

Simplifying Data-Driven Modeling of the Volume-Flow-Pressure Relationship in Hydraulic Soft Robotic Actuators

Sang-Yoep Lee, *Member, IEEE*, Leonardo Zamora Yañez, Jacob Rogatinsky, Vi T. Vo, Tanvi Shingade, and Tommaso Ranzani, *Member, IEEE*,

Abstract—Soft robotic systems are known for their flexibility and adaptability, but traditional physics-based models struggle to capture their complex, nonlinear behaviors. This study explores a data-driven approach to modeling the volume-flow-pressure relationship in hydraulic soft actuators, focusing on low-complexity models with high accuracy. We perform regression analysis on a stacked balloon actuator system using exponential, polynomial, and neural network models with or without autoregressive inputs. The results demonstrate that simpler models, particularly multivariate polynomials, effectively predict pressure dynamics with fewer parameters. This research offers a practical solution for real-time soft robotics applications, balancing model complexity and computational efficiency. Moreover, the approach may benefit various techniques that require explicit analytical models.

Index Terms—Soft robotics, nonlinear dynamics, regression analysis, system identification, and proprioceptive sensing

I. INTRODUCTION

SOFT robotic technologies have gained increasing interest over the past decade, leading to more adaptable, safe, and resilient robotic devices [1]. Recent studies provide compelling evidence that a deeper understanding and integration of high-order dynamics and internal state variables in soft robots can significantly broaden their functionality and range of applications [2], enabling a shift from kinematics and quasi-statics to kinetics and dynamics [3]. However, these aspects have often been neglected due to challenges in observing and utilizing them. Unlike rigid robotics, where a standard equation of motion exists, there is no gold-standard mathematical framework for soft robotic behavior, which typically involves hysteresis, elastic deformation, and fluid dynamics [4].

Sensor integration is essential not only for accurate modeling and control but also for robust, real-time dynamic operation of soft robots [5]. However, integrating sensors in soft robots is challenging as they undergo large, continuous,

and nonlinear deformations that require a theoretically infinite number of measurement points and hinder reliable sensor attachment [6]. Even with high-resolution measurements from a limited number of sensors, their dynamic properties remain inherently undersampled. Thus, it can be critical to augment the observability of a soft robot's dynamics by leveraging intrinsic measurements from its fundamental subsystems such as actuation unit, beyond solely relying integrated sensors.

Among the actuation techniques for soft robots, fluidic actuation has been widely employed because of their large strokes, fast response, and high force transmission efficiency [7]. There are two main classes of fluidic actuators: elastomeric and sheet-based designs. Elastomeric fluid actuators achieve motion through stretching of their constituent materials, following the work input from a fluidic source that often reaches to high actuation pressures (i.e., hundreds of kPa) [8]. Sheet-based fluidic actuators, by contrast, are fabricated by selectively adhering thin, less-extensible films, and achieve motion via the geometric unfolding of these layers at much lower fluidic pressures (i.e., tens of kPa) [9].

A prominent example of the sheet-based approach is the stacked balloon actuator (SBA) [10]. SBAs are built from thin sheet materials so they occupy almost no volume when deflated but produce large, pre-programmed strokes when inflated. By integrating multiple SBA stacks in parallel, multi-directional bending is achievable for applications ranging from beating-heart surgery [11], to dexterous manipulation [12], to reconfigurable robot locomotion [13].

To describe the frequency-dependent behavior of a single soft fluidic actuator, Mosadegh et al. introduced an analytic framework based on volume–pressure curves (V–P curves; denoted *PV curves* in the original article) [14]. The V–P curve shows a unique profile for each actuator design, actuation condition, and environment. Due to pressure uniformity in the actuation fluid, the pressure can be measured remotely (e.g., in the fluidic supply line) without interfering the soft actuator's behavior. Since the pressure reflects changes in stroke and velocity, it can be used as a primary indicator of the actuator's dynamic state.

In conventional rigid-body robotics, system dynamics and mechanical impedance are analyzed by quantifying how twist and its time derivatives correlate with wrench. In fluidically actuated soft systems, an analogous framework can be built on fluid volume and its time derivatives to capture motion, and on pressure to capture force. This approach not only leverages

This work was supported by the National Institute of Biomedical Imaging and Bioengineering of the NIH, award R01EB035574. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH. (*Corresponding author: Tommaso Ranzani.*)

Sang-Yoep Lee, Leonardo Zamora Yañez, Jacob Rogatinsky, Vi T. Vo, and Tanvi Shingade are with the Department of Mechanical Engineering, Boston University, Boston, MA 02215 USA (e-mail: slee6@bu.edu; zamora@bu.edu; jrogat@bu.edu; vitvo@bu.edu; tshing26@bu.edu).

Tommaso Ranzani is with the Departments of Mechanical Engineering, Biomedical Engineering, and the Materials Science and Engineering Division, Boston University, Boston, MA 02215 USA (e-mail: tranzani@bu.edu).

established analytical tools in robotics but also establishes a foundation for the effective control and performance optimization of soft robots. However, the nonlinear nature of the soft material combined with fluid dynamics complicates establishing such relationship using simple physics-based laws. Moreover, factors such as fabrication defects or air pockets in hydraulic channels introduce actuator-specific variability, which underscores the need for adaptive, data-driven modeling.

In this study, we introduce a modeling approach to achieve the (fluidic-)actuation-to-pressure relationship in an inflatable soft actuator that balances prediction accuracy and model complexity. Taking a three-degree-of-freedom (3-DoF) SBA (Fig. 1(a)) based on [11] as a benchmark platform to collect datasets, three types of models, physics-inspired exponential, neural network, and multivariate polynomial models, were proposed and navigated. After identifying the model that offers the optimal balance between accuracy and complexity, we used it to tackle three key challenges associated with inflatable soft robots: dynamic characterization of SBAs to detect the presence of air pockets, differentiation of SBAs varying in size and design through tailored characterization, and proprioceptive estimation of external forces. Along with the potential to expand soft robotic analysis toward dynamics using the proposed modeling techniques, we also share important findings used throughout the study, such as the fact that fluid volume and flow rate (without acceleration) dominantly governs the pressure change and that the prediction performance can be improved by involving previous step terms.

II. MODELING VOLUME-FLOW-PRESSURE RELATION

To investigate the volume-flow-pressure relationship, a 3-DoF SBA based on the design introduced in [11] was selected as the benchmark SBA. The SBA actuation system and experimental setup for the 3-DoF SBA are shown in Fig. 1(b). For actuating the three chambers of the SBA, we used three actuation units, one per chamber. Each unit consisted of a stepper motor (NEMA 11, STEPPERONLINE), lead screw, medical-grade syringe (7649-01, Hamilton), and pressure sensor (P-7100, Nidec Components). Directly attached to each motor shaft, the lead screw stages were connected to the plunger of the syringe with a 2 mm diameter tube. The syringe volumes were controlled with a precision of $0.04 \text{ mm}^3 \text{ mm}$ per step, following the motors' specifications. Hydraulic actuation was used throughout the study. The data were collected by injecting and withdrawing water from 0 mm^3 to 550 mm^3 while varying the volume flow rate up to $100 \text{ mm}^3/\text{s}$. Beyond this volume or flow rate, the SBA may rupture due to over-pressurization [10]. The data points were sampled at 25 Hz. Minimum 20 inflation/deflation cycles were recorded for each experimental condition (e.g., inflation volume, flow rate, and acceleration) to account for hysteresis in the V-P profile.

In Fig. 1(c), volume-pressure profiles for five flow rates from $20 \text{ mm}^3/\text{s}$ to $100 \text{ mm}^3/\text{s}$ are shown. To evaluate the extent to which pressure $P(t)$, fluid volume $v(t)$ and its time derivatives are linearly associated, we first computed Pearson

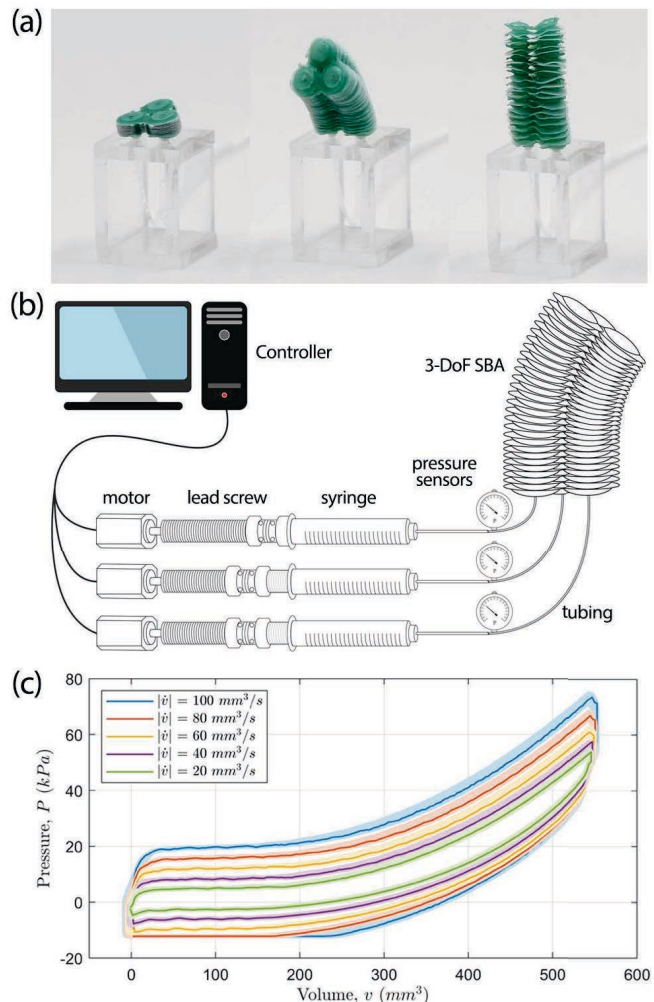


Fig. 1. **Overview of the experimental setup with a 3-DoF SBA.** (a) Fully deflated, partially inflated, and fully inflated 3-DoF SBA used in experiments. (b) Schematic overview of the actuation system for 3-DoF SBA. (c) V-P plot for a benchmark SBA across multiple inflation and deflation cycles at different $|\dot{v}|$. The shaded areas cover the range of actual trajectories for each flow rate, while the solid lines represent the average trajectory.

correlation coefficients¹ between synchronized samples of $P(t)$ and $v(t)$, $P(t)$ and $\dot{v}(t)$, and $P(t)$ and $\ddot{v}(t)$. The three correlations were: $\text{corr}(P, v) = 0.8029$, $\text{corr}(P, \dot{v}) = 0.4496$, and $\text{corr}(P, \ddot{v}) = 0.0965$. Considering the components used in hydraulic actuation such as tubing, these results indicate that the dynamics are primarily governed by elastic contribution $v(t)$ and viscous contribution $\dot{v}(t)$, with little contribution from inertial term $\ddot{v}(t)$.

Principal Component Analysis (PCA) was also performed with the dataset. The dataset was defined as:

$$X = \begin{bmatrix} v(t_1) & \dot{v}(t_1) & \ddot{v}(t_1) & P(t_1) \\ v(t_2) & \dot{v}(t_2) & \ddot{v}(t_2) & P(t_2) \\ \vdots & \vdots & \vdots & \vdots \\ v(t_N) & \dot{v}(t_N) & \ddot{v}(t_N) & P(t_N) \end{bmatrix} \quad (1)$$

¹ $\text{corr}(X, Y) = \text{cov}(X, Y) / \sigma_X \sigma_Y$, where $\text{cov}(X, Y)$ is the covariance between X and Y , and σ_X and σ_Y are the standard deviations of X and Y , respectively.

TABLE I
PRELIMINARY ANALYSIS: PCA RESULTS

		PC1	PC2	PC3	PC4
V	$v(t)$	0.61	-0.31	-0.38	-0.62
	$\dot{v}(t)$	0.34	0.70	-0.53	-0.35
	$\ddot{v}(t)$	-0.13	0.64	0.75	0.02
	$P(t)$	0.70	0.05	0.06	0.71
	$\vec{\lambda}_{norm}$	0.48	0.26	0.24	0.02

where $X \in \mathbb{R}^{N \times 4}$ consisting of columns $[v, \dot{v}, \ddot{v}, P]$ over N time points, with each row representing the corresponding values of volume $v(t)$, volumetric flow rate $\dot{v}(t)$, volumetric acceleration $\ddot{v}(t)$, and pressure $P(t)$ at a specific $t = t_i$. PCA was performed on the normalized data matrix $X' = (X - \mu)/\sigma$, where μ and σ as the mean and standard deviation of each column of X , respectively. The eigenvectors V and normalized eigenvalues $\vec{\lambda}_{norm}$ were computed according to $(\frac{1}{N-1}X'^T X')V = \Lambda V$ and $\vec{\lambda}_{norm} = \frac{\vec{\lambda}}{tr(\Lambda)}$, where N is the number of observations, $\Lambda = \text{diag}(\vec{\lambda})$ is the matrix of eigenvalues, and V is the matrix of eigenvectors.

Table I summarizes the PCA results, reporting both the loadings and the variance explained by each component. PC1 (48%) exhibits large loadings for $P(t)$ (0.70) and for $v(t)$ (0.61), along with a moderate loading for $\dot{v}(t)$ (0.34). PC2 (26%) and PC3 (24%) show large loadings for $\dot{v}(t)$ and $\ddot{v}(t)$ but near-zero loadings for $P(t)$ (0.05 and 0.06, respectively), indicating that these components do not account for variation in pressure. PC4 (2%) displays notable loadings for $P(t)$ (0.71) and $v(t)$ (-0.62), but given its minimal contribution to the overall variance, it is considered negligible and not used to draw conclusions.

These PCA findings are consistent with the Pearson correlation coefficients. Pressure co-varies primarily with volume and flow rate, with little association to acceleration. These results motivate the inclusion of higher-degree cross terms, such as $v(t) \cdot \dot{v}(t)$, to better capture the underlying nonlinear dynamics since the dominant components of $P(t)$ arose from combinations of $v(t)$ and $\dot{v}(t)$. This insight justify the need to

investigate more complex models to fully capture the system's dynamics.

Based on these PCA results, we modeled $P(t)$ using $v(t)$ and $\dot{v}(t)$, excluding $\ddot{v}(t)$. The mathematical form of the models used in the following sections are summarized in Table II.

A. Physics-Inspired Exponential Model

From a physical standpoint, two dominant behaviors govern the pressure inside a soft robotic system. First, the hyper-elastic properties of the material drive a nonlinear expansion as the internal volume changes [15]. Second, the fluid flow through the tubing can be assumed to be governed by Hagen–Poiseuille's law, indicating that the flow rate influences the fluid pressure [16]. We model these two effects as additive contributions: one term addressing the chamber's elastic response, and another accounting for the flow resistance.

Numerous analytic basis functions can serve to approximate the complex and nonlinear nature of soft robotic materials. In this section, we focused on exponential terms in volume, deferring a polynomial model description to Section II-C. The sum of exponential terms was used to capture the elastic expansion of the SBA chamber as a representation of the hyper-elastic relationship, while the affine term in \dot{v} addresses the Hagen–Poiseuille's law.

By analyzing the impact of numbers of exponential terms, this model was used to assess the best achievable fit accuracy without volume-flow interaction terms.

B. Neural Network Model

1) *Baseline Neural Network Model*: Neural network models were introduced to investigate whether allowing greater modeling flexibility could lead to improved regression performance, without imposing an explicit analytical relationship between $v(t)$ and $\dot{v}(t)$. While large and complex neural networks may yield higher accuracy, we constrained the number of parameters to be of the same order of magnitude as those of the other models to ensure model simplicity. These baseline neural networks used fully connected hidden layers with a fixed width

TABLE II
PRESSURE ESTIMATION MODELS

Models	Equations
Physics-inspired exponential model	$\hat{P}_{exp}(t) = \sum_{i=1}^k \alpha_i \cdot \exp(\beta_i v(t)) + \gamma \dot{v}(t) + \delta,$ k : the number of exponential terms; $\alpha_i, \beta_i, \gamma, \delta$: model parameters determined from data.
Neural network model (baseline and expanded)	$\hat{P}_{NN}(t) = f_{NN}(v(t), \dot{v}(t); \theta_{NN}),$ f_{NN} : a neural network with two inputs $\{v(t), \dot{v}(t)\}$ and one scalar output $\hat{P}_{NN}(t)$; θ_{NN} : the parameter vector including trained weights and biases.
Autoregressive neural network model	$\hat{P}_{NN_AR}(t) = f_{NN_AR}(\{v(t - \Delta t_i), \dot{v}(t - \Delta t_i)\}_{i=0 \dots p}; \theta_{NN_AR})$ f_{NN_AR} extends f_{NN} to include p previous time steps $v(t - \Delta t_i), \dot{v}(t - \Delta t_i)_{i=0 \dots p}$, with $\Delta t_0 = 0$.
Multivariate polynomial model	$\hat{P}_{poly}(t) = \sum_{i=0}^n \sum_{j=0}^m \alpha_{i,j} (v(t))^i (\dot{v}(t))^j$ n, m : the highest degrees of v and \dot{v} , respectively; $\alpha_{i,j}$: a parameter to be determined from experimental data.
Autoregressive multivariate polynomial model	$\hat{P}_{poly_AR}(t) = \sum_{i=0}^n \sum_{j=0}^m \alpha_{i,j} (v(t))^i (\dot{v}(t))^j + \sum_{k=1}^p (\beta_{k,1} v(t - \Delta t_k) + \beta_{k,2} \dot{v}(t - \Delta t_k))$ $\alpha_{i,j}, \beta_{k,1}, \beta_{k,2}$: parameters to be determined from experimental data.

of two neurons, and the network depth was varied from one to fifteen layers. This architectural choice was informed by physical insights derived from preliminary correlation analysis and principal component analysis (PCA). Training was conducted in MATLAB using `fitnet` with the Levenberg–Marquardt (LM) optimizer and a rectified linear unit (ReLU) activation function. The dataset was internally partitioned into training, validation, and test sets with a 70:15:15 split. A maximum of 3,000 epochs and a learning rate of $1 \cdot 10^{-7}$ were used as stopping criteria.

Neural network models with autoregressive inputs were also explored to assess whether incorporating short-term memory of state changes could enhance estimation performance, particularly in the presence of temporal factors such as measurement delays. Up to five autoregressive terms (corresponding to 0.2 s) were appended to the input vector to address potential signal-level delays without substantially increasing the model's degrees of freedom. To maintain consistency in network architecture and training, the number of hidden layer nodes was proportionally scaled to match the expanded input dimension.

2) *Expanded Neural Network Model*: The neural network search space across architecture and training parameters were systematically expanded to examine how much predictive performance could be gained by relaxing model constraints. The hyperparameter search space is summarized in Table III, and a total of 200 optimization trials were performed—100 using random search and 100 using Bayesian optimization with Gaussian process priors. Random search was performed manually in MATLAB, while Bayesian optimization was conducted using MATLAB's `bayesopt` function.

Another variation was explored by incorporating Long Short-Term Memory (LSTM) layers. As a type of recurrent neural network, LSTMs are commonly employed to handle time-series datasets. These models were used to cross-validate the performance of neural network models with autoregressive inputs by expanding the temporal window used to predict the current state. Total 50 trainings were performed, using MATLAB's `trainNetwork` function.

C. Multivariate Polynomial Model

Based on insights from the results of physics-inspired exponential and neural network models, multivariate polynomial models were introduced to explore the effect of higher-degree cross-terms of v and $\dot{v}(t)$, while retaining an explicit analytical form. Compared to the physics-inspired exponential model, the way we capture hyper-elastic behavior was changed from using exponential terms to using polynomial terms of v and \dot{v} .

Multivariate polynomial models with autoregressive terms were also evaluated. Rather than considering all possible combinations of v , \dot{v} , and their previous terms, only linear autoregressive terms were included. This choice kept the number of terms below 20 while still accounting for temporal effects from the previous terms.

D. Model Evaluation Criteria

To assess both the predictive accuracy and the complexity of the candidate models, we employed the following four metrics:

TABLE III
HYPERPARAMETER SEARCH SPACE FOR EXPANDED NEURAL NETWORK MODELS

Hyperparameter	Range
number of hidden layers	1 to 9
first-layer width	2 to 10
width tapering ratio between layers	0.4 to 1.0
learning rate	10^{-4} to 10^{-2}
mini-batch size	2^5 to 2^{10}
L2-regularization coefficient	10^{-6} to 10^{-2}
dropout rate	0 to 0.5
maximum number of training epochs	60 to 250
activation functions*	ReLU, leakyReLU, tanh, sigmoid, swish, ELU, GELU

*ReLU: $\max(0, x)$; leakyReLU: $\max(0.01x, x)$; tanh: $\tanh(x)$; sigmoid: $\sigma(x) = 1/(1 + e^{-x})$; swish: $x \sigma(x)$;

ELU: $\begin{cases} x & x > 0 \\ e^x - 1 & x \leq 0 \end{cases}$;

GELU: $x \Phi(x)$ where $\Phi(x)$ is the standard normal cumulative distribution function (CDF).

- **Root Mean Squared Error (RMSE)**. This measures the typical magnitude of the prediction error:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (P(t_i) - \hat{P}(t_i))^2}, \quad (2)$$

where \hat{P} denotes the predicted value, and N is the total number of data points. A lower RMSE indicates smaller residuals on average.

- **Adjusted R^2** . Unlike the standard R^2 value, the adjusted R^2 penalizes model complexity. For a model with k predictors fit to N data samples, the adjusted R^2 is defined as:

$$R_{\text{adj}}^2 = 1 - \frac{\sum_{i=1}^N (P(t_i) - \hat{P}(t_i))^2}{\sum_{i=1}^N (P(t_i) - \bar{P})^2} \cdot \frac{N - 1}{N - k - 1}, \quad (3)$$

where $\bar{P} = \sum_{i=1}^N P(t_i)/N$. The adjusted R^2 increases only if a newly added predictor improves the model more than would be expected by chance.

- **Corrected Akaike Information Criterion (AICc)**. As a model selection tool, the Akaike information criterion (AIC) estimates the prediction error to compare the relative quality of candidate models for a given dataset [17]. It balances the goodness of fit against model simplicity, addressing both overfitting and underfitting. The standard form can be written as $\text{AIC} = 2\nu - 2\ln(\mathcal{L})$, where ν is the number of estimated parameters and $\ln(\mathcal{L})$ is the log-likelihood of the model. Under the assumption of Gaussian regression, the log-likelihood term $\ln(\mathcal{L})$ is approximated by the following equation [18]:

$$\ln(\mathcal{L}) \approx -\frac{N}{2} \left[\ln \frac{2\pi}{N} \sum_{i=1}^N (P(t_i) - \hat{P}(t_i))^2 + 1 \right]. \quad (4)$$

In this work, we employed the AICc to more strongly penalize model complexity [17]. For finite sample sizes, the AICc introduces an additional penalty term:

$$\text{AICc} = \text{AIC} + \frac{2\nu(\nu + 1)}{N - \nu - 1}. \quad (5)$$

Lower values of AICc indicate a better trade-off between fit quality and parameter count, especially when N is not significantly larger than ν . Aside from AICc, there have been several studies that utilized the Bayesian information criterion (BIC), given by $BIC = \nu \ln(N) - 2 \ln(\mathcal{L})$, for model optimization. BIC generally emphasizes more on larger values of N than AICc. However, for our datasets, the absolute values of AICc and BIC tended to be nearly identical, while AICc showed higher sensitivity to ν . This difference can be shown with their partial derivatives:

$$\frac{\partial AICc}{\partial \nu} = 2 + \frac{(4\nu + 2)(n - \nu - 1) + 2\nu(\nu + 1)}{(n - \nu - 1)^2} \quad (6)$$

and

$$\frac{\partial BIC}{\partial \nu} = \ln(N). \quad (7)$$

Thus, AICc was more appropriate in our case.

- Time Complexity.** For real-time state estimation and control of systems, optimizing computation efficiency is important. Given the number of data points N , the number of parameters ν , the data matrix $X \in \mathbb{R}^{N \times \nu}$, and the regression vector $y \in \mathbb{R}^N$, the time complexity for all the physics-based exponential models and multivariate polynomial models is estimated as $O(N \cdot \nu^2)$ due to computation of least squares parameters $(X^T X)^{-1} X^T y$. On the other hand, the time complexity for the neural network model can be written as $O(\text{epochs} \cdot N \cdot \nu)$ where the number of parameters ν equals the number of network weights, which grows with neurons and layers. For reference, all computations in this work were performed on a single desktop PC equipped with an Intel Xeon E5-1660 CPU (3.30 GHz)², 64 GB RAM, and a GeForce GTX 1080 GPU, using MATLAB R2024b on Windows 10.

E. Model Optimization

For each model, the optimal parameters were determined by minimizing a joint cost function that linearly combines the evaluation criteria introduced in the previous section:

$$\theta^* = \arg \min_{\theta} \left[w_1 \text{RMSE} + w_2 (1 - R_{\text{adj}}^2) + w_3 \text{AICc} \right] \quad (8)$$

where w_1, w_2, w_3 are weights balancing the relative importance of predictive error and model complexity. Note that one's complement of the term R_{adj}^2 , $1 - R_{\text{adj}}^2$ was used for minimization consistency. In many cases, reducing RMSE simultaneously lowered $(1 - R_{\text{adj}}^2)$ and AICc, and encouraged the removal of unnecessary components (e.g., dropping negligible terms or neurons) when possible. Throughout this procedure, the model reduction was performed without harming the model's general form (i.e. the number of exponential terms or highest exponents of $v(t)$ and \dot{v}).

²Commercially available embedded systems operate at frequencies range from 16 MHz.

TABLE IV
SUMMARY OF REGRESSION PERFORMANCES

Physics-inspired exponential model				
k	ν	RMSE	R_{adj}^2	AICc
1	4	1.2986	0.9934	2.7573e+5
2	6	1.1100	0.9952	2.4999e+5
3*	8	1.0998	0.9952	2.4849e+5
4	10	1.0998	0.9952	2.4848e+5
5	12	1.0997	0.9952	2.4847e+5
Baseline neural network model				
d	ν	RMSE	R_{adj}^2	AICc
1	5	4.4011	0.9239	4.7604e+5
2	9	0.7067	0.9980	1.7591e+5
			...	
7	29	0.4292	0.9993	9.4124e+4
8*	33	0.3213	0.9996	4.6604e+4
9	37	0.3150	0.9996	4.3371e+4
			...	
14	57	0.2947	0.9997	3.2574e+4
15	61	0.2945	0.9997	3.2335e+4
Autoregressive neural network model				
(p, d)	ν	RMSE	R_{adj}^2	AICc
(0, 8)	33	0.3319	0.9995	2.6367e+4
(1, 8)	41	0.3324	0.9995	2.6649e+4
(2, 8)	57	0.3302	0.9995	2.6027e+4
(3, 8)	73	0.3304	0.9995	2.6020e+4
(4, 8)	89	0.3330	0.9995	2.6616e+4
(5, 8)*	105	0.3321	0.9995	2.6321e+4
Multivariate polynomial model				
(n, m)	ν	RMSE	R_{adj}^2	AICc
(2, 1)	6	1.4291	0.9920	2.9146e+5
(2, 2)	9	1.2165	0.9942	2.6503e+5
(2, 3)	12	1.4097	0.9922	2.8922e+5
(3, 1)	8	1.0644	0.9955	2.4312e+5
(3, 2)*	12	0.6884	0.9981	1.7160e+5
(3, 3)	16	1.1861	0.9945	2.6090e+5
(4, 1)	10	2.6945	0.9715	3.9554e+5
(4, 2)	15	2.6926	0.9715	3.9542e+5
Autoregressive multivariate polynomial model				
(p, n, m)	ν	RMSE	R_{adj}^2	AICc
(0, 3, 2)	12	0.4603	0.9990	5.4029e+4
(1, 3, 2)	14	0.4521	0.9991	5.2339e+4
(2, 3, 2)	16	0.4450	0.9991	5.0829e+4
(3, 3, 2)	18	0.4388	0.9991	4.9480e+4
(4, 3, 2)	20	0.4332	0.9992	4.8242e+4
(5, 3, 2)*	22	0.4280	0.9992	4.7063e+4

k : number of exponential terms; ν : degree(s) of freedom or number of parameters; d : depth of fully connected neural network; p : order of autoregressive model; n : maximum exponent of volume; m : maximum exponent of volume change rate.

^a Models with * are pseudo-optimal sets within each model type.

^b AICc scores can be compared within each model type.

^c All the metrics are in-sample (from the full dataset) performances.

III. MODEL EVALUATION

In this section, we evaluate and discuss the fitting performance of each model. RMSE was the primary goodness-of-fit metric, and AICc values were plotted together to visualize relative model parsimony within each model category for visualization. The R_{adj}^2 values and error bars were omitted for clarity. The evaluation scores are summarized in Tables IV and V³.

³Detailed statistical metrics are provided in the Supplementary Material; see the Data and Code Availability section for access.

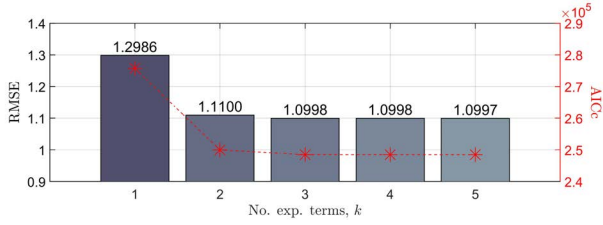


Fig. 2. **Regression performances using a physics-inspired exponential model.** The bar plot shows the RMSE values, while the red stars indicate the corresponding AICc values for varying numbers of exponential terms (k).

A. Physics-Inspired Exponential Model

Fig. 2 shows results for physics-inspired exponential models with up to five exponential terms ($k = 1, \dots, 5$). The estimation performance showed only a minor improvement ($\Delta\text{RMSE} < 0.01$) beyond one exponential term. Including more than five exponential terms did not lead to significant improvement in performance ($\Delta\text{RMSE} < 0.0001$). AICc values followed the trend of RMSE values, which may result from the increase in model complexity being canceled out by the decrease in error. Using three exponential terms showed the best balance of accuracy and simplicity (eight parameters, RMSE of 1.0998, R_{adj}^2 of 0.9952, and AICc of $2.4849e+5$).

Since the term $\dot{v}(t)$ was constrained to an affine form, increasing the number of exponential terms only increased the complexity in the $v(t)$ component of the model. This result indicated that a more intricate coupling of $v(t)$ and $\dot{v}(t)$ is needed, as hinted by the PCA study.

B. Neural Network Model

1) *Baseline Neural Network Model:* The baseline neural network models were evaluated by varying the number of layers up to 15 ($d = 1, \dots, 15$; Fig. 3(a)). A drastic decrease in the RMSE from 4.40 to 0.71 was observed after increasing the depth beyond one, as the network began to differentiate from a least-squares model of v and \dot{v} with a nonlinear activation function. The performance gradually improved until a depth of eight (RMSE of 0.71 with depth two and RMSE of 0.23 with depth eight), after which the performance saturated. Overall, the neural network models achieved better performance (lower RMSE and AICc) than the physics-inspired exponential models. As for the physics-inspired exponential models, the AICc values followed a similar trend to the RMSE values, showing a drastic improvement after depth seven (AICc of $1.7591e+5$ with depth two and AICc of $4.6604e+4$ with depth eight).

Fig. 3(b) shows the results from the neural network models with autoregressive inputs. Models with up to eight layers and autoregressive order p up to five were evaluated. Incorporating at least one previous time step significantly improved performance (reducing the RMSE up to 36%, when increasing the AR order p from 0 to 1). Adding more than one autoregressive input did not lead to a substantial improvement (RMSE of 0.3324 with AR order one and 0.3321 with AR order five).

Two key insights were drawn from the results of the neural network models. First, including at least one prior time step is important for pressure estimation. Second, a regression model

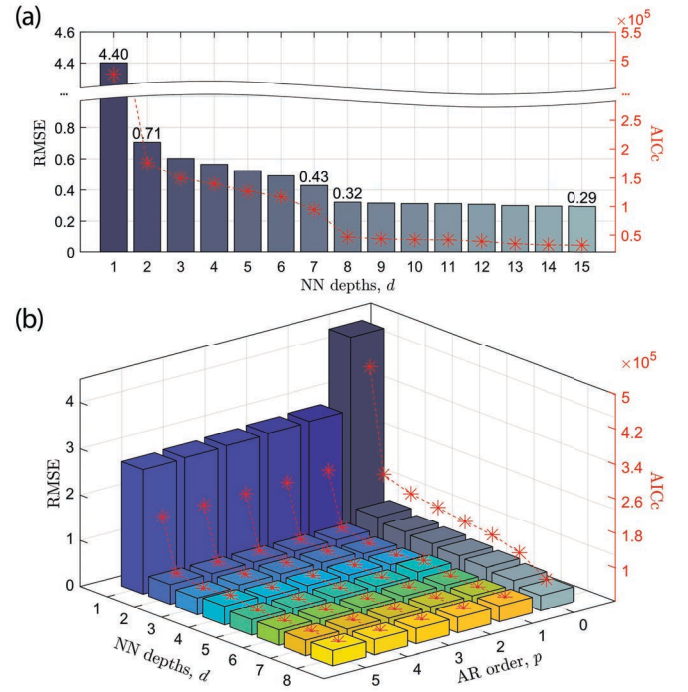


Fig. 3. **Regression performances using neural network models.** (a) The bar plot shows the RMSE values, while the red stars indicate the corresponding AICc values for varying numbers of neural network depths, d , without autoregressive inputs. (b) The bar plot displays the RMSE values, while the red stars represent the corresponding AICc values for varying network depths, d , and autoregressive orders, p . The gray bars ($p = 0$) represent the performance of the neural network models without autoregressive inputs shown in (a).

can be established without using more than 100 parameters (i.e., the simple neural network model with depth of 15 showed RMSE of 0.2945, R_{adj}^2 of 0.9997, and AICc of $3.2335e+4$, which was the best performance compared to all the other models).

2) *Expanded Neural Network Model:* The prediction performance of the expanded neural network models is summarized in Table V. The network architectures obtained through optimization techniques (e.g., random search and Bayesian optimization) exhibited greater degrees of freedom compared to the baseline neural network models. In these cases, the AICc values were mostly undefined because the number of parameters exceeded the limit imposed by the data size ($N - \nu - 1 \leq 0$). The top-performing models using ReLU and tanh activation functions achieved RMSE values between 0.34 and 0.35, which is comparable to the baseline neural network models of depth greater than eight, where RMSE values fell below 0.33. When compared with baseline neural network models incorporating autoregressive inputs, both RMSE and R_{adj}^2 remained similar, at approximately 0.33 and 0.9995, respectively.

The LSTM models exhibited performance trends similar to the physics-inspired exponential models, with RMSE values exceeding 1.0801 and R_{adj}^2 values below 0.9954. Overall, the LSTM models performed worse than the baseline neural network models with autoregressive inputs. This inferior performance likely stems from a mismatch between the LSTM architecture and the characteristics of the dataset: the limited

TABLE V
SUMMARY OF REGRESSION PERFORMANCES OF EXPANDED NEURAL NETWORK MODELS

Architecture*	ν^\dagger	RMSE	R_{adj}^2
[1024, 909, 794]	ReLU	1.6581e+06	0.3407 0.9995
[128, 122, 116]	ReLU	30,507	0.3420 0.9995
[128]	ReLU	513	0.3494 0.9995
[1024, 801]	tanh	8.249e+05	0.3528 0.9994
[1024, 941, 858]	ReLU	1.7767e+06	0.3571 0.9994
[256, 213, 170]	tanh	92,060	0.3839 0.9994
[256]	tanh	741	0.3937 0.9993
		...	
[1024, 1000, 976, 952, 928, 904, 880,]	leakyReLU	6.2107e+06	0.4584 0.9991
[128, 107, 85]	swish	23,453	0.6079 0.9983
		...	
[32, 30, 28, 26, 24, 22, 20]	GELU	4,387	2.3917 0.9741
[128, 120]		5.3069e+5	1.0801 0.9954
[64, 60, 57]	biLSTM [‡]	2.3360e+5	1.3609 0.9927
[256, 225]		2.1100e+6	1.3637 0.9927

*Bracketed list shows hidden-layer widths from input to output; the scalar output layer is omitted.

[†]AICc is not reported since the term is mostly undefined in these models ($\because N - \nu - 1 \leq 0$).

[‡]biLSTM incorporated both sigmoid and tanh.

sequence length and relatively small data size reduce the benefits of temporal modeling while increasing the risk of overfitting due to the larger number of trainable parameters. Consequently, the LSTM models could not generalize as effectively as simpler autoregressive neural networks in this setting.

C. Multivariate Polynomial Model

The multivariate polynomial models were evaluated by varying both the maximum exponents for $v(t)$ and $\dot{v}(t)$ from one up to seven for each ($n = 1, 2, \dots, 7$ and $m = 1, 2, \dots, 7$), as shown in Fig. 4(a). The highest-degree term's coefficients were constrained to be nonzero to prevent degenerate solutions (producing a reduced-degree model). A drastic decrease in performance (increase in both RMSE and AICc values; around 138% increase in RMSE) was observed after including terms with more than degree five.

Using a degree of three for $v(t)$ and a degree of two for $\dot{v}(t)$ ($n = 3, m = 2$) yielded significantly better performance than all the other polynomial models. Notably, this model's RMSE was 35% to 74% lower than that of models with nearby degree combinations ($n = 2, m = 1$ through $n = 4, m = 3$). This result suggests that beyond a certain complexity, increasing the degree does not significantly improve the model's ability to describe the system's behavior. It also points to the potential for simpler models that still capture the essential dynamics of the system effectively.

To further investigate the identified optimal model ($n = 3, m = 2$), the performance was evaluated after excluding one or more terms based on their contributions to the model, which were assessed by the magnitude of the coefficients and p-values from a two-sided t-statistic, with the null hypothesis that each coefficient is zero. The results are summarized in Table VI. In all cases, excluding a term degraded performance. The exclusion of terms was extended to adjacent models with

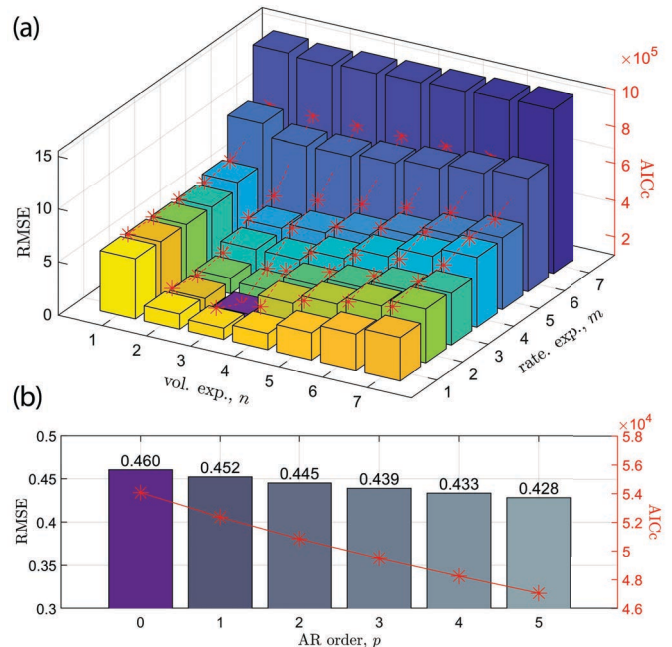


Fig. 4. Regression performances using multivariate polynomial models. (a) The bar plot shows the RMSE values, while the red stars represent the corresponding AICc values for varying maximum exponents of volume, n , and maximum exponents of volume change rate, m . The purple bar ($n = 3, m = 2$) indicates the best results among the adjacent models. (b) Regression performances using multivariate polynomial models with autoregressive inputs, where the maximum exponent of volume is three and the maximum exponent of volume change rate is two ($n = 3, m = 2$). The bar plot displays the RMSE values, while the red stars represent the corresponding AICc values for varying autoregressive orders, p . The purple bar ($p = 0$) shows the best performance multivariate polynomial model without autoregressive inputs.

different complexities, but no model outperformed the original model. Even when a parameter's magnitude is very small ($< 10^{-10}$), it can be concluded that all the terms remained essential for accurately representing the dataset.

Taking insight from the neural network model with autoregressive inputs, multivariate polynomial models with linear autoregressive terms up to a degree of five were inspected for the optimal case ($n = 3, m = 2$). The results are shown in Fig. 4(b). Unlike the autoregressive neural network models (Fig. 3(b)), here RMSE and AICc kept improving as the autoregressive order increased. Models with more autoregressive terms were not considered due to concerns about overfitting and a long initial transient.

D. Comparative Performance Review

Throughout our results, the number of parameters used in the physics-inspired exponential models and multivariate polynomial models was equal to or less than 22 ($\nu \leq 22$). On the other hand, the baseline neural network models involved a larger number of parameters ($\nu \leq 105$) and required up to 3,000 training epochs to converge. The expanded neural network models involved even more parameters ($\nu \geq 500$), implying a higher capacity to address complex nonlinearities. The actual computational time for inspecting all the possible models within each model category (e.g., physics-inspired ex-

TABLE VI
COEFFICIENTS OF THE OPTIMAL MULTIVARIATE POLYNOMIAL MODEL
AND REGRESSION PERFORMANCES OF REDUCED VARIANTS

Multivariate polynomial model: ($n = 3, m = 2$)					
term	coeff.	term	coeff.	term	coeff.
1	0	\dot{v}	1.7660e-1	\dot{v}^2	2.3010e-4
v	1.2695e-2	$v\dot{v}$	-1.3896e-4	$v\dot{v}^2$	-3.0150e-6
v^2	-8.0664e-5	$v^2\dot{v}$	6.6527e-7	$v^2\dot{v}^2$	1.3800e-8
v^3	4.1269e-7	$v^3\dot{v}$	-7.3542e-10	$v^3\dot{v}^2$	-1.3773e-11

Reduced multivariate polynomial models				
excluded term(s)	ν	RMSE	R^2_{adj}	AICc
$v\dot{v}$	11	0.6934	0.9981	1.7279e5
$v^3\dot{v}$	11	0.6943	0.9981	1.7301e5
$v\dot{v}, v^2\dot{v}$	10	0.6971	0.9981	1.7367e5
$v\dot{v}, v^3\dot{v}$	10	0.6943	0.9981	1.7301e5
$v\dot{v}, v^3\dot{v}$	10	0.6943	0.9981	1.7301e5
$v^2\dot{v}, v^3\dot{v}$	10	0.6962	0.9981	1.7345e5

The least-contributing terms among the 12 terms of the multivariate polynomial model ($n = 3, m = 2$) were selectively excluded based on the p-value of the t-statistic for a two-sided test with the null hypothesis that the coefficient is zero.

ponential model, neural network model, and multivariate polynomial model) matched the time complexity analysis in the previous section ($O(N \cdot \nu^2)$ for the physics-inspired exponential models and multivariate polynomial models; $O(\text{epochs} \cdot N \cdot \nu)$ for the neural network models). The physics-inspired exponential models required 2.31 s to optimize models up to 10 exponential terms, and multivariate polynomial models took 13.03 s to optimize 49 models with the maximum degrees of $v(t)$ and $\dot{v}(t)$ ranging from one to seven, respectively. In the case of baseline neural network models, 225.38 s were required to optimize models with network depths up to 15. Training a single expanded neural network model required at least 500 s under the same computational conditions, exceeding the total optimization time required by all other model categories. Considering an average time for optimizing one model from each category, the neural network models required more time than the physics-inspired exponential and multivariate polynomial models because of the many training epochs required to achieve an accurate prediction of $P(t)$.

The accuracy in the pressure prediction for each model category was also assessed, excluding the expanded neural network models due to excessive degrees of freedom. For comparison, one benchmark (pseudo-optimal) model was selected from each model category that compromises between pressure estimation performance and model complexity. The selected benchmark models were marked with * in Table IV and listed below:

- 1) **exp**: A physics-inspired exponential model with three exponential terms of $v(t)$ ($k = 3$)
- 2) **NN**: A baseline neural network model with eight fully connected layers ($d = 8$)
- 3) **NN-AR**: An autoregressive neural network model with an autoregressive order of five and eight fully connected layers ($p = 5, d = 8$)
- 4) **poly**: A multivariate polynomial model with a maximum exponent of three for $v(t)$ and a maximum exponent of two for $\dot{v}(t)$ ($n = 3, m = 2$)
- 5) **poly-AR**: An autoregressive multivariate polynomial

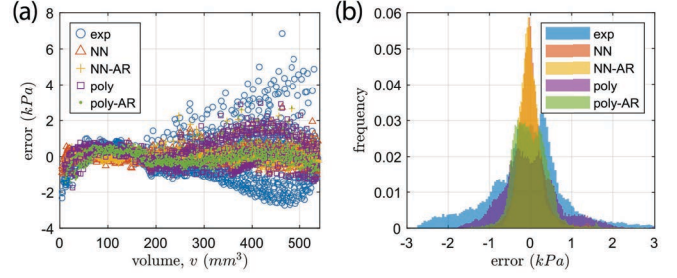


Fig. 5. **True vs. estimated residual plots.** (a) Scatter plot of errors vs. volume, for the optimal models of each model type. (b) Error distributions for optimal models in each model type.

model with an autoregressive order of five, a maximum exponent of three for $v(t)$, and a maximum exponent of two for $\dot{v}(t)$ ($p = 5, n = 3, m = 2$)

In Fig. 5(a), errors are scatter-plotted with respect to $v(t)$ for each benchmark model. The physics-inspired exponential model exhibited greater errors in the larger volume regions with a maximum error of 6.84 kPa. The average and standard deviation of absolute errors were 0.76 ± 0.81 kPa. For volumes greater than 300 mm^3 , the hyper-elastic behavior has a greater influence than the fluid dynamics, and the model's limited number of parameters cannot capture this behavior. In contrast, the baseline neural network (0.22 ± 0.23 kPa) and multivariate polynomial models (0.52 ± 0.45 kPa) displayed a more even error distribution across volumes, with the neural network model generally producing lower errors. In both cases, models that included autoregressive terms provided lower errors (0.21 ± 0.25 kPa in NN_AR and 0.33 ± 0.28 kPa in poly_AR). This is presumably due to better handling of measurement noise and delays by using previous-state information. In Fig. 5(b), error distribution plots for the benchmark models are shown. The frequency of errors was obtained by generating histograms with a bin size of 0.05 kPa. Notably, the errors from the neural network models were closer to a normal distribution, whereas the other models exhibited more model-specific error variations distributed across different regions.

IV. PRACTICAL MODEL UTILIZATION CASES

We demonstrate three practical use cases to illustrate how our model contributes to extending soft robots into dynamic regimes. We focused on the benchmark multivariate polynomial model because its mathematical form made the forthcoming analyses convenient. For example, the multivariate polynomial model in Table II can be rewritten in the form below:

$$\begin{aligned}
 P(t) &= \sum_{i=0}^n \sum_{j=0}^m \alpha_{i,j} (v(t))^i (\dot{v}(t))^j \\
 &= \left(\sum_{i=0}^n \sum_{j=1}^m \alpha_{i,j} (v(t))^i (\dot{v}(t))^{j-1} \right) \dot{v}(t) + \sum_{i=0}^n \alpha_{i,0} (v(t))^i \\
 &= C(v(t), \dot{v}(t)) \dot{v}(t) + K(v(t)),
 \end{aligned} \tag{9}$$

which shows a clear connection to a standard equation of motion with negligible inertia. Moreover, conventional math-

ematical procedures, such as taking partial derivatives with respect to $v(t)$ or $\dot{v}(t)$, can be easily performed since the equation explicitly includes polynomial (cross-)terms.

The benchmark multivariate polynomial model was further assessed for computational efficiency and data dependency. The prediction process over a total of 100,000 data points, randomly selected from the modeling dataset, required 0.32 s on a standard computational environment. This corresponds to an average of 3.2 μ s per prediction, demonstrating the model's compatibility with real-time applications and its deployability on conventional embedded systems. The model's sensitivity to dataset size was also examined by analyzing performance across varying numbers of inflation/deflation cycles. The results showed that prediction performance saturated after approximately three full cycles (\approx 38,000 data points), with additional data yielding marginal improvements. Leveraging these features, we explore three direct applications.

A. Stiffness/Damping Analysis with Air Pocket

Because of water's incompressibility, hydraulic actuation offers precise control, high load-bearing capacity, and fast response. However, maintaining a completely watertight system is difficult, and over time air infiltration can alter the actuator's dynamic properties. An air pocket within the hydraulic actuation tubing may reduce the stiffness and damping of actuation due to its compressibility (Fig. 6(a)). If the presence of air pockets can be diagnosed and identified in real time, the model and control system can be optimally adjusted.

For a quantitative study, data were collected by injecting different amounts of air into the syringe using the same 3-DoF SBA from the modeling experiments, as shown in Fig. 6(b). First, we computed partial derivatives of the multivariate polynomial model in Table II with respect to $v(t)$ and $\dot{v}(t)$ at each data point as $k_i = \frac{\partial P}{\partial v} \Big|_{(v(t_i), \dot{v}(t_i))}$ and $c_i = \frac{\partial P}{\partial \dot{v}} \Big|_{(v(t_i), \dot{v}(t_i))}$, which we interpret as instantaneous stiffness k_i and damping c_i . Then, average stiffness, \bar{k} , and damping, \bar{c} , values were computed as $\bar{k} = \frac{1}{N} \sum_{i=1}^N k_i$ and $\bar{c} = \frac{1}{N} \sum_{i=1}^N c_i$. These averaged stiffness and damping were also computed separately for the inflation and deflation phases. Fig. 6(c) plots these average k and c for different air volumes, and Fig. 6(d) shows the distribution of pointwise k_i and c_i . Both plots show that larger air pockets result in lower overall stiffness and damping. These results not only demonstrated that air infiltration can be detected and managed by observing the actuation-to-pressure relationship but also that system dynamics can be monitored in real time through high-resolution measurements of stiffness and damping.

B. Identification of Different SBA Designs

SBAs can vary widely in geometry, materials, fabrication, and size, which may lead to distinct dynamic behaviors. Characteristic SBA design parameters are depicted in Fig. 7(a). To assess volume-flow-pressure modeling performance across different SBAs, we fabricated four distinct 3-DoF SBAs (Fig. 7(b)) and collected data for each to cross-validate the model. The design parameters of the four SBAs are listed in Table VII. We collected data over multiple inflation/deflation

TABLE VII
DESIGN PARAMETERS OF DIFFERENT SBAS

No.	chamber shape	d_o	d_c (d_{c1}, d_{c2})	h	max. vol.	max. press.	main material
1	reniform	14	7.4, 4.3	30	550	80	TPE
2	circular	12	5.3	32	450	70	TPU
3	reniform	11.3	6.2, 2.4	20	200	35	TPE
4	circular	48	23.5	120	7000	40	TPU
		mm	mm	mm	mm ³	kPa	

d_o : outer width; d_c : chamber diameter (d_{c1} and d_{c2} in the reniform cases); h maximum expandable height.

cycles at various flow rates (Fig. 7(c)). The maximum volume and flow rate for each SBA were selected to remain within safe operating limits. For validation, all 49 multivariate polynomial models with maximum exponents for $v(t)$ and $\dot{v}(t)$ from one to seven, respectively, were optimized on the dataset from each SBA. The combination $n = 3$ and $m = 2$ exhibited the best performance in all four cases, with each model containing 12 coefficients. Since all models share the same structure, averaged stiffness and damping values could be computed consistently across trajectories. The results including evaluation performances and averaged stiffness/damping coefficients are shown in Table VIII.

In addition to evaluating regression performance, the Chow test was applied to every combination of two models from different SBAs. The Chow test assesses whether two regression models are statistically different [19] or not. The analysis involves computing F-statistics based on the ratio of covariances from an integrated model to those from two independent models. Since all four SBAs were modeled using the same set of basis functions (e.g., the multivariate polynomial model with $n = 3$ and $m = 2$), each model is represented by its corresponding 12 coefficients. Rather than simply differentiating the models using these coefficients, the Chow test was employed to quantitatively evaluate the ability to distinguish between datasets from different SBAs. With a clear distinction between models from different SBAs, it becomes possible to identify a specific SBA even from a small dataset. The results are shown in Table IX. All pairwise tests between models from SBA1 through SBA4 produced significant differences, with p-values less than 10^{-7} , thereby rejecting the hypothesis that any two models are identical. To further illustrate the significance, the F-statistics are listed in Table IX. Since even the smallest F-statistic value ($F = 1118.32$) was greater than the critical value of 5.9 at a significance level of $\alpha = 0.0005$, the null hypothesis that any two models are identical was decisively rejected.

These results support the expanded applicability of the proposed volume-flow-pressure modeling approach, enabling cross-comparison among different SBAs using a single mathematical model form. Moreover, the results suggest a link between the multivariate polynomial model with $n = 3$ and $m = 2$ and the underlying complex nonlinear physics governing SBA actuation.

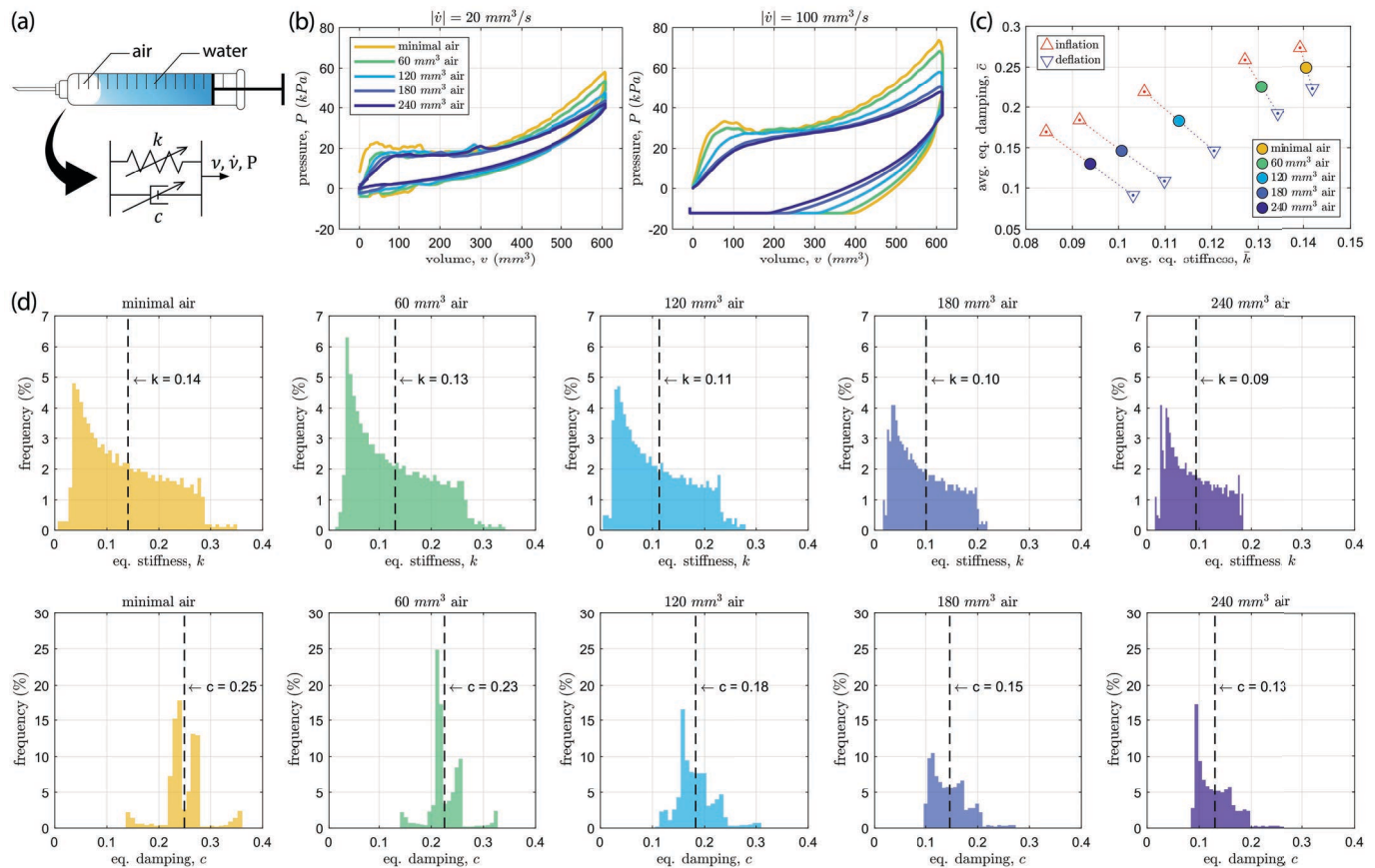


Fig. 6. **Stiffness and damping analysis with different volume of air pockets.** (a) Schematic of an air pocket inside the actuation syringe. The air pocket functions as a spring-damper. (b) V-P plots for different volumes of air in a chamber across multiple inflation and deflation cycles. The left plot includes trials with $|\dot{v}| = 20 \text{ mm}^3/\text{s}$, and the right plot includes trials with $|\dot{v}| = 100 \text{ mm}^3/\text{s}$. (c) The plot displays the averaged equivalent stiffness (\bar{k}) and damping coefficients (\bar{c}) for different volumes of air in a chamber. The red and blue triangles represent averaged equivalent stiffness and damping coefficients achieved only across inflation and deflation cycles, respectively. (d) Distribution of pointwise k_i and c_i for different air volumes in a chamber.

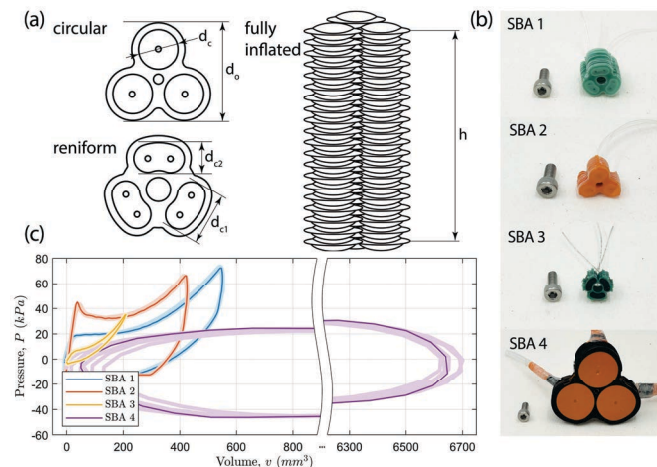


Fig. 7. **Different characteristics of four SBAs.** (a) Geometric parameters of SBA. (b) Four different SBAs tested in this study. As a reference, M4×10 mm hex screw was placed next to each SBA. (c) V-P curves of the four different SBAs.

C. External Force Estimation

The models presented so far were based solely on intrinsic measurements from actuating SBAs, without any influence

TABLE VIII
REGRESSION PERFORMANCES FOR DIFFERENT SBAs USING MULTIVARIATE POLYNOMIAL MODEL

No.	RMSE	R^2_{adj}	AICc	k	\bar{c}
1	1.4230	0.9943	3.5669e5	0.1561	0.1750
2	3.8981	0.9783	1.5228e4	0.1774	0.2308
3	0.5837	0.9974	2.0698e4	0.2305	0.5345
4	7.1198	0.9440	7.4670e3	0.0003	0.9176

kPa/mm³ kPa · s/mm³

TABLE IX
CHOW TEST RESULTS

	SBA1	SBA2	SBA3	SBA4
SBA1	-	12301.55	6052.92	1118.32
SBA2	12301.55	-	18891.99	6527.82
SBA3	6052.92	18891.99	-	2477.83
SBA4	1118.32	6527.82	2477.83	-

The noted values are F-statistics, and the critical values for each case were 5.9 (at significance level of $\alpha = 0.0005$).

from external forces. However, the pressure measurement will be affected by external forces. A force sensor is typically required to decouple actuation-induced pressure from externally induced pressure, but sensing integration is challenging due to

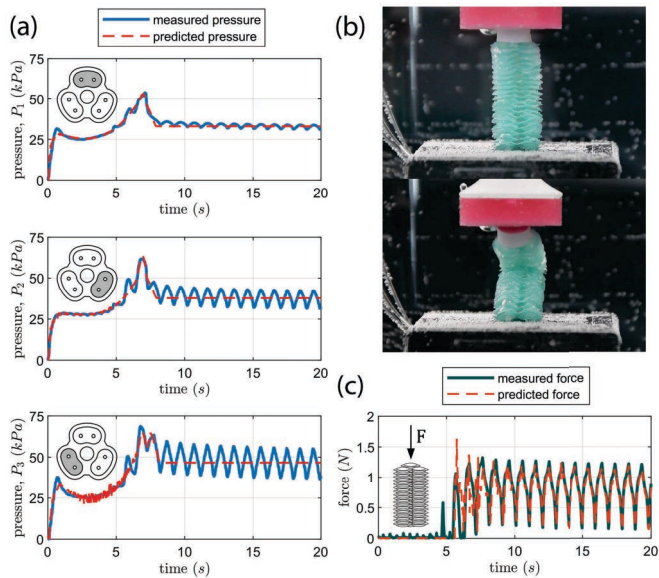


Fig. 8. **Proprioceptive external force estimation based on pressure.** (a) Measured and estimated pressure in each of the three chamber. (b) Experiment for estimating the net external force applied to the SBA. (c) The plot displays the measured and estimated net force applied to the SBA over time.

the SBA's form factor, material properties, and operating conditions. Since the optimized models were proven to accurately predict the actuation-induced pressure, the difference between the estimated and measured pressures can be correlated to an external force. Based on this hypothesis, we proposed a sensorless (or "proprioceptive") external force estimation method. Assuming our model gives the true pressure in the absence of an external load, the pressure residual in each chamber is integrated to calculate the magnitude of the external force.

Using the procedure reported in Section IV-B, we identified a volume-flow-pressure model for each of the 3-DoF SBA's three chambers. We then inflated the SBA with 500 mm³ of water. A robot arm (UR5e, UNIVERSAL ROBOTS) equipped with a force sensor (9105-TW-NANO17-E, ATI) was controlled to dynamically push the tip of the SBA 10 mm repeatedly at a rate of 1 Hz. The real-time pressure measurements and estimations based on individualized models for each chamber are shown in Fig. 8(a). The force from the robot arm induced an asymmetric compression in the three chambers (Fig. 8(b)), leading to a different pressure in each of them. The magnitude of the external force was calculated by multiplying the pressure differences by the effective cross-sectional area (22 mm² for each chamber) and summing the results. The estimation performance was evaluated against the force sensor measurements, yielding a mean error of 1.60 mN and a standard deviation of 7.50 mN. Across repeated trials with varying initial configurations (i.e., distance and pose between the SBA and the robot arm), the mean estimation error remained at approximately 6.40 mN, with a standard deviation of 5.50 mN. Although only the magnitude of the force could be estimated, the method demonstrated high precision without the need for an additional sensor.

V. DISCUSSION AND CONCLUSION

Inflatable soft actuators, like most soft robots, exhibit complex, nonlinear behavior due to their inherent material properties. Despite its advantages based on the incompressibility, hydraulic actuation adds complex dynamics throughout the actuation chain. The inflatable soft actuators' behaviors become further complicated by the two practical issues frequently addressed in the field of soft robotics: fabrication inconsistencies (and/or defects) and difficulties in complete removal of air in the fluidic actuation chain. These challenges inevitably necessitates continuous empirical tuning of the control system. This is further exacerbated by the difficulty of integrating sensors for measuring such characteristics.

In this study, we introduced an approach to address these challenges by establishing a model that relates between the intrinsic measurements. The input fluidic volume ($v(t)$) and its time derivatives ($\dot{v}(t)$, $\ddot{v}(t)$, ...) as well as pressure ($P(t)$) are the most fundamental measurements that can be measured from the fluidic actuation chain without interfering the soft robotic behavior. Our preliminary investigation revealed that the system's pressure dynamics are primarily governed by elastic contributions from $v(t)$ and viscous contributions from $\dot{v}(t)$, which helped constrain the model's degrees of freedom.

The actuation-to-pressure model was established by testing three types of models (e.g., physics-inspired exponential, neural network, and multivariate polynomial models) with incorporation of a model-selection criterion Eq. (8) that balances between the prediction accuracy and model complexity. The two key findings from the preliminary study were that second- (and-higher-) order time derivatives of volume impose negligible effect on pressure dynamics, whereas cross-terms between $v(t)$ and $\dot{v}(t)$ play a critical role. Based on these features, optimal ($R^2 > 0.99$) yet simple ($\nu < 105$) models were achieved in each model type. The most notable outcome was that the multivariate polynomial model with a maximum exponent of three for $v(t)$ and a maximum exponent of two for $\dot{v}(t)$ showed optimal prediction performance compared to all the adjacent combinations, with only twelve coefficients. This directly lead to that the model is realizable and can be implemented on deployable, resource-limited embedded systems (e.g., Arduino or Raspberry Pi), enabling real-time operation without the need for intensive computational resources. Furthermore, as the model consists of explicit polynomials of $v(t)$ and $\dot{v}(t)$, this framework accepts conventional robotics techniques—from Kalman filtering to Koopman operator theory [20]. This intuition was shown with Eq. 9.

Since the optimal multivariate polynomial model ($n = 3, m = 2$) obtained with the benchmark SBA also yielded the highest pressure prediction performance across different SBAs, variations in SBA characteristics could be captured through system identification process while maintaining the same model structure. Owing to its simple form, real-time, continuous model identification was feasible, as demonstrated in the air-pocket diagnosis and SBA differentiation studies. We envision that additional functionalities, such as fault detection and tracking time-varying degradation, can be realized through an extended modeling framework that builds upon

this foundational approach, potentially incorporating transfer learning to generalize across devices or conditions. A follow-up study will systematically address robustness and sensitivity across actuator designs, environmental conditions, and operational scenarios. Furthermore, the model's accurate pressure predictions enabled precise characterization of internal kinetics, thereby enabling precise proprioceptive force estimation without integrating a sensor.

In conclusion, we emphasize that the potential of this study is not limited to a specific form factor—SBA. Since our models accommodate a range of nonlinearities such as hyper-elastic behavior, hysteresis effects, and measurement delays, they can be applied to elastomeric soft actuators and more complex soft robotic systems. Building on this analytical framework, future work will focus on designing model-based dynamic controllers.

DATA AND CODE AVAILABILITY

All datasets and code used in this study are publicly available at https://github.com/Sangyoep/SBA_VQP_modeling.

REFERENCES

- [1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.
- [2] C. Della Santina, R. K. Katzschmann, A. Bicchi, and D. Rus, "Model-based dynamic feedback control of a planar soft robot: trajectory tracking and interaction with the environment," *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 490–513, 2020.
- [3] D. A. Haggerty, M. J. Banks, E. Kamenar, A. B. Cao, P. C. Curtis, I. Mezić, and E. W. Hawkes, "Control of soft robots with inertial dynamics," *Science robotics*, vol. 8, no. 81, p. eadd6864, 2023.
- [4] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1728–1748, 2023.
- [5] H. Wang, M. Totaro, and L. Beccai, "Toward perceptive soft robots: Progress and challenges," *Advanced Science*, vol. 5, no. 9, p. 1800541, 2018.
- [6] H. Lipson, "Challenges and opportunities for design, simulation, and fabrication of soft robots," *Soft Robotics*, vol. 1, no. 1, pp. 21–27, 2014.
- [7] P. Polygerinos, N. Correll, S. A. Morin, B. Mosadegh, C. D. Onal, K. Petersen, M. Cianchetti, M. T. Tolley, and R. F. Shepherd, "Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction," *Advanced engineering materials*, vol. 19, no. 12, p. 1700016, 2017.
- [8] M. Cianchetti, T. Ranzani, G. Gerboni, T. Nanayakkara, K. Althoefer, P. Dasgupta, and A. Menciassi, "Soft robotics technologies to address shortcomings in today's minimally invasive surgery: the stiff-flop approach," *Soft robotics*, vol. 1, no. 2, pp. 122–131, 2014.
- [9] R. Niiyama, X. Sun, C. Sung, B. An, D. Rus, and S. Kim, "Pouch motors: Printable soft actuators integrated with computational design," *Soft Robotics*, vol. 2, no. 2, pp. 59–70, 2015.
- [10] T. Ranzani, S. Russo, F. Schwab, C. J. Walsh, and R. J. Wood, "Deployable stabilization mechanisms for endoscopic procedures," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1125–1131.
- [11] J. Rogatinsky, D. Recco, J. Feichtmeier, Y. Kang, N. Kneier, P. Hammer, E. O'Leary, D. Mah, D. Hoganson, N. V. Vasilyev *et al.*, "A multifunctional soft robot for cardiac interventions," *Science Advances*, vol. 9, no. 43, p. eadi5559, 2023.
- [12] H. D. Yang and A. T. Asbeck, "A layered manufacturing approach for soft and soft-rigid hybrid robots," *Soft Robotics*, vol. 7, no. 2, pp. 218–232, 2020.
- [13] V. T. Vo, L. Z. Yañez, C. Muter, A. M. Moran, M. Saxena, G. Matthews, and T. Ranzani, "Soft, fiber-reinforced bellow actuators," *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 1896–1903, 2025.
- [14] B. Mosadegh, P. Polygerinos, C. Keplinger, S. Wennstedt, R. F. Shepherd, U. Gupta, J. Shim, K. Bertoldi, C. J. Walsh, and G. M. Whitesides, "Pneumatic networks for soft robotics that actuate rapidly," *Advanced functional materials*, vol. 24, no. 15, pp. 2163–2170, 2014.
- [15] R. W. Ogden, *Non-linear elastic deformations*. Courier Corporation, 1997.
- [16] P. Breitman, Y. Matia, and A. D. Gat, "Fluid mechanics of pneumatic soft robots," *Soft Robotics*, vol. 8, no. 5, pp. 519–530, 2021.
- [17] C. M. Hurvich and C.-L. Tsai, "A corrected akaike information criterion for vector autoregressive model selection," *Journal of time series analysis*, vol. 14, no. 3, pp. 271–279, 1993.
- [18] J. Vanhatalo, P. Jylänki, and A. Vehtari, "Gaussian process regression with student-t likelihood," *Advances in neural information processing systems*, vol. 22, 2009.
- [19] G. C. Chow, "Tests of equality between sets of coefficients in two linear regressions," *Econometrica: Journal of the Econometric Society*, pp. 591–605, 1960.
- [20] D. Bruder, X. Fu, R. B. Gillespie, C. D. Remy, and R. Vasudevan, "Data-driven control of soft robots using koopman operator theory," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 948–961, 2020.