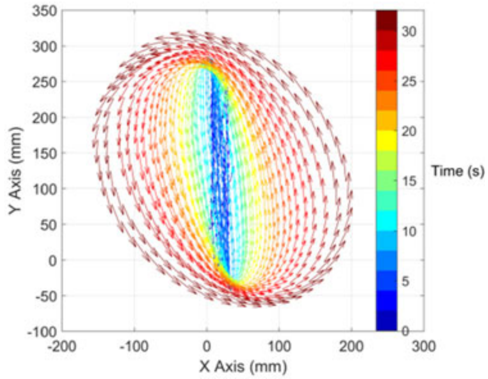

Fig. 6. Chaotic motion of the manipulator observed in the planar task.

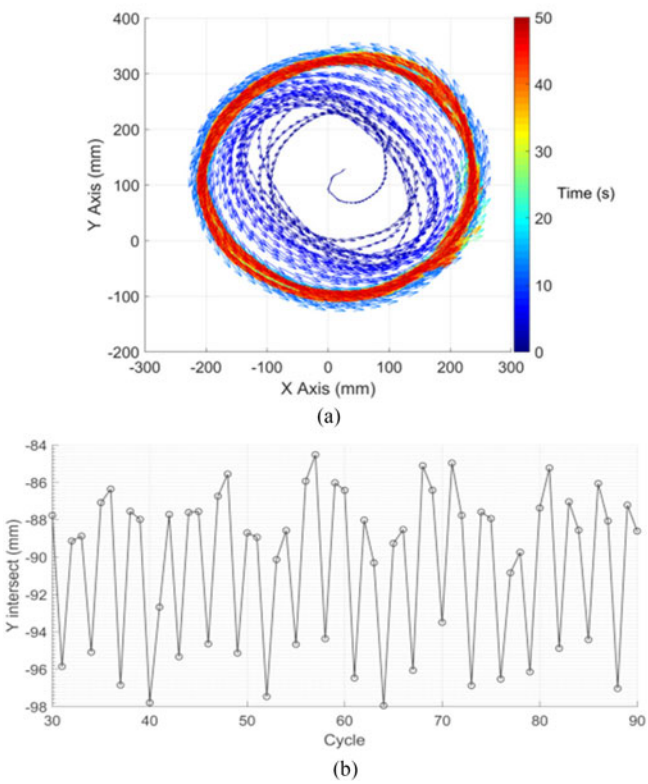

Fig. 7. (a) Long term behavior of the circular motion. (b) Return map obtained at a line draw at $X = 0$.

paths. The commanded pressure inputs to the regulators are given in Fig. 5(c). The step-like solution obtained is because of the dimensionality reduction step mentioned previously in Section III.B. The last 4 seconds of this input signal is repeated to obtain the periodic behavior.

B. Stability Analysis

To examine the long term behavior of the soft manipulator to the derived periodic inputs we perform numerical and graphical analysis of the end effector position over time. For the planar trajectory, the observed motion is very similar to experimental observations on a spherical pendulum [13] (See Fig. 6).

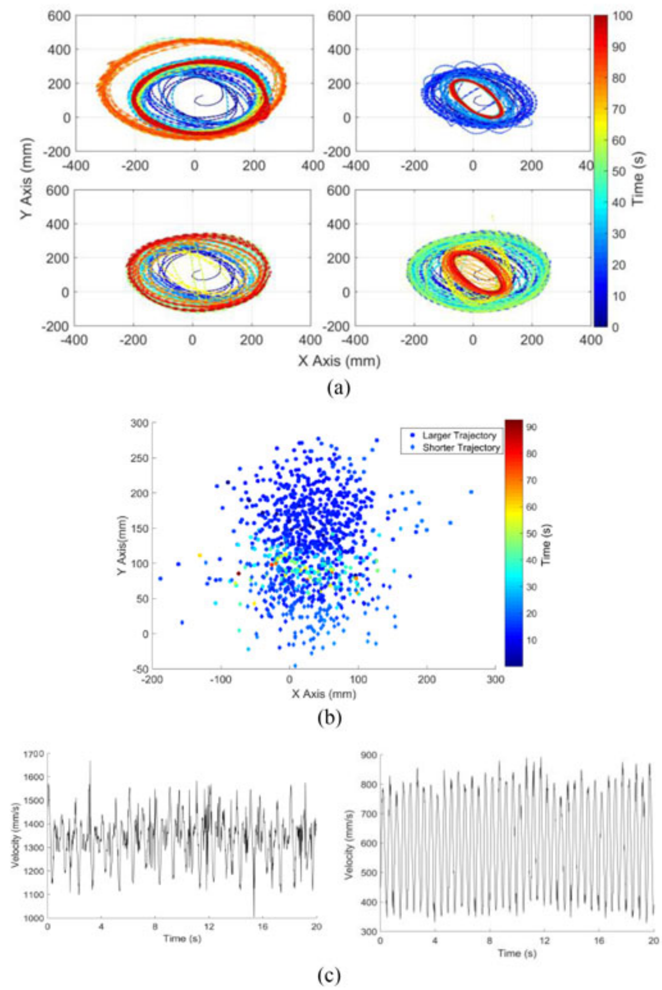

Fig. 8. (a) Two observed limit cycles for the circular task. (b) Region of attraction for the two trajectories observed in the circular task. (c) Velocity plots for the corresponding limit cycles.

Non periodic and non-repeatable motion is observed even when devoid of external disturbances.

The circular task on the other hand converges to a periodic orbit given enough time (See Fig. 7(a)). Observing the return map in the Y coordinate at a line passing through $X = 0$ and moving in the positive X direction (Fig. 7(b)), we can see that the period of the orbit is same as the period of the repeating signal indicating that the cycle is more complex. The repeatability of the motion is also high as observed from the return map. Similar return maps are observed for subsequent trajectories also.

To study the stability of this periodic orbit random external disturbances are applied on the manipulator and the manipulator is allowed to stabilize again. For all the disturbances applied, the manipulator always showed a bi-stable periodic behavior as shown in Fig. 8(a). This was further validated using the learned forward model by observing the long term behavior of the trajectories for different initial conditions (See Fig. 8(b)). The region of attraction is obtained for points throughout the dynamic workspace and the time for convergence is set at the instance the trajectory stays within 1 mm of the final stable trajectory. The velocity plots of the two limit cycles are shown

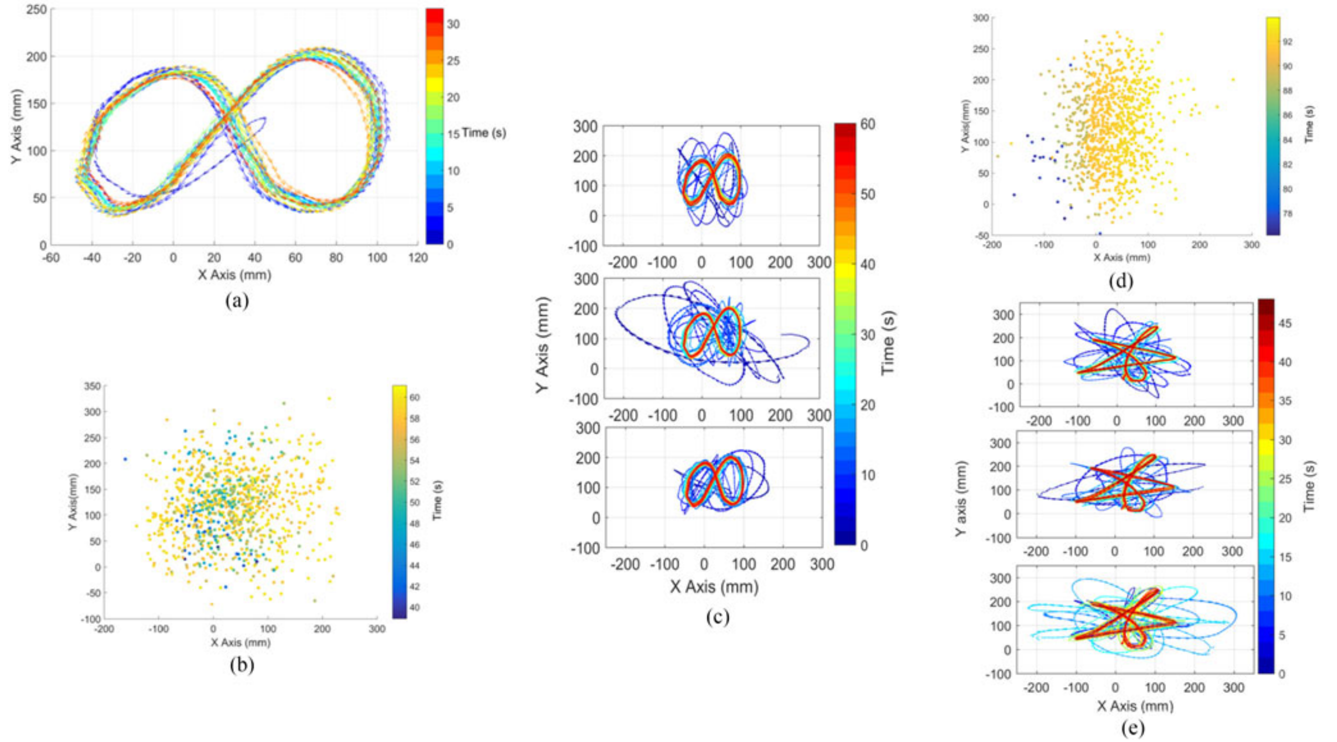

Fig. 9. (a) Undisturbed long term behavior in the Fig. 8 task. (b) Time to convergence for the Fig. 8 trajectory. For all initial conditions, the motion converged. (c) Convergence to the periodic orbit under external disturbances for the Fig. 8 task. (d) Time to convergence for the hypotrochoid trajectory. For all initial conditions, the motion converged. (e) Convergence to the periodic orbit under external disturbances for the hypotrochoid task.

in Fig. 8(b). This behavior is interesting for application where a transition of task is required based on the initial conditions.

For the other two trajectories a high stable periodic orbit is obtained for all the possible conditions that we manually performed (See Fig. 9). Simulation results provide the same results, although the convergence time is higher in simulations. Experimentally, the time taken to return to the stable orbit is also faster compared to the circular task. Therefore it is likely that the apparent high stability of these trajectories could be because they are nonplanar and in a low energy state. Also worth noting is that the actual trajectories are not axially symmetric like in the case of the spherical pendulum. This could be because of asymmetries in manipulator itself or the control policy. Nevertheless, this implies that more complex trajectories can be generated by either modifying the control objective or by design modifications. The repeatability of the periodic orbit is also high and we obtained return maps with the same period as the control policy just like the circular task.

V. Conclusion

In this letter we examine the possibility of stable dynamic control of a soft robotic manipulator without any sensory feedback. Our findings advocate the potential application of soft robotic manipulators for repetitive tasks, where sensory feedback is expensive. They could be ideal in unstructured environments where sensing is difficult and safe interactions are essential. Although the learning process requires a feedback system, this is done offline. It was observed that numerous open loop stable trajectories can be obtained without enforcing any convergence criterion during the trajectory optimization process. We conjecture that this is due to the richness of the manipulator dynamics.

The accuracy of the controller can be evaluated in terms of the prediction error. Even with a very short sampling period (240 seconds) and a single layer recurrent neural network, we were able to achieve long term prediction accuracies in the range of 1–3 centimeters, depending on the task. Higher accuracy can be further achieved by iterative learning techniques specific to the task. Once a desired cycle is obtained the motion is highly repeatable. This means that with fine tuning of the open loop policies, very accurate motions can be obtained. Depending on the trajectory, highly stable motions are observed with very large basin of attraction.

Even though the accuracy obtained is good for a soft robotic manipulator and comparable to other works on open loop stable manipulation [8], there are still some details to be addressed. Our analysis is based on the assumption that the forward model is known and does not change during the task, but in a typical pick and place task, this is not true. This could be solved by learning a model with the object or finding trajectories with invariant end points with added mass. Online learning methods like reinforcement learning could also be employed to do the same. Another important question to be addressed is the role of the manipulator morphology in determining the shape and stability of the trajectories. For this accurate analytical models are essential. Future works would involve development of closed loop controllers which would be essential for accurate global tracking. This could be done through a Model Predictive Control framework provided the optimization time could be reduced or by direct policy learning approaches.

REFERENCES

[1] E. Brown *et al.*, "Universal robotic gripper based on the jamming of granular material," *Proc. Nat. Acad. Sci. USA*, vol. 107, no. 44, pp. 18809–18814, 2010.

[2] S. Collins, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, no. 5712, pp. 1082–1085, 2005.

[3] M. Calisti, E. Falotico, and C. Laschi, "Hopping on uneven terrains with an underwater one-legged robot," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 461–468, Jan. 2016.

[4] S. Kim, J. Clark, and M. Cutkosky, "iSprawl: Design and tuning for high-speed autonomous open-loop running," *Int. J. Robot. Res.*, vol. 25, no. 9, pp. 903–912, 2006.

[5] J. Ackerman and J. Seipel, "Energy efficiency of legged robot locomotion with elastically suspended loads," *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 321–330, Apr. 2013.

[6] F. Ilievski, A. Mazzeo, R. Shepherd, X. Chen, and G. Whitesides, "Soft robotics for chemists," *Angewandte Chemie Int. Edit.*, vol. 50, no. 8, pp. 1890–1895, 2011.

[7] M. Plooij, W. Wolfslag, and M. Wisse, "Open loop stable control in repetitive manipulation tasks," in *Proc. 2014 IEEE Int. Conf. Robot. Autom.*, Hong Kong, 2014, pp. 949–956.

[8] T. G. Thuruthel *et al.*, "Learning dynamic models for open loop predictive control of soft robotic manipulators," *Bioinspiration Biomimetics*, vol. 12, no. 6, 2017, Art. no. 066003.

[9] M. Manti, A. Pratesi, E. Falotico, M. Cianchetti, and C. Laschi, "Soft Assistive Robot for personal care of elderly people," in *Proc. 2016 6th IEEE Int. Conf. Biomed. Robot. Biomechatronics*, Jun. 26, 2016, pp. 833–838.

[10] W. Wolfslag, M. Plooij, R. Babuška, and M. Wisse, "Learning robustly stable open-loop motions for robotic manipulation," *Robot. Auton. Syst.*, vol. 66, pp. 27–34, 2015.

[11] J. Miles, "Stability of forced oscillations of a spherical pendulum," *Quart. Appl. Math.*, vol. 20, no. 1, pp. 21–32, 1962.

[12] G. T. Thomas, F. Egidio, M. Mariangela, P. Andrea, C. Matteo, and L. Cecilia, "Learning closed loop kinematic controllers for continuum manipulators in unstructured environments," *Soft Robot.*, vol. 4, no. 3, pp. 285–296, Sep. 2017, https://doi.org/10.1089/soro.2016.0051

[13] D. Tritton, "Ordered and chaotic motion of a forced spherical pendulum," *Eur. J. Phys.*, vol. 7, no. 3, pp. 162–169, 1986.

[14] Y. Ansari, M. Manti, E. Falotico, Y. Mollard, M. Cianchetti, and C. Laschi, "Towards the development of a soft manipulator as an assistive robot for personal care of elderly people," *Int. J. Adv. Robot. Syst.*, vol. 14, no. 2, 2017, doi 10.1177/172988141668713.

[15] A. Marchese, R. Tedrake, and D. Rus, "Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator," *Int. J. Robot. Res.*, vol. 35, no. 8, pp. 1000–1019, 2015.

[16] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1109–1122, Oct. 2014.