IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Structure-Preserving Model Order Reduction of Slender Soft Robots via Autoencoder-Parameterized Strain

Abdulaziz Y. Alkayas, Anup Teejo Mathew, Daniel Feliu-Talegon, Yahya Zweiri, Thomas George Thuruthel, and Federico Renda

***Abstract*—While soft robots offer advantages in adaptability and safe interaction, their modeling remains challenging. This letter presents a novel, data-driven approach for model order reduction of slender soft robots using autoencoder-parameterized strain within the Geometric Variable Strain (GVS) framework. We employ autoencoders (AEs) to learn low-dimensional strain parameterizations from data to construct reduced-order models (ROMs), preserving the Lagrangian structure of the system while significantly reducing the degrees of freedom. Our comparative analysis demonstrates that AE-based ROMs consistently outperform proper orthogonal decomposition (POD) approaches, achieving lower errors for equivalent degrees of freedom across multiple test cases. Additionally, we demonstrate that our proposed approach achieves computational speed-ups over the high-order models (HOMs) in all cases, and outperforms the POD-based ROM in scenarios where accuracy is matched. We highlight the intrinsic dimensionality discovery capabilities of autoencoders, revealing that HOM often operate in lower-dimensional nonlinear manifolds. Through both simulation and experimental validation on a cable-actuated soft manipulator, we demonstrate the effectiveness of our approach, achieving near-identical behavior with just a single degree of freedom. This structure-preserving method offers significant reductions in the system degrees of freedom and computational effort while maintaining physical model interpretability, offering a promising direction for soft robot modeling and control.**

***Index Terms*—Modeling, control, learning, Reduced order modeling, Strain parameterization, Deep Learning Methods, Cosserat Rod.**

Received 25 March 2025; accepted 17 August 2025. Date of publication 4 September 2025; date of current version 15 September 2025. This article was recommended for publication by Associate Editor F. Chen and Editor C. Laschi upon evaluation of the reviewers' comments. This work was supported in part by the US Office of Naval Research Global under Grant N62909-21-1-2033 and in part by Khalifa University under Award RIG-2023-048 and Award RC1-2018-KUCARS. *(Corresponding author: Abdulaziz Y. Alkayas.)*

Abdulaziz Y. Alkayas is with the Department of Mechanical and Nuclear Engineering, Khalifa University, Abu Dhabi 52799, UAE, and also with the Department of Computer Science, University College London, WC1E 6BT London, U.K. (e-mail: abdulaziz.alkayas@ku.ac.ae).

Anup Teejo Mathew and Federico Renda are with the Department of Mechanical and Nuclear Engineering, Khalifa University, Abu Dhabi 52799, UAE, and also with the Khalifa University Center for Autonomous Robotic Systems (KUCARS), Khalifa University, Abu Dhabi 127788, UAE (e-mail: anup.mathew@ku.ac.ae; federico.renda@ku.ac.ae).

Daniel Feliu-Talegon is with the Department of Mechanical and Nuclear Engineering, Khalifa University, Abu Dhabi 52799, UAE, and also with the Department of Cognitive Robotics, Delft University of Technology, 2628 CN Delft, Netherlands (e-mail: d.feliutalegon@tudelft.nl).

Yahya Zweiri is with the Khalifa University Center for Autonomous Robotic Systems (KUCARS), Khalifa University, Abu Dhabi 127788, UAE, and with the Department of Aerospace Engineering, Khalifa University, Abu Dhabi 52799, UAE, and also with the Advanced Research and Innovation Center (ARIC), Khalifa University, Abu Dhabi 52799, UAE (e-mail: yahya.zweiri@ku.ac.ae).

Thomas George Thuruthel is with the Department of Computer Science, University College London, WC1E 6BT London, U.K. (e-mail: t.thuruthel@ucl.ac.uk).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3606389, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3606389

I. Introduction

SOFT robots, inspired by biological organisms and evolutionary designs [1], [2], offer advantages like adaptability and safe interaction, making them ideal for applications in agriculture, medicine, and inspection [3], [4], [5]. However, their flexibility also complicates modeling and control, leading to developments that benefit wider engineering and computational applications.

Various modeling approaches have been developed for soft robotic systems, including Finite Element Models (FEM), rod-based models, and Machine Learning (ML) techniques [6], [7]. FEM offers high accuracy but is often computationally intensive due to its high-dimensional nature. ML methods can capture complex nonlinear behaviors efficiently but may lack interpretability and generalizability. Rod-based models strike a balance between accuracy and computational efficiency, making them a widely used alternative. Among these, the Piecewise Constant Curvature (PCC) model [8] was extended to the Piecewise Constant Strain (PCS) model to include torsion and shear [9]. The Geometric Variable Strain (GVS) model [10], [11] further generalized this framework using a finite set of possibly non-constant strain bases, with a recent formulation supporting nonlinear, state- and time-dependent strains [12]. While more compact than FEM, these models can still result in high-dimensional systems that are computationally demanding to solve.

To address the high dimensionality of soft robotic models, researchers have increasingly turned to structure-preserving Reduced Order Modeling (ROM) techniques [13]. These methods aim to reduce computational cost while maintaining essential physical behavior and interpretability, which is a hybrid approach that is neither purely physics-based nor purely data-driven. A widely used ROM tool is Proper Orthogonal Decomposition (POD), which extracts dominant orthogonal modes from data to project high-dimensional systems onto reduced subspaces. POD has proven particularly effective when applied to FEM, achieving significant speedups without sacrificing accuracy [14]. More recently, a hybrid approach combined POD with

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

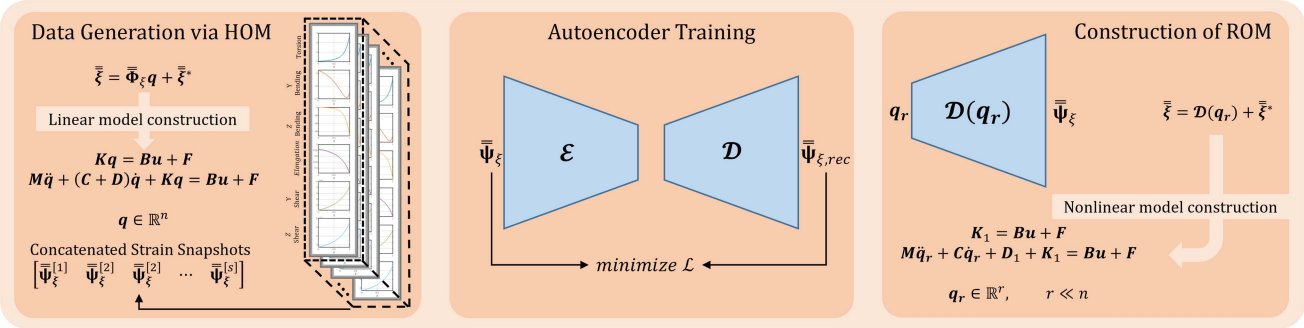

Fig. 1. Overview of our proposed model order reduction approach.

the GVS model by learning optimal strain bases directly from data to construct compact dynamic models [15]. In addition to POD, Autoencoder (AE) neural networks have emerged as powerful tools for nonlinear dimensionality reduction, providing latent representations that capture complex system dynamics. AE-based ROMs have shown promise in modeling high-dimensional mechanical systems [16], [17] and deformable solids [18], [19], with growing interest in developing efficient latent spaces for planar and 3D deformable objects to enable high-fidelity simulations at reduced computational cost [20], [21], [22]. To the best of our knowledge, AE-based dimensionality reduction has not yet been applied to strain-based rod models, leaving an open opportunity for exploration in this direction.

In this letter, we employ autoencoders (AEs) to learn low-dimensional strain parameterizations for slender soft robots. These autoencoder-based parameterizations are then utilized to construct the reduced-order model (ROM) while preserving the Lagrangian structure of the GVS model. It is worth noting that our proposed ROM method is neither purely physics-based nor purely data-driven; rather, it adopts a hybrid approach that leverages data-driven tools to enhance physics-based models. We demonstrate that AE-based ROMs outperform Proper Orthogonal Decomposition (POD)-based approach [15], achieving lower errors for the same number of Degrees of Freedom (DOFs). To further assess their effectiveness, we compare the behavior of multiple systems under both AE and POD formulations across different number of DOFs. We also show that our method consistently reduces computation time compared to the high-order models and outperforms the POD-based ROM when evaluated at equivalent accuracy levels. Additionally, we highlight the intrinsic dimensionality discovery capability of AEs [23]. While Higher-Order Models (HOMs) operate in a high-dimensional state space, their essential dynamics often live in lower-dimensional nonlinear manifolds. We illustrate this phenomenon through visualizations and discussions of the lower-dimensional manifolds in the latent spaces of various slender soft robotic systems. Finally, we experimentally validate the AE-based ROM on a single-tendon soft manipulator, demonstrating near-identical behavior with just a single DOF.

II. Geometric Variable Strain Model

The GVS model is a strain-based reduced-order method for modeling Cosserat rods (Fig. 2). The generalized coordinates $\boldsymbol{q}$ of the model are parameters that define the rod's strain field. We provide a summary of the model's kinematics and the Lagrangian form of the governing equations. A detailed derivation of the model, along with analytical expressions for all its components, is available in [12].

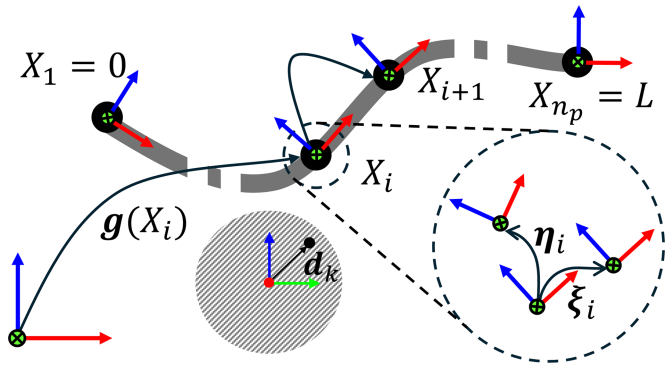

Fig. 2. Cosserat rod and its discretization. The rate of change of $\boldsymbol{g}$ with respect to X and t defines the local velocity twist $\boldsymbol{\eta}$ and the rod strain $\boldsymbol{\xi}$. Rod cross-section with the actuator routing, given by $\boldsymbol{d}_k(X)$, is shown in the inset.

A. Kinematics

The kinematics involves determining the position and orientation of a reference frame associated with the cross-sections of the rod. We represent this using a homogeneous transformation matrix $\boldsymbol{g}(X) \in SE(3)$, where, $X \in [0, L]$, L being the length of the rod. The rod strain $\boldsymbol{\xi}$ and the local velocity twist $\boldsymbol{\eta}$ are defined by the rate of change of $\boldsymbol{g}$ with respect to X and time t (Fig. 2).

$$\boldsymbol{g}' = \boldsymbol{g}\widehat{\boldsymbol{\xi}}\,, \tag{1a}$$

$$\dot{\boldsymbol{g}} = \boldsymbol{g}\widehat{\boldsymbol{\eta}}\,, \tag{1b}$$

where, $\widehat{\bullet}$ is the isomorphism from $\mathbb{R}^6$ to $\mathfrak{se}(3)$. The rod strain is typically represented as a linear function of the strain basis $\boldsymbol{\Phi}_\xi(X)$.

$$\boldsymbol{\xi}(X, \boldsymbol{q}) = \boldsymbol{\Phi}_\xi(X)\boldsymbol{q} + \boldsymbol{\xi}^*(X)\,, \tag{2}$$

where, $\boldsymbol{q} \in \mathbb{R}^n$ is the generalized coordinates of the robot and $\boldsymbol{\xi}^*$ represents the reference (stress-free) strain of the Cosserat rod.

However, nonlinear or coordinate-dependent strain bases were proposed in [12]. Accordingly, the strain field is parameterized as follows:

$$\boldsymbol{\xi}(X, \boldsymbol{q}) = \boldsymbol{\Psi}_\xi(X, \boldsymbol{q}) + \boldsymbol{\xi}^*(X)\,. \tag{3}$$

Using (3) in (1a) and integrating in X we obtain the kinematic map from $\boldsymbol{q}$ to $\boldsymbol{g}(X)$.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Considering the equality of the mixed partial derivatives of $\boldsymbol{g}$, we obtain the derivatives of velocity and acceleration twists with respect to X:

$$\boldsymbol{\eta}' = \dot{\boldsymbol{\xi}} - \mathrm{ad}_{\boldsymbol{\xi}}\boldsymbol{\eta}\,, \tag{4}$$

$$\dot{\boldsymbol{\eta}}' = \ddot{\boldsymbol{\xi}} - \mathrm{ad}_{\dot{\boldsymbol{\xi}}}\boldsymbol{\eta} - \mathrm{ad}_{\boldsymbol{\xi}}\dot{\boldsymbol{\eta}}\,, \tag{5}$$

where $\mathrm{ad}_{(\cdot)} \in \mathbb{R}^{6\times 6}$ is the adjoint operator of $\mathfrak{se}(3)$ and the time derivatives of $\boldsymbol{\xi}$ are given by:

$$\dot{\boldsymbol{\xi}} = \frac{\partial \boldsymbol{\Psi}_{\xi}}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}}\,, \tag{6}$$

$$\ddot{\boldsymbol{\xi}} = \frac{\partial \boldsymbol{\Psi}_{\xi}}{\partial \boldsymbol{q}}\ddot{\boldsymbol{q}} + \left(\frac{\partial^2 \boldsymbol{\Psi}_{\xi}}{\partial \boldsymbol{q}^2}\dot{\boldsymbol{q}}\right)\dot{\boldsymbol{q}} \tag{7}$$

The integration of (4) and (5) leads to the definition of the geometric Jacobian $\boldsymbol{J}(\boldsymbol{q}, X)$ and its time derivative $\dot{\boldsymbol{J}}(\boldsymbol{q}, \dot{\boldsymbol{q}}, X)$.

B. Dynamics and Statics

The free dynamics of the Cosserat rod are projected onto the space of generalized coordinates using the geometric Jacobian, according to the D'Alembert-Kane method, to derive the Lagrangian form of the dynamic equations:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{D}_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{K}_1(\boldsymbol{q}) = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{u} + \boldsymbol{F}(\boldsymbol{q}) \tag{8}$$

where, $\boldsymbol{M}$ is the generalized mass matrix, $\boldsymbol{C}$ is the generalized Coriolis matrix, $\boldsymbol{D}_1$ is the generalized damping force, $\boldsymbol{K}_1$ is the generalized elastic force, $\boldsymbol{B}$ is the generalized actuation matrix, $\boldsymbol{u}$ is the vector of actuation forces, and $\boldsymbol{F}$ is the generalized external force. The generalized static equilibrium equations are obtained from (8) by setting $\ddot{\boldsymbol{q}} = \boldsymbol{0}$ and $\dot{\boldsymbol{q}} = \boldsymbol{0}$.

$$\boldsymbol{K}_1(\boldsymbol{q}) = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{u} + \boldsymbol{F}(\boldsymbol{q}) \tag{9}$$

Since the free dynamics of the soft body are expressed in a continuum (distributed) form, the projected quantities $\overline{\boldsymbol{Q}}$ are integrated along the rod domain using a quadrature method.

$$\int_0^L \overline{\boldsymbol{Q}}(X)dX = \sum_{i=1}^{n_p} w_i \overline{\boldsymbol{Q}}(X_i) \tag{10}$$

where, n_p is the number of quadrature points and w_i are quadrature weights.

C. Actuation With Frictional Dissipation

The actuation wrench in the local frame is obtained by calculating the moment and force acting on the rod's centerline:

$$\boldsymbol{\mathcal{F}}_a(X) = \sum_{k=1}^{n_a} \begin{bmatrix} \boldsymbol{d}_k \times \boldsymbol{t}_k \\ \boldsymbol{t}_k \end{bmatrix} \alpha_k u_k = \boldsymbol{\Phi}_a(\boldsymbol{q}, X)\boldsymbol{u} \tag{11}$$

where $\boldsymbol{\Phi}_a(\boldsymbol{q}, X) \in \mathbb{R}^{6\times n_a}$ is the actuation basis, n_a is the number of actuators, $\boldsymbol{d}_k(X) \in \mathbb{R}^3$ is the distance from the center-line to actuator k, and $\boldsymbol{t}_k(X)$ is the unit vector tangent to the actuator's path. In the presence of friction, the cable tension decreases along the rod due to the interaction between the tendon and the soft body. This effect has been studied in previous work [24]. In such case, the reduction in cable tension is accounted for by an attenuation factor $\alpha_k(X)$ given by:

$$\alpha_k(X) = e^{-\mu \int_0^X \phi_k(s)ds} \tag{12}$$

where μ is the coefficient of friction, and $\phi_k(X)$ is the curvature of the actuator.

III. Reduction via Autoencoders

Our aim is to find the strain function $\boldsymbol{\Psi}_{\xi}$ that would appropriately describe the rod's strain field. For generic soft rod systems, selecting such a function is non-trivial. We propose to learn this parameterization from strain data generated by a ground-truth HOM. The decoder of a trained autoencoder then serves as the nonlinear strain function $\boldsymbol{\Psi}\xi$. An overview of our proposed approach is summarized in Fig. 1. To integrate the reduction approach with the discrete domain implementation of the GVS model (see [12]), we sample the 6D strains at p points along the Cosserat rod abscissa, a concatenated vector of strains and twists can be defined as:

$$\bar{\bar{\boldsymbol{\Psi}}}_{\xi} := \left[\boldsymbol{\xi}_1^T - \boldsymbol{\xi}_1^{*T}, \boldsymbol{\xi}_2^T - \boldsymbol{\xi}_2^{*T}, \ldots, \boldsymbol{\xi}_p^T - \boldsymbol{\xi}_p^{*T}\right]^T \in \mathbb{R}^{6p} \tag{13}$$

where $\boldsymbol{\xi}_j - \boldsymbol{\xi}_j^*$ is the deviation from reference strain at the j^{th} computational point. The computational points include both the quadrature points used for spatial integration in (10) and those used for kinematic computations.

Now, multiple samples of $\bar{\bar{\boldsymbol{\Psi}}}_{\xi}$ generated by the HOM is the data that will be used in training the AE. It is worth noting that the proposed AE-based ROM is broadly applicable, requiring only representative strain fields. The key requirement for the amount of training data is to reflect the system's expected operating conditions -for instance, contact scenarios- and cover thoroughly the actuation/input space. The AE's function is to reconstruct the input as follows:

$$\bar{\bar{\boldsymbol{\Psi}}}_{\xi,rec} = \mathcal{D}(\mathcal{E}(\bar{\bar{\boldsymbol{\Psi}}}_{\xi})), \tag{14}$$

while minimizing the regularized reconstruction loss:

$$\mathcal{L} = \|\bar{\bar{\boldsymbol{\Psi}}}_{\xi,rec} - \bar{\bar{\boldsymbol{\Psi}}}_{\xi}\|_2^2 + \lambda \sum_i \|\boldsymbol{W}_i\|_2^2, \tag{15}$$

where $\mathcal{D}$ is the decoder function, $\mathcal{E}$ is the encoder function, $\mathcal{L}$ is the reconstruction loss, λ is the regularization parameter, and $\boldsymbol{W}_i$ is the weight matrix of the i^{th} layer of the AE. We include the L2 regularization term to avoid overfitting with $\lambda = 10^{-4}$.

We use $\tanh$ activation functions for all layers in the AEs we train. Such function is convenient as the first and second derivatives are continuous, facilitating the use of numerical solvers and avoiding potential issues that might arise when using discontinous functions. We use two architectures as strain parameterizations in our examples, mainly a shallow AE of 3 hidden layers, and a deeper AE of 7 hidden layers. To systematically evaluate their performance, we analyze the reconstruction error (RMSE) across varying latent space dimensions (1, 2, and 3 DOFs) and AE depths (3, 5, and 7 hidden layers). The results of this study, conducted for the cases in Sections IV-A and IV-B, are summarized in Table I. These two examples were selected for their suitability for both analysis and visualization. The findings illustrate how the theoretical minimum number of DOFs required by a system influences the AE's representational capability. For example, using a single DOF for the inherently 2D system in Section IV-B leads to a significant reconstruction error, as evidenced by the sharp error reduction when increasing to 2 DOFs, unlike the 1D case in Section IV-A, which is effectively captured even with a single DOF.

The encoder and decoder are symmetric, with layer sizes following a geometric progression: the encoder halves the size at each layer, while the decoder doubles it. For example, the sizes of the decoder layers of the deeper AE variant are $r,\ 6p/8,\ 6p/4,\ 6p/2,\ 6p$. The "six" here is due to the six strain modalities in general, however it can be less if some strains are

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I
RMS RECONSTRUCTION ERRORS ACROSS THE FULL DATASET FOR DIFFERENT AE ARCHITECTURES AND LATENT SPACE DIMENSIONS (DOFS) FOR THE SCENARIOS IN SECTIONS IV-A AND IV-B

Single Cable Manipulator			
Hidden Layers	**1 DOF**	**2 DOFs**	**3 DOFs**
3	0.1447	0.0936	0.0870
5	0.0556	0.0336	0.0321
7	0.0383	0.0310	0.0248
Soft Beam Under External Moment			
Hidden Layers	**1 DOF**	**2 DOFs**	**3 DOFs**
3	7.6686	2.6016	1.5544
5	6.4376	0.8733	0.6452
7	4.8100	0.5923	0.5318

not excited. The latent space layer size is chosen based on the reduced number of generalized coordinates $\boldsymbol{q_r}$, that is r. Then the strain parameterization would be:

$$\bar{\bar{\boldsymbol{\xi}}}(\boldsymbol{q}_r) = \mathcal{D}(\boldsymbol{q}_r) + \bar{\bar{\boldsymbol{\xi}}}^* \,, \tag{16}$$

and from here the derivation for the whole model can begin. The decoder's first derivative and the vector product of the second derivative with $\dot{\boldsymbol{q}}_r$ are needed as seen in (6) and (7). The first derivative (the network jacobian) can be easily acquired analytically using the chain rule. To avoid computing the expensive second derivative of the decoder (i.e., the network Hessian), which is required in (7), we instead utilize the following relation:

$$\frac{\partial^2 \boldsymbol{\Psi}_\xi}{\partial \boldsymbol{q}^2}\dot{\boldsymbol{q}} = \frac{\partial \dot{\boldsymbol{\xi}}}{\partial \boldsymbol{q}}, \tag{17}$$

which is once again computed analytically using the chain rule.

The analytically computed strain gradients from smooth AE decoders, combined with the differentiability of the GVS-based Lagrangian formulation [25], ensured stable convergence across all static and dynamic simulations presented in this work.

IV. SIMULATIONS

To evaluate the performance of the proposed AE-ROM method, we conducted a series of simulation studies on three different systems. For solving statics, we used MATLAB's fsolve with its default trust-region-dogleg algorithm with default tolerances. Alternative algorithms like Levenberg-Marquardt and trust-region were also tested and yielded identical results with varying computation times. For dynamics, we used the semi-implicit solver ode15 s, as explicit solvers like ode45 and ode113 led to significantly slower simulations due to system stiffness. The PC specifications for all computations presented here are as follows: 13th Gen Intel(R) Core(TM) i9-13900HX, 2.20 GHz, 64.0 GB RAM. The MATLAB version used is R2024a.

A. Single-Tendon Soft Manipulator

Here, we study the case of a soft manipulator actuated by a single tendon under gravitational load in a cantilever position. Table II includes the dimensions and material parameters of the manipulator. Frictional effects are accounted for as presented earlier in (11) and (12), with a coefficient of 0.88. The cable path is linear, defined by $\boldsymbol{d}_1(0) = [0, 0, 1]^T\ cm$ and $\boldsymbol{d}_1(L) = [0, 0, 0.3]^T\ cm$ at the base and tip of the manipulator, respectively. As this is a planar deformation scenario, only three strains will be excited: bending about y, shear along z, and elongation. Each of these 3 strains is parameterized using a 10^{th}

TABLE II
PARAMETERS OF THE CABLE DRIVEN SOFT MANIPULATORS

	Single-Actuator	Six-Actuators
Length	25 cm	60 cm
Base Radius	1.25 cm	2 cm
Tip Radius	0.5 cm	1 cm
Density	1121 kg/m^3	1000 kg/m^3
Young's Modulus	0.617 MPa	1 MPa
Poisson's Ratio	0.5	
Damping Constant	$10^4\ Pa \cdot s$	

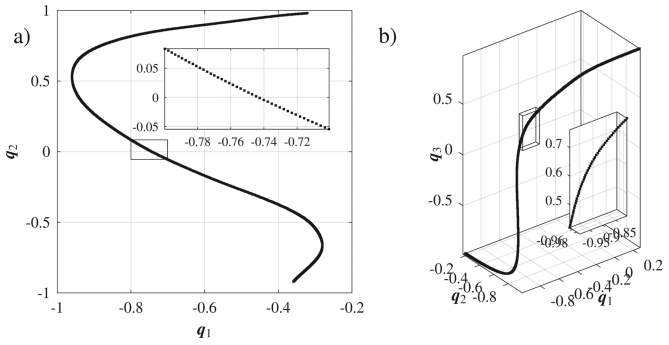

Fig. 3. Latent space visualization for the trained AE network using a) 2 DOFs and b) 3 DOFs. Each black dot represents the latent space coordinates for a training sample.

order Legendre polynomial, resulting in a total of 33 DOFs for the manipulator.

Multiple strain snapshots are collected and organized according to the methodology discussed in Section III. Each snapshot is associated with an actuation value, with 1000 levels of actuation ranging from -6 N to 3 N. We define an abscissa discretization of 102 points, making each snapshot of a vector of 306 dimensions. The snapshots are then used as the training samples for the AE network. Three AEs are trained, with a latent space of dimension 1,2 and 3. The reconstruction errors for multiple AE depths and latent space dimensions are detailed in Table I.

Before using the decoder of the AE as a reduced nonlinear basis, we infer the encoder with the training samples to inspect the latent space. We know that in the presented case of static equilibrium that a single DOF is capable of describing the behaviour of the system. Indeed we observe, as seen in Fig. 3(a) and (b), that the latent space coordinates of the training samples form a 1D curve in 2D and 3D spaces. This behaviour arises naturally with complete unsupervision in the training process, hinting at the AE's ability to discover the intrinsic dimensionality of the underlying system.

The single DOF AE is then utilized as the reduced nonlinear basis for the manipulator, and the static equilibrium is solved at different actuation levels. It can be seen from Fig. 4 that using only one DOF, the ROM is able to produce almost the same strain field as the HOM. The figure also shows how the AE-based ROM shows superior performance compared to the POD-based ROM, despite having 2 DOFs.

At 1 DOF, the AE-based ROM achieved less than 1% strain error with respect to the HOM, with the Mean Absolute Error (MAE) error evaluated at all quadrature points along the rod. We used this as the accuracy threshold for a fair comparison and increased the number of POD modes until the same criterion was met. This occurred at 7 DOFs. The corresponding strain profile

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

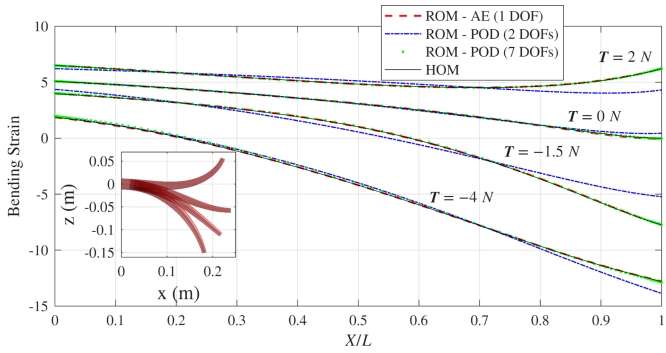

Fig. 4. Comparison between the AE-based 1 DOF and the POD-based 2, 7 DOFs ROMs to the HOM. As inset, the configurations of the manipulator at different actuation levels using the AE ROM.

is also shown in Fig. 4. In this comparable accuracy setting, the AE-based ROM completed the computation in 42 ms on average, whereas the POD-based ROM required 165 ms on average, resulting in a speed-up factor of 3.9. Both offer substantial gains over the HOM, which took 670 ms on average.

B. Soft Beam Under External Moment

We move to a different system, where a soft beam undergoes bending due to an applied external moment of different magnitudes at different locations. The beam has a length of 0.8 m, a radius of 1.5 cm, and a Young's modulus of 1 MPa. The beam is only subjected to a moment that acts along the local y-axis and is distributed in the shape of a triangular pulse, centered at a specific point with a base width equal to 10% of the beam's total length. The analytical strain solution to this problem is a sigmoid function. Thus, in order to capture this highly nonlinear strain profile, we employ local strain bases that resemble linear FEM shape functions. Correspondingly, a 50 DOFs local strain parameterization is used to generate the HOM strain data used for the AE training. The dataset is generated from 5000 solutions of moment locations $X/L \in [0, 1]$ and magnitudes $\mathcal{M}_e \in [-5, 5]\ N.m$.

We then move forward to test the intrinsic dimension discovery property of the AE. We train AEs to see how the resulting manifold appears, and we report the reconstruction errors across different numbers of hidden layers and latent space dimensions in Table I. We see that the resulting latent space is indeed a surface in 3D space, due to having two main contributors to the data, location and magnitude. For simplicity, we show in Fig. 5 the latent space of a simpler shallow AE as the surface increases in complexity when the AE becomes deeper. However, we employ a deeper AE as the ROM strain function with better reconstruction capabilities, and proper visualization of it's latent space is present in the multimedia attachment of this letter. Fig. 5(a) and (b) show how the training samples construct the system's intrinsic manifolds in both 2D and 3D space respectively. We observe that the latent space coordinates form a perfectly 2D surface in 3D space. Symmetry can also be seen around the origin in both latent spaces, due to the positive and negative duality in the dataset for positive and negative moments. Additionally, this symmetry point is the bottleneck between both positive and negative halves, and exhibits a high density of points around it. That is due to the fact that sigmoids of small magnitudes (close to zero) look similar regardless of the location, and consequently mapped to a similar region to the zero magnitude point. Particularly at zero magnitude, information about the location is completely lost, and thus mapped to a singularity, resulting in ill-conditioning of the problem. Similar issues arise in nonlinear strain parametrization, where at some coordinates the basis matrix can be singular as discussed in [12]. This issue can be avoided by using an alternative strain parameterization, such as global or local linear bases, although this typically comes at the expense of introducing many additional DOFs.

The decoder of the deeper AE is then employed as the nonlinear strain parametrization for the system, with a 2 dimensional latent space. The same data used for the AE training is decomposed using POD and the bases are to build a 2 dimensional POD-based ROM. We then compare the performance of both ROMs for two different loading conditions, 4 N.m applied at $X/L = 0.25$, and 2 N.m applied at $X/L = 0.75$. Both the AE-based and the POD-based ROMs are then solved and the performance is compared in Fig. 6. It can be seen that the AE-based ROM is capable of producing strains very similar to the HOM and much better than the POD-based one using the same number of generalized coordinates.

We used the same evaluation metric as before to assess computational speed. For this highly nonlinear strain profile, 19 POD modes were required to achieve a comparable strain error ($< 1\%$), as shown in Fig. 6. In this comparable accuracy setting, the AE-based ROM completed the computation in 37ms on average, whereas the POD-based ROM required 239ms on average, resulting in a speed-up factor of 6.5. Once again, both reduction approaches offered substantial gains over the HOM, which took 810ms on average.

Additionally, we train the autoencoder on noisy strain data to assess its impact on the resulting latent space manifold. We find that noise locally introduces an additional dimension to the manifold; however, it largely retains the same global structure as in the noise-free case. Interestingly, the performance of the AE as a strain parameterization remains almost unaffected, which we attribute to the inherent denoising capabilities of autoencoders. Visualizations of the latent space manifolds for the AEs trained on noisy data, corresponding to the scenarios in Sections IV-A and IV-B are provided in the video attachment accompanying this letter.

C. Six-Tendon Multisection Soft Manipulator

This section introduces a six-actuator manipulator with a different actuation configuration: three actuators span the entire body, while the remaining three operate over half its length. The actuators are arranged 60 degrees apart in an alternating pattern of full- and half-body actuation domains, with straight paths on the manipulator's surface. The dimensional and material parameters for the manipulator are summarized in Table II, and the tendon interactions with the body is assumed to be frictionless.

Strain discontinuities arise where cables terminate midway along the manipulator. Since continuous approximation functions may fail to capture these, the HOM framework partitions the domain of the rod into two segments, each with independent strain bases. Torsional, elongation, and shear strains use first-order (linear) Legendre polynomials, while bending strains use second-order (quadratic) ones. This parameterization yields 28 DOFs for the full manipulator.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

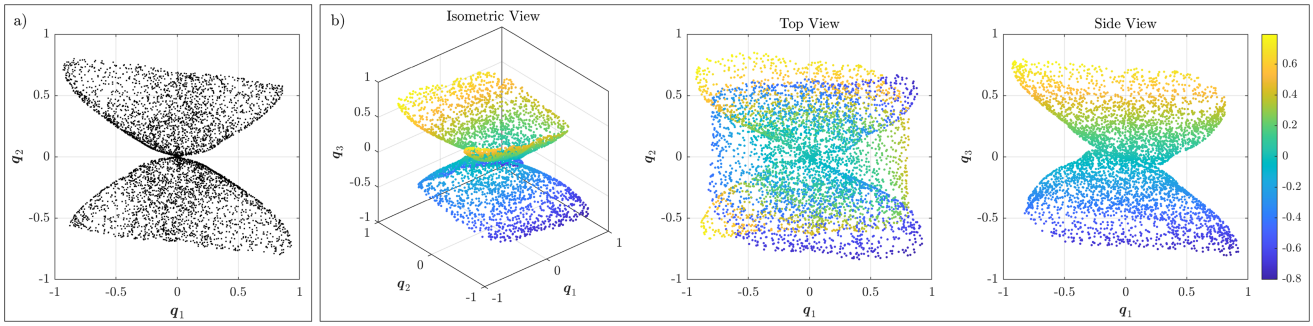

Fig. 5. Latent space visualization for the trained AE network using (a) 2 DOFs and (b) 3 DOFs for the soft beam under external moment. Each dot shows the latent coordinates of a training sample, using a shallower AE than in simulation for ease of visualization. The color gradient is purely for visualization purposes and is correlated with the value of q_3. The surfaces resulting from deeper AEs can be seen in the accompanying multimedia material.

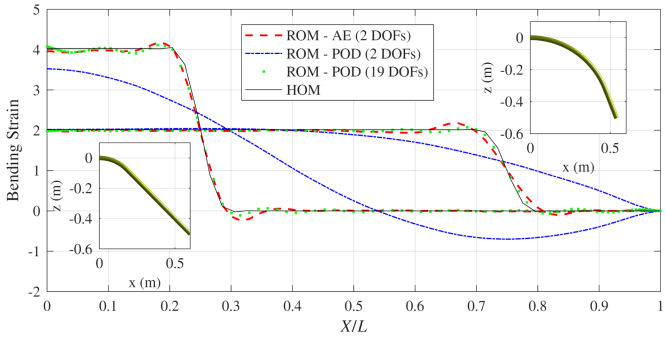

Fig. 6. Comparison between the AE-based 2 DOFs and the POD-based 2, 19 DOFs ROMs to the HOM. As inset, the configurations of the soft beam at when subjected to two different moments at both $X/L = 0.25$ and 0.75 using the AE-based ROM.

1) Statics: As demonstrated in [15], uniform sampling -or grid sampling- of the actuation domain is the most effective strategy for generating HOM data in the context of POD-based ROMs, despite its inherent exponential complexity. However, when applied to training an AE, this approach fails to adequately describe the underlying manifold. To illustrate this, we trained two AEs using the same amount of data but with different sampling strategies: one with uniform sampling of the actuation space and the other with random sampling, both having a 6D latent space. We then visualized the latent space by projecting the 6D points of the training samples onto the first three principal components. As depicted in Fig. 8, the uniformly sampled data produces a sparse manifold with significant uncharted regions, with no guarantees that these regions result in the proper behavior or not. Whereas the randomly sampled data yields a more evenly parameterized manifold. This distinction is critically important, as a well-parameterized manifold facilitates the iterative load ramping approach for finding static equilibrium solutions. An important aspect to note in such systems is the critical role of the initial guess provided to the solver. Using well-informed initial conditions can significantly improve convergence and may eliminate the need for techniques like load ramping altogether.

We adopt the random sampling approach to generate a static equilibrium dataset comprising 20,000 samples, with actuation forces randomly distributed within the range of $[-30, 30]$ N. We trained an AE with a 6D latent space—matching the system's six actuators—and compared it against a POD-based ROM with 6 modes. Simulation results for five arbitrary actuation inputs were used to evaluate the computational performance and accuracy of both ROMs at equal DOFs. Fig. 7 illustrates the resulting configurations and tip position errors for both ROMs relative to the HOM (28 DOFs). While the POD-based ROM was slightly faster, averaging 94 ms per static equilibrium evaluation compared to 106 ms for the AE, the AE-based ROM achieved significantly higher accuracy. The tip position error was 2.32 cm (3.9% of the manipulator length) for AE, versus 4.95 cm (8.3%) for POD. For reference, the HOM required 276 ms on average.

2) Dynamics: We adopt a similar approach to [15] and use actuator babbling to generate the dynamic simulation dataset. Random actuation values within the range $[-15, 0]$N are applied to each actuator, with holding times randomly varied between 0.5 and 1s, independently for each actuator.

We simulate three minutes of variable-hold babbling, sampling the strain field evolution at 100 Hz, which yields 18,000 strain snapshots. These snapshots are used to train multiple AEs with varying latent space dimensions. The decoders of these AEs are then integrated as the nonlinear strain parametrization. The resulting AE-based ROMs are evaluated on a new random actuation sequence lasting 10 seconds, with changes occurring every 1 s. Their performance is compared to that of a POD-based ROM with the same number of DOFs and the HOM. The results of these tests are summarized in Fig. 9, which depicts the average tip position errors between the ROMs and the HOM over the 10-second test. The AE-based ROM consistently outperforms the POD-based ROM across all tested DOFs, achieving reductions in average errors of up to 80%. As shown in the inset of Fig. 9, the trace of the tip position at 7 DOFs highlights the superior qualitative performance of the AE-based ROM. In terms of computational speed, similar trends to the static case were observed. For the same number of DOFs, the POD-based ROM was consistently faster, with evaluation time increasing almost linearly with the number of modes. In contrast, the AE-based ROM exhibited comparable runtimes at lower DOFs but showed a higher-order growth in evaluation time as the DOFs increased. For instance, at 7 DOFs, the POD-based ROM completed the simulation in 11.6 seconds, while the AE-based ROM required 17.2 seconds, approximately 48% slower. However, this came with a significant improvement in accuracy: the tip position error was reduced from 7% for POD to 1.9% for AE. In comparison, the HOM simulation took 114.3 seconds, highlighting the substantial efficiency gains achieved by both ROMs. Furthermore, a video comparison between the proposed ROM and the HOM for the 10-second dynamics test

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

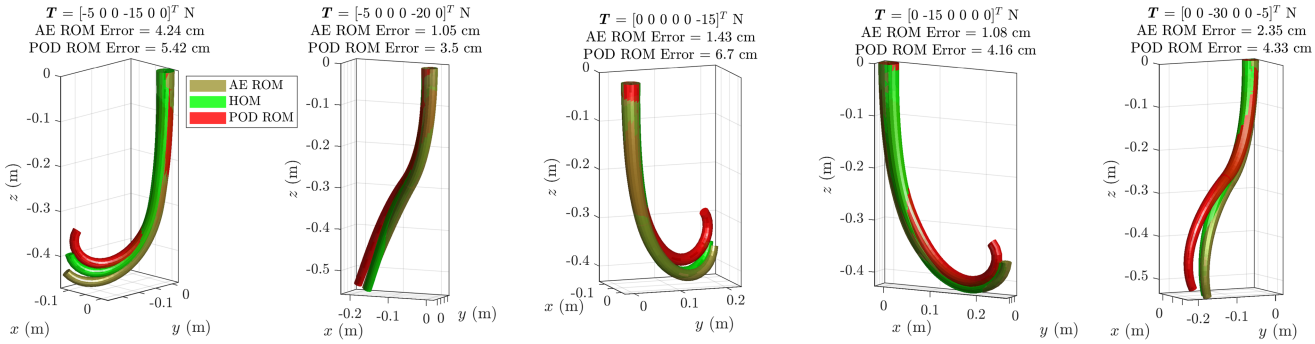

Fig. 7. Comparison of the AE-based ROM (gold), POD-based ROM (red), and HOM (green) for the six-actuator manipulator's static equilibrium under different actuation force. Each figure is accompanied by the corresponding actuation forces and tip position errors of the ROMs relative to the HOM.

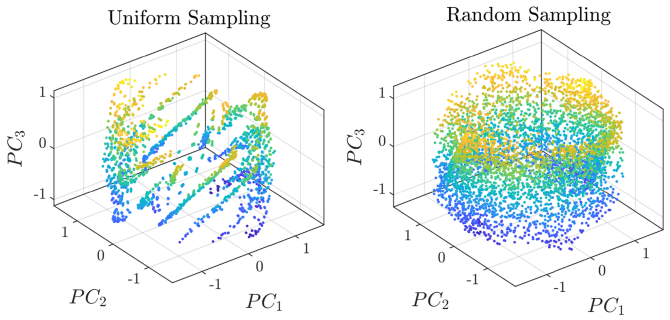

Fig. 8. Projections of the 6D latent space coordinates for the training samples of the six-actuator manipulator, comparing uniform (grid) and random sampling strategies. The color gradient is for visualization purposes and represents the value of PC_3.

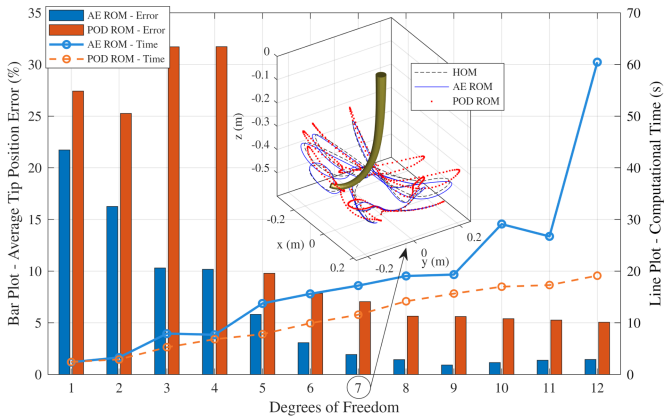

Fig. 9. Comparison of computational times and average tip position errors (expressed as a percentage of the body length) between the AE-based and POD-based ROMs relative to the HOM, evaluated for varying numbers of DOFs[1]. The inset illustrates the trace of the tip positions for the three models over a 10-second test, as discussed in Section IV-C2, specifically for 7 DOFs ROMs.

across multiple DOFs can be examined in the accompanying multimedia material.

V. Experimental Validation

In this section, the performance of the proposed ROM is assessed in an experimental prototype in order to demonstrate the validity of our approach in real scenarios. Material properties for the prototype were identified experimentally, yielding the parameters detail in Table II, and a friction coefficient of 0.88.

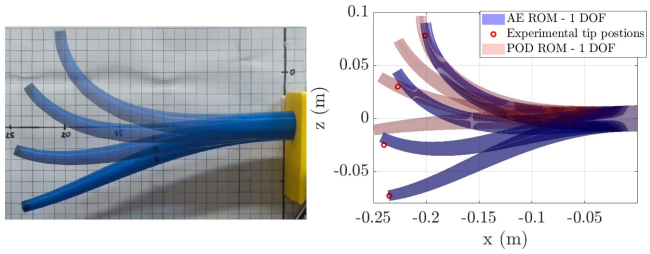

Fig. 10. (a) Snapshots of the experimental prototype taken at different actuation values. (b) Resulting configurations of the ROM using a single DOF, with the AE-based ROM (blue) and the POD-based ROM (red). The actual experimental tip positions are indicated with the red circles.

TABLE III
Summary of AE- and POD-Based ROMs Computational Performance Under Different Evaluation Criteria. Time is in Seconds. † Indicates 7 DOFs

Same Accuracy Level (<1% strain error)					
	POD		AE		
Case	DOF	Time	DOF	Time	HOM Time
IV-A	7	0.165	1	0.042	0.670
IV-B	19	0.239	2	0.037	0.810
Same DOF Study					
	POD		AE		
Case	Error	Time	Error	Time	HOM Time
IV-C1	8.3 %	0.094	3.9 %	0.106	0.276
IV-C2†	7.0 %	11.6	1.9 %	17.2	114.3

The actuation system consists of a Blue Bird BMS-A922+ servo motor and a highly accurate Micro S-type Load Cell B313-20 N for tension measurement at the base. Specific servo motor rotations were commanded to exert tensions measured at $(-1.12, -2.35, -3.7, -5.05)$ N. The resulting manipulator configurations are illustrated in Fig. 10(a). Tip positions were captured using a camera system for comparison with the ROM.

For the ROM, simulation data are generated over the same range of actuation forces presented in Section IV-A, with both gravity and friction taken into account. Once the AE was trained, the ROM is constructed based on the decoder as the nonlinear strain parametrization, and is solved and compared with the experimental results. As seen in Fig. 10(b), the errors for the AE-based ROM are small, with an average error of 0.87 cm and a maximum error of 1.43 cm for the tip position, which represents 3.48% and 5.72% of the body length respectively. In

comparison, the POD-based ROM exhibits significantly higher errors, with an average of 4.76 cm and a maximum of 6.42 cm, representing 19.04% and 25.68% of the body length respectively. This showcases state-of-the-art performance for modeling such a system with a single DOF.

VI. Conclusion

In this letter, we introduced an autoencoder-based approach for model order reduction of slender soft robots that preserves the underlying Lagrangian structure of the system. By identifying the nonlinear strain parameterizations through autoencoders, the method achieved substantial improvements over POD-based reduction across diverse test cases involving three systems: a single-actuator soft manipulator, a soft beam under external moments, and a six-actuator multisection manipulator in both static and dynamic scenarios.

We highlighted the ability of autoencoders to naturally discover the intrinsic dimensionality of systems, as evidenced by the emergence of lower-dimensional manifolds in the latent space. This capability enables more efficient representations of complex strain fields without requiring prior knowledge of the underlying structure. Table III summarizes the performance of the AE and POD ROMs. In the simulation studies, AE achieved faster computation than POD when targeting the same strain error relative to the HOM. In contrast, for the same number of DOFs, POD was faster, but its accuracy was significantly lower than that of AE, with both the AE- and POD-ROMs offering speed-ups compared to the HOM.

Experimental validation on a single-cable, cable-actuated soft manipulator demonstrated the practical applicability of our approach, achieving an average tip position error of just 3.48% of the body length with a single DOF.

While the nonlinear parameterization inherent to AE-based ROMs introduces additional computational complexity in derivative calculations compared to linear models, our results show that it performs better computationally at the same accuracy level, facilitating its applicability to real-time simulation for control purposes. This trade-off, combined with the reduced system dimensionality, is particularly beneficial for model-based motion planning and shape estimation using limited sensing. Future work may explore extending this approach to hybrid soft–rigid systems, further broadening its practical impact.

References

[1] A. T. Mathew et al., "ZodiAq: An isotropic flagella-inspired soft underwater drone for safe marine exploration," *Soft Robot.*, vol. 12, pp. 410–422, 2025.

[2] Z. Wang, N. M. Freris, and X. Wei, "SpiRobs: Logarithmic spiral-shaped robots for versatile grasping across scales," *Device*, vol. 3, 2024, Art. no. 100646.

[3] C. Armanini et al., "Soft robotics for farm to fork: Applications in agriculture & farming," *Bioinspiration Biomimetics*, vol. 19, no. 2, 2024, Art. no. 021002.

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

[5] L. Groo, A. T. Juhl, and L. A. Baldwin, "Toward soft robotic inspection for aircraft: An overview and perspective," *MRS Commun.*, vol. 14, no. 5, pp. 741–751, 2024.

[6] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1728–1748, Jun. 2023.

[7] D. Kim et al., "Review of machine learning methods in soft robotics," *PLoS One*, vol. 16, pp. 1–24, Feb. 2021.

[8] I. Webster, J. Robert, and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1661–1683, 2010.

[9] F. Renda, F. Boyer, J. Dias, and L. Seneviratne, "Discrete cosserat approach for multisection soft manipulator dynamics," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1518–1533, Dec. 2018.

[10] F. Renda, C. Armanini, V. Lebastard, F. Candelier, and F. Boyer, "A geometric variable-strain approach for static modeling of soft manipulators with tendon and fluidic actuation," *IEEE Robot. Automat. Lett.*, vol. 5, no. 3, pp. 4006–4013, Jul. 2020.

[11] F. Boyer, V. Lebastard, F. Candelier, and F. Renda, "Dynamics of continuum and soft robots: A strain parameterization based approach," *IEEE Trans. Robot.*, vol. 37, no. 3, pp. 847–863, Jun. 2021.

[12] A. T. Mathew, D. Feliu-Talegon, A. Y. Alkayas, F. Boyer, and F. Renda, "Reduced order modeling of hybrid soft-rigid robots using global, local, and state-dependent strain parameterization," *Int. J. Robot. Res.*, vol. 44, pp. 129–154, 2025. doi: 10.1177/02783649241262333.

[13] S. Sadati, S. E. Naghibi, L. da Cruz, and C. Bergeles, "Reduced order modeling and model order reduction for continuum manipulators: An overview," *Front. Robot. AI*, vol. 10, 2023, Art. no. 1094114.

[14] O. Goury and C. Duriez, "Fast, generic, and reliable control and simulation of soft robots using model order reduction," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1565–1576, Dec. 2018.

[15] A. Y. Alkayas, A. T. Mathew, D. Feliu-Talegon, P. Deng, T. G. Thuruthel, and F. Renda, "Soft synergies: Model order reduction of hybrid soft-rigid robots via optimal strain parameterization," *IEEE Trans. Robot.*, vol. 41, pp. 1118–1137, 2025.

[16] M. Lepri, D. Bacciu, and C. D. Santina, "Neural autoencoder-based structure-preserving model order reduction and control design for high-dimensional physical systems," *IEEE Contr. Syst. Lett.*, vol. 8, pp. 133–138, 2024.

[17] J. S. Hesthaven, C. Pagliantini, and N. Ripamonti, "Structure-preserving model order reduction of hamiltonian systems," 2021, *arXiv:2109.12367*.

[18] S. Shen et al., "High-order differentiable autoencoder for nonlinear model reduction," *ACM Trans. Graph.*, vol. 40, Jul. 2021, Art. no. 68.

[19] L. Fulton, V. Modi, D. Duvenaud, D. I. W. Levin, and A. Jacobson, "Latent-space dynamics for reduced deformable simulation," *Comput. Graph. Forum*, vol. 38, no. 2, pp. 379–391, 2019.

[20] N. Sharp et al., "Data-free learning of reduced-order kinematics," in *Proc. ACM SIGGRAPH 2023 Conf.*, 2023, pp. 1–9.

[21] Z. Zong et al., "Neural stress fields for reduced-order elastoplasticity and fracture," in *Proc. SIGGRAPH Asia 2023 Conf. Papers*, 2023, pp. 1–11.

[22] A. Lyu, S. Zhao, C. Xian, Z. Cen, H. Cai, and G. Fang, "Accelerate neural subspace-based reduced-order solver of deformable simulation by Lipschitz optimization," *ACM Trans. Graph.*, vol. 43, no. 6, pp. 1–10, 2024.

[23] K. Zeng, C. E. P. De Jesús, A. J. Fox, and M. D. Graham, "Autoencoders for discovering manifold dimension and coordinates in data from complex dynamical systems," *Mach. Learn.: Sci. Technol.*, vol. 5, May 2024, Art. no. 025053.

[24] I. Hussain et al., "Compliant gripper design, prototyping, and modeling using screw theory formulation," *Int. J. Robot. Res.*, vol. 40, no. 1, pp. 55–71, 2021.

[25] A. T. Mathew, F. Boyer, V. Lebastard, and F. Renda, "Analytical derivatives of strain-based dynamic model for hybrid soft-rigid robots," *Int. J. Robot. Res.*, 2025. doi: 10.1177/02783649251346209.