university of groningen / faculty of mathematics and natural sciences / artificial intelligence

Dynamic Modeling of a P(VDF-TrFE-CTFE)-Composite Soft Actuator

Master's Research Project
(Computational Intelligence)

Niklas Erdmann (s2869896)

July 9, 2022

Internal Supervisor(s): Prof. Dr. Raffaella Carloni (Artificial Intelligence, University of Groningen)
Prof. Dr. Herbert Jaeger (Artificial Intelligence, University of Groningen)
Riccardo D'Anniballe MSc. (Artificial Intelligence, University of Groningen)

Artificial Intelligence
University of Groningen, The Netherlands

Table of Contents

Acknowledgements

Due to this project being a rather lengthy and work-intensive endeavour, I would like to use this opportunity to express some appreciation for the people who made it possible:

First, a very big thanks for a relaxed and close work atmosphere and never ending support to Riccardo D'Anniballe. A large thanks for providing helpful directions, trust and needed criticism to Raffaella Carloni and Herbert Jaeger. Additionally, I would like to extend an even larger thanks to Lauren Hansen-Manguikian, who forced me to actually finish it. I would also like to thank Lauren Hansen-Manguikian, Sebastian Prehn, Lars Cordes, Kimberly Wickborn and Wouter Kruijne, who, despite being incredibly busy, somehow had enough time for motivational or emotional support and editorial comments. Lastly, I would like to thank my parents for being so supportive and patient with a child that seemingly can go on studying forever.

university of groningen

faculty of mathematics and natural sciences

artificial intelligence

Abstract

Soft actuators, in contrast to traditional actuators, are made from soft and compliant materials. A category of soft materials are electro-active, showing a response to an applied electric field. Electrostrictive materials, a type of electro-active material, have a quadratic response between the applied electric field and the strain they produce. This project attempts to model an electro-active soft actuator, composed of an active layer of electrospun aligned nanofibers made from the P(VDF-TrFE-CTFE) electrostrictive polymer integrated in a PDMS silicone matrix. The actuator features a passive, Kapton® layer and electrodes made from a conductive carbon powder silicone mixture. The soft actuator shows complex dynamic responses to electric field stimuli and natural individual differences between samples. These complications motivate a novel dynamical modeling attempt, employing an Echo State Network. The trained model, tested on new data, displays a normalized root mean square error (non-normalized in parenthesis) of 0.433 (0.106 mm) and 0.280 (0.019 mN) towards modeling tip deflection and blocking force exertion respectively. To show applicability and transferability of the model, the blocking force network is tested in a more functional context: A two actuator gripper. Despite noisy task and observational settings, the modeling and the gripper experiment suggest Echo State Networks work very well with electrostrictive soft actuators.

1 Introduction

Soft robotics is a relatively new and emerging field, deploying compliant (elastic) appliances to robotic settings. Robotics deals with the interaction and manipulation of an environment by a robot or actuator and soft robotics builds on this with a more biologically inspired manipulation. This perspective can provide significant advantages over traditional methods (Manti, Cacucciolo, & Cianchetti, 2016): Compliant robots may have a higher adaptability to a changing environment and are naturally safer for human-machine interactions. In addition, compliant mechanisms employed in robotic systems are able to act more like biological organisms, as organic mechanisms tend to be soft by default. This might even increase the potential intelligence of a system. This proposed effect is called embodied intelligence, and states that as environmental interactions become less rigid and restricted, more intelligence can be displayed by the system (Cangelosi, Bongard, Fischer, & Nolfi, 2015).

A major area of soft robotics is dedicated to the development and testing of soft actuators as the central robotic component enabling movement and interaction (El-Atab et al., 2020). The field features a wide range of fabrication designs and material variations, as compliance can be achieved with a diverse range of approaches. While approaches may vary widely, they are always reliant on a stimulus to produce a response (e.g. air pressure, temperature, magnetic fields, or pH value). Soft actuators are multi-domain dynamic systems, meaning they display electrical, mechanical, chemical and thermal interaction effects which are difficult to combine, or approximate. Electric-responsive soft actuators are a major category within soft actuators. This project focuses on a dielectric elastomer based system which exerts a strain when applying an electric field, hence displays behaviour as a reaction to the application of an electric field. The actuator in question is fabricated from electrostrictive materials, that show a quadratic relation between the strength of the electric field applied on them and their strain response.

As observable in other approaches (Gu, Gupta, Zhu, Zhu, & Zhu, 2017; Moss, Krieg, & Mohseni, 2021; Zhou et al., 2019) using dielectric soft actuators, to model and control an actuator's behaviour is an ongoing challenge. The soft materials used do not exhibit the same rigid and fixed degree of freedom structure as regular, stiff robotic actuators, but are visco-elastic. Often, soft actuators may also display time-dependent differences in behaviour. Soft actuators, similar to the one in this project, which use the mechanical strain induced by an external electric field, have often been modeled with finite element analysis (Chee, Mah, & Ali, 2016; Kuhring, Uhlenbusch, Hoffstadt, & Maas, 2015; McGough, Ahmed, Frecker, & Ounaies, 2014) or with analytical models (Zhou et al., 2019; Youn et al., 2020; Zhao et al., 2018). The inherent non-linearity and time-dependent differences in behaviour may explain why such modeling can be inherently demanding when using an analytical model. As such, data driven black box models may be a more flexible choice compared to analytical modeling. Due to a data-based approach drawing from underlying observations, it is possible to take material- and fabrication-related differences into account and even discern between actuators, which would not be possible with an analytical model. While analytical models require both insight into underlying effects and experimental data to validate the results, data driven methods can be used with sufficient experimental data. Data driven methods can also be more efficient in deployment after initial training, as only the mathematical function of the observed behaviour is approximated.

In analytical models, mechanics and behaviour are simulated based on their properties and relationships of properties, which is inherently more demanding. Data driven methods have been frequently used to try approximate actuator responses given prior observations of said responses (a selection of these are given in Table 1, where used actuator types, model architectures, sensors and data types are listed).

Actuators might also feature natural variability in their behaviour across different samples of the same actuator design model. This can depend largely on the fabrication method and/or material types employed in the construction of said actuators. Further, as the physical properties of actuators tend to often still be under development, they will show variability between different versions of the actuator. These differences prove to be natural challenges in terms of accurately modeling behaviour for analytical models. However, differences in behaviour can be captured in observational data and thus modeled in data driven methods. Data driven methods require a lot of diverse observations to arrive at a satisfactory estimation of an actuators behaviour. On the other side, analytical models also require observational data to compare their results to. Further, actuator samples need to be tested in terms of their respective performance anyways, providing opportunity for data gathering.

Table 1 shows an attempted taxonomy of current projects related to soft actuators and soft actuator modeling. From this table, it is clear that the majority of approaches are concerned with exact positional control/path tracking of their given actuators. Further, a majority of actuators currently employed in the field are pneumatic. This may be because pneumatic actuators require less sophisticated construction methodology. This enables the design of relatively more complex systems with overall higher blocking force exertion or actual 3D navigation. These factors may make development of modeling and control mechanisms related to such actuators more practical. Similarly, pneumatic and cable-based actuators feature actual path tracking, while dielectric polymer based research focuses more on one dimensional positional estimation. That being the case, dielectric polymer actuators are exclusively using recurrent neural networks for their analysis, perhaps related to the prominent need to model dynamical data.

While it is possible to compare attributes of soft actuators, i.e. being multi-domain dynamic systems, many soft actuation approaches in the literature are inherently different from each other and are not easily generalizable. This is usually because they operate via exploiting different physical phenomena. For instance, the behavior of moving segments in an air inflated, segmented arm (Jiang et al., 2017) differ from that of a polymer based actuator (Shintake, Schubert, Rosset, Shea, & Floreano, 2015) which in turn differs from that of a cable driven approach (Giorelli et al., 2013). Moreover, there are even fibrous actuators, aimed to behave like artificial muscles (Gotti, Sensini, Zucchelli, Carloni, & Focarete, 2020). Actuators can also be of different sizes, feature different amount of joints and generally be designed to fit into different roles or be more adept at specific purposes. These differences cause different actuators to feature high variation in method and domain of application, leading to diverse approaches and projects within the field.

Application of data driven models in terms of modeling the behaviour of a soft actuator with a neural network is often only oriented towards achieving applicable performance within a setting at hand. Few efforts are made to facilitate comparison of data driven analysis across research projects. This does require some additional considerations to guarantee comparability between approaches across actuators. Comparable error terms would be necessary, as well as a more unified/standardized

Table 1: Data driven architectural choices in different soft robotic actuators. Explanation for Architecture terms: Feed Forward Neural Network (FNN), Regression (Reg), Recurrent neural network (RNN). Disp. refers to measures of actuator tip displacement.

Architecture	Authors	Year	Actuator	Datatype	Sensors	Control target
FNN	Giorelli, Renda, Ferri, and Laschi	(2013)	Cable	Disp.	Simulation	position control
FNN	Giorelli et al.	(2015)	Cable	Disp.	Camera	path tracking
FNN	Jiang et al.	(2017)	Pneumatic	Disp.	Camera	pose and path tracking
FNN, Reg	Elgeneidy, Lohse, and Jackson	(2018)	Pneumatic	Disp.	Camera	bending angle prediction
FNN	Braganza, Dawson, Walker, and Nath	(2007)	Pneumatic	Disp.	Camera	path tracking
FNN, RNN	Melingui, Lakhal, Daachi, Mbede, and Merzouki	(2015)	Pneumatic	Disp.	Camera	path tracking
Reg, FNN	Reinhart, Shareef, and Steil	(2017)	Pneumatic	Disp.	Camera	position and path tracking
RNN	Thuruthel, Falotico, Renda, and Laschi	(2017)	Cable	Disp.	Simulation	path and force tracking
RNN	Truong and Ahn	(2015)	Dielectric	Disp.	Laser	position control
RNN	D'Anniballe, Paoletta, and Carloni	(2021)	Dielectric	Disp., Force	Camera, Load Cell	position control
RNN	Thuruthel, Shih, Laschi, and Tolley	(2019)	Pneumatic	Disp., Force	Camera, Load Cell	path tracking
RNN	Jiang, Li, and Wang	(2022)	Dielectric	Disp	LVDT Sensor	position control
RNN	Xiao, Wu, Ye, and Wang	(2020)	Dielectric	Disp	Laser	position control

task, observation setting, and data preprocessing methodology.

Given the current state of the field, one can argue that even though many papers already use data driven techniques for modeling, there is still a need for more unifying investigations. The formalization of processes and aims of designing a data driven model for a soft actuator may also enable clearer comparisons in modeling across actuators.

While there are many articles examining actuators in conjunction with data driven methods, not many of the actuators are comparable to the performance or behavior of the P(VDF-TrFE-CTFE)-composite actuator examined here. The aim of this project is to show the value in using a data driven approach of an Echo State Network with this specific actuator. By providing a step by step description of this process, this project also aims to provide a basis for a standardized methodology that could be used by future projects modeling soft actuators.

In order to provide practical evidence for the viability and flexibility of data driven models, the behavioral observations and model insights are applied in a functional context. To do this, a simple two actuator gripper is designed, capable of lifting a small polystyrene cube. The Echo State Network is then used to approximate the electric field needed to successfully grasp the object. This demonstrates the transferability and ease of adopting this type of modeling in practice.

Specifically, this project is answering the question whether modeling behaviour of a P(VDF-TrFE-CTFE)-composite actuator is possible with an Echo State Network. This includes all necessary considerations and steps, starting with a physical actuator, gathering behavioural data, and ultimately arriving at a simple application of the same model in form of a gripper made from the same actuators. The network is being used to dynamically model tip deflection and blocking force exertion of the actuators from the same step in time, based on the applied electric field strength. Similarly, the gripper control is established by estimating the required input electric field strengths using a model trained on previously observed blocking force data. The performance of the model on tip deflection and blocking force fitted to novel observations, is assessed based on a comparison to a baseline linear regression. Additionally, the modeling of gripper input voltages is assessed based on differences to observed input voltages during a successful lift of a cube.

Parts of this project, namely the methods and result sections related to blocking force and tip displacement, are being simultaneously published as contribution to the 2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) (D'Anniballe, Erdmann, Selleri, & Carloni, 2022). Sections explaining mechanical properties and fabrication of the soft actuator are directly derived from Riccardo d'Anniballe's work.

2 Theory

2.1 Soft Actuators

The content of this project aims to further the development of a soft actuator (D'Anniballe, Zucchelli, & Carloni, 2021, 2022; Burawudi, D'Anniballe, Langius, & Carloni, 2021; Carloni, Lapp, Cremonese, Belcari, & Zucchelli, 2018) acting as cantilever beam. This actuator consists of a mat of aligned nanofibers made from P(VDF-TrFE-CTFE), integrated in a polydimethylsiloxane (PDMS) matrix. The nanofibers produce an electric field induced strain, resulting in a deformation of the material. The actual actuator consists of a combination of an active layer (integrated nanofibers), inter-

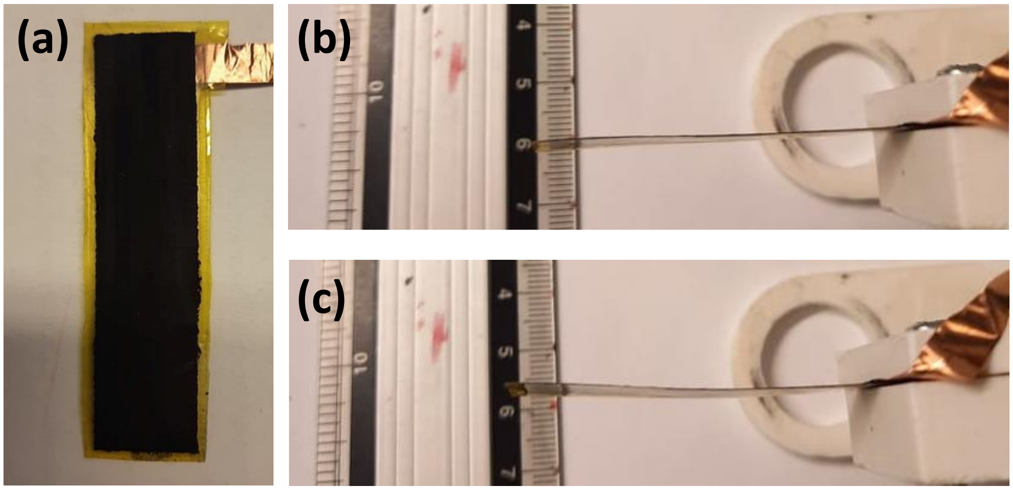


Figure 1: a) Actuator sample as seen from above. b) Same actuator sample as seen in a), fastened in an experimental setup. c) Actuator sample, now being exposed to an electric field, causing it to display a deflection at its tip.

leaved between two electrodes of PDMS and carbon powder, and a passive layer (adhesive/Kapton® tape) (Carloni et al., 2018). Contrary to previous work, the nanofibers used here are immersed in PDMS silicone to obtain stable electric properties while maintaining the actuator elasticity (L. Liu et al., 2020). With the actuators dimensions of 70x20x0.06 mm, one can imagine the material as looking and behaving similar to a piece of tape. For an impression of a single soft actuator, see Figure 1. A cross section representation of a soft actuator can be found in Figure 2a. When exposed to an external electric field, the P(VDF-TrFE-CTFE) fibers produce a strain along their longitudinal axes due to the synergistic effects of Maxwell stress and electrostriction. Maxwell stress represents the electrodes in the material attracting each other which causes a compression in the thickness direction and an expansion in the longitudinal direction. Electrostriction is an electric field-induced conformation change of the polymeric chains as a result of a realignment of dipoles within the nanofibers. This produces a large strain in the thickness direction. During the nanofiber mat's response to an electric field, the Kapton® tape (the passive layer) causes the material to bend in a specific direction. This is due to the passive layer causing a directed imbalance in the stiffness of the actuator. This effect can also be seen in Figure 1: The actuator bends always to the side which is lacking a passive layer, which is ”up” in the picture. More formally, the total strain S induced in the P(VDF-TrFE-CTFE) nanofibers as response to an applied electric field E can be described as follows:

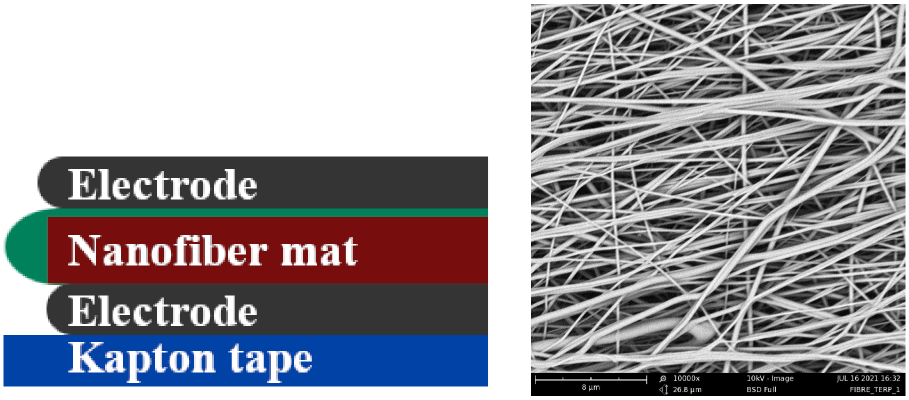
$$S = S_{Maxwell} + S_{electrostriction} = -\frac{1}{2}\frac{\varepsilon_0\varepsilon_r E^2}{Y} + QP'^2 \tag{1}$$

where Y is the P(VDF-TrFE-CTFE) Young's modulus, representing its resistance to deformation under load. ε_r is the polymer dielectric constant, which is the electric permittivity of the polymer.

In other words, ε_r is the ratio of charge stored in an insulating material between two electrodes in relation to amount of charge stored when the insulator is a vacuum. ε_0 is the vacuum permittivity, so the amount of charge stored with a vacuum as insulator. Q is the electrostrictive coefficient, denoting the strength of the electrostrictive effect. Lastly, P' is the phase transformation-induced polarization of the material, i.e.:

$$P = D\varepsilon_0 E \tag{2}$$

In which D represents the electric displacement field, describing the displacement effects of an electric field on the charges within a dielectric material.



(a) Cross section of Actuator Layers. The red nanofiber mat is immersed in green PDMS.

(b) SEM picture of the electrospun P(VDF-TrFE-CTFE) aligned nanofibers.

Figure 2: A representation of actuator layers and an SEM picture of the P(VDF-TrFE-CTFE) nanofibers.

Fabricating a single actuator is a multi-stage process, involving the electrospinning of the nanofiber mat, immersion of the mat in PDMS, adhering the PDMS/ carbon powder electrodes and lastly attaching the Kapton® tape. The electrospinning of the nanofiber mats and the subsequent fabrication of the actuator samples are performed by members of the University of Groningen, Robotics Lab (`https://www.roboticsresearchlab.nl/`) in the framework of the MAGNIFY Project (`https://www.magnifyproject.eu/`).

All of these steps are dependent on thorough and practiced handling in order to arrive at a satisfactory product and thus likely to introduce some variation between actuator samples. Especially the electrospinning process is prone to produce fiber mats of varying alignment, confounding later actuator behaviour. This phenomenon can be observed in Figure 2b, which was obtained with the Phenom ProX Desktop scanning electron microscope (SEM) (Thermo Fisher Scientific, Waltham (MA), USA, `www.thermofisher.com`).

Variations between actuator samples are not simply captured in analytical approaches, as they consider each actuator to be an identical specimen. However, differences can be observed and hence related based on the experimentally observed behavioural data.

2.2 Echo State Networks

Soft actuators exhibit nonlinear behaviour, which may be modeled analytically, but also by a black box, or data driven approach. The term black box includes a wide array of methods geared towards approximating the mathematical function underlying the relationships between a set of observations. While there are a lot of solutions fitting this description, they are often designed differently depending on the type of observation, or data being examined. In other words, the model type is dictated by the task. In this case, the variable of interest (actuator behaviour observations) show nonlinear dynamical relationships with temporal dependencies, for which Recurrent Neural Networks (RNN) are a popular choice. RNNs are a class of neural networks which are distinctive through their use of recurrent connections between neurons. In contrast to conventional feed-forward networks in which connections only go in one direction, RNNs may have connection in several directions and within themselves. Through the subsequent recurrent activation of their neurons, RNNs are able to sustain concurrent activation across time. Due to this, they are able to retain activation partially independent from the current input. This makes it possible to model different events in time with the assumption that they are not independent from each other and even keep a dynamical memory of past events. Because of this, a RNN is considered a universal approximator of dynamical systems (Funahashi & Nakamura, 1993). On the other side, one of the most prominent problems with RNNs is their reliance on gradient descent learning methods. Specifically, convergence can be heavily influenced by the presence of bifurcations (Doya, 1993). Additionally, RNNs often use many different hyper-parameters and as each training phase is computationally heavy to complete, optimization can be a rather tedious and inefficient procedure.

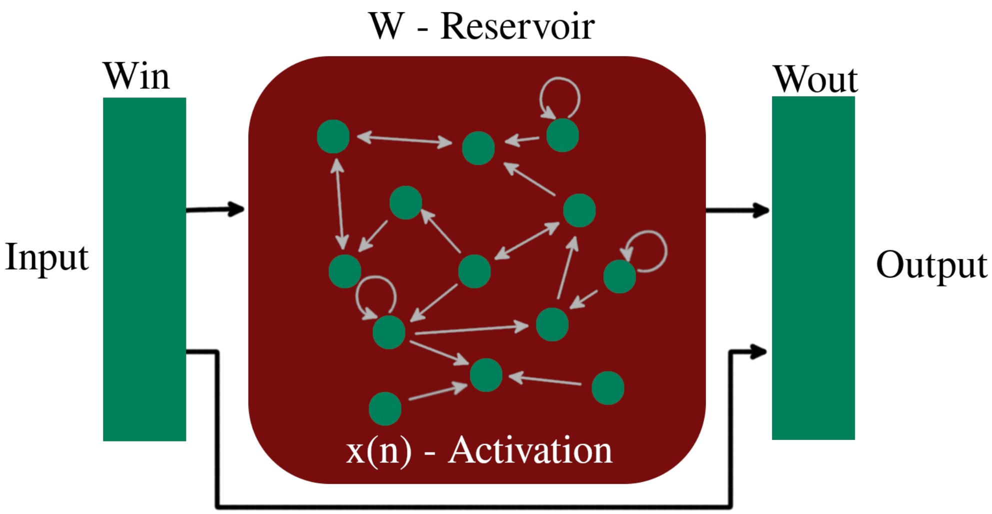


Figure 3: Representation of an Echo State Network.

Echo State Networks (ESN) (Jaeger, 2001) have been developed to remedy past problems with RNNs while still being able to accurately model nonlinear dynamical systems. During the time of ESNs invention, RNN convergence in terms of gradient descent was heavily influenced by the presence of bifurcations (Doya, 1993). Although this shortcoming in RNNs is no longer of concern, ESNs remain a system combining simplicity of use as well as fast and computationally light training with high performance. Reservoir Computation methods (in which ESNs are included in) are distinct from other Neural networks in that they consist of a random hidden layer of interconnected, recurrent neurons, also called a reservoir or liquid. The activation of that layer is then interpreted and used to model the dynamical relationship between input and output. See Figure 3 for a representation of a simple ESN. This project focuses on ESNs, therefore further descriptions of the reservoir computing approach will be made from that particular perspective.

ESNs are designed to match their output $y(n)$ to a target output $y^{target}(n)$, thereby n is representing each discrete time-step $n = 1,...,N$ and N being the total number of data-points. The model is trained on a set D consisting of pairs of observations, relating input to output $D = [u(n), y^{target}(n)]$, marking ESN training as an inherently supervised technique. Training is achieved by minimizing a measure of error, marking the lowest difference from $y(n)$ to $y^{target}(n)$.

Structure wise, an ESN has three layers of neurons, or weights: An input layer, an output layer and the reservoir in between. Out of these layers, only the output weights are optimized during training. Weights from input layer and the reservoir are randomly initiated and not changed during training. One can imagine the output weights interpreting the recurrent activation in the network, which represent the underlying relationships in the data. Thus the network first extrapolates on a lower dimensional input and then analyses the result in addition to the raw input, subsequently enabling itself to create an interpretation of a given activation in context of a backdrop of previous activation. Consequently, the output of the network $y(n)$ is determined via multiplying the internal network activation $x(n)$, but also the original input $u(n)$ and a bias value with the output weights $W^{out}(n)$, i.e.:

$$y(n) = W^{out}(n)[1; u(n); x(n)] \tag{3}$$

Internal activation of the network, is computed by the sum of the multiplication of current input and input weights and the multiplication of the input of the previous time step with the reservoir weights, i.e.:

$$x(n) = (1-\alpha) \times x(n-1) + \alpha \times tanh(W^{in}(u(n)) + Wx(n-1)) \tag{4}$$

This is then added towards previous activation augmented with a leaking rate α, with which one can influence the impact of previous time steps. The *tanh* function is the commonly used sigmoid activation function for ESNs (Lukoševičius, 2012).

In order to optimize W^{out} as output weights, usually a ridge regression is used, as it features a build-in regularization parameter (β), controlling for overfit of the model via including the weight vector in the fitting process. First, all internal activation for the current weights and training data are collected in $X \in \mathbb{R}^{L \times N}$, L being the sum of reservoir size, input size and one, and N the size of the training data. Then, the regression can be applied in conjunction with the respective target values Y^{target}, ie.:

$$W^{out} = Y^{target} X^T (XX^T + \beta I)^{-1} \tag{5}$$

The regression, as problem of mean square error minimization, can be done in one step, hence being significantly faster than the gradual adaption of gradient descent.

The trained ESN can then be used to generate new output with a test set in order to assess performance. Optimization of the model is done via hyper-parameter setting optimization, which are explained in the method and result sections.

Currently, ESNs are used to in a diverse range of applications, such as Electroencephalogram (EEG) interpretation (L. Sun et al., 2019)(Fourati, Ammar, Jin, & Alimi, 2020), industrial sensors (Lemos et al., 2021) (Patanè & Xibilia, 2021), stock market predictions (Lin, Yang, & Song, 2011) (Z. Liu, Liu, Song, Gong, & Chen, 2017) and medical data (C. Sun, Song, Hong, & Li, 2020), however there have not been many applications of ESNs across applications in soft robotics. ESNs have been used in conjunction with several projects using pneumatic based, soft actuators (Soliman, Mousa, Saleh, Elsamanty, & Radwan, 2021; Li, 2013; Cao, Huang, & Xiong, 2021; Sakurai et al., 2020). According to the investigation of the author, even though ESNs also seem excellently suited to actuators based on dielectric polymers, they have not been used in that capacity as of yet.

In this project, ESNs are used to dynamically model tip-deflection and blocking force behaviour of the soft actuators. In comparison to analytical approaches, data driven methods, such as ESNs make a good fit to the task specification here. Specifically, ESNs are good in modeling time sensitive memory effects and nonlinear behaviour of soft actuators. Another advantage of data-driven approaches to be investigated, is their potential transferability to another problem in the same context.

2.3 Gripper Control

In order to show the transferability and general applicability of the ESN model to a soft actuator in a different task setting, a gripper is designed. This gripper is made with two actuators mounted opposite from each other. The task is to successfully lift an object, so the actuators need to generate a sufficient force to overcome the weight of said object. In order to achieve this, the ESN trained on blocking force data will be used to predict the input voltage need for a successful lift. Experimental observations of the gripper actually lifting objects will then be compared to the predictions of the ESN. While sufficient input voltages could simply be saved and retrieved from a table, the ESN was not trained on these values, but on previously gathered blocking force data. Hence, the setup demonstrates a transferability in data modeling while using ESN on a soft actuator.

Following the task description, it should be noted that while this is labeled as a task to control correct gripper behaviour, it is too simple to be equal to traditional control problems. This being the case, much of the more in-depth aspects of control theory may not apply to the current implementation, and will not be addressed here.

In contrast to other modelling problems, control problems are characterized by the desire to control behaviour, or the output of a system instead of simply modeling it. This is usually described as minimizing the differences between a target behaviour and the current behaviour of a system (Distefano, Stubberud, & Williams, 1967). Depending on the setting and task requirements, there are different ways of establishing a form of output control over a system. Main differences are whether there is a form of feedback relating back to the control algorithm, or not. Respective systems are categorized as open- or closed loop controllers.

Categories for control may be stability, controlability and observability (Ogata et al., 2010). That are the resistance of a system to disruption, its sensitivity to control manipulations and the ease of observing its current state.

Neural Networks can be used for control by modeling the required input for a system to reach certain state, without having to model the system itself. Depending on the requirements of the system, this may also be done as a forecast where a current time-step is used to make inferences about a future output.

In this case, the dielectric polymer based actuators do not feature integrated sensors which are able to report on their behaviour. Moreover, the observational data utilized to train the ESN will be the blocking force data. This is recorded with the actuator directly touching a load cell and exerting its force on it. This setting is replicated in the gripper via setting up the actuators as already touching the object before a voltage is applied. As such, the external camera can only report on gripping success and not on things like actuator displacement in relation to the object. The force exertion from the actuator on the object is similarly also not measurable as that would require a load cell between object and actuator. Thus control can only be established in an open-loop setting.

As will be more deeply explained in section 3.5, the gripper is only formally controlled in terms of an approximated voltage to achieve a successful lift. Further, the gripper was only observed in terms of which voltages were required for a successful lift. The blocking force trained ESN was used to approximate these values. A further problem is that the task only requires discrete voltages, thus any continuous signal of gripper observations and ESN will be transformed to a discrete value.

3 Method

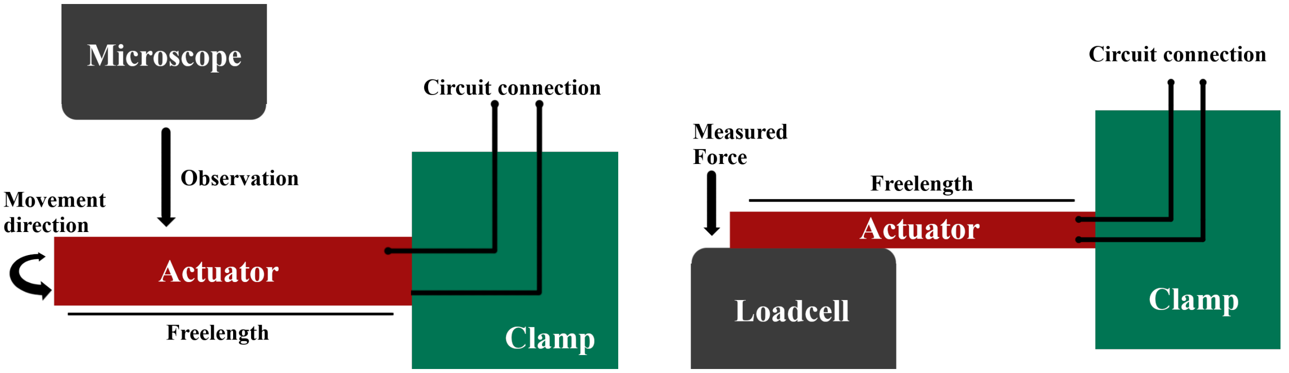
In order to map out and model the dynamic and nonlinear behaviour of the soft P(VDF-TrFE-CTFE)-composite actuator, several different experimental setups are employed. The first goal is to model dynamical behaviour of the actuators based on an applied electric field, using an Echo State Network. To do this, observations of the actuator's tip deflection and blocking force exertion are required. Subsequently, the applicability of said network to a novel setting is tested by letting the network model input voltage to lift up a cube with a gripper consisting of two actuators.

With the exception of the algorithm used for video tracking during the tip deflection experiment which is implemented in Matlab (MATLAB, 2021), all other code is implemented in the Python programming language (Van Rossum & Drake Jr, 1995).

3.1 Experimental Setup and Data Collection

Across different experiments, three identically fabricated actuator samples are being used. In all experiments, an input signal of varying waveform is used to elicit different behaviours in the actuator. This input signal is produced via the RIGOL DG1022 waveform generator (RIGOL Technologies, Beaverton, Oregon, USA, `www.rigolna.com`). The waveform generator is connected to the TRek Model 10/10B-HS high voltage amplifier (Trek Inc., Lockport, New York, USA, `www.trekinc.com`), which is needed to supply a sufficiently high voltage to the actuator (up to 500 V - 2000 V for a strong enough response). Figure 5 shows a representation of the input signal pipeline.

The actuator itself is fixated via a 3D printed clamp made from ABS material. Clamp-internal copper tapes are used to connect the actuator to the electrical circuit. All actuators are fixed to the



(a) The actuator is moving horizontally, with a microscope filming from above.

(b) The actuator is in contact with a load cell that registers the exerted force.

Figure 4: Experimental setup for tip deflection (a) and blocking force exertion (b) data collection. Both setups feature a free length of 45mm.

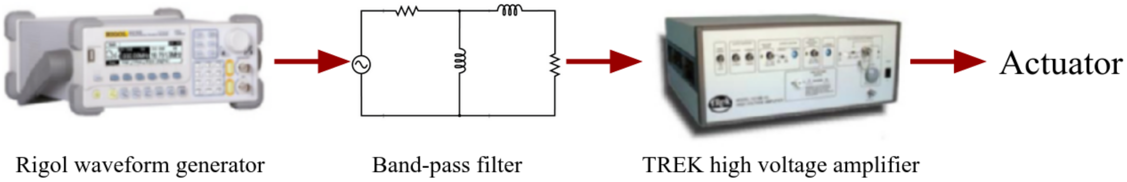


Figure 5: Input signal pipeline to the actuator. The band-pass filer constraints the frequency of the input signal which is further explained in section 3.2.

clamp with a free length of 45mm. Free length has a direct influence in tip deflection and force exertion responses of the actuators, thus needs to be kept constant. When measuring tip deflection of an actuator, the actuator-clamp arrangement is positioned such that the actuator bends horizontally. This allows tip deflection to be measured from above with a digital AM7915MZT(L) 5 MPx microscope (Dino-Lite, AnMo Electronics Corp, Taipei, Taiwan, `www.dino-lite.com`). In order to measure the blocking force exertion, the Instron^{TM} ElectroPuls E1000 test instrument (`www.instron.us`, Norwood, MA, USA) is used to position the clamp above the Instron™ static load cell 2530-5N (capacity of ±5 N and sensitivity of 1.6 mV/V to 2.4 mV/V at static rating), such that the tip of the actuator, touching the load cell, lays flat on its surface. Figure 4 shows the respective experimental setups of tip deflection and blocking force measurements.

Measurements are taken over the duration of an input signal (see section 3.2 for the exact times) and with a sampling frequency depending on observation medium (50 Hz for load cell/ blocking force- and 20 Hz for microscope/tip deflection measurements). These measurement are conducted using the following steps: The experimental setup is constructed and assessed for correct assembly and proper electrode connections on the actuator(s). Then, the recording of variables of interest (voltage, tip deflection or blocking force) is started and subsequently the voltage signal is administered to the actuator. After the signal, the recording is stopped and the next measurement can be set up. The experimental data collection resulted in a collection of time-series which is aggregated and used for model training and testing.

3.2 Signals and Actuator Responses

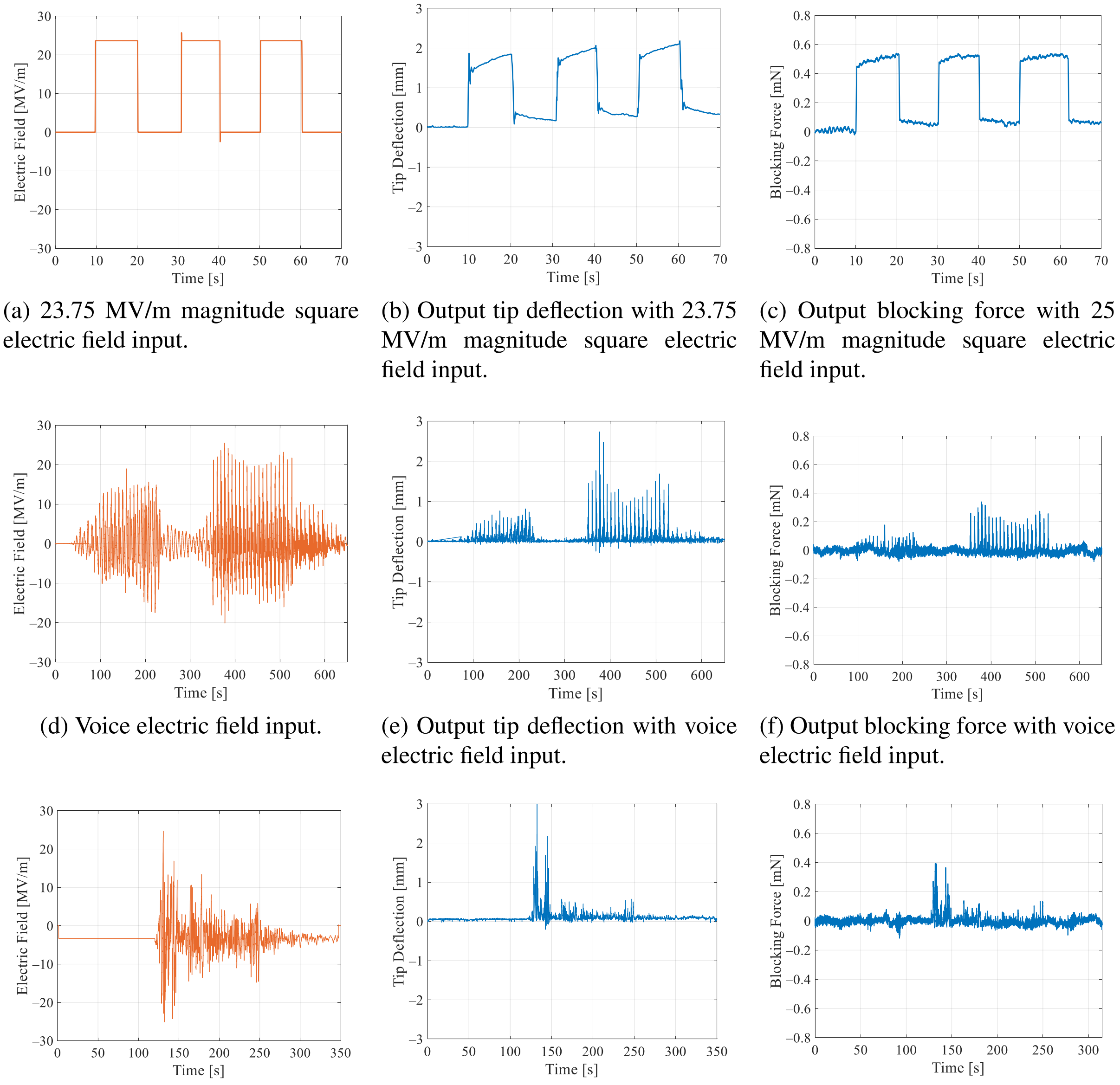
To facilitate later training of the dynamic model, three different functions are used as input for the actuator samples. The more complex and non-cyclic the function, the more information can be drawn from it by a model approximating the underlying reactive behaviour of the actuators. Due to limitations in the testing equipment, internal functions of the waveform generator are used for this purpose (otherwise preferred would be a manually generated non-cyclic function containing several different frequency signals in a range desired for observation). They are, square functions of different magnitudes (ranging from 200 to 2000 V input strength, or up to 23.75 MV/m electric field strength), a voice function emulating a human voice pattern, and a quake function (both quake and voice are scaled such that their highest peak reach 2000 V as well). The three different input functions can be examined in Figure 6a, d and g. Voice and quake are selected for their relatively widely distributed frequency range which provides rich information on the actuator behaviour to the network. Square functions are used instead of sines, as more abrupt changes deliver more information during model training, than a cyclic, gradual change.

In each condition, 22 functions are applied to each of the three actuator samples. 16 functions are varying magnitude square functions. Voice and quake are repeated three times each. In total, that makes 66 functions per condition. For each condition, a random half of each of the function types is later used as training data for the model (24 squares, 5 quakes and 5 voice). The remaining half is used to test model results on novel data. The sample rate for the microscope is 20 Hz and for the load cell is 50 Hz. This sample rate resulted in 256000 (3.55 h) and 763900 (4,24 h) raw data points for tip deflection and blocking force respectively.

In addition to the three input signals, a selection on respective actuator responses in terms of tip deflection and blocking force are shown in Figure 6. Actuator responses are generally noisy compared to the noise-free input voltages. Further, actuators are not affected by the polarity of the input function, as the electrostrictive nanofibers display a quadratic relationship between strain response and electric field. In the square function responses, time dependent and dynamic effects can be observed. For instance, actuator oscillations are visible in the tip deflection response. One can also observe visible delayed responses in the square functions of tip deflection and blocking force observations.

Keeping the sample rates (20 Hz and 50 Hz) in mind, some considerations for the input signals are made: The Nyquist rate (Shannon, 1949) specifies: if the input frequencies are larger than twice the size of the sampling frequency, there is a loss of information due to aliasing. As the signals for both experimental conditions are wanted to be the same, input signals are kept at a range from 0 Hz to 10 Hz.

There are several ways in which the input signal frequency is manipulated. The waveform generator permits length modification of a signal's period, meaning voice and quake could be drawn out to a longer time frame to reach the target frequency range (800 s for voice and 600 s for quake). Further, a band pass filter is implement between waveform generator and voltage amplifier. The band pass filter not only suppresses higher frequencies, but also smooths out the frequency distribution in



(a) 23.75 MV/m magnitude square electric field input.

(b) Output tip deflection with 23.75 MV/m magnitude square electric field input.

(c) Output blocking force with 25 MV/m magnitude square electric field input.

(d) Voice electric field input.

(e) Output tip deflection with voice electric field input.

(f) Output blocking force with voice electric field input.

(g) Quake electric field input.

(h) Output tip deflection with quake electric field input.

(i) Output blocking force with quake electric field input.

Figure 6: Electric field input functions (left column), tip deflection output observations (middle column) and blocking force exertions (right column).

the target range, such that it is more evenly distributed. Notably, a dielectric actuator acts as resistor capacitor differentiator (D'Anniballe, Zucchelli, & Carloni, 2021), which is a high-pass filter. Due to this and the need to test for correct manipulations, the frequency distribution is tested on whether it is within the target range of 0-10 Hz. Consequently, a Fourier transform is applied on the actuator responses to the voice and quake signals to examine their frequency ranges. The respective results are shown in section 4.2.

3.3 Actuator Video Tracking

Given the task setting of observing actuator tip deflection with a microscope, a way to efficiently and accurately track actuator movement in the video files needed to be found. A previous implementation of such an algorithm (D'Anniballe, Paoletta, & Carloni, 2021) used a form of scale invariant feature transform (SIFT) (Lowe, 1999). The SIFT algorithm can track and identify key-points across several different images. It does so via identifying key-points featuring high gradient. Then SIFT characterizes and recognizes key-points via their surroundings in the image and in smoothed versions of the same image. In this application, SIFT is used to track key-points near the tip of the actuator across different frames in the video footage and computing the average tip deflection between the same key-points. While this approach functions well on low velocity movement of the actuator, fast and high deflection can cause the key-points to be lost progressively. Exposure time (shutter speed) of the microscope tends to cause smears in the image, changing the gradient distribution on the surface of the actuator and making it impossible for SIFT to successfully relocate a given key-point in the next frame. As a solution, one can also find new feature vectors after a given one has been lost. However, doing so causes a shift in the average tip deflection as the distribution of identified points across the actuator changes.

With the goal to mitigate the problem of dynamically changing gradient values, a different approach is used for the current version of the actuator tracking algorithm.

Intuitively, grey values and grey value differences between actuator and background will be similar across frames and videos. Hence, a difference in grey value can be tracked to reveal the relative position of the actuator. The main problem here is still the changing grey values of the actuator when it is moving very fast through the frame. To overcome this, grey value information can be accumulated across several frames. This provides clearer information on where the grey value of interest (the actuator) clusters in the frame. As these clusters can be modeled with a linear regression, a regression line can be used as approximation of the actuator movement from frame to frame. This idea is also shown in Figure 7, where angular change signifies the angular rotation from one regression line to the next.

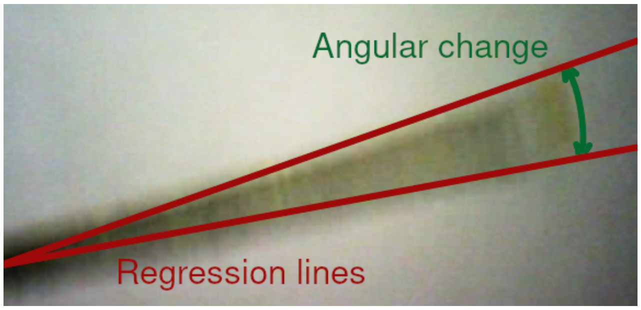


Figure 7: Actuator as captured by the Dino-lite microscope overlaid with the regression lines which are used to approximate its angular rotation from one frame to the next.

For this approach, a video S is considered a set of frames $S = (X_1, ..., X_F)$ with F being the number of frames in the video. All frames in the video are transformed to grey-scale. Two subsequent frames in the video, X_f and X_{f+1} are each evaluated in terms of their grey values respective to the average grey value of the actuator. With t being the observed average grey value of the actuator at the start of the video, one can compute a set of weights which will be highest for values that are similar to t via:

$$greyDiff(X_f,t) = 1/(abs(X_f - t)^2) \tag{6}$$

These weights are then used to highlight the actuator position in the two subsequent frames X_f and X_{f+1}, i.e.:

$$Y_f = X_f \cdot greyDiff(X_f,t) + X_{f+1} \cdot greyDiff(X_{f+1},t) \tag{7}$$

The sum of both weighted frames Y_f now contains a cloud of consistently higher values than random noise in the background. From this point cloud, all the positions of values above a threshold $d = 0.001$ (most noise seems to appear below that value) are then extracted as a set of two variables $[Index_x, Index_y]$. A linear regression line is fitted to the set of indices in order to approximate the position of the actuator in the image. Actuator displacement in the image can now be estimated via comparing two regression lines.

To measure angular displacement of the actuator tip, the formula to calculate arc length l, is used:

$$l = \theta \times r \tag{8}$$

r is the radius of a circle and θ the angle between the two lines. r can be estimated knowing the length of the actuator in the image. Given the slopes m_1 and m_2 of both regression lines, the θ can be calculated via the formula:

$$\theta = tan^{-1}(|\frac{m_1 - m_2}{1 + m_1 m_2}|) \tag{9}$$

Subsequently, angular change in position l is calculated and scaled to fit real distances measured in centimeters. This is done via a reference image, depicting a ruler. This picture is done with the same distance, resolution and zoom level of the microscope as used in the original video. Using the picture, the length of a centimeter in pixel values is determined by drawing a line across a centimeter in the ruler and determining the pixel difference between both ends of the line. As a last step in the algorithm, any displacements higher than 2 cm or point clouds with less than two members are discarded. This measure filters out artifacts related to random noise in the images.

Testing revealed that the output of the new approach to be very similar to the old SIFT algorithm with the extension of being able to interpret actuator movement in higher velocities.

Some observations however showed that the algorithm does not work well in combination with all actuators samples. Curved actuators (due to variance in the fabrication process) are likely to show variations in the detected direction of the displacement. This is presumably the case as, a curve in the actuator translates to two prominent actuator slopes in the image. The linear regression can switch on which slope along the curve of the actuator it fits itself, thus recording abnormal (and not real) angular displacement.

3.4 Network Setup and Assessments

The ESN is set up in accordance with the general guidelines compiled by Lukoševičius (2012). According to the goal of modeling the behaviour of the soft actuator, the model is needed to establish the relation between input and output. Here, Output is a behavioral observation of the actuators (blocking force f or tip deflection d). Input has 3 values; input voltages v, actuator label l and bias, i.e.: $f : x(n) = [1, v(n), l(n)] \rightarrow y(n) = f(n)$, or $y(n) = d(n)$. The model approximates output $y(n)$ from the same time step n as the input $x(n)$. This is done, as the goal is not to predict what happens in the future based on the input, but what the behaviour of the actuator is for the current input. This means, each time-step of Voltage is matched with the behaviour observation of the same step in time.

The variable actuator label is introduced due to the need to control for variations in behaviour of different actuator samples. To do this, actuator behavior across a set condition (average maximal tip deflection at 2000 V, or average maximal blocking force at 2000 V) is used to construct a substitute coding variable. This second input variable functioned as a label for the actuator sample that the algorithm is currently examining. The inclusion of such additional information makes modeling of behaviours easier, and enables the model to distinguish between different actuators. The variable itself is of size $1 \times length(dataset)$ and holds the values $[6, 4.5, 3]$ for blocking force and $[1.8, 0.7, 0.3]$ for tip displacement. The label values are simplified behavioral values, constructed by taking only the first and second digit of the value into account.

As explained in section 3.2, in the tip deflection and the blocking force condition, the time-series of all measured observations is split first. The split is done according to the number of each signal type in the overall sample, such that an equal amount of signal types are assigned to a training set and a test set. The test set is then later used to assess the ESN model's capability to predict actuator behaviour based on observation it has never been exposed to before. In addition to this, a linear regression model is trained and tested on the same data split. In this way, performance of the ESN can be assessed in comparison to a baseline. While the input variables remain the same for the linear regression, the time step that the ESN receives is adapted to a time window, i.e.: $f : x(w) \rightarrow y(n)$ with $w = (n, n-1, ..., n-h)$. h as window size parameter is optimized in the same way as the ESN parameters (which is explained in section 4.1).

To start constructing the network, input- and reservoir weights are randomly generated from a Gaussian distribution. The input weights W are scaled with an input scaling parameter and the reservoir weights are adjusted according to a target spectral radius, i.e.:

$$W_{new} = W_{old} \times (\rho_{target}(W) / \rho_{current}(W)) \tag{10}$$

The spectral radius $\rho(W)$ is used to scale the maximal eigenvalue of the generated reservoir weights, thus determining the spread of values in the matrix. As it introduces more chaos into the system, there exists some controversy in the literature if $\rho(W)$ should be set above one. This is because it can lead to violation of the Echo state property in which the activation should be defined by the past inputs into the network and not its initial conditions. However, this is not a hard constraint. As long as the inputs to the network $u(n) \neq 0$, the input will be influential enough in the computation of the network activation (see equation 4). At this state, Input Scale and Reservoir Size are two other hyperparameters to be considered. Input Scale concerns the scaling of each column of the input weights, which can either be optimized separately, or holistically, to minimize on parameters. Generally, Input Scale influences the linearity of the network. With lower values in the input, the *tanh* activation function behaves more linearly, while in turn higher values will result in binary switching of values. Next, Reservoir size is mainly a trade-off in higher accuracy against generalization to novel data.

However, concerns about generalizations may be mitigated by regularization methods. Another concern is to estimate how difficult the given problem is to describe and how much data is available to do so. With lower complexity and a smaller data set, the reservoir may not need to be very large.

After the initial setup of input weights and reservoir, activation based on the training data are gathered in $X(n)$ with $n = (1,..,N)$ and N being the length of the training data. In conjunction with the target values, $X(n)$ is used to generate the output weights via ridge regression. The activation of the reservoir (equation 4) harbors the Leaking rate parameter α, which is an extension to the original Echo State Networks (Jaeger, Lukoševičius, Popovici, & Siewert, 2007), determining the impact of new time steps on the reservoir activation. The setting of this parameter is highly dependent on how fast effects develop in time, which can vary for different problems. Further, the training via ridge regression requires setting of the regularization term β, which penalizes large output weights and should make the network more generalizeable. Testing for the optimal regularization parameter can be done with the validation set only and does not require rerunning of the training as it does not influence the impact of other parameters.

After applying the ridge regression, the trained weights can be used to let the network generate a final output for the training- and testing data. Subsequently, the difference between predictions and target values is used to assess performance. Network performance is assessed with a normalized root mean squared error (NRMSE), i.e.:

$$RMSE = \sqrt{\frac{1}{N}\Sigma_{i=1}^{N}\left(Y_i - Y_i^{target}\right)^2}, NRMSE = \frac{RMSE}{\sigma^{Y target}} \tag{11}$$

To calculate the NRMSE score, the root of the squared average of the differences between each prediction Y_i and respective target Y_i^{target} are divided by the average standard deviation of the target variable $\sigma^{Y target}$.

This is a highly recommended performance measurement, as it permits understanding of the accuracy of the algorithm without being forced to take the underlying properties of the data into account. An acceptable model should show a NRMSE score between 0 and 1, as that means that the RMSE score is below the natural occurring standard deviation of the data.

Prior to being used to train the ESN, the data needs to be preprocessed as well. Specifically, all variables are normalized making value distributions more comparable between independent settings. Prior to normalization, blocking force observations are first centered, then detrended, using a linear regression. Detrending is only done for blocking force data, as the load cell tends to loose its initial calibration over a longer observation period. The loss of calibration leads to trends in the data. After normalization, the blocking force data is smoothed with a sliding window average filter of size three. This is done to make the signal less noisy. Noteworthy here is that only samples including voice and quake input functions are detrended. This is the case as the linear regression detrend does not work with the square functions and is not strictly necessary. Square input functions are shorter, thus are less likely to feature significantly large trends.

For hyper-parameter optimization, all parameters are adjusted independently from each other while testing performance of the network. This is done, as each parameter change may influence performance of other parameters as well. The dependence is bidirectional, meaning that an already optimized parameter might need more testing after subsequent adaptions of other parameters. Following the recommended approach on optimizing parameters (Lukoševičius, 2012), the value range and step size of parameters is kept large, in order to test a non-exhaustive variety of values. The reason for this is that the network's performance is not sensitive to minuscule variation in parameters, but can already be well tuned with rather coarse changes. Generally, testing is done via a fresh setup of the network. This includes activation gathering and retraining of the network. The exception of this is β which only needed a retraining of the output weights.

3.5 Soft Gripper

In order to test and validate the applicability and transferability of the trained model, a gripper task has been designed. Therein, input voltages required for the gripper to lift objects of different weight are observed. An ESN trained with previously observed blocking force data is used to approximate the experimentally determined input voltages for the gripper.

The gripper behaviour is observed with different electrical field strengths and object weights, in order to record the electrical field strength needed to achieve a successful lift.

For the object, a polystyrene cube is chosen because it features a very low density of 0.96–1.0g/cm^3, enabling the lift for the P(VDF-TrFE-CTFE)-composite soft actuators. In addition, polystyrene is widely available.

The input for the system is the weight of a cube which needs to be translated to the required force to lift it. The blocking force observations, thus the ESN, only show the relationship between blocking force and electric field strength. Consequently, the relationship between object weight and required blocking force is derived analytically. This arrangement results in the processing pipeline shown in Figure 8. As the application does represent a combination of black box and analytical modeling, it could be considered a grey box approach.

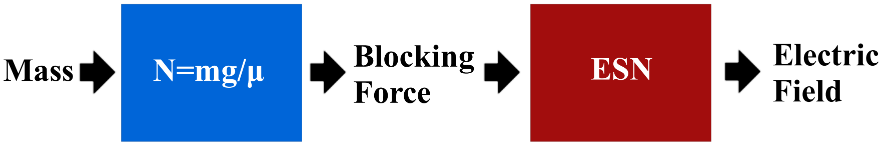


Figure 8: Gripper control pipeline. Given mass, the goal is to return an adequate electric field strength to achieve a successful lift of the cube.

For the analytical part, a completely straight angle of contact between both actuator and the cube is assumed. In addition, the setting also assumes a similarly sized surface area covered between both actuators. Adhering to both of these assumptions, a parallel gripping system is realized. Given this, the following equation (also seen in Figure 8) can be used:

$$N = mg/\mu \tag{12}$$

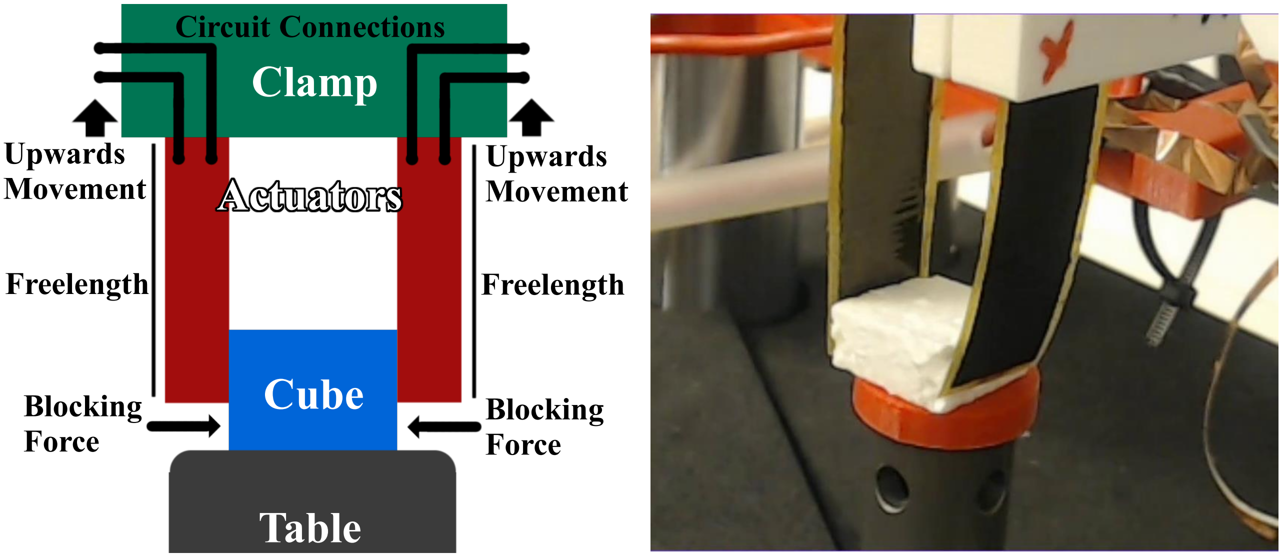
Here, N represents the blocking force that the gripper needs to reach. m is the mass of the cube and g is gravity acting on the mass. μ is the friction coefficient which is the ratio between the frictional force and the normal force. The frictional force is resisting the the motion of two objects which are touching each other. The normal force on the other hand is pressing the objects together. Generally speaking, the higher the friction coefficient between two surfaces, the higher the friction. The more friction, the more energy is required to move objects across each other

Herμ is categorizing the influence of friction between polystyrene and Kapton® surfaces. In equation 12 N and μ are unknown. To arrive at an approximation for N, the experimental data is analyzed in terms of an lowest approximated force needed to arrive at a successful lift. μ is determined experimentally, via placing a polystyrene cube on a Kapton® slope and raising the slope until the cube starts to slide. In this conditions, one can assume the frictional force to be equal to the gravitational force, i.e.:

$$mg\ sin(\theta) = \mu\ mg\ cos(\theta) \tag{13}$$

m and g are still mass, and gravity respectively and θ represents the angle from which each force influences the cube. In simplifying this equation, one arrives at $\mu = sin(\theta)/cos(\theta)$, or $\mu = tan(\theta)$. Calculating μ based on the experimental results returns $\mu = 0.84$.

3.5.1 Experimental Setup



(a)

(b)

Figure 9: Experimental setup as scheme (a) and picture (b) for the two actuator, parallel gripper experiment. Both actuators are fastened at a free length of 45mm.

The gripper is set up, with the two actuators in parallel, each being fastened to a 3D-printed clamp made from ABS material. The clamps are also housing the conductive copper tapes which are used to apply a voltage to the electrodes on each actuator (see Figure 9). The clamps are both mounted on a 3D printed hanger, facing each other. The hanger itself is mounted to the Instron™ElectroPuls E1000 test instrument (`www.instron.us`, Norwood, MA, USA) which can be used to move the frame up and down. The polystyrene cube is placed between the two actuators, such that both actuators lie flat on the side of the cube. Experimentally, a specific electric field is applied to the actuators, then the frame is moved upwards in which case the object is either lifted, or remains grounded when the applied electric field is not strong enough. The whole procedure is documented with the Logitech^{TM} C930E HD camera (`www.logitech.com`, USA). A range of different input voltage strengths between 1500 and 2000 V are tested for successful lifts and labeled depending on experimental outcome. In order to estimate a range of force requirements, polystyrene cubes of 3 different weights (42 μg, 56 μg, 66 μg) are tested. The range of weights are chosen such that lift-off remains possible, but at different requirements in minimal input voltage height. The cubes' respective input voltage requirements were 1400 V, 1700 V and 1900 V. These values still fit within the range used to observe blocking force behaviour (200 V - 2000 V). They also are nicely spread around the voltage range where sufficiently high blocking forces were expected given the observational data. With these considerations in mind, the required input voltages to a successful lift-off are suitable to be approximated based on the blocking force observations.

3.5.2 ESN Setup

Now, the goal is to approximate input voltage from a given blocking force using the already acquired blocking force data. That means that parts of input and output are reversed compared to prior modeling, so input is $x(n) = [1, f(n), l(n)]$ with f being the blocking force data and l being the actuator label. $x(n)$ is used to approximate $y(n)$, i.e.: $f : x(n) \rightarrow y(n) = v(n)$, where v is the previous input voltage. Due to some input voltages within the experimental blocking force observations being negative, the absolute values of the voltage variable are used. In addition the other preprocessing steps for blocking force are applied as discussed in the section 3.4. Both the train and test set was used for the training process. As parameters are optimized during cross validation and fit of the model is assessed based on observed gripper behaviour instead of the blocking force data used for training, there is no need for a distinctive test set. However, the train set was filtered by actuators used in the gripper and functions relevant for the task. Thus, only the blocking force data of the two actuator samples used in the gripper and only the square signals are taken into account for the given task. Consequently, the data set is of length 125790 and the actuator label only contain $l = [4.5, 3]$.

A major point to address is that the current task requires static outputs, while the given model is dynamical in nature. Hence, dynamical conditions need to be emulated in order for the model to be able to approximate correct voltage requirements. For this purpose, input cube weight is first transformed to required blocking force to lift the cube. This is done in the analytical step explained in the previous section (also see Figure 8). Blocking force is transformed from a singular value into a continuous, square signal of a similar form used to train the dynamical ESN model, i.e.: $f : x \rightarrow x(n)$. A singular square signal is defined by 10 s of 0 values, followed by 10 s of the force input value followed by 10 s of 0 values. The square function is repeated twice, because the responses of two different actuator needs to be estimated. This signal is in turn repeated three times for each cube weight. With a 50 Hz sampling rate, that makes $1500 \times 2 \times 3 = 9000$ data points for the simulated signal. A noise function is added on top of the constructed blocking Force, making the signal more realistic. Lastly, the resulting signal is normalized. In addition to the input force, the actuator label input variable (categorizing differences in actuator sample behaviour) is constructed utilizing the same length and alternating the actuator label for each square.

As to be expected, the output of the ESN needs to be transformed into stationary values again, i.e.: $f : y(n) \rightarrow y$. For this purpose, 40 data points adjacent to the center of each square peak are sampled and a mean for each actuator is calculated. This results in a distinct value for each actuator. However, the voltage amplifier used to supply voltage to the actuators only has one output, permitting no distinct actuator control. Hence results for each actuator are summarized in an average for both to arrive at a final estimation.

4 Results

4.1 Parameter Optimization

The ESN and baseline linear regression hyper-parameters are adjusted via n-fold cross validation with $n = 10$. The goal of cross-validation is to optimize generalization in model optimization. Normally, only novel data can be used to assess generalizability of a model. Cross-validation represents a work-around, in which different combinations of the given data set are used to assess model performance. Before starting the cross-validation, a random selection of half the signals of each signal type was drawn from the overall data. This was done for both tip deflection and force exertion data, to be able to validate final optimizations on real, novel data. For tip deflection, the data pairs of input and output S are divided into $S^{train} \in \mathbb{R}^{4\times79740}$ and $S^{test} \in \mathbb{R}^{4\times67315}$. Blocking force is divided into $S^{train} \in \mathbb{R}^{4\times395024}$ and $S^{test} \in \mathbb{R}^{4\times213860}$. There is some variation in set length, as no signal is exactly the same length and signal types differ in length as well. The gripper data set is not divided, but filtered for relevant actuators and signal types, arriving at $S \in \mathbb{R}^{4\times125790}$.

Cross-validation consists of splitting the data sets S into n different subsets D. In n steps, D_{n-1} randomly drawn subsets are then used as a unified training set for a single model configuration, while the remaining set is used as validation set. Here, signal type distribution was not controlled, as all different sample configurations are tested either way. Now, performance for this model configuration on the validation set is averaged. This process is repeated for all configurations to be tested. Afterwards, the best performing configuration can be determined and trained on the whole data set. Lastly, the remaining half of the data is used as a final test set for the best performing model configuration. Within each training run, the ESN model is set up by first drawing input and reservoir weights from a Gaussian distribution, then network activation of the training data are harvested. The resulting activation and target values are subsequently used to train the output weights.

The process of the cross-validation is repeated with a range of different hyper-parameters. These are repeatedly and individually optimized, as hyper-parameters can mediate the effectiveness of other hyper-parameters. Furthermore, reservoir activation patterns and the mean absolute output weights are monitored, as weak or inconsistent activation and extreme output weights may indicate problems in the optimization of the network (pointing out a need to further modify hyper-parameters). Finally, plots featuring target and predicted values of the test set are examined and used as final test of validation towards a good fit in terms of overall model performance.

The explored hyper-parameter ranges for the ESN are reported in Table 2 while the respective optimized parameters for tip-deflection, blocking force and the gripper experiment are reported in Table 3. The linear regression baseline model is tested on performance with and without an intercept.

Linear regression is optimized with window length $w = 20$ for blocking force and $w = 30$ for tip deflection.

Table 2: Value ranges during optimization of hyper-parameters.

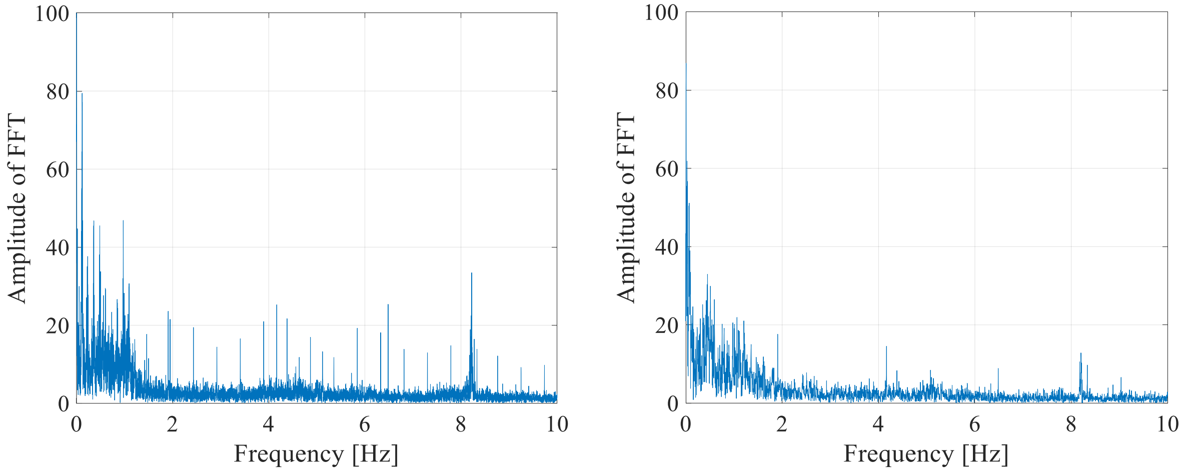
Hyper-Parameter	Value Range
Reservoir Size	$10-500$
Input Scaling	$0.01-10$
Spectral Radius	$0.4-2.5$
Leakiness	$0.1-0.99$
Regularization	1e−8 − 1e+5

Table 3: Optimized hyper-parameter settings per each respective experimental setting.

Hyper-Parameter	Tip-Deflection	Blocking Force	Gripper (Voltage)
Reservoir Size	100	500	100
Input Scaling	0.5	1	0.8
Spectral Radius	0.9	0.6	0.7
Leakiness	0.1	0.3	0.03
Regularization	0.01	0.01	0.01

4.2 Signal Frequency Observations

Due to prior considerations of appropriate distribution of input signals in the frequency domain, a Fast Fourier Transform (FFT) is employed on actuator observations after applying respective signal functions (voice and quake). Respective plots are shown in Figure 10. The plot shows amplitude peaks widely distributed between 0 and $10Hz$ for voice and quake inputs, which represents the range of interest in this setting. There does not seem to exist obvious application of modeling a type of actuator such as the given one in higher frequencies. As such, the employed manipulation of signals in their frequency seems successful.



(a) Output blocking force with voice input signal.

(b) Output blocking force with quake input signal.

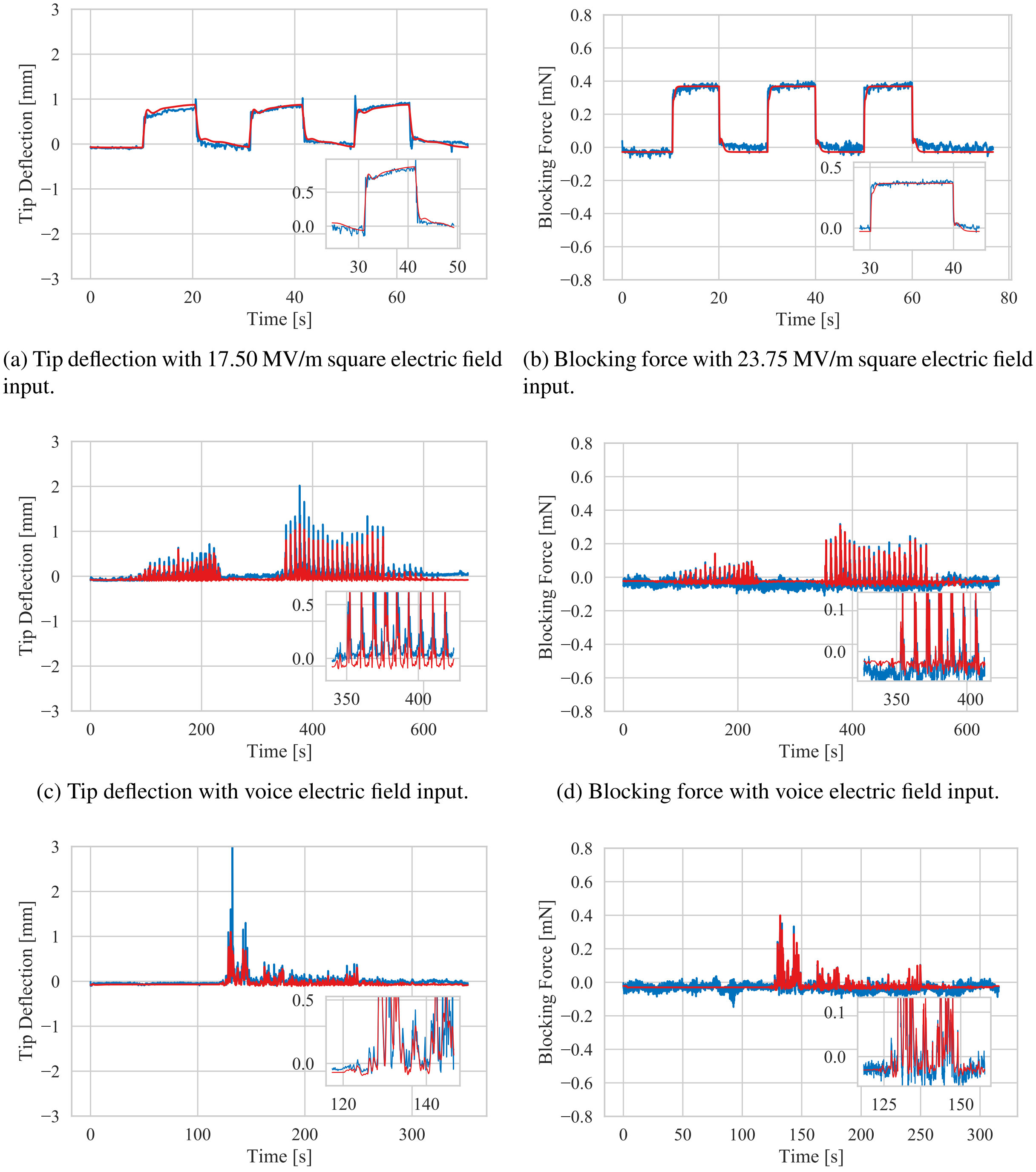
Figure 10: Fourier Transform amplitudes of actuator blocking force after exposure to either the voice or quake input signal.

4.3 Network Performance

Table 4: Final NRMSE (RMSE) values after optimizing respective parameters and modeling Tip Deflection or Blocking Force exertion. ESN results are compared to a Baseline linear regression model.

Error type	Tip Deflection		Blocking Force	
	ESN	Baseline	ESN	Baseline
Training error	0.255(0.062mm)	0.704(0.171mm)	0.222(0.015mN)	0.778(0.061mN)
Testing error	0.433(0.106mm)	0.754(0.183mm)	0.280(0.019mN)	1.034(0.081mN)

Final model fit for blocking force and tip deflection can be examined in Table 4 which shows NRMSE scores and RMSE scores in parentheses. Testing error shows ESN performance on completely new data, which is higher than the training scores, but still show NRMSE values below one. Tip deflection shows consistently higher error rates than blocking force which may be related to the higher potential for noise due to the need to transform the related data from a video. Also visible in Table 4 are results of the optimized baseline linear regression. Linear regression fit in both conditions is worse than the ESN. In tip deflection, test NRMSE is still below one, but not so for blocking force. In terms of RMSE, tip deflection of the baseline is almost twice as high as the ESN. Baseline blocking force RMSE is four times higher than the respective ESN error. Figure 11 shows a selection of ESN model outputs for previously unseen testing data of both the blocking force and tip deflection condition. Compared to the target, the model displays barely any noise. This is because these predicted values are generated based on electric field measurements which contain less noise than the behavioural observations. This can be observed when comparing voltage input signal from Figure 3.2 with Figure 11. On the other side, voice and quake predictions in the deflection model



(a) Tip deflection with 17.50 MV/m square electric field input.

(b) Blocking force with 23.75 MV/m square electric field input.

(c) Tip deflection with voice electric field input.

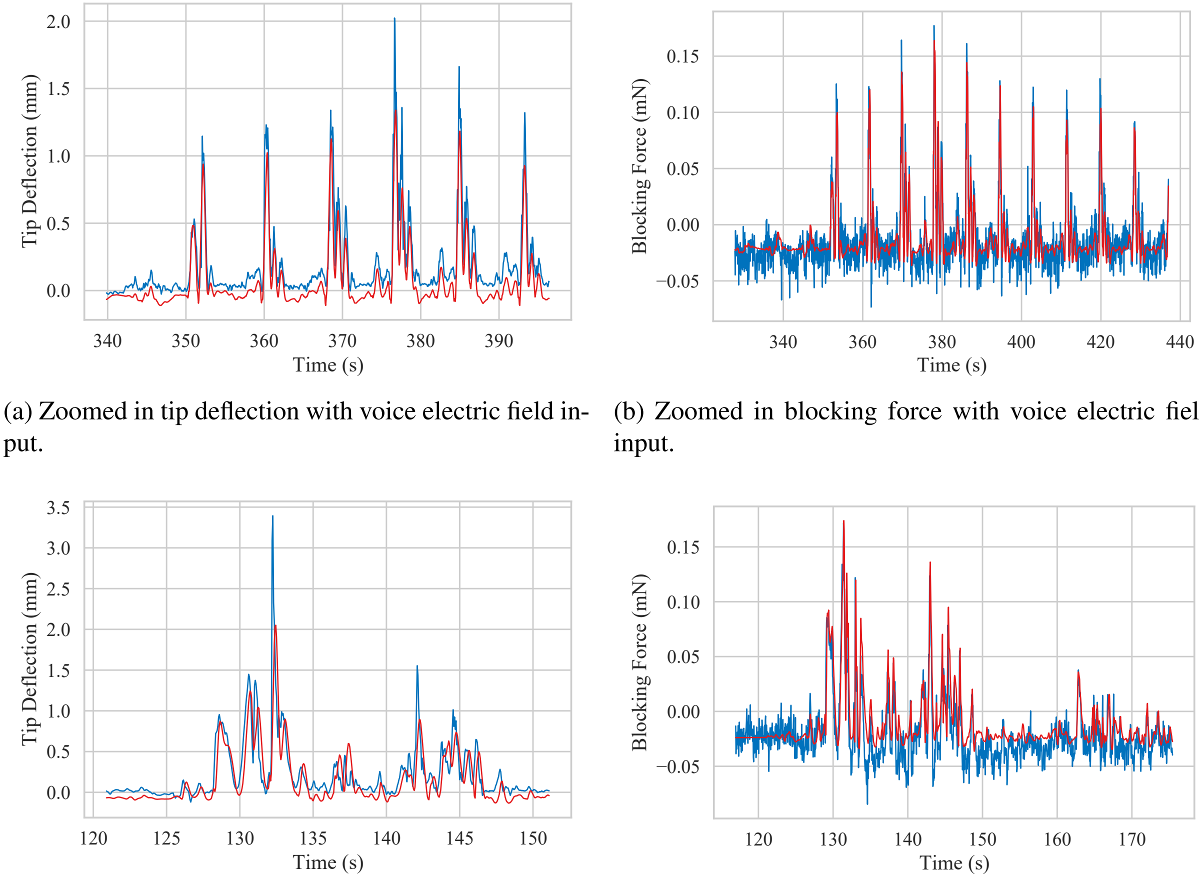
(d) Blocking force with voice electric field input.

(e) Tip deflection with quake electric field input.

(f) Blocking force with quake electric field input.

Figure 11: Tip deflection (left) and blocking force (right) ESN model output (red) compared with the experimentally observed results (blue) with different electric field input signals.

seem to exhibit overall lower tip deflection than the target. This might be due to the video tracking



(a) Zoomed in tip deflection with voice electric field input.

(b) Zoomed in blocking force with voice electric field input.

(c) Zoomed in tip deflection with quake electric field input.

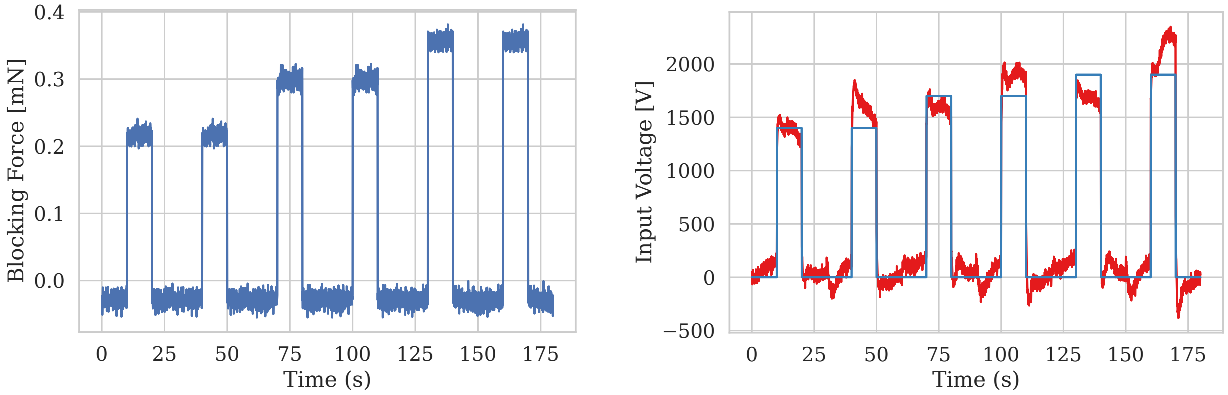
(d) Zoomed in blocking force with quake electric field input.

Figure 12: Zoomed in model results, showing dynamical detail in the model results. ESN model output is (red) compared with the experimentally observed results (blue) with different electric field input signals.

script overestimating real movements when they occur fast and with a sudden onset. Alternatively, the lower sample rate of the microscope might result in aliasing effects, which may result in less consistent deflection observations in high frequency settings. Notably, the network can easily follow the dynamic behaviour of even the more complex input functions voice and quake. The model results of voice and quake can be more closely examined in Figure 12 which shows zoomed-in responses to both signal types.

4.4 Gripper

In order to determine the lowest required input voltage to arrive at a strong enough grip for a successful grip and lift-off of the three differently weighted cubes, the gripper is tested with input voltages ranging from 1500 to 2000 V in increments of 100 V. Noteworthy for this analysis, the lowest voltage with a confirmed lift-off is considered. However, involved components are receptive



(a) Required Blocking Force for each of the two actuators and all three different weighted cubes.

(b) Approximated input voltage according to the ESN (red) and simulated target voltages (blue).

Figure 13: Input and Output of the trained ESN in the Gripper experiment. Two neighboring squares are representative of the two actuators of the gripper with a cube of specific weight. The first two square peaks correspond to a cube of 42 μg the next two to 56 μg and the last to 66 μg

Table 5: Different cube sample weights and their related lowest recorded Voltage input on the gripper for a successful lift and the results of the ESN.

Variables	Samples		
Cube Weight	42 μg	56 μg	66 μg
lowest Voltage Input for Lift	1400 V	1700 V	1900 V
Model Output	1518 V	1763 V	1946 V

to external influences, so lift-off can not be replicated consistently for any input voltage. Observed lowest required input voltage is compared to the input voltages predicted by the ESN explained in section 3.5.2. The NRMSE of training the ESN on the blocking force data-set is 0.172. The summarized value comparison between observed and modeled values is shown in Table 5. Generally, input voltages which the ESN-based system would have been employing are similar, but larger than the experimentally observed values. In the experimental observations, the lightest polystyrene cube is successfully lifted with more than a 100 V less compared to the model results. However, the other modeled input voltages seem to fit well with the observations in the gripper experiment. Input and output to the ESN arrangement for the gripper can be examined in Figure 13. The Figure also contrasts predicted input voltage with experimental observations. While value range outputs are not fully stable, the ESN successfully distinguishes between both actuators. This is also reflected in the actuators behaviour, as one tended to show stronger blocking forces than the other.

5 Discussion

In this project, a novel electrostrictive actuator is observed in terms of its blocking force exertion and tip deflection responses to complex signals of distributed frequency. The actuator consists of an active layer of electrospun P(VDF-TrFE-CTFE) nanofibers, submerged into a PDMS silicone matrix, a passive Kapton® layer and electrodes made from a silicone carbon mixture. The acquired observational data is used to train an ESN. A second variable is introduced to label different actuator samples according to their behaviour at an input voltage of 2000 V. The resulting models reveal a NRMSE below 1 on both conditions and shows generally low deviation to the previously withheld second half of the observational data. The ESN model error measures are also significantly lower than a baseline linear regression fitted to the same data. Results show a good fit, thereby demonstrating the effectiveness of ESNs in modeling dynamical behaviour of a P(VDF-TrFE-CTFE)-composite based, electrostrictive soft actuators.

In order to show transferability and applicability of the data-driven ESN approach, a simple, two actuator gripper task is designed. The gripper task consists of a successful lift of a polystyrene cube of varying weight. This requires a specific input voltage such that the actuators can exert a high enough force to lift the cube. Voltage is predicted using the related blocking-force observations to train an ESN. Results show a close fit to the observed, required input voltages for a successful lift of a cube. In addition, the implementation demonstrates how the network differentiates performance of different actuators due to the second input variable.

The ESN seems to have frequent trouble with underestimating maximal tip deflection. This issue is less prevalent, but still present, with blocking force data. Likewise, this effect is reflected in a consistent overestimation of input Voltage in the gripper experiment. The differences between experimental observations and model predictions may be explained by the presence of noise in the data. Specifically, data collection suffers from sensitivity limitations in terms of sensors and external influences. On the one side, the load cell detects noise in addition to actuator blocking forces, which may stem from it not being sensitive enough for the low blocking forces of the actuators. In addition, noise may be generated from external vibrations as well. This may have warranted a set-up which prevents vibrations transfer towards the load cell. On the other side, the microscope has similar problems with noise. Moving air, or other external vibrations may cause the actuator to oscillate. Additionally, the settings for the recordings could have been more optimized towards the set-up, with better contrasts between fore- and background and even brighter lighting. On the hardware side, the microscope's 20 frame per second recording limitation, as well as its long shutter time, resulted in further sources of uncertainty in the analysis of the actuator movement. This being the case, the algorithm interpreting actuator deflection does not remain free of its own problems either. The current implementation assumes the actuator behaving as a straight line when moving. During testing, this assumption is shown to hold, but it has been faced with counterexamples later on. Due to their fabrication process, no two actuators have perfectly the same shape and some feature a slight natural bend. This bend can throw off the regression estimation of current actuator placement, as via a bend essentially two alternative choices in angle directions are shown. When the actuator moves, the algorithm is presented with a choice of which angle to adhere its linear regression to. This makes it possible for the algorithm to rapidly shift in angle and thus record nonexistent movement of the actuator. Data which is visibly corrupted by misinterpretation of angle alignment is removed from the training and testing set, elevating concerns of unreliable data but resulting in less data applicable to the network. This being the case, a curved regression fit would have been significantly better when taking these issues into account.

As future direction, either stronger actuators or more precise conditions may be able to produce even better modeling results. Other fields, featuring dynamical systems, have over time developed a more unified methodology (e.g., speech recognition (Ibrahim, Odiketa, & Ibiyemi, 2017), stock market forecasting (Kumar, Jain, & Singh, 2021) , video analytics (Zhang, Sun, Wu, & Zhong, 2019)) to deal with inherent difficulties of applying data driven methods to their respective challenges. At the moment, soft actuators still appear to be in the progress of converging onto such a unified task and processing pipeline, which makes it hard to quantitatively compare methodology in data gathering, preprocessing and modeling across actuator projects. This work may contribute a part towards inspiring the construction of a unified methodology to make such comparisons possible.

In conclusion, present results show that an ESN can indeed be successfully employed in modeling and controlling a P(VDF-TrFE-CTFE)-Composite Actuator. While modeling results did not always fit perfectly, this may be explained by inherent difficulties in the methodology on how the measurements are obtained and processed. More systematic and unified methodologies of data gathering and processing could potentially increase increase ESN model efficacy further in future investigations.

References

Braganza, D., Dawson, D. M., Walker, I. D., & Nath, N. (2007). A neural network controller for continuum robots. *IEEE Transactions on Robotics*, *23*(6), 1270-1277. doi: 10.1109/TRO.2007.906248

Burawudi, K. K., D'Anniballe, R., Langius, R. G., & Carloni, R. (2021). A comparative study of predictive models for nafion-117 ipmc soft actuators. In *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)* (p. 1124-1129). doi: 10.1109/AIM46487.2021.9517466

Cangelosi, A., Bongard, J., Fischer, M., & Nolfi, S. (2015, 01). Embodied intelligence. *Springer Handbook of Computational Intelligence*, 697-714. doi: 10.1007/978-3-662-43505-2_37

Cao, Y., Huang, J., & Xiong, C. (2021). Single-layer learning-based predictive control with echo state network for pneumatic-muscle-actuators-driven exoskeleton. *IEEE Transactions on Cognitive and Developmental Systems*, *13*(1), 80-90. doi: 10.1109/TCDS.2020.2968733

Carloni, R., Lapp, V. I., Cremonese, A., Belcari, J., & Zucchelli, A. (2018). A variable stiffness joint with electrospun P(VDF-TrFE-CTFE) variable stiffness springs. *IEEE Robotics and Automation Letters*, *3*(2), 973-978. doi: 10.1109/LRA.2018.2793348

Chee, P. S., Mah, C. K., & Ali, M. S. M. (2016). Soft dielectric elastomer actuator for micropump application. In *2016 IEEE 29th International Conference on Micro Electro Mechanical Systems (MEMS)* (p. 561-564). doi: 10.1109/MEMSYS.2016.7421687

D'Anniballe, R., Erdmann, N., Selleri, G., & Carloni, R. (2022). Dynamic modeling of P(VDF-TrFE-CTFE)-based soft actuators via echo state networks. In *2022 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM).*

Distefano, J., Stubberud, A., & Williams, I. (1967). *Theory and problems of feedback and control systems, Schaum's outline series.* Mcgraw-Hill New York.

Doya, K. (1993). Bifurcations of recurrent neural networks in gradient descent learning. *IEEE Transactions on Neural Networks*, *1*(75), 218.

D'Anniballe, R., Paoletta, G., & Carloni, R. (2021). A polyurethane-based electrospun nanofiber bundle soft actuator: Fabrication, modeling, and control. In *2021 IEEE 4th international conference on soft robotics (robosoft)* (pp. 393–398).

D'Anniballe, R., Zucchelli, A., & Carloni, R. (2021). Towards poly(vinylidene fluoride-trifluoroethylene-chlorotrifluoroethylene)-based soft actuators: Films and electrospun aligned nanofiber mats. *Nanomaterials*, *11*(1). doi: 10.3390/nano11010172

D'Anniballe, R., Zucchelli, A., & Carloni, R. (2022). The effect of morphology on poly(vinylidene fluoride-trifluoroethylene-chlorotrifluoroethylene)-based soft actuators: Films and electrospun aligned nanofiber mats. *Sensors and Actuators A: Physical*, *333*, 113255. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0924424721007184` doi: https://doi.org/10.1016/j.sna.2021.113255

El-Atab, N., Mishra, R. B., Al-Modaf, F., Joharji, L., Alsharif, A. A., Alamoudi, H., ... Hussain, M. M. (2020). Soft actuators for soft robotic applications: A review. *Advanced Intelligent Systems*, *2*(10), 2000128. doi: https://doi.org/10.1002/aisy.202000128

Elgeneidy, K., Lohse, N., & Jackson, M. (2018). Bending angle prediction and control of soft pneumatic actuators with embedded flex sensors – a data-driven approach. *Mechatronics*, *50*, 234-247. doi: https://doi.org/10.1016/j.mechatronics.2017.10.005

Fourati, R., Ammar, B., Jin, Y., & Alimi, A. M. (2020). EEG feature learning with intrinsic plasticity based deep echo state network. In *2020 International Joint Conference on Neural Networks (IJCNN)* (p. 1-8). doi: 10.1109/IJCNN48605.2020.9207464

Funahashi, K., & Nakamura, Y. (1993). Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, *6*(6), 801-806. Retrieved from https://www.sciencedirect.com/science/article/pii/S089360800580125X doi: https://doi.org/10.1016/S0893-6080(05)80125-X

Giorelli, M., Renda, F., Calisti, M., Arienti, A., Ferri, G., & Laschi, C. (2015). Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature. *IEEE Transactions on Robotics*, *31*(4), 823-834. doi: 10.1109/TRO.2015.2428511

Giorelli, M., Renda, F., Ferri, G., & Laschi, C. (2013). A feed-forward neural network learning the inverse kinetics of a soft cable-driven manipulator moving in three-dimensional space. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (p. 5033-5039). doi: 10.1109/IROS.2013.6697084

Gotti, C., Sensini, A., Zucchelli, A., Carloni, R., & Focarete, M. L. (2020). Hierarchical fibrous structures for muscle-inspired soft-actuators: A review. *Applied Materials Today*, *20*, 100772. Retrieved from https://www.sciencedirect.com/science/article/pii/S2352940720302201 doi: https://doi.org/10.1016/j.apmt.2020.100772

Gu, G.-Y., Gupta, U., Zhu, J., Zhu, L.-M., & Zhu, X. (2017). Modeling of viscoelastic electromechanical behavior in a soft dielectric elastomer actuator. *IEEE Transactions on Robotics*, *33*(5), 1263-1271. doi: 10.1109/TRO.2017.2706285

Ibrahim, Y. A., Odiketa, J. C., & Ibiyemi, T. S. (2017). Preprocessing technique in automatic speech recognition for human computer interaction: an overview. *Annals. Computer Science Series*, *15*(1), 186–191.

Jaeger, H. (2001). The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, *148*(34), 13.

Jaeger, H., Lukoševičius, M., Popovici, D., & Siewert, U. (2007). Optimization and applications of echo state networks with leaky- integrator neurons. *Neural Networks*, *20*(3), 335-352. Retrieved from https://www.sciencedirect.com/science/article/pii/S089360800700041X (Echo State Networks and Liquid State Machines) doi: https://doi.org/10.1016/j.neunet.2007.04.016

Jiang, H., Wang, Z., Liu, X., Chen, X., Jin, Y., You, X., & Chen, X. (2017). A two-level approach for solving the inverse kinematics of an extensible soft arm considering viscoelastic behavior. In *2017 IEEE International Conference on Robotics and Automation (ICRA)* (p. 6127-6133). doi: 10.1109/ICRA.2017.7989727

Jiang, Z., Li, Y., & Wang, Q. (2022). Intelligent feedforward hysteresis compensation and tracking control of dielectric electro-active polymer actuator. *Sensors and Actuators A: Physical*, *341*, 113581.

Kuhring, S., Uhlenbusch, D., Hoffstadt, T., & Maas, J. (2015). Finite element analysis of multilayer

DEAP stack-actuators. In *Electroactive Polymer Actuators and Devices (EAPAD) 2015* (Vol. 9430, pp. 395–407).

Kumar, G., Jain, S., & Singh, U. P. (2021). Stock market forecasting using computational intelligence: A survey. *Archives of Computational Methods in Engineering*, *28*(3), 1069–1101.

Lemos, T., Campos, L. F., Melo, A., Clavijo, N., Soares, R., Câmara, M., ... Pinto, J. C. (2021). Echo state network based soft sensor for monitoring and fault detection of industrial processes. *Computers & Chemical Engineering*, *155*, 107512. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0098135421002908` doi: https://doi.org/10.1016/j.compchemeng.2021.107512

Li, T. (2013). *Learning from the octopus: sensorimotor control of octopus-inspired soft robots* (Doctoral Thesis, University of Zurich). Retrieved from `https://www.zora.uzh.ch/id/eprint/91630/1/20142087.pdf`

Lin, X., Yang, Z., & Song, Y. (2011). Intelligent stock trading system based on improved technical analysis and echo state network. *Expert Systems with Applications*, *38*(9), 11347-11354. Retrieved from `https://www.sciencedirect.com/science/article/pii/S095741741100409X` doi: https://doi.org/10.1016/j.eswa.2011.03.001

Liu, L., Lei, Y., Zhang, Z., Liu, J., Lv, S., & Guo, Z. (2020). Fabrication of PDA@SiO2@rGO/PDMS dielectric elastomer composites with good electromechanical properties. *Reactive and Functional Polymers*, *154*, 104656. doi: https://doi.org/10.1016/j.reactfunctpolym.2020.104656

Liu, Z., Liu, Z., Song, Y., Gong, Z., & Chen, H. (2017). Predicting stock trend using multi-objective diversified echo state network. In *2017 Seventh International Conference on Information Science and Technology (ICIST)* (p. 181-186). doi: 10.1109/ICIST.2017.7926754

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE International Conference on Computer Vision* (Vol. 2, pp. 1150–1157).

Lukoševičius, M. (2012). A practical guide to applying echo state networks. In G. Montavon, G. B. Orr, & K.-R. Müller (Eds.), *Neural Networks: Tricks of the Trade: Second Edition* (pp. 659–686). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/978-3-642-35289-8_36

Manti, M., Cacucciolo, V., & Cianchetti, M. (2016). Stiffening in soft robotics: A review of the state of the art. *IEEE Robotics Automation Magazine*, *23*(3), 93-106. doi: 10.1109/MRA.2016.2582718

MATLAB. (2021). *version 9.11.0 (r2021b)*. Natick, Massachusetts: The MathWorks Inc.

McGough, K., Ahmed, S., Frecker, M., & Ounaies, Z. (2014). Finite element analysis and validation of dielectric elastomer actuators used for active origami. *Smart Materials and Structures*, *23*(9), 094002.

Melingui, A., Lakhal, O., Daachi, B., Mbede, J. B., & Merzouki, R. (2015). Adaptive neural network control of a compact bionic handling arm. *IEEE/ASME Transactions on Mechatronics*, *20*(6), 2862-2875. doi: 10.1109/TMECH.2015.2396114

Moss, A., Krieg, M., & Mohseni, K. (2021). Modeling and characterizing a fiber-reinforced dielectric elastomer tension actuator. *IEEE Robotics and Automation Letters*, *6*(2), 1264-1271. doi: 10.1109/LRA.2021.3056349

Ogata, K., et al. (2010). *Modern control engineering* (Vol. 5). Prentice Hall Upper Saddle River, NJ.

Patanè, L., & Xibilia, M. G. (2021). Echo-state networks for soft sensor design in an sru process. *Information Sciences*, *566*, 195-214. Retrieved from https://www.sciencedirect.com/science/article/pii/S0020025521002504 doi: https://doi.org/10.1016/j.ins.2021.03.013

Reinhart, R. F., Shareef, Z., & Steil, J. J. (2017). Hybrid analytical and data-driven modeling for feed-forward robot control. *Sensors*, *17*(2), 311. doi: 10.3390/s17020311

Sakurai, R., Nishida, M., Sakurai, H., Wakao, Y., Akashi, N., Kuniyoshi, Y., ... Nakajima, K. (2020). Emulating a sensor using soft material dynamics: A reservoir computing approach to pneumatic artificial muscle. In *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)* (p. 710-717). doi: 10.1109/RoboSoft48309.2020.9115974

Shannon, C. (1949). Communication in the presence of noise. *Proceedings of the Institute of Radio Engineers*, *37*(1), 10–21.

Shintake, J., Schubert, B., Rosset, S., Shea, H., & Floreano, D. (2015). Variable stiffness actuator for soft robotics using dielectric elastomer and low-melting-point alloy. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (p. 1097-1102). doi: 10.1109/IROS.2015.7353507

Soliman, M., Mousa, M. A., Saleh, M. A., Elsamanty, M., & Radwan, A. G. (2021). Modelling and implementation of soft bio-mimetic turtle using echo state network and soft pneumatic actuators. *Scientific Reports*, *11*(1), 1–11.

Sun, C., Song, M., Hong, S., & Li, H. (2020). A review of designs and applications of echo state networks. *CoRR*, *abs/2012.02974*. Retrieved from https://arxiv.org/abs/2012.02974

Sun, L., Jin, B., Yang, H., Tong, J., Liu, C., & Xiong, H. (2019). Unsupervised EEG feature extraction based on echo state network. *Information Sciences*, *475*, 1-17. Retrieved from https://www.sciencedirect.com/science/article/pii/S0020025518307692 doi: https://doi.org/10.1016/j.ins.2018.09.057

Thuruthel, T. G., Falotico, E., Renda, F., & Laschi, C. (2017, oct). Learning dynamic models for open loop predictive control of soft robotic manipulators. *Bioinspiration & Biomimetics*, *12*(6), 066003. doi: 10.1088/1748-3190/aa839f

Thuruthel, T. G., Shih, B., Laschi, C., & Tolley, M. T. (2019). Soft robot perception using embedded soft sensors and recurrent neural networks. *Science Robotics*, *4*(26). doi: 10.1126/scirobotics.aav1488

Truong, B. N. M., & Ahn, K. K. (2015). Inverse modeling and control of a dielectric electro-active polymer smart actuator. *Sensors and Actuators A: Physical*, *229*, 118-127. doi: https://doi.org/10.1016/j.sna.2015.03.032

Van Rossum, G., & Drake Jr, F. L. (1995). *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.

Xiao, H., Wu, J., Ye, W., & Wang, Y. (2020). Dynamic modeling for dielectric elastomer actuators based on lstm deep neural network. In *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)* (pp. 119–124).

Youn, J.-H., Jeong, S. M., Hwang, G., Kim, H., Hyeon, K., Park, J., & Kyung, K.-U. (2020). Dielectric elastomer actuator for soft robotics applications and challenges. *Applied Sciences*, *10*(2), 640.

Zhang, Q., Sun, H., Wu, X., & Zhong, H. (2019). Edge video analytics for public safety: A review. *Proceedings of the IEEE*, *107*(8), 1675-1696. doi: 10.1109/JPROC.2019.2925910

Zhao, H., Hussain, A. M., Duduta, M., Vogt, D. M., Wood, R. J., & Clarke, D. R. (2018). Compact dielectric elastomer linear actuators. *Advanced Functional Materials*, *28*(42), 1804328.

Zhou, F., Zhang, F., Cao, X., Zhang, Z., Chen, X., Xiao, Y., … Xu, Z. (2019). Fabrication and modeling of dielectric elastomer soft actuator with 3d printed thermoplastic frame. *Sensors and Actuators A: Physical*, *292*, 112-120. Retrieved from `https://www.sciencedirect.com/science/article/pii/S092442471831968X` doi: https://doi.org/10.1016/j.sna.2019.02.017