

# Neural Autoencoder-Based Structure-Preserving Model Order Reduction and Control Design for High-Dimensional Physical Systems

Marco Lepri, Davide Bacciu<sup>ID</sup>, *Senior Member, IEEE*, and Cosimo Della Santina<sup>ID</sup>, *Senior Member, IEEE*

**Abstract**—This letter concerns control-oriented and structure-preserving learning of low-dimensional approximations of high-dimensional physical systems, with a focus on mechanical systems. We investigate the integration of neural autoencoders in model order reduction, while at the same time preserving Hamiltonian or Lagrangian structures. We focus on extensively evaluating the considered methodology by performing simulation and control experiments on large mass-spring-damper networks, with hundreds of states. The empirical findings reveal that compressed latent dynamics with less than 5 degrees of freedom can accurately reconstruct the original systems' transient and steady-state behavior with a relative total error of around 4%, while simultaneously accurately reconstructing the total energy. Leveraging this system compression technique, we introduce a model-based controller that exploits the mathematical structure of the compressed model to regulate the configuration of heavily underactuated mechanical systems.

**Index Terms**—Hamiltonian dynamics, model order reduction, autoencoders.

## I. INTRODUCTION

SEVERAL application domains exhibit high-dimensional dynamics, e.g., continuum mechanics, fluid dynamics, quantum systems, financial markets. In such contexts, a useful approach for effective control, which often relies on system-specific expertise, is to find low-dimensional approximations of these systems that preserve their key structural properties [2], [3], [4]. This letter concerns itself with automatic discovery of these approximations using machine learning.

In machine learning, a wealth of research focuses on approximating complex nonlinear dynamical systems while

Manuscript received 13 September 2023; revised 15 November 2023; accepted 30 November 2023. Date of publication 19 December 2023; date of current version 23 January 2024. This work was supported by EU EIC Project EMERGE under Grant 101070918. Recommended by Senior Editor P. Tesi. (Corresponding author: Davide Bacciu.)

Marco Lepri was with the Department of Computer Science, University of Pisa, 56125 Pisa, Italy. He is now with the Biomedical AI Group, NEC Laboratories Europe, 69115 Heidelberg, Germany (e-mail: marco.lepri@neclab.eu).

Davide Bacciu is with the Department of Computer Science, Università di Pisa, 56125 Pisa, Italy (e-mail: davide.bacciu@unipi.it).

Cosimo Della Santina is with the Department of Cognitive Robotics, Delft University of Technology, 2628 CD Delft, The Netherlands, and also with the Institute of Robotics and Mechatronics, German Aerospace Center, 82234 Weßling, Germany (e-mail: c.dellasantina@tudelft.nl).

Digital Object Identifier 10.1109/LCSYS.2023.3344286

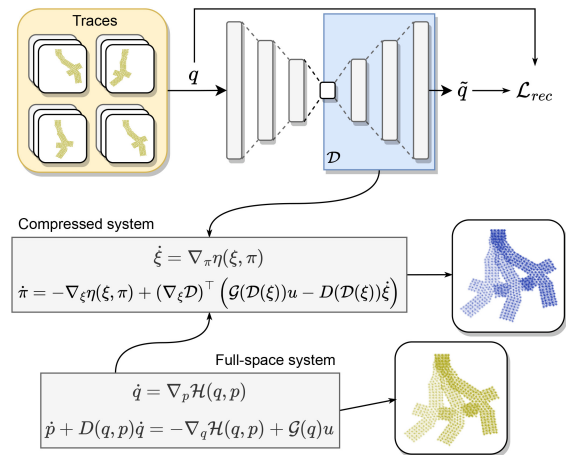


Fig. 1. The proposed strategy is a two-step process. First, we compress the configuration space  $q$  of the physical system into latent representations via deep neural autoencoders. We then generate a compressed dynamical system that uses the learned latent representation while maintaining the Hamiltonian structure of the complete system.

ensuring the learned dynamics fulfill specific structural properties [5], [6], [7], [8], which enabled application to model-based control [9], [10]. The case of direct learning of a compressed dynamics of an high-dimensional system has also been thoroughly investigated in the literature and applied to model-based control [11], [12], [13], [14].

A relevant alternative to directly learn the dynamics combines analytic models with machine learning [15]. An established strategy is to project the dynamics into a latent space using principal component analysis (PCA) [16]. Nonlinear counterparts of PCA, such as neural autoencoders (AE), have been considered only in recent years in [17] and [18]. These pioneering works target the rendering of deformable objects in computer graphics, only providing qualitative analysis of simulation behavior.

In this letter, we make a further step in that direction by combining deep learning with structure-preserving model order reduction [19], [20]. Our approach is schematized in Figure 1. We exploit AEs, investigating both flat [21] and graph-based AEs [22], to extract compressed representations of the system's configuration directly from evolution traces. Then, by combining the decoder of the autoencoder model with the original system specification, we derive a new set of dynamic equations describing the system's dynamics, while at the same time maintaining their original Hamiltonian or

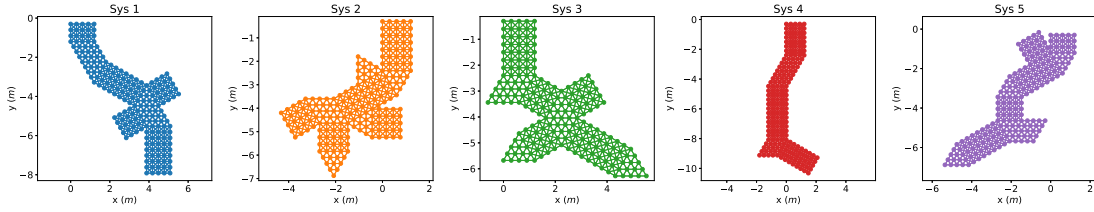


Fig. 2. The five planar models of deformable objects considered in this works, in their rest position when no gravity is present. Each model is a mass-spring-damper network composed of 205 masses and 636 connections.

Lagrangian form. Relying on such a structure, we also propose a closed-loop controller that can regulate the configuration of high-dimensional systems relying on a small amount of inputs. We thoroughly test these methodologies on networks of masses interconnected by springs and dampers that can be seen as a finite element approximation a continuous mechanical system. We conduct numerical simulations in the latent space, focusing on the adherence of the reduced dynamics to the original system and its physical principles. In addition, we explore the capability of the reduced system to approximate the real system under highly constrained latent space dimensions. Finally, experiments on planar posture regulation are performed, exploiting the learned representations and developed controller.

This approach holds potential for diverse applications involving soft robots [23] or deformable objects [24]. For instance, it could address control tasks on soft robots, leveraging approximations of their state to deal with their high dimensionality and limited actuation. Similarly, it may find application in the manipulation of deformable objects, where analogous limitations and challenges exist.

## II. DEEP PHYSICAL COMPRESSION

A generic Port-Hamiltonian<sup>1</sup> system is defined as

$$\dot{x} = [\mathcal{J}(x) - \mathcal{R}(x)]\nabla_x \mathcal{H}(x) + \mathcal{G}(x)u, \quad (1)$$

where  $x$  is the system's state, and  $u$  the input.  $\mathcal{J}$  is a skew-symmetric matrix that specifies the interconnection structure,  $\mathcal{R}$  is a semi-positive definite dissipation matrix,  $\mathcal{G}$  is the input field, and  $\mathcal{H}$  is the Hamiltonian of the system - i.e., its total energy. We consider here systems whose state can be represented as  $x = (q, p) \in \mathbb{R}^{2n}$ , with  $q$  being the configuration and  $p$  the generalized momenta, and having the following structure

$$\dot{q} = \nabla_p \mathcal{H}(q, p), \dot{p} = -D(q)\nabla_p \mathcal{H}(q, p) - \nabla_q \mathcal{H}(q, p) + \mathcal{G}(q)u, \quad (2)$$

where  $D(q) \in \mathbb{R}^{n \times n}$  is a dissipation matrix, assumed positive definite. The term  $D(q)\nabla_p \mathcal{H}(q, p)$  is a common way to describe effects that make the energy strictly decrease in time, as mechanical friction. Note indeed that  $\mathcal{H} = -\dot{q}^\top D(q)\dot{q} \leq 0$ . The control action  $u$  is assumed to be of size  $a$  and thus  $\mathcal{G} \in \mathbb{R}^{n \times a}$ . For the sake of clarity of derivations, we assume the Hamiltonian  $\mathcal{H} : \mathbb{R}^{2n} \rightarrow \mathbb{R}$  to be quadratic in the generalized momenta

$$\mathcal{H}(q, p) = \frac{1}{2}p^\top M^{-1}(q)p + V(q), \quad (3)$$

where  $M : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  is a positive definite matrix and  $V : \mathbb{R}^n \rightarrow \mathbb{R}$  the potential energy. Note that  $\dot{\mathcal{H}} =$

<sup>1</sup>Analogous derivations would be possible in the Lagrangian one.

$-(Mp)^\top D(Mp) \leq 0$  as  $D \succeq 0$ . For example, mechanical systems have such a structure. In this case,  $M$  is called the inertia matrix.

We assume that a description of the system in the form (2) is available. Our goal is to obtain a new system with the same Hamiltonian structure but with a substantially smaller state space, leveraging the concepts described in the following.

### A. Autoencoders

We propose to use a neural autoencoder [21] to compress the configuration space representation from dimension  $n$  to dimension  $m \ll n$ . An autoencoder is composed of two parts; an *encoder network*  $\mathcal{E} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  that compresses  $q$  into its latent representation  $\xi \in \mathbb{R}^{m \ll n}$ , and a *decoder network*  $\mathcal{D} : \mathbb{R}^m \rightarrow \mathbb{R}^n$  which maps  $\xi$  in an approximation of  $q$ . An ideal autoencoder is one such that  $\mathcal{E}(\mathcal{D}(\xi))$  is close to the identity function, despite  $m \ll n$ . Since we want to solely assess the robustness of the deep compressor, we use a simple MSE loss without task-specific regularizations

$$\mathcal{L}_{\text{rec}}(q) = \|q - \mathcal{D}(\mathcal{E}(q))\|_2^2. \quad (4)$$

### B. Compressed System

We perform derivations by assuming an ideal autoencoder, i.e., one for which the loss in (4) is close to zero. We will discuss this hypothesis later. We want to give to the latent dynamics the same Hamiltonian structure of the complete system (2)-(3). We thus impose the following latent dynamics

$$\dot{\xi} = \nabla_\pi \eta(\xi, \pi), \dot{\pi} + \Delta(\xi)\nabla_\pi \eta(\xi, \pi) = -\nabla_\xi \eta(\xi, \pi) + \Gamma(\xi)u, \quad (5)$$

with  $\pi \in \mathbb{R}^{m \ll n}$  being the generalized momenta associated to the latent space configuration  $\xi \in \mathbb{R}^m$  introduced in the previous subsection. The terms  $\Delta, \Gamma$  describe the latent-space dissipation and input field respectively. The latent Hamiltonian/energy is

$$\eta(\xi, \pi) = \frac{1}{2}\pi^\top M_\eta^{-1}(\xi)\pi + V_\eta(\xi), \quad (6)$$

with  $M_\eta$  and  $V_\eta$  being latent space counterparts of  $M$  and  $V$ . We now need to derive all the unknowns from the knowledge of the original dynamics and of the autoencoder. We start by relating the time derivative of the latent configuration with the one of the full configuration by the chain rule  $\dot{q} = \nabla_\xi \mathcal{D}(\xi)\dot{\xi}$ , where  $\nabla_\xi \mathcal{D}$  is the Jacobian of the decoder. Combining the first equation in (5) and (6), we also get  $\pi = M_\eta(\xi)\dot{\xi}$ . We then impose that the latent energy is the same as the total energy of the system

$$\eta(\xi, \pi) = \mathcal{H}(\mathcal{D}(\xi), p(\xi, \pi)) \quad (7)$$

where  $p(\xi, \pi)$  is a mapping from the latent state to  $p$ , and  $\mathcal{H}$  is defined as in (3). The following choices of compressed

potential  $V_\eta$  and inertia matrix  $M_\eta$  fulfill the constraints imposed by (3), (6), and (7)

$$V_\eta(\xi) = V(\mathcal{D}(\xi)) \quad (8)$$

and

$$\begin{aligned} \pi^\top M_\eta^{-1}(\xi)\pi &= p^\top M^{-1}(\mathcal{D}(\xi))p, \\ \Rightarrow \dot{\xi}^\top M_\eta M_\eta^{-1} M_\eta \dot{\xi} &= \dot{\xi}^\top (\nabla_\xi \mathcal{D})^\top M M^{-1} M \nabla_\xi \mathcal{D} \dot{\xi}, \\ \Rightarrow M_\eta(\xi) &= (\nabla_\xi \mathcal{D}(\xi))^\top M(\mathcal{D}(\xi)) \nabla_\xi \mathcal{D}(\xi). \end{aligned} \quad (9)$$

Comparing (2) and (5) and following similar steps as for the energy yields the following expressions for the input field and the dissipation

$$\Gamma(\xi) = (\nabla_\xi \mathcal{D}(\xi))^\top \mathcal{G}(\mathcal{D}(\xi)), \quad \Delta(\xi) = (\nabla_\xi \mathcal{D}(\xi))^\top D(\mathcal{D}(\xi)). \quad (10)$$

To conclude, combining together (5), (6), (8), (9), and (10) yields the compressed Hamiltonian system in (11) as shown at the bottom of the page.

This is a low-dimensional dynamical system with the same mathematical structure of the original high-dimensional one (2). The two models will represent similar behaviors<sup>2</sup> if  $\mathcal{L}_{\text{rec}} \simeq 0$ .

### III. LEARNING COMPRESSED REPRESENTATIONS

Compressed representations are obtained as the latent representations of a neural autoencoder whose architecture depends on the nature of the input information used to encode the uncompressed system. To show the flexible formulation of our approach, in the following we consider two alternative autoencoder configurations. The former is a flat autoencoder, comprising dense feed-forward layers in the encoder and decoder, where the uncompressed system in input is represented by the configuration vector  $q$ . The second is a graph autoencoder which leverages a structured representation of the physical systems meant to highlight their composing parts (e.g., masses) and the relationships existing between them (the adjacency constraints).

Deep learning for graphs (DLG) [22] deals with the adaptive processing of information represented in a structured form. These models typically work by learning to represent the structural elements (nodes, edges) or the full graphs in embedding vectors  $\mathbf{h}$  which can then be used for predictive, descriptive, or generative purposes. The most popular DLG paradigm leverages a message passing scheme [25] exploiting local information exchanges between neighboring nodes and exploits a layered neural architecture (where layering can be defined also by unfolding in time) to promote effective information diffusion across the graph. More formally, the encoding of the  $v$ -th node at layer  $l+1$  is obtained as

$$\mathbf{h}_v^{l+1} = \phi^{l+1} \left( \mathbf{h}_v^l, \Psi(\{\psi^{l+1}(\mathbf{h}_u^l) \mid u \in \mathcal{N}_v\}) \right) \quad (12)$$

<sup>2</sup>Similar is to be intended as the error between the real and reconstructed transients and steady-states being small.

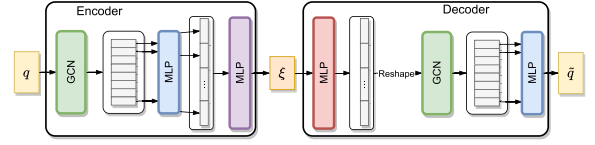


Fig. 3. Graph Autoencoder architecture. Encoder and decoder are implemented as a combination of graph convolutional networks (GCN), to naturally process the graph, and multi-layer perceptrons (MLP), to process the obtained node embeddings.

where  $\phi^{l+1}$ ,  $\psi^{l+1}$  are parameterized neural layers (linear/nonlinear), and  $\Psi$  is a permutation invariant function defined over the embeddings  $\mathbf{h}_u^l$  of the nodes  $u$  in the neighborhood  $\mathcal{N}_v$  of  $v$ , computed at previous step  $l$ . The general formulation in equation (12) can be specialized to cover a wide variety of DLG models, as shown in [22]. Within the scope of this letter, we use a graph autoencoder with the architecture in Figure 3 where both encoder and decoder are implemented with a specialization of (12) using SAGE [26] for neighborhood aggregation followed by an ELU non-linearity in  $\phi$  (GCN block in the figure). The encoder obtains the latent embedding  $\xi$  of the full graph through MLPs, which are also used in the decoder to reconstruct the node features  $\tilde{q}$ .

We comment here on the assumption, made in Section II-B, that the autoencoder achieves close-to-zero loss. In general, this is not trivial to achieve, nor to validate, for any given configuration, apart for those in the training set. However, the dissipative nature of the considered systems, guarantees that the set of *reasonable* configurations is just a portion of the full configuration space. Therefore, it is much more reasonable to assume close-to-zero loss only on that subset, which can be more easily validated using dense enough simulated data as external validation/test set.

### IV. LATENT SPACE CONTROL

We consider under-actuated posture regulation - i.e., we want to generate a control action  $u \in \mathbb{R}^a$  such that the high-dimensional configuration  $q \in \mathbb{R}^n$  of system (2) reaches  $\tilde{q} \in \mathbb{R}^n$ , with  $a < n$ . Call  $\bar{\xi} = \mathcal{E}(\tilde{q}) \in \mathbb{R}^m$  the compressed encoding of  $\tilde{q}$  and assume that the system state  $(q, \dot{q})$  is compressed online into  $(\xi, \dot{\xi})$  through  $\mathcal{E}$  and its Jacobian. The controller we propose has the form:

$$u(\bar{\xi}, \xi, \dot{\xi}) = A^L(\bar{\xi}) \left( \underbrace{\frac{\partial V(\mathcal{D}(\xi))}{\partial \xi}(\bar{\xi})}_{\text{FeedForward}} + \underbrace{\alpha(\bar{\xi} - \xi) - \beta\pi}_{\text{Feedback}} \right), \quad (13)$$

where  $A = (\nabla_\xi \mathcal{D}(\xi))^\top \mathcal{G}(\mathcal{D}(\xi))$ , with  $A^L$  its left inverse, and  $\alpha, \beta \in \mathbb{R}^+$  are positive control gains. This controller is essentially operating an output regulation when taking  $\mathcal{E}(x)$  as output. The task-space<sup>3</sup> closed loop generated by (11)

<sup>3</sup>As common in mechanical systems control literature [27], we refer to task-space dynamics as the second order dynamics describing the evolution of a function of the configurations;  $\mathcal{E}(q)$  in this specific case.

$$\begin{aligned} \dot{\xi} &= \nabla_\pi \eta(\xi, \pi), \quad \dot{\pi} = -\nabla_\xi \eta(\xi, \pi) + (\nabla_\xi \mathcal{D}(\xi))^\top (\mathcal{G}(\mathcal{D}(\xi))u - D(\mathcal{D}(\xi))\nabla_\pi \eta(\xi, \pi)), \\ \text{with } \eta(\xi, \pi) &= \underbrace{\frac{1}{2}\pi^\top \left( (\nabla_\xi \mathcal{D}(\xi))^\top M(\mathcal{D}(\xi)) \nabla_\xi \mathcal{D}(\xi) \right)^{-1} \pi}_{\text{Latent Space Kinetic Energy}} + \underbrace{V(\mathcal{D}(\xi))}_{\text{L.S. Potential E.}}. \end{aligned} \quad (11)$$

and (13) is  $\dot{\pi} = [(\nabla_{\xi}\eta(\xi, \pi) - \nabla_{\xi}\eta(\bar{\xi}, 0)) + \alpha(\bar{\xi} - \xi)] - [(\nabla_{\xi}\mathcal{D}(\xi))^{\top} (D(\mathcal{D}(\xi))\nabla_{\pi}\eta(\xi, \pi) + \beta\pi)]$ , where we used that  $AA^L = I$  and that  $\nabla_{\xi}\eta(\xi, 0) = \nabla_{\xi}(V \circ \mathcal{D})(\xi)$ . The convergence follows with standard arguments that we do not report here for the sake of space, under the assumption that  $\alpha I + (\nabla_{\xi}\eta(\xi, \pi) - \nabla_{\xi}\eta(\bar{\xi}, 0))$  is positive definite in  $\bar{\xi}, 0$ . The closed loop energy  $\eta(\xi, \pi) + \alpha(\bar{\xi} - \xi)^{\top}(\xi - \bar{\xi})$  can be used as Lyapunov candidate, which has time derivative  $\dot{V} = \pi^{\top}[\nabla_{\xi}\mathcal{D}(\xi)^{\top} D(\mathcal{D}(\xi))\nabla_{\xi}\mathcal{D}(\xi) + \beta]\pi \leq 0$ , and invoking La Salle principle. In turn, convergence to the task-space equilibrium implies  $\|\mathcal{E}(q_t) - \mathcal{E}(\bar{q})\| \rightarrow 0$ . Note that we could leverage this common arguments [28] because our learning technique is such that the task-space dynamics of  $\mathcal{E}(x)$  is an Hamiltonian system.

## V. SIMULATIONS

### A. Setup

We evaluate the performance of the approach in compressing high-dimensional mechanical systems. We focus on continuously deformable planar bodies subject to a gravitational field and generic external perturbing forces. This class of systems is quite relevant from an application perspective as it is central in robotic manipulation of deformable objects [24] and control of soft robots [3]. We consider mass-spring-damper models as high-dimensional models of these systems [23]. This is a widely used technique to approximate soft-body dynamics by discretizing their volume as a set of masses (nodes) interconnected by ideal springs and dampers (edges). We use simulation data of the systems to train an autoencoder to reconstruct their configuration  $q$ . Then, exploiting the learnt latent representation  $\xi$ , we simulate the corresponding full-space system by solving the compressed dynamics equation (11).

1) *Data*: We use five high-dimensional, randomly generated systems in the form of (2). Figure 2 shows the considered systems in their rest position. Each system is made of 200 masses and  $e = 636$  connections resulting in  $n = 400$  degrees of freedom captured in the configuration vector  $q$ . We consider systems immersed in a constant gravitational field. For each system, we perform 7 simulations to generate training data and 28 simulations for test purposes. Gravity conditions change for each simulation. The initial configuration  $q(0)$  is randomly generated, while the systems always start at rest, i.e.,  $p(0) = 0$ . More details on the data generation can be found in Section A.1 of the supplementary material.

2) *Training and Model Selection*: A flat autoencoder and a graph autoencoder are trained for each system, with a latent size of 5 units. The same neural architecture is used across systems, while hyperparameters are selected via grid search for each system. Training, validation and test reconstruction scores (MSE) of the best model, for each system, are reported in Table I. More details on the training and model selection can be found in Section A.2 of the supplementary material.

### B. Simulation Results

In order to validate and evaluate the approach, we use the trained models to reconstruct the test full-space simulations, and we compare results in terms of reconstructed state  $(q, \dot{q})$  and energy  $\mathcal{H}$ . In particular, we evaluate the approach in two different ways. First, for *pointwise evaluation*, we reconstruct the system state and energy at each step of the real simulation by applying the autoencoder end-to-end to the real system configuration  $\mathcal{D}(\mathcal{E}(q))$  and velocity  $\nabla_{\mathcal{E}(q)}\mathcal{D} \cdot \nabla_q \mathcal{E} \dot{q}$ . Second,

TABLE I  
TRAINING, VALIDATION AND TEST MSE SCORE FOR THE BEST FLAT AND GRAPH AUTOENCODERS, FOR EACH SYSTEM

System	Model	Train. mse	Valid. mse	Test mse
1	Flat AE	2.17e-4	4.73e-4	<b>4.24e-4</b>
	Graph AE	3.40e-4	5.24e-4	7.25e-4
2	Flat AE	1.13e-4	2.48e-4	<b>3.08e-4</b>
	Graph AE	12.24e-4	15.37e-4	17.94e-4
3	Flat AE	1.69e-4	4.41e-4	<b>3.75e-4</b>
	Graph AE	3.73e-4	8.13e-4	7.37e-4
4	Flat AE	4.75e-4	18.08e-4	12.51e-4
	Graph AE	5.56e-4	13.72e-4	<b>9.71e-4</b>
5	Flat AE	8.53e-4	11.7e-4	12.70e-4
	Graph AE	7.41e-4	9.48e-4	<b>9.59e-4</b>

we use equation (11) to simulate the evolution in the compressed space and reconstruct the configuration  $\mathcal{D}(\xi)$ , velocity  $(\nabla_{\xi}\mathcal{D})\dot{\xi}$  and energy  $\eta(\xi, \pi)$  from the latent variables at each simulation step (*compressed evaluation*).

Figure 4 shows the average pointwise and compressed MSE w.r.t. time for each system. The two models have similar results in the compressed simulations, with the graph autoencoder having a slightly higher error, although it largely depends on the considered system. In all the cases, the compressed error on  $q$  is a few orders of magnitude higher than the pointwise error. Interestingly, the error on  $\dot{q}$  follows a similar behaviour although the models are never explicitly trained to reconstruct this information.

The reconstructed energy  $\eta$  has the opposite trend: it is better reconstructed in the compressed case than in the pointwise case. This is due to the fact that pointwise reconstructing the state does not allow energy variations to govern the dynamics of the system, while this is possible in the compressed case. As a further evidence, Figure 5 shows the evolution of the kinetic, potential, and total energy in a simulation. While the single kinds of energy might not be reconstructed as precisely in the compressed case, the total energy maintains the same non-increasing behaviour typical of dissipative systems. Qualitatively, compressed simulations are stable and loyal to the real ones, with a good match of the full transient. Figure 6 shows some frames from an example simulation. The portions where the real and reconstructed systems are not perfectly aligned are also those that exhibit major and more varied oscillations, such as the bottom portion of the central chain or some lateral structures. Videos can be found here.

### C. Compression Analysis

We further analyze how much the approach can compress the systems' state. Figure 7 shows the test MSE on the five systems varying the latent state size for both flat and graph autoencoders. In most cases, 3 variables are enough to efficiently represent the system's state with a reasonable approximation error, while using more variables typically results in small or marginal improvements in the error. Using 2 variables seems to not be enough to effectively capture all systems' behaviours as can also be observed from the reconstructed trajectory of some of the masses in Figure 8. We can notice that, with 2 variables, minor oscillations are not correctly reconstructed, and there is a consistent gap between the real and reconstructed trajectory. This does not happen when using 3 or more variables. The 2 variables case is also useful to show what happens when the assumption that

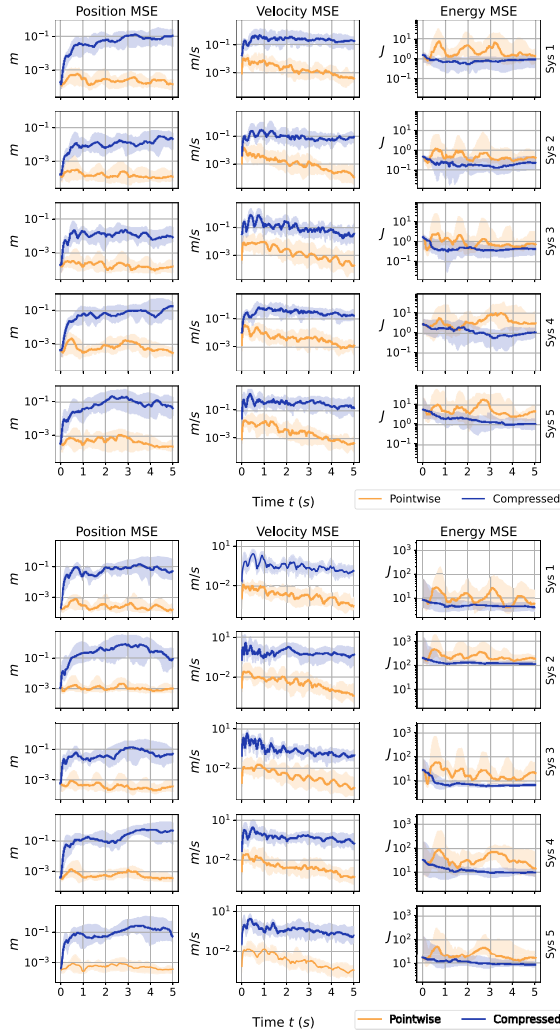


Fig. 4. MSE on the reconstructed position  $\mathcal{D}(\xi)$ , velocity  $(\nabla_{\xi}\mathcal{D})\dot{\xi}$  and total energy  $\eta(\xi, \pi)$  of the systems in the test trajectories using: a flat autoencoder (left) and a graph autoencoder (right). For each plot, the bold line is the median error over the trajectories, while the shaded area represents the 20-80 percentiles.

the model achieves close-to-zero loss does not hold. Indeed, the reduced model still approximate the real-space model, although losing the ability to represent some of its particularity. We also believe the models can be successfully used for latent space simulations, although these could deviate much more from the real ones.

#### D. Control Experiments

1) *Setup*: We test the proposed controller by simulating its application for planar posture regulation on the second spring network in Figure 2. The system is actuated by a generalized force  $\tau \in \mathbb{R}^2$  applied to a single mass at each simulation step. Our controller employs an autoencoder with latent size  $m = 2$ , trained as in the previous sections. The actuation matrix is therefore the selection matrix  $\begin{bmatrix} 0 & \dots & 0 & I & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{2 \times 2n}$ ,

where  $a$  is the index of the actuated mass and  $I \in \mathbb{R}^{2 \times 2}$  is the identity matrix. We perform 50 simulations, randomly selecting the target configuration among the configurations  $\bar{q}$  in the training/validation simulations. The actuation mass is randomly selected among three fixed candidates chosen in correspondence with the lateral structures and as far as

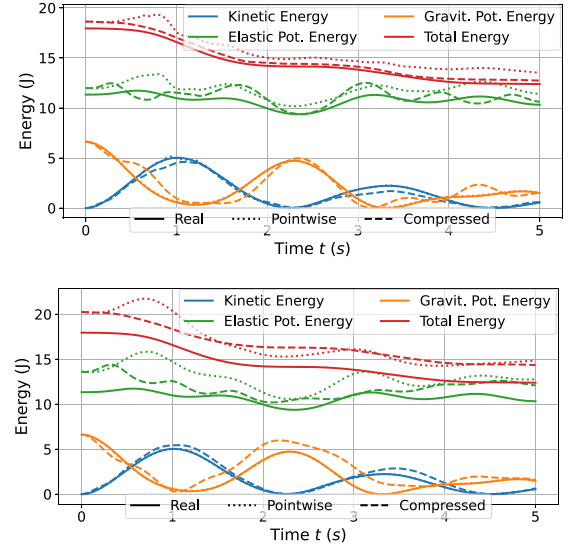


Fig. 5. Example of evolutions of the kinetic, potential and total energies in a simulation for the full-space and reduced systems using: a flat autoencoder (top) and a graph autoencoder (bottom). The total energies are the Hamiltonians  $\mathcal{H}$  and  $\eta$  introduced in (3) and (7) respectively. The solid line is the energy of system (2), while the dotted and dashed lines are the pointwise reconstructed and compressed energy.

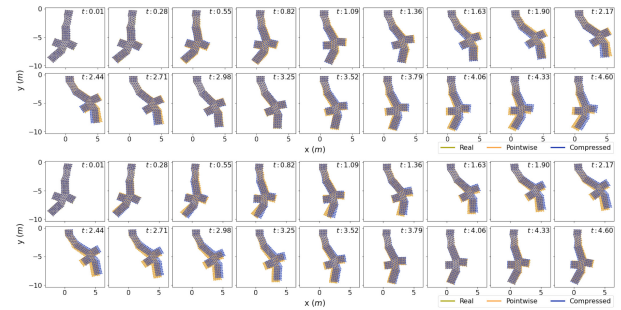


Fig. 6. Comparison between frames from a real simulation (yellow), a pointwise reconstructed simulation (orange), and a reduced simulation (blue) using a flat autoencoder (above) and a graph autoencoder (below).

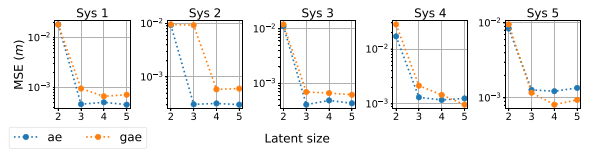


Fig. 7. Test MSE on the five considered systems varying the latent state size of a flat autoencoder and a graph autoencoder.

possible from the structure edges. The initial state is always the rest position with zero initial velocity. The simulations are 5 seconds long.

2) *Results*: We evaluate the controller according to the MSE between the full-space system configuration at time  $t$  and the target configuration  $\bar{x}$ . We report the resulting Figure 9. The dashed line represents the median error among the considered simulations, while the band represents the 25-75 percentiles of the error.

## VI. CONCLUSION

This letter investigated the application of deep autoencoders to the compression of high-dimensional dynamical systems, while maintaining Hamiltonian/Lagrangian structural

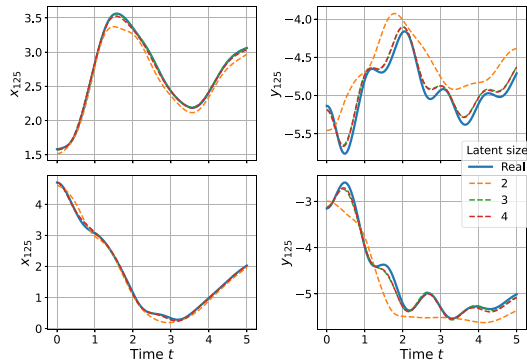


Fig. 8. Example of reconstructed trajectory with different sizes of latent state.

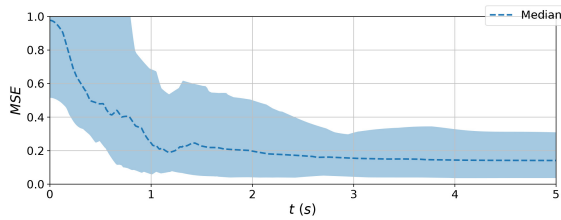


Fig. 9. Evolution of the normalized MSE through time and across several simulation, calculated between the reference configuration  $\bar{q}$  and the actual configuration  $q(t)$ . The shaded are refers to the first and third quartile.

properties in the low-dimensional approximation. The approach was extensively validated and evaluated over several high-dimensional mass-spring-damper models. The reduced systems were exploited to perform simulation in the latent space, from which the original complete space evolution were reconstructed. We also proposed a possible usage of such compressed representations for planar posture regulation of highly underactuated systems, evaluating the developed controller in simulations. Future work will focus on developing and testing data-driven control algorithms for high-dimensional systems where model-based strategy are used in conjunction with the learned latent model. We will also dive into extending the approach to systems whose Hamiltonian differs from (3), including fluid dynamics [29] and astronomy [30].

#### ACKNOWLEDGMENT

For additional detailed information related to this letter, please refer to the supplementary material accessible at [1].

#### REFERENCES

- [1] M. Lepri, D. Bacciu, and C. D. Santina, "Neural autoencoder-based structure-preserving model order reduction and control design for high-dimensional physical systems," 2023, *arXiv:2312.06256*.
- [2] A. Agrachev and A. Sarychev, "Navier–Stokes equations: Controllability by means of low modes forcing," *J. Dyn. Control Syst.*, vol. 7, pp. 108–152, Mar. 2005.
- [3] C. Della Santina, C. Duriez, and D. Rus, "Model-based control of soft robots: A survey of the state of the art and open challenges," *IEEE Control Syst. Mag.*, vol. 43, no. 3, pp. 30–65, Jun. 2023.
- [4] A. A. Kaptanoglu, K. D. Morgan, C. J. Hansen, and S. L. Brunton, "Physics-constrained, low-dimensional models for magnetohydrodynamics: First-principles and data-driven approaches," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 104, no. 1, Jul. 2021, Art. no. 015206.
- [5] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–16.
- [6] M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho, "Lagrangian neural networks," in *Proc. ICLR Workshop Integr. Deep Neural Models Differ. Equ.*, 2020, pp. 1–9.
- [7] T. Beckers, J. Seidman, P. Perdikaris, and G. J. Pappas, "Gaussian process port-Hamiltonian systems: Bayesian learning with physics prior," in *Proc. IEEE 61st Conf. Decis. Control (CDC)*, 2022, pp. 1447–1453.
- [8] G. Evangelisti and S. Hirche, "Physically consistent learning of conservative lagrangian systems with Gaussian processes," in *Proc. IEEE 61st Conf. Decis. Control (CDC)*, 2022, pp. 4078–4085.
- [9] J. Liu, P. Borja, and C. D. Santina, "Physics-informed neural networks to model and control robots: A theoretical and experimental investigation," 2023, *arXiv:2305.05375*.
- [10] M. Lutter and J. Peters, "Combining physics and deep learning to learn continuous-time dynamics models," *Int. J. Robot. Res.*, vol. 42, no. 3, pp. 83–107, 2023.
- [11] M. T. Gillespie, C. M. Best, E. C. Townsend, D. Wingate, and M. D. Killpack, "Learning nonlinear dynamic models of soft robots for model predictive control with neural networks," in *Proc. IEEE Int. Conf. Soft Robot. (RoboSoft)*, 2018, pp. 39–45.
- [12] S. L. Brunton, "Applying machine learning to study fluid mechanics," *Acta Mechanica Sinica*, vol. 37, pp. 1–9, Jan. 2022.
- [13] R. Perera, D. Guzzetti, and V. Agrawal, "Graph neural networks for simulating crack coalescence and propagation in brittle materials," *Comput. Methods Appl. Mech. Eng.*, vol. 395, May 2022, Art. no. 115021.
- [14] F. Mählknecht et al., "Using spectral submanifolds for nonlinear periodic control," in *Proc. IEEE 61st Conf. Decis. Control (CDC)*, 2022, pp. 6548–6555.
- [15] R. Swischuk et al., "Projection-based model reduction: Formulations for physics-based machine learning," *Comput. Fluids*, vol. 179, pp. 704–717, Jan. 2019.
- [16] K. Willcox and J. Peraire, "Balanced model reduction via the proper orthogonal decomposition," *AIAA J.*, vol. 40, no. 11, pp. 2323–2330, 2002.
- [17] 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, May 2019.
- [18] S. Shen et al., "High-order differentiable autoencoder for nonlinear model reduction," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–15, 2021.
- [19] B. Karasözen, S. Yildiz, and M. Uzunca, "Structure preserving model order reduction of shallow water equations," *Math. Methods Appl. Sci.*, vol. 44, no. 1, pp. 476–492, Jan. 2021.
- [20] J. S. Hesthaven, C. Pagliantini, and N. Ripamonti, "Structure-preserving model order reduction of hamiltonian systems," 2021, *arXiv:2109.12367*.
- [21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: <http://www.deeplearningbook.org>
- [22] D. Bacciu, F. Errica, A. Micheli, and M. Podda, "A gentle introduction to deep learning for graphs," *Neural Netw.*, vol. 129, pp. 203–221, Sep. 2020.
- [23] 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.
- [24] J. Zhu et al., "Challenges and outlook in robotic manipulation of deformable objects," *IEEE Robot. Autom. Mag.*, vol. 29, no. 3, pp. 67–77, Sep. 2022.
- [25] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 3, 2017, pp. 2053–2070.
- [26] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.
- [27] X. Wu, C. Ott, A. Albu-Schäffer, and A. Dietrich, "Passive decoupled multitask controller for redundant robots," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 1, pp. 1–16, Jan. 2022.
- [28] R. Ortega, J. G. Romero, P. Borja, and A. Donaire, *PID Passivity-Based Control of Nonlinear Systems With Applications*. Hoboken, NJ, USA: Wiley, 2021.
- [29] R. Maulik, B. Lusch, and P. Balaprakash, "Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders," *Phys. Fluids*, vol. 33, no. 3, Mar. 2021, Art. no. 037106.
- [30] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, "Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy," *Nat. Phys.*, vol. 18, no. 1, pp. 112–117, 2022.