Model Learning With Backlash Compensation for a Tendon-Driven Surgical Robot

Francesco Cursi, *Member, IEEE*, Weibang Bai, Eric M. Yeatman, *Fellow, IEEE*, and Petar Kormushev

***Abstract*—Robots for minimally invasive surgery are becoming more and more complex, due to miniaturization and flexibility requirements. The vast majority of surgical robots are tendon-driven and this, along with the complex design, causes high nonlinearities in the system which are difficult to model analytically. In this work we analyse how incorporating a backlash model and compensation can improve model learning and control. We combine a backlash compensation technique and a Feedforward Artificial Neural Network (ANN) with differential relationships to learn the kinematics at position and velocity level of highly articulated tendon-driven robots. Experimental results show that the proposed backlash compensation is effective in reducing nonlinearities in the system, that compensating for backlash improves model learning and control, and that our proposed ANN outperforms traditional ANN in terms of path tracking accuracy.**

***Index Terms*—Model learning, backlash compensation, tendon-driven robots, minimally invasive surgery.**

I. Introduction

In RECENT years, a major translation in Robotic Assisted Minimally Invasive Surgery (RAMIS) has been towards flexible surgical robots, in order to ensure miniaturization, flexibility, and precision while navigating inside a confined human anatomy and performing complicated surgical tasks. The advancements in design and manufacturing have enabled the possibility to build highly complex structures such as highly articulated and parallel robots [1], [2], continuum robots [3], and soft robots [4]. These structures use complex actuation mechanisms such as tendon sheath, wire drives, or customized joint designs which make them very dexterous, flexible, and well-fitted for navigation during surgical procedures [5], [6]. Despite the promising capabilities of these systems, their applications in real clinical scenarios are still limited due to many limitations, such as lack of effective motion control strategies and proper modelling caused by their high complexities [7].

Manuscript received 28 February 2022; accepted 22 June 2022. Date of publication 30 June 2022; date of current version 6 July 2022. This letter was recommended for publication by Associate Editor N. Figueroa and Editor J. Kober upon evaluation of the reviewers' comments. This work was supported by Engineering and Physical Sciences Research Council (EPSRC) programme grant Micro-Robotics for Surgery under Grant EP/P012779/1. *(Corresponding author: Weibang Bai.)*

Francesco Cursi is with the Hamlyn Centre, Imperial College London, SW7 2BX London, U.K., and also with the Robot Intelligence Lab, Imperial College London, SW7 2BX London, U.K. (e-mail: cursifrancesco@gmail.com).

Weibang Bai and Eric M. Yeatman are with the Hamlyn Centre, Imperial College London, SW7 2BX London, U.K. (e-mail: wbbai@imperial.ac.uk; e.yeatman@imperial.ac.uk).

Petar Kormushev is with the Robot Intelligence Lab, Imperial College London, SW7 2BX London, U.K. (e-mail: p.kormushev@imperial.ac.uk).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2022.3187519, provided by the authors.

Digital Object Identifier 10.1109/LRA.2022.3187519

The main modelling difficulties come from the mapping from actuation to configuration space (such as joint space), caused by the complex transmission, and from configuration space to Cartesian space, due to the highly articulated design.

Different works have focused on analytically modelling and compensating for tendon nonlinearities [8]–[10] by using friction models such as LuGre or Wouc-Ben. Yet, properly modelling these nonlinearities is usually not generalizable, as it requires knowledge about the specific robot design, and it also requires measuring both motor and joint positions, which might be tedious for highly articulated systems.

Baek *et al.* [11] employed computer vision to estimate the joint positions of a single-segment 2-Degree-of-Freedom (DOF) continuum robot from camera images and exploit a friction model to compensate for the system's nonlinearities. Using cameras, however, has the major limitation of being highly affected by light reflection, shadows, and occlusions that might prevent the approach from properly estimating the robot's state.

Other researchers have instead focused on offline modelling approaches, that thus do not rely on any external sensors during the operation. In [12] the authors present a simplified piece-wise linear model to compensate non-linear hysteresis of both backlash and dead zone together. This approach might work only for specific systems where such an approximation is valid. In [13] the authors tackle backlash compensation by estimating the size of the deadzones, removing the deadzone jumps, and modelling the remaining mapping from actuation to configuration space. Similarly to the other works, this approach was validated on a one-DOF robotic catheter only.

Because of the complexities in modelling such systems, researchers have also shifted towards using black-box data-driven approaches. In [14], [15] Long Short-Term Memory Neural Network (LSTM) is employed, given its capability to consider sequences of data for model learning. In [14], LSTM is used in an end-to-end fashion to learn only the forward kinematics of their one-DOF unidirectional catheter and thus evaluate how well it can predict the robot's pose. In [15] it is used still in end-to-end fashion to learn the robot's inverse kinematics, with the model trained on task-specific data. In [16], instead, the authors used feedforward Artificial Neural Networks (ANN) to learn directly the inverse kinematics of their 2-DOF continuum surgical robot and compare the control results with using the pure learned model or an adaptive closed-loop PID feedback controller.

Most of the state-of-the-art work have focused on learning directly the inverse kinematics of their robots, which generally have low DOFs. Learning the inverse kinematics is a challenging task as there exist multiple solutions [17], it does not allow the exploitation of redundancy or including motion constraints [18], makes it more challenging to control surgical robotic tools when in conjunction with serial-link manipulators [19], [20],

and it requires information about the robot pose, which might be unavailable during use due to the lack of sensors, as in the case of robotic surgery. On the other hand, learning the forward kinematics of the robot can be beneficial to overcome these limitations [21], [22].

The common approaches are end-to-end which means that, if applied to tendon-driven robots, the learning technique should be able to also learn an intrinsic compensation of the tendon nonlinearities. Yet, this might make the modelling more tedious and less accurate. Thus far these nonlinearities have generally been neglected or treated as unmodelled noise [23]. Additionally, state-of-the-art learning techniques only model the mapping between the control variables and the robot pose, but neglect information that comes from the differential relationship between positions and velocities. As shown in [24], [25] including the differential relationships can help improving robot model learning, but it has never been tested on highly nonlinear systems like tendon-driven surgical robots. For these reasons, the contributions of this manuscript are:

- employing a Feedforward ANN architecture ($AugNet$), incorporating physical differential relationships during training to model the forward kinematics of a tendon-driven surgical robot;
- proposing an offline backlash identification and compensation approach for highly articulated robots with limited sensor measurements;
- analysing how black-box end-to-end model learning with ANN compares to modelling with additional a priori knowledge of the backlash model and compensation.

Compared to the state-of-the-art research, our proposed approach is here tested on the highly articulated Micro-IGES tendon-driven surgical robot [26]. The a priori backlash compensation is used to limit the effects of time and motion-dependent nonlinearities, aiming to simplify model learning. We are also carrying out an additional comparison to the results obtained when controlling the Micro-IGES robot through LSTM-based inverse kinematics modelling [15].

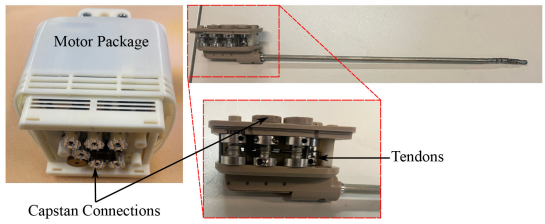
The manuscript is thus structured as follows. Section II presents the Micro-IGES robot, describing its design and kinematic structure. Section III-B describes the method to identify and compensate for backlash in the system. Section IV presents the model learning technique and how it is used to learn the robot kinematic model with and without backlash compensation. Section V shows the robot modelling and experimental control results and, finally, conclusions are drawn in Section VI.

II. Robot Description

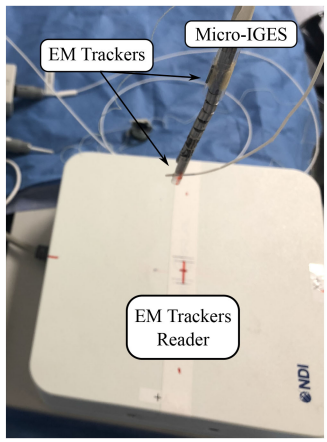
In this section, the Micro-IGES surgical robot is presented, describing its design and its kinematic model.

A. Robot Description

The Micro-IGES (Figs. 1 and 2) is a custom-made surgical robotic tool, composed of a rigid shaft and a flexible section. In total it has 6 DOF for motion control, with the two elbows consisting of a pair of coupled joints. As shown in Fig. 1(a), the robot's base is attached to the motor package by means of pinned connectors. Each motor drives two antagonistic tendons, which have multiple routings on the capstans. The coupling of the two pairs of joints of the elbows occurs directly at the distal driving unit. For a more thorough description, please refer to [15]. The



(a) The Micro-IGES components



(b) The Micro-IGES surgical tool and experimental setup

Fig. 1. The Micro-IGES robot with: 1(a) the robot's components; 1(b) experimental setup.

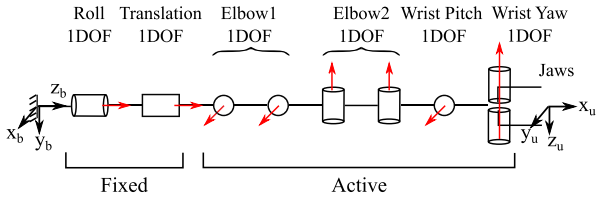


Fig. 2. Micro-IGES kinematic model.

system is not equipped with any sensor, except for the motor encoders in the motor pack. Due to the current setup that prevents the use of the shaft, in this work only 4 DOF are considered (Elbow 1, Elbow 2, Wrist Pitch, Wrist Yaw) and the system's state is hereby expressed by $\boldsymbol{\theta} = \begin{bmatrix}\theta_{e1} & \theta_{e2} & \theta_W & \theta_{or}\end{bmatrix}$. Additionally, due to the designed tendon routing of the robot, backlash effect is predominant only in these joints.

B. Robot Kinematic Model and Control

In tendon-driven robots, tendons are connected to the motors on one end and to the joints on the other. The routing of the tendons inside the robot causes nonlinearities in the mapping from motor to joint space, described as $\boldsymbol{q} = \boldsymbol{f}(\boldsymbol{\theta})$, being $\boldsymbol{\theta} \in \mathbb{R}^4$ the vector of motor positions, $\boldsymbol{q} \in \mathbb{R}^4$ the vector of joint positions, and $\dot{\boldsymbol{\theta}} \in \mathbb{R}^4$ the vector of motor velocities. The control problem for the Micro-IGES can therefore be formulated as a function of the motor values, which can be directly measured and controlled.

Algorithm 1: Offline procedure to identify deadzone sizes for each motor.

procedure IdentifyDeadzones($Data$)
  $t = 0$
  while $t <= T$ **do**
    $\theta_t, \dot{\theta}_t \leftarrow$getMotorValues($Data$)
    $\boldsymbol{P}_t, \dot{\boldsymbol{P}}_t \leftarrow$getTipPosition&Velocity($Data$)
    if $|\dot{\theta}_t| > \epsilon$ **AND** $||\dot{\boldsymbol{P}}_t|| < \epsilon_v$ **then**
      $\theta_{init} = \theta_t$
      $t = t + 1$
      while $t <= T$ **do**
        $\theta_t, \dot{\theta}_t \leftarrow$getMotorValues($Data$)
        $\boldsymbol{P}_t, \dot{\boldsymbol{P}}_t \leftarrow$getTipPosition&Velocity($Data$)
        if $|\dot{\theta}_t| >= \epsilon$ **AND** $||\dot{\boldsymbol{P}}_t|| >= \epsilon_v$ **then**
          $\theta_{fin} = \theta_t$
          break
        end if
        $t = t + 1$
      end while
      $\delta_{set} \leftarrow$ Append $(|\theta_{fin} - \theta_{init}|)$
      $\theta_{set} \leftarrow$ Append (θ_{init})
    end if
    $t = t + 1$
  end while
  return $\delta_{set}, \theta_{set}$
end procedure

Modelling the robot's inverse kinematics is a tedious task due to the non convexity of the problem [27]. If a forward kinematic model outputting the predicted robot's tip position $\hat{\boldsymbol{P}}(\boldsymbol{\theta})$ is available, given a desired Cartesian trajectory specified by $\tilde{\boldsymbol{P}}(t) = \tilde{\boldsymbol{P}}_t, \dot{\tilde{\boldsymbol{P}}}(t) = \dot{\tilde{\boldsymbol{P}}}_t$, the control problem at each timestep t can be formulated at the velocity level as a Quadratic Program such that:

$$\dot{\boldsymbol{\theta}}_t^* = \arg\min_{\dot{\boldsymbol{\theta}}} \frac{1}{2} ||\dot{\tilde{\boldsymbol{P}}}_t - \hat{\boldsymbol{J}}(\boldsymbol{\theta}_t)\dot{\boldsymbol{\theta}}||^2$$
$$\text{s.t. } \boldsymbol{\theta}_m - \boldsymbol{\theta}_t \leq \dot{\boldsymbol{\theta}}\Delta t \leq \boldsymbol{\theta}_M - \boldsymbol{\theta}_t$$
$$\text{and } \quad \boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \dot{\boldsymbol{\theta}}_t^* \Delta t \tag{1}$$

where $\hat{\boldsymbol{J}} = \frac{\partial \hat{\boldsymbol{P}}}{\partial \boldsymbol{\theta}}$, $\boldsymbol{\theta}_M, \boldsymbol{\theta}_m$ are the motor position bounds, and Δt is the sampling time. This formulation allows optimally solving for robot's redundancies and guarantees satisfaction of possible motion constraints.

III. Backlash Compensation

In this section we present how backlash is identified and compensated for.

A. Offline Backlash Identification

The main causes of backlash in tendon-driven robots are effects like friction, tendon elongation, tendon slacking, which cause a delay in the motion of the robot whenever there is a change in motion direction. During the motion reversals, if the system's state is in deadzone, the robot would not move, despite commanding non-null commands. Approaches to compensate for backlash act in general on the motor to joint mapping. However, surgical robots do not have any sensors on the joints, due to sterilization and miniaturization requirements, and accurately measuring the joint positions by means of external sensors may be tedious. Similarly to [13], in this work we use only measurements of the tip position to estimate the backlash size. For highly articulated robots, the contribution of each joint can still lead to zero tip's velocity, even with non-null motor commands. Therefore, in order to identify the size of the deadzone for each motor j of the robot and avoid mis-classifications of deadzones, sinusoidal waves with linearly varying amplitude are commanded to each motor independently for a total amount of time $T = 500$ s. Electromagnetic (EM) trackers are here employed to collect the corresponding 3D tip position $\boldsymbol{P}$, given a desired motor command.

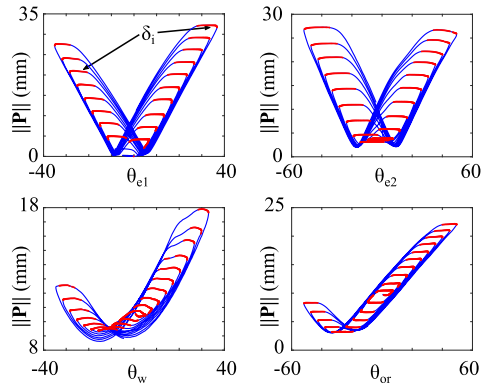


Fig. 3. Backlash identification for each motor, with $\boldsymbol{\theta} = [\theta_{e1} \;\; \theta_{e2} \;\; \theta_W \;\; \theta_{or}]$ corresponding to Elbow 1, Elbow 2, Wrist Pitch, Wrist Yaw. The red dots indicate the identified points in the deadzones. For visualization purposes, the tip position norm is computed neglecting the robot length at its zero configuration.

In this work we are assuming a simplified backlash model, where the deadzones are considered to be only functions of the current robot's configuration. To estimate the sizes of the deadzones, the procedure described in Algorithm 1 is used. We iteratively search through the collected datapoints to find the regions where, despite commanding non null motor velocities ($|\dot{\theta}_j| > \epsilon$), the tip is not moving ($||\dot{\boldsymbol{P}}|| < \epsilon_v$). ϵ and ϵ_v are two user-defined thresholds. Thanks to the design of the Micro-IGES, all tendons pass through the center of the cross-section area, thus minimizing crosstalk and motion interference. For this reason, the identification can be conducted for each motor independently, neglecting any contribution to the tip motion of the other motors. The deadzone regions will be identified by the initial motor position $\theta_{j,init}$, when the change of motion occurs and the tip stops moving, and $\theta_{j,fin}$, when instead the motor goes out of the deadzone and the tip starts moving again. The size of the deadzone at location $\theta_{j,i} = \theta_{j,init}$ is then computed as $\delta_{j,i} = |\theta_{j,fin} - \theta_{j,init}|$. Since the change of motion occurs at different locations, the sizes of the deadzones and the corresponding motor positions are stored in $\delta_{j,set} = \{\delta_{j,i}, i = 1 \dots I\}$ and $\theta_{j,set} = \{\theta_{j,i}, i = 1 \dots I\}$, where I is the number of times deadzones were identified. This procedure is repeated for each motor independently. Fig. 3 shows a represenatation of the identification of the deadzones for each motor.

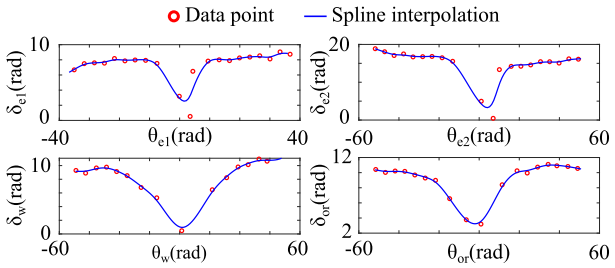


Fig. 4. Mapping from motor values to deadzone size for the four motors $\boldsymbol{\theta} = [\theta_{e1} \;\; \theta_{e2} \;\; \theta_W \;\; \theta_{or}]$ corresponding to Elbow 1, Elbow 2, Wrist Pitch, and Wrist Yaw, as a function of the motor positions.

B. Backlash Modelling and Compensation

Given the two sets $\delta_{j,set}$ and $\theta_{j,set}$, for each motor j it is then possible to construct a function to estimate the size of the deadzone at each motor position $\delta_j = \delta_j(\theta_j)$. In this work we employed cubic splines to find this mapping, and results are shown in Fig. 4. The obtained function allows estimating the size of the deadzone at each possible motor position during the motion. Once the mapping for each motor is obtained, it is then possible to compensate for the backlash by simply adding or removing the deadzone size whenever there a desired change of motion is commanded, meaning:

$$\theta_{j,comp}(t) = \begin{cases} \theta_j(t) - \delta_j(\theta_j(t)), \text{if } \dot{\theta}_j(t) < 0 \wedge \dot{\theta}_j(t-1) > 0 \\ \theta_j(t) + \delta_j(\theta_j(t)), \text{if } \dot{\theta}_j(t) > 0 \wedge \dot{\theta}_j(t-1) < 0 \\ \theta_j(t), \text{ otherwise} \end{cases} \tag{2}$$

IV. Neural Networks for Model Learning

In this section, the approach to model the robot's kinematics by means of Artificial Neural Networks is presented.

A. Feedforward Neural Networks

In this work, a standard ANN ($FFNet$) is employed to directly map the motor positions to the tip Cartesian position $\boldsymbol{\theta} \rightarrow \hat{\boldsymbol{P}}$, where $\hat{\boldsymbol{P}}$ is the network output (Fig. 5(a)). By learning the motor to tip mapping, we inherently learn the motor to joint mapping, which is generally very tedious to retrieve analytically. Even though the mapping from joints to tip could be easily computed from the geometry of the robot, joint data are hard to collect compared to the tip position, which, instead, can be measured directly through external sensors. The network weights are thus computed by minimizing the position loss function $c_p = \sum_{\mathcal{D}} \frac{1}{2} ||\hat{\boldsymbol{P}} - \boldsymbol{P}_{\mathcal{D}}||^2$, where $\boldsymbol{P}_{\mathcal{D}}$ are the measured positions. From the learnt motor to tip position mapping, it is also possible to have an estimate of the Jacobian $\hat{\boldsymbol{J}}$ after the training.

B. Augmented Feedforward Neural Networks

In order to optimally control a robotic system using the techniques described in Section II-B, the forward kinematics of a robot should be computed, obtaining both a mapping from the control variables to the tip position and information about the robot Jacobian. Even though $FFNet$ allows obtaining the Jacobian by derivation, information on the velocity mapping is not included in the training. This might, however, be beneficial for the network to learn a more accurate model. For this reason an augmented network $AugNet$ (Fig. 5(b)) is proposed. $AugNet$ takes as input both the motor positions $\boldsymbol{\theta}$ and velocities $\dot{\boldsymbol{\theta}}$ and outputs both the estimated tip positions $\hat{\boldsymbol{P}}$ and velocities $\dot{\hat{\boldsymbol{P}}}$. The decoupling layer is only used to split the inputs into two sub-vectors. The motor positions $\boldsymbol{\theta}$ are fed into a feedforward layer to estimate the tip position. The derivative layer computes the derivatives of the network, given the current set of weights and inputs, to estimate the Jacobian $\hat{\boldsymbol{J}}$ during the network training. This layer, however, does not add additional weights to the model. The Jacobian is then multiplied by the motor velocities to produce the expected tip velocities $\dot{\hat{\boldsymbol{P}}} = \hat{\boldsymbol{J}}\dot{\boldsymbol{\theta}}$. The cost function to train the network's weights is then defined as $c_{pv} = \sum_{\mathcal{D}} \frac{w_p}{2} ||\hat{\boldsymbol{P}} - \boldsymbol{P}_{\mathcal{D}}||^2 + \frac{w_{dp}}{2} ||\dot{\hat{\boldsymbol{P}}} - \dot{\boldsymbol{P}}_{\mathcal{D}}||^2$, where $\dot{\boldsymbol{P}}_{\mathcal{D}}$ are the measured tip velocities, and $w_p = 10^{-3}, w_{dp} = 1$ are user-defined weights.

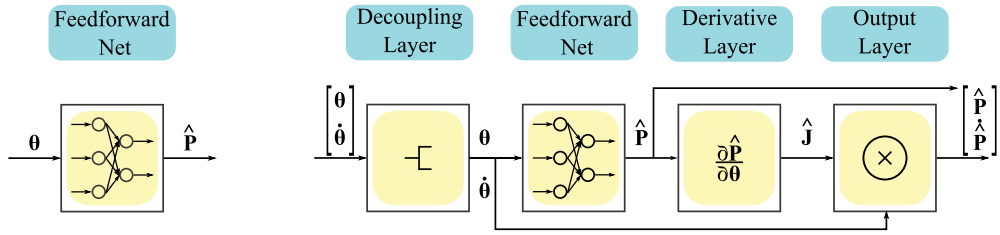
Even if for the training phase $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$, and are needed, for the inference in testing and control it is necessary to only use $\boldsymbol{\theta}$, as only the feedforward layer will be retained and used for estimations of $\hat{\boldsymbol{P}}$.

C. Model Learning and Control Approach

To evaluate the effects of backlash compensation, two different strategies are tested, employing both presented network architectures $FFNet$ and $AugNet$. It is worth mentioning that all the control is performed in open-loop, without any on-board sensory feedback. For this reason, the input to the controller is the desired tip position and the only output is the model's predicted position.

1) Learning and Control Without Compensation: In this case, the mapping from motor to tip is built without compensating the motor positions to send to the robot, but in a pure black-box fashion. Training data is collected without the compensation and consists of the desired motor positions $\boldsymbol{\theta}$ and corresponding measured tip positions $\boldsymbol{P}_{\mathcal{D}}$. Once the kinematic model is learnt, the control strategy in Fig. 6(a) is employed. The learnt model outputs the expected tip position $\hat{\boldsymbol{P}}$ and Jacobian $\hat{\boldsymbol{J}}$, which are then used in the inverse kinematics solver (1) to compute the motor values, given a desired tip position $\tilde{\boldsymbol{P}}$. The resulting motor positions from the solver are then directly commanded to the robot.

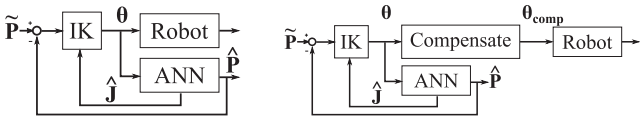
2) Learning and Control With Compensation: In this case, the training data to learn the kinematic model is collected including the backlash compensation described in III-B. Because of the deadzones, non-null desired motor commands would correspond to null tip velocities, making the mapping less univocal and more challenging to invert. The a priori backlash compensation is used to isolate time-dependent nonlinearities, simplify the model and the computation of its derivatives, and reduce the occurrence of uncertain states due to the motor inputs being in the deadzones. The input of the network is still the desired motor positions $\boldsymbol{\theta}$, yet they are first compensated before being sent to the robot and the corresponding tip position $\boldsymbol{P}_{\mathcal{D}}$ is then measured. Similarly to the case without compensation, the learnt model predicts the expected tip position and Jacobian to use in the inverse kinematics solver. Since the network's inputs are the uncompensated desired motor commands, the resulting motor



(a) Standard Feedforward Neural Network ($FFNet$)

(b) Augmented Neural Network ($AugNet$)

Fig. 5. The architectures for the two artificial neural network models: 5(a)) standard feedforward neural network; 5(b)) augmented neural network.



(a) Control without backlash compensation

(b) Control with backlash compensation

Fig. 6. Control strategy for the case: 6(a)) without compensation; 6(b)) with compensation.

TABLE I
AVERAGE DEADZONE SIZES (IN RAD) FOR EACH MOTOR, WITH AND WITHOUT COMPENSATION

Motor Motion	Compensation	Elbow 1	Elbow 2	Wrist Pitch	Jaws
Sinusoidal	**without**	7.28	12.83	7.28	8.95
	with	0.84(-88%)	0.72(-94%)	0.76(-90%)	0.95(-89%)
Exponential	**without**	6.94	13.38	5.15	7.15
	with	6.53(-6.0%)	7.49(-44%)	4.51(-12%)	4.50(-37%)
Random	**without**	8.64	15.54	9.29	11.14
	with	5.10(-41%)	11.49(-26%)	5.28(-43%)	7.85(-29%)

The values in brackets are the percentage reduction with respect to the case without compensation.

positions are first compensated and then commanded to the robot (Fig. 6(b)).

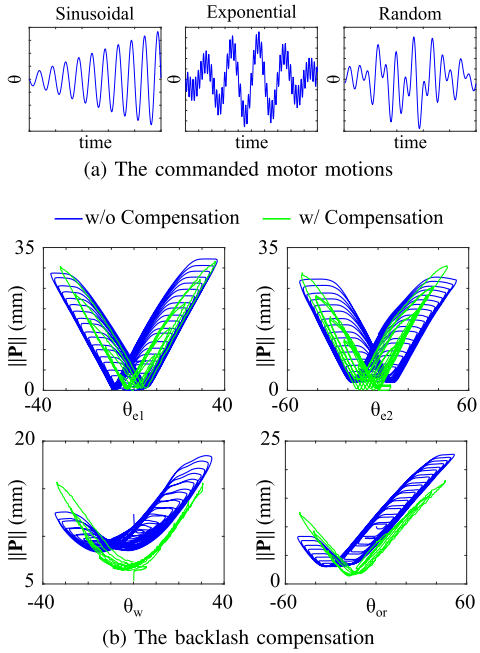
V. RESULTS

In this section, results are presented, showing tests to validate the backlash compensation, the accuracy of the learnt models, and the comparison of the control tasks with the two different learned models.

A. Validation of Backlash Compensation

To validate the proposed backlash compensation, three different kinds of motions were commanded to each motor independently: sinusoidal motion with linearly increasing amplitude (same as the one used to estimate the deadzones), sinusoidal motion with exponentially varying amplitude, random motion (Fig. 7(a)).

Fig. 7(b) shows the comparison of the motion obtained with and without backlash compensation when commanding the sinusoidal motion, whereas Table I reports the sizes of the deadzones identified as described in III-A for each motion when setting $\epsilon = \epsilon_v = 10^{-4}$ for each motor. Results prove that the proposed approach ensures good compensation of the deadzones, thus largely reducing motion lag at each change of direction. The offsets and discrepancies in the tip position are due to the initial



(a) The commanded motor motions

(b) The backlash compensation

Fig. 7. Example of the commanded motions for validating the backlash compensation and comparison of the backlash compensation for $\boldsymbol{\theta} = [\theta_{e1} \;\; \theta_{e2} \;\; \theta_W \;\; \theta_{or}]$ corresponding to Elbow 1, Elbow 2, Wrist Pitch, and Wrist Yaw with sinusoidal motion.

misalignment when running the different tests. In fact, at the beginning of each test the robot needs to be manually brought to the home configuration, and this causes inaccuracies. Some of the overshooting, instead, may be caused by an overestimation of the deadzone size due to the spline approximation of the motor to deadzone function.

B. Robot Model Learning

To generate the data, the Micro-IGES motors are excited with sinusoidal motion of the type $\theta_{u,i}(t) = 0.8\frac{\theta_{u,i,\max}-\theta_{u,i,\min}}{2}(\sin(2\pi\psi\frac{t}{T}) + 0.2\sin(20\pi\psi\frac{t}{T}))$. In order to explore as much as possible of its workspace, each motor is excited with two possible choices $\psi = \{2, 4\}$, resulting in a total of 2^4 combinations, with an additional independent excitation for each single motor also included. In total $2^4 + 4 = 20$ excitations were commanded, each one lasting $T = 40$ s and with a sampling rate $\Delta t = 200$ ms. Consequently, 4000 data points

TABLE II
ROOT MEAN SQUARE ERROR (RMSE) FOR THE TRAINED MODELS IN THE TRAINING AND TEST SETS, FOR BOTH CASE WITH AND WITHOUT BACKLASH COMPENSATION

		$\mathbf{RMSE_P}$				$\mathbf{RMSE_{dP}}$			
		Train		**Test**		**Train**		**Test**	
Model	**Compensation**	**x**	**y**	**x**	**y**	**x**	**y**	**x**	**y**
AugNet	**with**	3.59	2.31	3.55	2.18	1.03	0.73	1.04	0.65
	without	3.21	2.01	3.40	2.06	1.20	0.86	1.22	0.84
FFNet	**with**	3.12	1.77	3.26	1.91	1.38	1.24	1.23	1.25
	without	2.67	1.72	2.60	1.84	1.46	1.16	1.31	1.16

$RMSE_P$ is the error on the tip positions and $RMSE_{dP}$ on the velocities per unit of time. All measures are in mm.

were collected and randomly split into a training (80%) and test (20%) set.

The sampling rate is the same later used for control, and its choice was dictated by the acquisition frequency of the EM trackers. Due to the reduced number of DOF available, only the Cartesian x, y components are controlled and, therefore, modelled. Fig. 1 shows the experimental setup. Two EM trackers are used to collect the tip position data, with a reference sensor at the Micro-IGES base and one on the tip, which allows referring all measurement with respect to the robot's base directly. To learn the forward kinematics, two networks are used to model independently the x and y Cartesian components and, for both $FFNet$ and $Augnet$, the feedforward network consists of one input layer with 4 neurons (corresponding to the robot motor values $\boldsymbol{\theta}$), a single hidden layer with 30 neurons, and a one-dimensional output layer. $sigmoid$ activation function is employed, the training learning rate was set to 10^{-4}, and 20000 epochs were run. These parameters were heuristically chosen on a trial-and-error basis. The same procedure for data collection and the same network architecture are also used for the modelling when no backlash compensation is implemented.

Table II reports the Root Mean Square Error (RMSE) in the training and test sets for each model. Due to the weights in the cost function for $AugNet$, the model accuracy on the tip position is generally lower than $FFNet$. However, since $FFNet$ neglects any information about the derivative of the kinematics, and therefore about the velocity mapping, its error at the velocity level is generally higher. The slightly smaller errors on the positions in the models without compensation may be due to the fact that, when the motors are in deadzones, multiple close motor positions correspond to the same tip position, resulting in a lower prediction error. Yet, without the compensation non-null motor velocities might correspond to null tip velocities, which might be more challenging for the network to capture as demonstrated by the larger errors at the velocity level using $AugNet$.

C. Control Tests

The robot is required to follow four different paths (two ellipses with axes of 12 and 20 mm and two rectangles with sides of 12 and 20 mm) starting from the home straight configuration. We compared the tracking accuracy when employing the learnt $AugNet$ and $FFNet$ models and the currently implemented geometric model based on Denavit-Hartenberg (DH) convention, both with and without the backlash compensation. Fig. 8 plots the comparison of the tracking tasks and the norm of the tracking error $\boldsymbol{\epsilon}_P = \tilde{\boldsymbol{P}} - \boldsymbol{P}_{act}$, where $\boldsymbol{P}_{act}$ is the actual robot tip position measured with the EM trackers. It is worth mentioning that each test was conducted in open-loop and the EM trackers were only used to collect the tip position data.

Table III(a) reports the mean tracking error norm $\bar{\boldsymbol{\epsilon}_P}$ and the improvements in the case with backlash compensation over those without. $AugNet$ is the model resulting in the lowest tracking errors, with a mean of 3.74 mm over the four tests in the case with backlash compensation and 5.13 mm without, compared to 5.88 mm and 5.80 mm for $FFNet$ with and without compensation, and 5.39 mm and 6.35 mm for the DH model (with and without compensation). This means that $AugNet$ results in an overall improvement of 36% over $FFNet$ and 31% over DH model, in the case with backlash compensation. Even though $FFNet$ performs slightly better on the rectangles, it is $AugNet$ to lead to paths more similar to the desired ones, as reported by the ratio of the areas of the tracked paths over the desired ones. Additionally, adding derivatives information during training with $AugNet$ allows having smoother and less jerky motion compared to $FFNet$.

Our results also show that the backlash compensation is effective in improving the robot control. In the control tests without compensation, a large motion lag is noticeable, especially when performing the elliptical shapes, and the robot's paths result to be scaled (Table III(b)). This effect might be caused by the not proper tensioning of the tendons by the motors when commanding the desired motor commands. When employing the backlash compensation, this effect is reduced thanks to the additional rotations of the motors added to compensate for the deadzones. The backlash compensation leads to an overall improvement of 27% for $AugNet$ and 15% for DH model in terms of tracking error with respect to the control without compensation. For $FFNet$, instead, there is a slight deterioration, mostly caused by the poor performance over the second ellipse. With regards to the sizes of the areas, the compensation leads to an overall increase of 60% for $AugNet$, 57% for $FFNet$, and 22% for the DH model. Our results thus show that:

- the implemented backlash compensation helps improving the robot's control, even with traditional DH geometric model;
- adding a priori knowledge of the backlash results in better performance;
- adding differential relationships during training by using $AugNet$ improves model learning and control.

Even though the proposed backlash compensation and learning approach allow for improved modelling accuracy and tracking precision with respect to the currently implemented DH model, the control performances are still not perfect. In fact, most of the tracking errors occur at the beginning of the motion, where understanding if the system is in a deadzone is highly challenging due to manual homing and lack of sensors. Currently, because of its design (Fig. 1(a)), our system does not allow estimating tensions on the tendons due to the lack of sensors and to the fact that each motor drives two tendons, thus making it challenging to properly have an estimation of the tensions on each tendon. Including such information and employing more advanced modelling techniques capable of including more information about the hysteretic behaviour might lead to even further improvements.

D. Comparison to LSTM Learning

Because of the time and motion-dependent nonlinearities in tendon-driven robots, LSTM could be beneficial to learn robot's kinematics. However, if LSTM were used to learn the

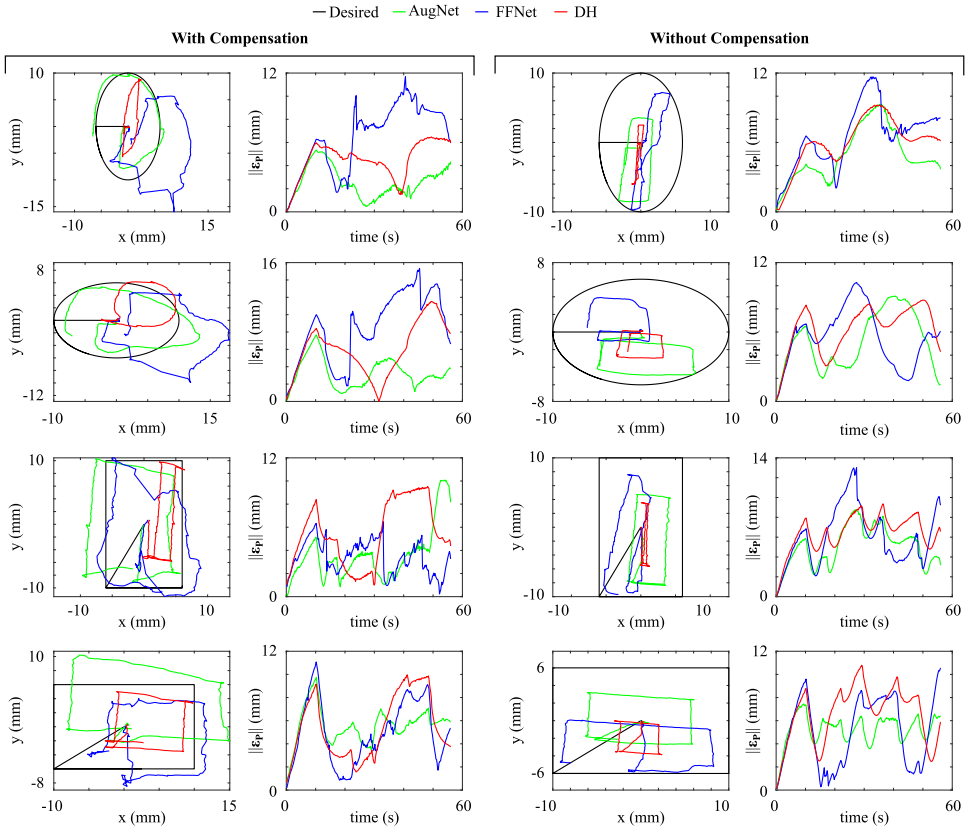


Fig. 8. Control test results on different paths for both cases with and without backlash compensation. ϵ_P is the error between the desired and the actual robot position.

TABLE III
RESULTS FOR THE PATH TRACKING TESTS WITH AND WITHOUT BACKLASH COMPENSATION: III IS THE TIP POSITION ERROR NORM; III IS THE RATIO BETWEEN THE AREAS OF THE ACTUAL AND DESIRED PATHS

(a) Mean tracking error norms $||\bar{\epsilon_P}||$ in mm.

Model	Compensation	Ellipse (12-20)	Ellipse (20-12)	Rectangle (12-20)	Rectangle (20-12)
AugNet	with	2.58 (−48%)	3.27 (−38%)	3.56 (−30%)	5.58 (+6%)
	without	4.96	5.26	5.05	5.26
FFNet	with	6.53 (0%)	8.58 (+55%)	3.41 (−43%)	4.99 (−1%)
	without	6.54	5.54	6.02	5.04
DH	with	4.55 (−25%)	6.01 (−5%)	5.52 (−14%)	5.49 (−17%)
	without	6.04	6.33	6.45	6.58

(b) Ratio (%) between the areas of the actual paths and the desired paths.

Model	Compensation	Ellipse (12-20)	Ellipse (20-12)	Rectangle (12-20)	Rectangle (20-12)
AugNet	with	82 (+57%)	84 (+63%)	89(+62%)	87(+58%)
	without	25	21	27	29
FFNet	with	86 (+72%)	87 (+70%)	89 (+56%)	57 (+28%)
	without	14	17	33	29
DH	with	14 (+12%)	38 (+30%)	20 (+18%)	34 (+28%)
	without	2	8	2	6

The values in brackets are the percentage improvements over the case without compensation.

forward kinematics, then extrapolating the network derivatives for Jacobian computation would be tedious and computationally expensive. In [15] LSTM was employed to directly learn the inverse kinematics of the Micro-IGES on task-specific data and perform Cartesian motion control, thus aiming to inherently learn a backlash compensation. The Micro-IGES is required to follow the 12×20 mm ellipse, the 20×12 mm rectangle, and a 10×10 mm square. The LSTM model-based mean tracking errors $||\bar{\epsilon_P}||$ result to be 3.69 mm, 3.60 mm, and 3.47 mm on the ellipse, rectangle, and square respectively, compared to 2.58 mm, 5.58 mm, and 2.73 mm for $AugNet$ with compensation. The current compensation and modelling technique based on $AugNet$ leads to an improvement in the mean tracking error of 30% and 21% on the ellipse and square paths, but to a less accurate tracking on the rectangle. One of the causes of the larger errors in tracking the rectangle might be due to the use of

a more sparse and less-task specific dataset for training. Future work will focus on comparing how backlash compensation and training on full workspace exploration data benefits an LSTM approach.

VI. Discussion and Conclusion

In this work we analysed how an offline a priori backlash compensation strategy can improve model learning. The proposed backlash compensation technique is employed in order for the ANN to bypass learning the compensation and just build a forward kinematic model that maps from desired motor positions to the robot's tip position. Incorporating the backlash compensation reduces the nonlinearities in the system, simplifying model learning. For better modelling, we employed the novel ANN architecture named $AugNet$ that includes differential relationships during training to additionally learn velocity mapping.

Results on the real Micro-IGES tendon-driven robot show that our proposed backlash compensation strategy is effective at reducing lags in the robot motion and at improving the model learning, leading to more accurate motion tracking. Furthermore, results show the proposed $AugNet$ model outperforms both standard ANN modelling ($FFNet$) and the generally used DH model. Nevertheless, the proposed backlash compensation is still based on some simplified assumptions and it does not manage to fully compensate for the nonlinearities in the system. Future work will focus on improving the compensation by exploring novel architectures, including additional information about motion history, and, eventually, implementing adaptive modelling strategies.

Acknowledgment

For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

References

[1] W. Bai *et al.*, "Anthropomorphic dual-arm coordinated control for a single-port surgical robot based on dual-step optimization," *IEEE Trans. Med. Robot. Bionics*, vol. 4, no. 1, pp. 72–84, Feb. 2022.

[2] B. Chen, Q. Cao, and W. Bai, "A design of surgical robotic system based on 6-DOF parallel mechanism," in *Proc. Int. Conf. Biol. Inf. Biomed. Eng.*, Shanghai, 2018, pp. 1–5.

[3] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1261–1280, Dec. 2015.

[4] M. Cianchetti, C. Laschi, A. Menciassi, and P. Dario, "Biomedical applications of soft robotics," *Nature Rev. Mater.*, vol. 3, no. 6, pp. 143–153, 2018.

[5] W. Bai, Y. Cao, P. Wang, Y. Kobayashi, M. G. Fujie, and Q. Cao, "Development of a new control system for the robotic single port surgery," in *Proc. 11th Anniversary Asian Conf. Comput. Aided Surg.*, 2015.

[6] X. Zhi, W. Bai, and M. Y. Eric, "Kinematic parameter optimization of a miniaturized surgical instrument based on dexterous workspace determination," in *Proc. IEEE Int. Conf. Adv. Robot. Mechatron.*, 2021, pp. 112–118.

[7] O. M. Omisore, S. Han, J. Xiong, H. Li, Z. Li, and L. Wang, "A review on flexible robotic systems for minimally invasive surgery," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 52, no. 1, pp. 631–644, Jan. 2020.

[8] W. Hong, A. Schmitz, W. Bai, P. Berthet-Rayne, L. Xie, and G. Z. Yang, "Design and compensation control of a flexible instrument for endoscopic surgery," in *Proc. - IEEE Int. Conf. Robot. Automat.*, 2020, pp. 1860–1866.

[9] Z. Sun, Z. Wang, and S. J. Phee, "Elongation modeling and compensation for the flexible tendon - sheath system," *IEEE/ASME Trans. Mechatron.*, vol. 19, no. 4, pp. 1243–1250, Aug. 2014.

[10] W. Xu, C. C. Poon, Y. Yam, and P. W. Chiu, "Motion compensated controller for a tendon-sheath-driven flexible endoscopic robot," *Int. J. Med. Robot. Comput. Assist. Surg.*, vol. 13, no. 1, 2017, Art. no. e1747.

[11] D. Baek, Y.-H. Nho, and D.-S. Kwon, "ViO-Com: Feed-forward compensation using vision-based optimization for high-precision surgical manipulation," *IEEE Robot. Automat. Lett.*, vol. 7, no. 1, pp. 263–270, Jan. 2022.

[12] D. H. Lee, Y. H. Kim, J. Collins, A. Kapoor, D. S. Kwon, and T. Mansi, "Non-linear hysteresis compensation of a tendon-sheath-driven robotic manipulator using motor current," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 1224–1231, Apr. 2021.

[13] T. Poignonec, P. Zanne, B. Rosa, and F. Nageotte, "Towards in situ backlash estimation of continuum robots using an endoscopic camera," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 4788–4795, Jul. 2020.

[14] D. Wu, Y. Zhang, M. Ourak, K. Niu, J. Dankelman, and E. V. Poorten, "Hysteresis modeling of robotic catheters based on long short-term memory network for improved environment reconstruction," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2106–2113, Apr. 2021.

[15] W. Bai *et al.*, "Task-based LSTM kinematic modelling for a tendon-driven flexible surgical robot," *IEEE Trans. Med. Robot. Bionics*, vol. 4, no. 2, pp. 339–342, May 2022.

[16] Z. Wang *et al.*, "Hybrid adaptive control strategy for continuum surgical robot under external load," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 1407–1414, Apr. 2021.

[17] D. Nguyen-Tuong and J. Peters, "Model learning for robot control: A survey," *Cogn. Process.*, vol. 12, no. 4, pp. 319–340, 2011.

[18] F. Cursi, V. Modugno, L. Lanari, G. Oriolo, and P. Kormushev, "Bayesian neural network modeling and hierarchical MPC for a tendon-driven surgical robot with uncertainty minimization," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2642–2649, Apr. 2021.

[19] F. Cursi and P. Kormushev, "Pre-operative offline optimization of insertion point location for safe and accurate surgical task execution," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2021, pp. 4040–4047.

[20] F. Cursi, W. Bai, E. M. Yeatman, and P. Kormushev, "Optimization of surgical robotic instrument mounting in a macro–micro manipulator setup for improving task execution," *IEEE Trans. Robot.*, to be published, doi: 10.1109/TRO.2022.3171097.

[21] F. Cursi, G. P. Mylonas, and P. Kormushev, "Adaptive kinematic modelling for multiobjective control of a redundant surgical robotic tool," *Robotics*, vol. 9, no. 3, Aug. 2020, Art. no. 68.

[22] C. Salaün, V. Padois, and O. Sigaud, "Control of redundant robots using learned models: An operational space control approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 878–885.

[23] K. Chin, T. Hellebrekers, and C. Majidi, "Machine learning for soft robotic sensing and control," *Adv. Intell. Syst.*, vol. 2, no. 6, 2020, Art. no. 1900171.

[24] F. Cursi, D. Chappell, and P. Kormushev, "Augmenting loss functions of feedforward neural networks with differential relationships for robot kinematic modelling," in *Proc. 20th Int. Conf. Adv. Robot.*, 2021, pp. 201–207.

[25] F. Cursi, W. Bai, W. Li, E. M. Yeatman, and P. Kormushev, "Augmented neural network for full robot kinematic modelling in SE(3)," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7140–7147, Jul. 2022.

[26] J. Shang *et al.*, "A single-port robotic system for transanal microsurgery–design and validation," *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1510–1517, Jul. 2017.

[27] M. I. Jordan and D. E. Rumelhart, "Forward models: Supervised learning with a distal teacher," *Cogn. Sci.*, vol. 16, no. 3, pp. 307–354, 1992.