TYPE Original Research
PUBLISHED 23 August 2023
DOI 10.3389/fphy.2023.1253642

Check for updates

OPEN ACCESS

EDITED BY
Viet-Thanh Pham,
Ton Duc Thang University, Vietnam

REVIEWED BY
Samaneh Soradi-zeid,
University of Sistan and Baluchestan, Iran
Oscar Castillo,
Instituto Tecnológico de Tijuana, Mexico
Fernando Serrano,
National Autonomous University of Honduras, Honduras, in collaboration with reviewer OC

*CORRESPONDENCE
Hajid Alsubaie,
h.alsubaie@tu.edu.sa

RECEIVED 05 July 2023
ACCEPTED 07 August 2023
PUBLISHED 23 August 2023

CITATION
Alsubaie H (2023), A neural state-space-based model predictive technique for effective vibration control in nano-beams.
Front. Phys. 11:1253642.
doi: 10.3389/fphy.2023.1253642

A neural state-space-based model predictive technique for effective vibration control in nano-beams

Hajid Alsubaie*

Department of Mechanical Engineering, College of Engineering, Taif University, Taif, Saudi Arabia

Model predictive control (MPC) is a cutting-edge control technique, but its susceptibility to inaccuracies in the model remains a challenge for embedded systems. In this study, we propose a data-driven MPC framework to address this issue and achieve robust and adaptable performance. Our framework involves systematically identifying system dynamics and learning the MPC policy through function approximations. Specifically, we introduce a system identification method based on the Deep neural network (DNN) and integrate it with MPC. The function approximation capability of DNN enables the controller to learn the nonlinear dynamics of the system then the MPC policy is established based on the identified model. Also, through an added control term the robustness and convergence of the closed-loop system are guaranteed. Then the governing equation of a non-local strain gradient (NSG) nano-beam is presented. Finally, the proposed control scheme is used for vibration suppression in the NSG nano-beam. To validate the effectiveness of our approach, the controller is applied to the unknown system, meaning that solely during the training phase of the neural state-space-based model we relied on the data extracted from the time history of the beam's deflection. The simulation results conclusively demonstrate the remarkable performance of our proposed approach in effectively suppressing vibrations.

KEYWORDS

model predictive control, data-driven MPC, nano system, robust control, NSG theorem

1 Introduction

Neural networks have brought about substantial changes in the handling of nonlinear systems, holding immense potential to revolutionize the control field. Their unique ability to model and interpret complex, high-dimensional dynamics positions them as key contributors in areas where traditional mathematical models typically face challenges [1, 2]. State-space models based on neural networks are capable of mapping the intricate relationships between the inputs, outputs, and internal states of nonlinear systems, using their capacity to approximate any continuous function [3, 4]. They utilize past and current data, learning the nonlinear dynamics, to predict the future states of a system based on the present state and control inputs.

MPC is a highly effective control approach widely employed in diverse engineering domains to achieve superior control performance compared to conventional methods [5, 6]. By utilizing a predictive model of the system, MPC optimizes control actions over a finite time horizon. MPC enables the consideration of future system behavior and constraints, allowing for more precise and robust control actions [7, 8]. Hence, to now, MPC has been widely used in various engineering domains to achieve superior control performance compared to conventional methods [9–12].

Nanostructures, including nano-beams, have generated considerable attention across a range of disciplines due to their superior mechanical attributes and the profound potential they possess for advancements in nanotechnology applications. Up to this point, an expansive corpus of research has been established within this particular field of study. Undeniably, it is of utmost importance to sustain these scholarly pursuits. Doing so will not only enhance our comprehension but will also enable us to leverage these findings more effectively for the greater benefit. For example, the study by Ohashi et al. [13] underscores the necessity for the stable delivery of nano-beams in facilitating advanced nanoscale analyses. However, the diminutive dimensions of these structures pose unique challenges pertaining to their dynamic behavior, notably when exposed to vibrational forces [14, 15].

Given the escalating demand for nanotechnology across diverse sectors, from medicine to information technology, it is clear that this area of research requires continued exploration and development [16–18]. However, the miniaturized scale of these structures brings forth distinct challenges related to their dynamic behavior, especially when exposed to vibrations. Consequently, the study and control of nano-beams have emerged as an integral field of study, aiming to ensure the reliability and operational efficacy of nano-devices. This focus is evident in the comprehensive review by Roudbari et al. [19], which emphasizes the significance of size-dependent continuum mechanics models for micro and nano-structures. Similarly, the research by Miandoab et al. [20] offers a nonlocal and strain gradient-based model for electrostatically actuated silicon nano-beams, thereby addressing specific control issues inherent in such structures.

Sliding mode controllers [21, 22] and other robust controllers [23] have been extensively investigated and suggested for nano and microsystems. However, when it comes to control in nano and microsystems, the application of MPC has not been adequately proposed. The main reason behind this limitation is the substantial amount of uncertainties present in these systems. Unlike other control methods, MPC relies on having an accurate model of the system, which is practically impossible to obtain in real-world nano and microsystems. Therefore, despite the potential advantages of MPC, its practical implementation in this domain remains unfeasible. As researchers continue to explore novel control approaches, finding ways to overcome these challenges and devise MPC strategies for nano and microsystems will be essential.

The quest for optimally controlling nonlinear and uncertain systems is a formidable challenge in modern control theory, where traditional methods like MPC and robust control present significant advantages but also face limitations. MPC's high computational costs and reliance on accurate system models make it less suited for real-time applications and systems with complex, uncertain dynamics. On the other hand, robust control handles uncertainties [24, 25] but often leads to suboptimal performance and does not directly account for state and control constraints. For example, in [26], MPC was suggested as a method for atomic force microscopy. However, this control strategy relies on the assumption that a complete and perfectly accurate system model is available, which is often not a reality in actual practice due to the unpredictability and complexity inherent in real-world scenarios.

In practice, obtaining a fully accurate model of the system is challenging due to inherent uncertainties and practical limitations. Thus, the assumptions made in the design of the controller do not hold true in practical applications. This emphasizes the need to develop control strategies that can effectively handle the uncertainties and limitations present in nano and microsystems without relying on perfect system models.

Recently, data-driven methods, as presented in studies such as [27–29] promise a more efficient and adaptive approach. These methods leverage machine learning to learn system dynamics and control policies, reduce the computational burden, and adapt to system changes. For instance, Li and Tong [30] applied an encoder-decoder neural network model for developing an MPC. Their focus was on the efficient control of an HVAC system, and their results showcased promising convergence. Also, in a more recent study, Bonassi et al. [31] offered a comprehensive discussion on the integration and evaluation of various recurrent neural network structures within the framework of MPC.

Despite the aforementioned advancements, some problems persist in the majority of studies within this field. Most notably, there is a consistent lack of guaranteed convergence and stability, which presents significant challenges for the advancement of machine learning-based MPC solutions. Hence, more research is needed to refine data-driven approaches for optimal control of nonlinear and uncertain systems, focusing on their performance, computational efficiency, robustness, convergence, generalizability, and data requirements. This challenge has served as a significant motivation for our current study. Recognizing the limitations of existing control techniques, particularly in the context of nano and microsystems, we are driven to explore innovative approaches that can overcome the hurdles associated with uncertainties in these systems. Through our study, we aspire to pave the way for practical implementation and real-world applications of advanced control methods in the realm of nano and microsystems.

We propose a neural state space-based model predictive control by integration of DNN with MPC. DNNs have the remarkable ability to learn complex patterns and capture intricate relationships from data [32, 33]. Therefore, we utilize Deep Neural Networks (DNNs) as neural state space models for the systems. Through training on accessible data, DNNs can construct nonlinear models that effectively approximate the system's behavior, even when uncertainties and disturbances are present. This provides a valuable advantage when dealing with nano-beam vibrations, where comprehensive knowledge of the system's dynamics may be elusive. The integration of DNNs with MPC enables the development of an intelligent control framework that effectively compensates for the limitations of MPC and suppresses vibrations in NSG nano-beams. In this study, we enhance the control strategy by integrating an additional control term, ensuring the robustness of the controller and promoting the convergence of the closed-loop system to the desired value. This synergistic combination of DNNs and MPC acts as a corrective component, elevating the stability and performance of the control system.

The structure of this paper is as follows: Section 2 offers a comprehensive introduction to the fundamental concepts and principles, setting the groundwork for our proposed framework. Subsequently, we present and validate our framework in subsequent sections. Section 3 focuses on the governing equations of the NSG nano-beam, taking into account its unique characteristics. In Section 4, we apply the proposed controller to the nano-beam and thoroughly investigate its performance through simulations. Finally, in Section 5, we present the concluding remarks summarizing the key findings and suggest areas for further improvements.

2 The proposed control scheme

In this section, we present some preliminaries and our control approach. Firstly, in Section 2.1 we describe the methodology used to construct a neural state-space-based model that captures the dynamics of the system accurately. Subsequently, in Section 2.2, we delineate the MPC policy employed in our framework. We outline the optimization problem formulation and the steps involved in generating control actions over a finite time horizon. Furthermore, in Section 2.3, we introduce the robustness term that is added to enhance the controller's stability and performance. Additionally, we depict the control scheme, illustrating how the neural state-space-based MPC policy and robustness term are integrated to form a cohesive control framework.

2.1 Neural state-space models

Neural state-space models encompass a category of models that employ neural networks to depict the functions that define the nonlinear state-space representation of a system. In traditional control theory, state-space models are used to describe the behavior of dynamic systems by representing the relationship between the system's inputs, outputs, and internal states. Suppose a general state-space form with the following mathematical representation. The mathematical form of the system is given by

$$\dot{x}(t) = f_c(x(t)) + h_c(x(t))u(t) \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector, and the input vector is represented by $u \in \mathbb{R}^m$. Also, f_c and h_c are two static non-linear mappings. The discrete-time formulation is given by

$$x(k+1) = f(x(k)) + h(x(k))u(k) \tag{2}$$

in linear systems, these equations are typically represented by linear functions.

Assumption 1. The system dynamics functions $f(x)$ and $h(x)$ are assumed to be Lipschitz continuous, indicating that there exists a Lipschitz constant that governs the behavior of $f(x)$ and $h(x)$ as follows

$$\| f(\bar{x}(k)) + h(\bar{x}(k))\bar{u}(k) - f(x(k)) + h(x(k))u(k) \| \leq \varepsilon_x \| \bar{x} - x \| + \varepsilon_u \| \bar{u} - u \| \tag{3}$$

in which $\varepsilon_x, \varepsilon_u \geq 0$ are constants values.

However, in many real-world scenarios, systems exhibit nonlinear behavior that cannot be accurately captured by linear models. Neural networks offer a powerful framework for representing and learning nonlinear relationships [34] making them well-suited for constructing state-space models for such systems. Here, we introduce a neural state-space model, where the state equation is represented by neural networks. The neural network represents the function that describes the behavior of the system's states, although here we used DNN, these networks can be designed as recurrent neural networks (such as LSTM or GRU), or other types of architectures depending on the characteristics of the system being modeled.

The DNN in the neural state-space model is trained using data from the system. This training involves optimizing the network parameters to minimize the discrepancy between the model's predictions and the observed behavior of the system. Various techniques, such as gradient descent or backpropagation, can be employed for this purpose. Once trained, the neural state-space model is used to simulate the behavior of the system, estimate its internal states based on available inputs and outputs, and predict future system responses. The mathematical form of the learned state space is given by

$$\hat{x}(k+1) = \hat{f}(\hat{x}(k)) + \hat{h}(\hat{x}(k))u(k) \tag{4}$$

where the variables $\hat{x}(k)$ and $u(k)$ correspond to the baseline state and control input, respectively, for the baseline model. The function $\hat{f}(\hat{x}(k))$ represents the baseline model dynamics (here is the neural network).

It is noteworthy that the learned state space model as represented in Eq. 4 can be backpropagated, and its derivatives are computable through the application of automatic differentiation. Here we assume $\hat{f}$ and $\hat{h}$ satisfy the conditions of Lipschitz continuity and general continuity. This assumption of Lipschitz continuity and general continuity for the dynamic functions is widely recognized in the field. The current study also acknowledges and incorporates this fundamental premise.

2.2 Nonlinear MPC for the baseline model

By data-driven NMPC we refer to establishing MPC policies based on the learned neural state space-based model.

The cost function of MPC associated with the neural state space model 4) is defined as follows

$$J_N = \sum_{k=0}^{N-1} Q(\hat{x}(k), u(k)) + T(\hat{x}(N)), \tag{5}$$

The cost function for plant 1) is determined by considering several factors. It incorporates the stage cost, represented by $Q(\hat{x}(k), u(k))$, which takes into account the current state x and control input u at each stage. Additionally, there is a terminal cost component denoted by $T(\hat{x}(N))$, which captures the cost associated with the final state. In the current study, the terminal cost was not applied, despite its mention within the theoretical formulations. This was intended to preserve the generality of the presentation. Also, The cost function is defined over a prediction horizon N, encompassing the control inputs $u(0, 1, \ldots, N-1)$, state variables $\hat{x}(0, 1, \ldots, N)$. Eq. 5 illustrates the system in its general form; however, the controller proposed in this study has been specifically designed for affine systems. By employing the baseline model 4) and initiating from an initial state $\hat{x}(0) = \hat{x}_0$, the data-driven NMPC is introduced. This approach aims to minimize the cost function 5) with respect to the baseline model 4) which ius given by

$$\begin{aligned}(\hat{x}^o, u^o) &= \arg\min_{\bar{x},\bar{u}} J_N(\hat{x}, u)\\ \text{s.t.}\quad \hat{x}(k+1) &= \hat{f}(\hat{x}(k)) + \hat{h}(\hat{x}(k))u(k)\\ \hat{x}(0) &= \hat{x}_0.\end{aligned} \tag{6}$$

the augmented cost function, denoted as J, expands the original cost function 5) to include the constraints and Lagrange multipliers. It is obtained by integrating the Hamiltonian function across the prediction horizon N. Therefore, the augmented cost function can be expressed as:

$$\begin{aligned}C(k) &= Q(\hat{x}(k), u(k)) + \lambda^T(k+1)\left[\hat{f}(\hat{x}(k)) + \hat{h}(\hat{x}(k))u(k)\right]\\ \bar{J}_N &= \sum_{k=0}^{N-1}\left(C(k) - \lambda^T(k+1)\hat{x}(k+1)\right) + T(\hat{x}(N)),\end{aligned} \tag{7}$$

In this context, $\lambda(k+1)$ represents the Lagrange multiplier associated with the dynamics of the baseline model 4). It is worth noting that these Lagrange multipliers, also known as co-states, play a significant role. By solving the NMPC problem (Eq. 6), we obtain the optimal trajectories for the baseline denoted as $\hat{x}^o(k)$ and $u^o(k)$. During this optimization process, the Karush-Kuhn-Tucker (KKT) conditions are employed to derive the necessary optimality conditions, ensuring that the augmented cost function 7) is effectively minimized. These optimality conditions can be expressed as follows:

$$C_u(k) = 0, k = 0, 1, \ldots, N-1 \tag{8 – a}$$

$$\lambda(k) = C_{\hat{x}}(k), k = 0, 1, \ldots, N-1 \tag{8 – b}$$

$$\lambda(N) = T_{\hat{x}}(\hat{x}(N)), k = 0, 1, \ldots, N-1 \tag{8 – c}$$

By utilizing the KKT conditions and the Lagrange multipliers $\lambda(k+1)$ as well as the baseline optimal solution $\hat{x}^o(k)$ and $u^o(k)$ can be computed online. Taking into account the KKT conditions, we have the following expressions:

$$\begin{aligned}C_{\bar{u}}(k) = Q_{\bar{u}}(\hat{x}^o(k), u^o(k)) + \lambda^T(k+1)\\ \left[\hat{f}_u(x^o(k)) + \hat{h}_u(\hat{x}^o(k))u^o(k)\right] = 0\end{aligned} \tag{9 – a}$$

$$\lambda(k) = Q_{\hat{x}}(\hat{x}^o(k), u^o(k)) + \lambda^T(k+1)\left[\hat{f}_{\hat{x}}(x^o(k)) + \hat{h}_{\hat{x}}(\hat{x}^o(k))u^o(k)\right], \tag{9 – b}$$

$$\lambda(N) = T_{\hat{x}}(\hat{x}^o(N)). \tag{9 – c}$$

By considering Eqs 8, 9, the Lagrange multipliers $\lambda(k+1)$ can be obtained through the following calculations:

$$\lambda(k+1) = Q_{\hat{x}}(k) + \lambda^T(k+1)\hat{f}_{\hat{x}}(k), \tag{10}$$

Additionally, the Lagrange multipliers (10) are considered to be the baseline optimal Lagrange multipliers denoted as $\lambda^o(k)$.

Consider the scenario involving system 1), baseline model 4), and data-driven NMPC 6). The data-driven NMPC yields an optimal solution denoted by $\hat{x}^o(k)$ and $u^o(k)$, which serves as the baseline solution. However, due to an error between the real systems and the baseline systems a perturbation $\tilde{N}(x,t)$ may arise as a result. If this perturbation does not affect the status of the constraints, the optimal solution for system 1) can be adjusted as $u = u_\delta(k) + u^o(k)$. The focus now lies on devising a robust NMPC framework capable of effectively addressing the challenges arising from unknown bounded disturbances and errors in the neural state space model. This objective is explicitly articulated in the following part.

2.3 Robust NMPC with guaranteed convergence

We define the error of the system as $e = x - x_d$, where x_d represents the desired reference trajectory. The robust control tracking control law can be expressed as follows:

$$\begin{aligned}u_\delta &= -\frac{u_r}{\hat{h}(x)} = -\frac{u_r}{h(x) + \tilde{M}}\\ u_r &= \hat{f}_c(\hat{x}(t)) + \dot{x}_d + \delta e + \mu\, sign\,(e)\\ \tilde{M} &= \hat{h}(x) - h(x)\end{aligned} \tag{11}$$

Here, δ and μ are user-defined parameters that need to be positive.

Theorem 1. Assuming that the compound uncertainty remains within established boundaries, the proposed control law, as described in Eq. 12, in conjunction with the MPC defined in Eq. 6 and derived from the neural state-space model in Eq. 4 handles residual errors in the tracking control, and guarantees the convergence of the states in the state-space model 1) towards the desired values.

Proof. Suppose that the error arising from the estimation of the system's dynamics and the error in the baseline initial condition can be combined into a single term denoted as $\tilde{N} = f_c(x) + h_c(x)u^o - \hat{f}(\hat{x}) + \hat{h}(\hat{x})u^o$. By substituting Eq. 11 and Eq. 1 in the time derivative of the defined error, we achieve:

$$\begin{aligned}\dot{e} = \dot{x} - \dot{x}_d &= f_c(x(t)) + h_c(x(t))u(t) - \dot{y}_d\\ &= f_c(x(t)) + h_c(x(t))(u_\delta + u^o) - \dot{x}_d\\ &= f_c(x(t)) + h_c(x(t))u_\delta + \hat{f}(\hat{x}) + \hat{h}(\hat{x})u^o + \tilde{N} - \dot{x}_d\end{aligned} \tag{12}$$

Now due to the optimality of u^o we know after a shoer period of time $\hat{f}(\hat{x}) + \hat{h}(\hat{x})u^o = 0$, also we know $h(x)\frac{u_r}{h(x)+\tilde{M}} = u_r - \frac{\tilde{M}}{h(x)+\tilde{M}}u_r$. Therefore we have

$$\begin{aligned}\dot{e} &= \dot{x} - \dot{x}_d\\ &= f_c(x(t)) - \hat{f}_c(\hat{x}(t)) + \frac{\tilde{M}}{h(x)+\tilde{M}}u_r - \delta e - \mu\, sign\,(e) + \tilde{N}\end{aligned} \tag{13}$$

We define $f_c(x(t)) - \hat{f}_c(\hat{x}(t)) + \frac{\tilde{M}}{h(x)+\tilde{M}}u_r = \tilde{M}_d$ and substitute in Eq. 13 which results in

$$\dot{e} = \dot{x} - \dot{x}_d = \tilde{M}_d - \delta e - \mu\, sign\,(e) + \tilde{N} \tag{14}$$

Now, let us consider a Lyapunov function candidate denoted as $V(x)$, which is expressed as:

$$V = \frac{1}{2}e^2 \geq 0 \tag{15}$$

The derivative of the Lyapunov function $V(x)$ with respect to time is expressed as

$$\dot{V} = e\dot{e} = e\left(\tilde{M}_d - \delta e - \mu\, sign\,(e) + \tilde{N}\right) \leq -\delta e^2 + \left(\tilde{M}_d + \tilde{N}\right)e - \mu|e| \tag{16}$$

parameters μ should be selected in a way that $\mu > |\tilde{M}_d + \tilde{N}|$, as a result, we have

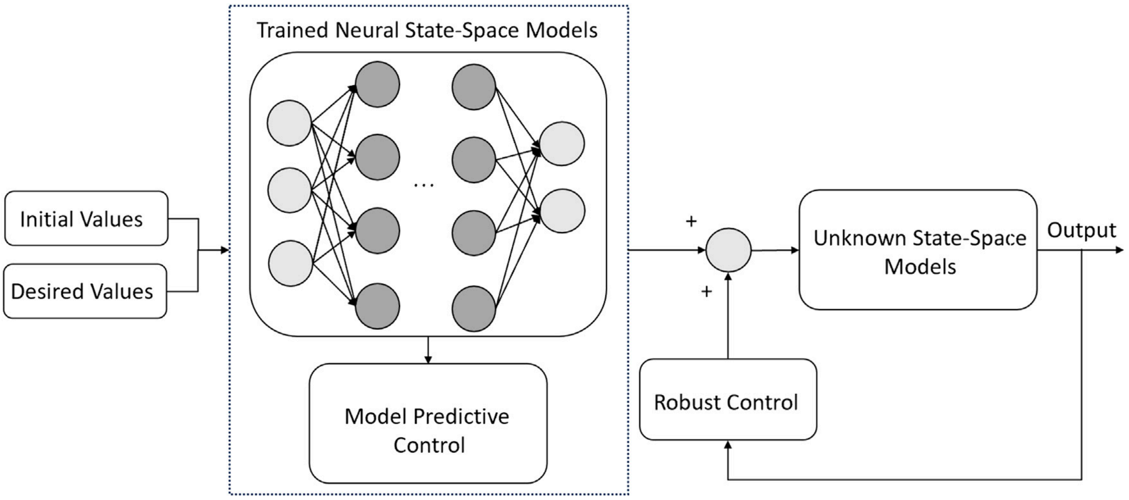

FIGURE 1
The proposed neural state-space-based model MPC.

$$\dot{V} = e\dot{e} = e\left(\tilde{M}_d - \vartheta e - \mu\, sign\ (e) + \tilde{N}\right) \leq -\delta e^2 \tag{17}$$

By utilizing Eq. 17, we can validate that the convergence of the states of the closed-loop system towards the equilibrium point is assured, in accordance with the Lyapunov stability theorem. This result completes the proof.

Remark 1. The parameters δ and μ, defined by the user, must adhere to predefined constraints in order to ensure the validity of the results and the stability of the model. Specifically, the parameter δ is required to maintain a positive value. Furthermore, the parameter μ, besides being positive, should satisfy an additional condition, namely, that $\mu > |\tilde{M}_d + \tilde{N}|$. This criterion is of paramount importance for maintaining Lyapunov stability, as elucidated in Eq. 17.

Remark 2. The deployment of the sign function in the controller design can give rise to non-smooth control inputs, leading to undesirable chattering. A prevalent and efficacious strategy to counteract such instances involves employing a continuous approximation instead of the sign function. In this context, the arctangent (atan) function emerges as a fitting option and can be used to result in smooth control inputs.

The block diagram presented in Figure 1 illustrates the proposed control technique. It incorporates robust control in Eq. 11, allowing for the inclusion of disturbances in the model. This design choice ensures that the controller is well-suited and resilient for controlling nanobeams.

3 NSG nano-beams

The Euler-Bernoulli displacement components of a hinged-hinged nanobeam are expressed as follows:

$$\begin{aligned} u_x &= d_a(x,t) - z\frac{\partial d_t(x,t)}{\partial x} \\ u_y &= 0 \\ u_z &= d_t(x,t) \end{aligned} \tag{18}$$

The nanobeam's x, y, and z displacements are symbolized by u_x, u_y, and u_z correspondingly. The axial and transverse deflections of any point on the neutral axis are represented by d_a and d_t, respectively. The independent spatial and time variables are denoted by x and t, respectively.

Here we use the NSG theorem to present the governing equation of nanobeam. Strain gradients refer to the variation of strain within a material [35, 36]. In traditional continuum mechanics, the strain is assumed to be constant throughout the material. However, at small scales, such as in microstructures or near material boundaries, the strain may vary significantly. Strain gradients take into account this variation and introduce additional terms to the constitutive equations to capture the effect. Nonlocal effects refer to the fact that the behavior of a material at a particular point depends not only on its immediate surroundings but also on a larger region. In other words, the material's response is influenced by the overall deformation state of the neighboring points. Nonlocal effects are particularly important in materials with characteristic length scales, such as granular materials or materials with microstructural features. When both strain gradients and nonlocal effects are considered together, the resulting theory is referred to as NSG theory. It provides a more accurate description of the mechanical behavior of materials at small scales and can be used to analyze phenomena such as size-dependent plasticity, fracture, and creep in microstructures [37, 38]. The formulation for the strain energy (U) of an isotropic linear elastic material, as provided by the NSG theory, can be expressed in the following manner:

$$U = \frac{1}{2}\int_V \left(\sigma_{11}\varepsilon_{11} + \sigma_{11}{}^{(1)}\nabla\varepsilon_{11}\right)dV \tag{19}$$

where σ_{11} denotes the classical stress, ${\sigma_{11}}^{(1)}$ and ε_{11} represent the stress and the normal strain. The differential operator is denoted by ∇ which is equivalent to the partial derivative with respect to x. Additionally, σ_{11} and ${\sigma_{11}}^{(1)}$ are defined as follows.

$$\sigma_{11} = \int_0^L E\xi_0 \varepsilon_{11}^{'}(x')dx' \qquad (20-\mathrm{a})$$

$${\sigma_{11}}^{(1)} = {l_s}^2 \int_0^L E\xi_1 \varepsilon_{11}^{'}(x')dx' \qquad (20-\mathrm{b})$$

$$t_{11} = \sigma_{11} - \nabla {\sigma_{11}}^{(1)} \qquad (20-\mathrm{c})$$

where the length of the nanobeam is symbolized by L and ξ_0 represent the principal attenuation kernel function, which combines the constitutive equations describing the nonlocal effects. l_s is the strain gradient length scale parameter, Alongside, we have ξ_1, an additional kernel function that relates specifically to the nonlocal effect. Young's modulus, denoted as E, is also a key factor in the equation. The underlying assumption of nonlocal elasticity theory is that the stress at a point in a body does not only depend on the strain at that point but also depends on the strain at other points. The use of the integral in Eq. 20 represents this nonlocal behavior. The integral sums up the contributions of the strain at all points (from 0 to L, the length of the nanobeam) in the body to the stress at a particular point.

The constitutive behavior of NSG can be described by the following general equation:

$$\frac{1}{E}\left[1-{\theta_0}^2\nabla^2\right]\left[1-{\theta_1}^2\nabla^2\right]t_{11} = \left[1-{\theta_1}^2\nabla^2\right]\varepsilon_{11} - {l_s}^2\left[1-{\theta_0}^2\nabla^2\right]\nabla^2\varepsilon_{11} \qquad (21)$$

where ∇^2 is Laplacian operator and the nonlocal parameters θ_0 and θ_1 are incorporated to acknowledge the significance of the nonlocal elastic stress field. These nonlocal parameters, modulate the influence of the stress field at distant points. They are typically chosen based on experimental observations or are calibrated using numerical methods to match the predictions of the nonlocal theory with the observed material behavior. Let $\theta_0 = \theta_1 = \theta$ (this is a valid assumption, for more detailed information and clarification, see [39], Eq. 21 can be reformulated as

$$\frac{1}{E}\left[1-\theta^2\nabla^2\right]t_{11} = (1 - l_S^2\nabla^2)\varepsilon_{11} \qquad (22)$$

Assuming $\boldsymbol{l_s = 0}$, it leads to the formulation of local elasticity theory as follows:

$$\frac{1}{E}\left[1-\theta^2\nabla^2\right]t_{11} = \varepsilon_{11} \qquad (23)$$

By setting $\theta = 0$ to zero, the strain gradient theory can be represented as follows:

$$t_{11} = E\left(1 - {l_s}^2\nabla^2\right)\varepsilon_{11} \qquad (24)$$

Remark 3: It should be underscored that the assumptions of $l_s = 0$ and $\theta = 0$ are not adopted in the present study. Their inclusion here is merely illustrative, employed with the explicit intent of elucidating the interconnections between nonlocal strain gradients, strain gradients, and local elasticity. When analyzing a straight Euler-Bernoulli nanobeam under the assumptions of large deflection and small slope, the nonlinear strain relationship derived from Von Karman's theory can be represented in the following manner:

$$\varepsilon_{11} = \frac{\partial d_a(x,t)}{\partial x} + \frac{1}{2}\left(\frac{\partial d_t(x,t)}{\partial x}\right)^2 - z\frac{\partial^2 d_t(x,t)}{\partial x^2} \qquad (25)$$

where ε_{11} represents the longitudinal strain, consequently one can achieve:

$$\delta\int_0^t U dt = \int_0^t\int_0^L\left[N_c\left(\frac{\partial\delta d_a}{\partial x} + \frac{\partial d_t}{\partial x}\frac{\partial\delta d_t}{\partial x}\right) - M_c\frac{\partial^2\delta d_t}{\partial x^2}\right]dxdt$$
$$+\int_0^t\left[N_{nc}\left(\frac{\partial\delta d_a}{\partial x} + \frac{\partial d_t}{\partial x}\frac{\partial\delta d_t}{\partial x}\right) - M_{nc}\frac{\partial^2\delta d_t}{\partial x^2}\right]\Bigg|_0^L dt \qquad (26)$$

in which N_{nc}, M_{nc}, N_c, and M_c are given by

$$N_{nc} = \int_A {\sigma_{11}}^{(1)}dA, M_{nc} = \int_A z{\sigma_{11}}^{(1)}dA, N_c = \int_A t_{11}dA, M_c$$
$$= \int_A zt_{11}dA, \qquad (27)$$

where M_c represents the classical normal moment, N_c represents the classical force, M_{nc} represents the non-classical normal moment, and N_{nc} represents the non-classical force. Furthermore, the work of the applied external forces is given by

$$\delta\int_0^t W dt = \int_0^t\int_L (f_a\delta d + f_t\delta d_t)dxdt \qquad (28)$$

In the provided equation, f_a and f_t denote the distributed axial and transverse loads, respectively. Additionally, the expression for the first variation of kinetic energy (K_e) is given as follows:

$$\delta\int_0^t K_e dt = \int_0^t\int_0^L I_A\left(\frac{\partial d_a}{\partial t}\frac{\partial\delta d_a}{\partial t} + \frac{\partial d_t}{\partial t}\frac{\partial\delta d_t}{\partial t}\right)dxdt \qquad (29)$$

where $I_A = \frac{1}{12}bh^3$. Note that this formulation is based on several common assumptions, which include: 1) The principle of virtual work holds true, which means that the virtual work done by the applied forces is equal to the change in kinetic and potential energy 2) The quantities d_a and d_t are assumed to be differentiable functions of time. 3) The moment of inertia I_A is constant across the length of the system. This would imply a uniform mass distribution along the system.

The given expression for Hamilton's principle, which is employed to derive the equations of motion, is as follows:

$$\delta\int_0^t [K_e - (U - W)]dt = 0 \qquad (30)$$

By applying Hamilton's principle (30) and considering the rotational inertia of the beam to be negligible, we obtain the governing equation for the nanobeam according to the NSG theory. The aforementioned equation can be represented in the following manner:

$$D_{11}l_m^2\frac{\partial^6 d_t}{\partial x^6}-D_{11}\frac{\partial^4 d_t}{\partial x^4}+\left[\frac{A_{11}}{2L}\int_0^L\left(\frac{\partial d_t}{\partial x}\right)^2dx-\frac{A_{11}l_m^2}{L}\int_0^L\left(\frac{\partial d_t}{\partial x}\frac{\partial^3 d_t}{\partial x^3}\right.\right.$$
$$\left.\left.+\left(\frac{\partial^2 d_t}{\partial x^2}\right)^2\right)dx\right]\times\left[\frac{\partial^2 d_t}{\partial x^2}-\theta^2\frac{\partial^4 d_t}{\partial x^4}\right]+I_A\frac{\partial^2}{\partial t^2}\left[(\theta)^2\frac{\partial^2 d_t}{\partial x^2}-d_t\right]$$
$$=\theta^2\frac{\partial^2 q_t}{\partial x^2}-f_t \tag{31}$$

To render Eq. 31 in a dimensionless manner, the subsequent variables are introduced:

$$\bar{x}=\frac{x}{L},\quad \bar{d}_t=\frac{d_t}{r},\quad \bar{z}=\frac{z}{h},\quad \bar{t}=t\sqrt{\frac{EI}{\rho AL^4}},\quad \nu=\frac{\theta}{L},\quad \kappa=\frac{l_m}{L} \tag{32}$$

By substituting t $r=\sqrt{\frac{I}{A}}$, the resulting equation becomes dimensionless, and the governing equation can be expressed as follows.

$$\beta^2\bar{D}_{11}\frac{\partial^6\bar{d}_t}{\partial\bar{x}^6}-\bar{D}_{11}\frac{\partial^4\bar{d}_t}{\partial\bar{x}^4}+$$
$$\left[\frac{\bar{A}_{11}}{2}\int_0^1\left(\frac{\partial\bar{d}_t}{\partial\bar{x}}\right)^2d\bar{x}-\kappa^2\bar{A}_{11}\int_0^1\left(\frac{\partial\bar{d}}{\partial\bar{x}}\frac{\partial^3\bar{d}_t}{\partial\bar{x}^3}+\left(\frac{\partial^2\bar{d}_t}{\partial\bar{x}^2}\right)^2\right)d\bar{x}\right]\frac{\partial^2\bar{d}_t}{\partial\bar{x}^2}$$
$$-\left[\frac{\alpha^2\bar{A}_{11}}{2}\int_0^1\left(\frac{\partial\bar{d}_t}{\partial\bar{x}}\right)^2d\bar{x}-\nu^2\kappa^2\bar{A}_{11}\int_0^1\left(\frac{\partial\bar{d}_t}{\partial\bar{x}}\frac{\partial^3\bar{d}_t}{\partial\bar{x}^3}+\left(\frac{\partial^2\bar{d}_t}{\partial\bar{x}^2}\right)^2\right)d\bar{x}\right]\frac{\partial^4\bar{d}_t}{\partial\bar{x}^4}$$
$$+\nu^2\bar{I}_A\frac{\partial^4\bar{d}_t}{\partial\bar{x}^2\partial\bar{t}^2}-\bar{I}_A\frac{\partial^2\bar{d}_t}{\partial\bar{t}^2}=\nu^2\frac{\partial^2\bar{q}}{\partial\bar{x}^2}-\tau \tag{33}$$

Given that we are dealing with a homogeneous nanobeam, it can be demonstrated that $\boldsymbol{\bar{A}_{11}=1}$, $\boldsymbol{\bar{D}_{11}=1}$, and $\boldsymbol{\bar{I}_A=1}$ (see [40] for detailed information). Now, we can employ the Galerkin approach to convert the partial differential equation into a nonlinear ordinary differential equation. This procedure entails separating the temporal and spatial components of $\boldsymbol{\bar{d}_t(\bar{x},\bar{t})}$ as outlined in [41].

$$\begin{gathered}\bar{d}_t(\bar{x},\bar{t})=\varrho(\bar{x})\psi(\bar{t})\\ \varrho(\bar{x})=sin(\pi\bar{x})\end{gathered} \tag{34}$$

Within the provided context, $\psi(\bar{t})$ signifies the temporal component that is yet to be determined, whereas $\varrho(\bar{x})$ represents the spatial component of the transverse deflection. Note that the mode shape $\varrho(\bar{x})=sin(\pi\bar{x})$, is a common choice for a beam that is hinged, or simply supported, at both ends. The primary rationale behind this selection is based on the boundary conditions of a simply supported beam and the mathematical properties of the sine function. Additionally, the concentrated force $\bar{q}(\bar{x},\bar{t})$ is defined as follows.

$$\bar{q}(\bar{x},\bar{t})=f_t(\bar{t})\delta\left(\bar{x}-\frac{1}{2}\right), \tag{35}$$

By combining Eqs 32, 33 with Eq. 38, and subsequently multiplying both sides of Eq. 38 by the spatial component $\varrho(\bar{x})$, and integrating over the length of the beam, an intriguing transformation is obtained. This transformation leads us to an ordinary differential equation as follows

$$\ddot{\psi}(\bar{t})+\beta_1\psi(\bar{t})+\beta_2\psi^3(\bar{t})=b\tau(\bar{t}) \tag{36}$$

while the coefficients β_1 and β_2 are determined by the following expressions

$$\beta_1=\frac{\beta^2\bar{D}_{11}\int_0^1\varrho^{(6)}\varrho d\bar{x}-\bar{D}_{11}\int_0^1\varrho^{(4)}\varrho d\bar{x}}{\alpha^2\int_0^1\varrho''\varrho d\bar{x}-\int_0^1(\varrho)^2d\bar{x}}$$
$$\beta_2=-\frac{\begin{gathered}\frac{\bar{A}_{11}}{2}\int_0^1(\varrho')^2d\bar{x}.\int_0^1\varrho''\varrho d\bar{x}-\kappa^2\bar{A}_{11}\int_0^1\varrho'''\varrho' d\bar{x}.\int_0^1\varrho''\varrho d\bar{x}\\ -\kappa^2\bar{A}_{11}\int_0^1(\varrho'')^2d\bar{x}.\int_0^1\varrho''\varrho d\bar{x}\end{gathered}}{\nu^2\int_0^1\varrho''\varrho d\bar{x}-\int_0^1(\varrho)^2d\bar{x}}$$
$$-\frac{\begin{gathered}\frac{\nu^2\bar{A}_{11}}{2}\int_0^1(\varrho')^2d\bar{x}.\int_0^1\varrho^{(4)}\varrho d\bar{x}-\nu^2\kappa^2\bar{A}_{11}\int_0^1\varrho'''\varrho' d\bar{x}.\\ \int_0^1\varrho^{(4)}\varrho d\bar{x}-\nu^2\kappa^2\bar{A}_{11}\int_0^1(\varrho'')^2d\bar{x}.\int_0^1\varrho^{(4)}\varrho d\bar{x}\end{gathered}}{\nu^2\int_0^1\varrho''\varrho d\bar{x}-\int_0^1(\varrho)^2d\bar{x}}$$
$$b=-(\nu^2\pi^2+1) \tag{37}$$

In the given context, $\varrho^{(i)}$ represent the i^{th} derivative of ϱ with respect to time. On the other hand, ϱ' refers to the first derivative of ϱ with respect to $\bar{x}$. Taking into account that $\psi(\bar{t})=x_1, \dot{\psi}(\bar{t})=\dot{x}_1=x_2$, we derive the following non-dimensional state-space equation of motion:

$$\begin{cases}\dot{x}_1=x_2\\ \dot{x}_2=-\beta_1x_1-\beta_2(x_1)^3-b\tau(\bar{t})\end{cases} \tag{38}$$

where x_1 denotes the non-dimensional deflection of the beam and x_2 represents its derivative.

4 Numerical results

Herein, we present the numerical simulation showcasing the stabilization of a nanobeam through the implementation of the proposed control scheme. The parameters used for the simulation of the nanobeam are $\nu=\kappa=0.1$. Considering the formulation in Appendix, we obtain the corresponding values of $\beta_1=97.4$, $\beta_2=-19.97$, and $b=1.09$, based on the given ν and κ values. The criteria for the design parameters of the controller are detailed in Remark 1. For the numerical simulations here, we have chosen the parameters such that μ equals 10 and δ equals 1.

To generate training data, we employed random inputs to stimulate the system, measuring and recording both the deflection and its derivative. Subsequently, the collected training data was used to train the neural network offline. For training, we employed 200-time histories of deflection and its corresponding derivative. An example of this 200-time history samples used in the training phase can be seen in Figure 2. In this study, random inputs have been utilized to facilitate the learning of the neural state space representation of the model. The core rationale behind this selection pertains to the enhancement of the model's generalization capabilities. By deploying random inputs, we can expose the model to a more extensive and varied spectrum of data, thereby augmenting the robustness of the learning process. This strategy ensures that the model experiences a wide variety of situations during the training phase, equipping it with the ability to better

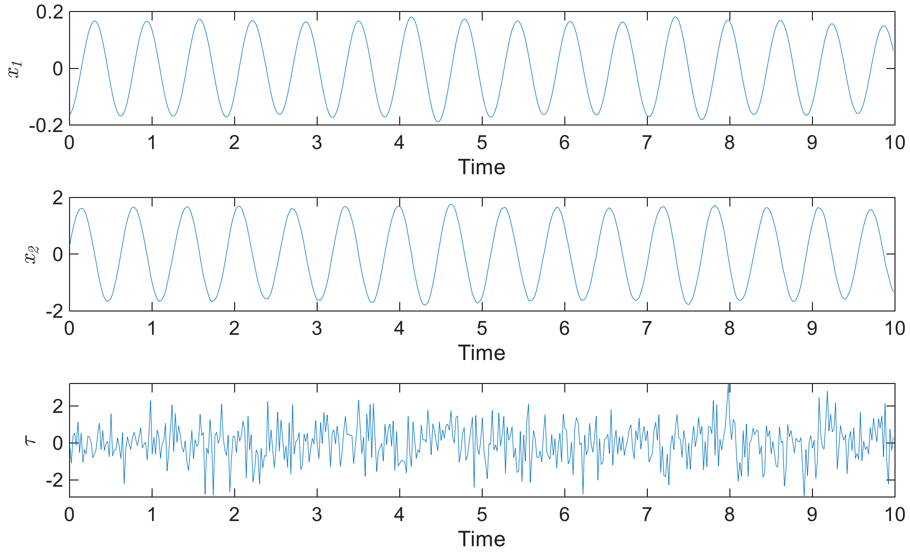

FIGURE 2
A training time history used for training of state space neural network.

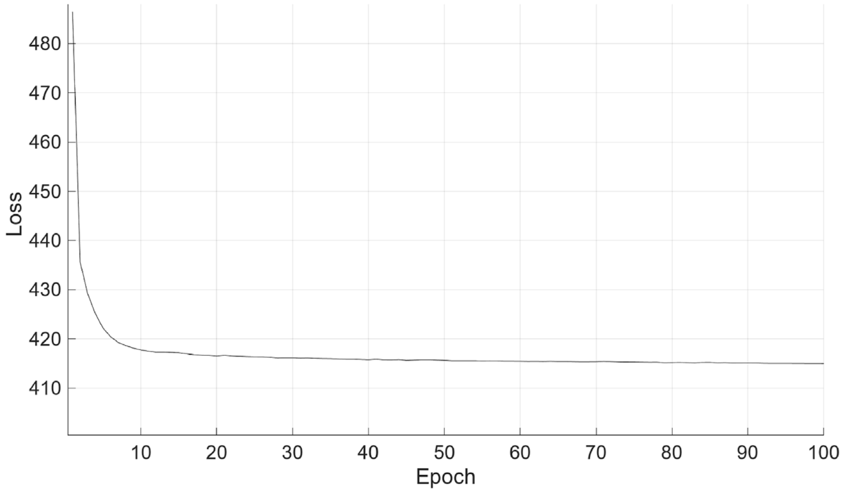

FIGURE 3
The loss function of neural state space model during training.

adapt to unforeseen scenarios when it is subsequently implemented in a real-world context. Utilizing a specific or limited type of input data for training could lead to the development of a bias in the model towards this data. This bias could adversely affect the model's performance when presented with diverse data or scenarios. To circumvent this potential bias and guarantee the broad generalizability of our model, we have chosen to employ random inputs.

The software used for the simulations is MATLAB 2022a. In the initial phase of model learning, the computational cost is primarily dependent on the number of training samples. However, considering the low-dimensional nature of the system, these costs are relatively moderate compared to typical regression and classification problems tackled by feed-forward neural networks. Once the state-space model has been learned, the computational expenditure for implementing the controller aligns with that of a typical MPC application. Hence, while the pre-training phase causes additional computational costs, the operational costs of the controller do not significantly deviate from conventional MPC approaches.

Figure 3 illustrates the loss function of the neural network training for the neural state-space model representation of the system. This loss function provides insight into the optimization process of the neural network. By monitoring the loss function, we can assess the progress and convergence of the training process,

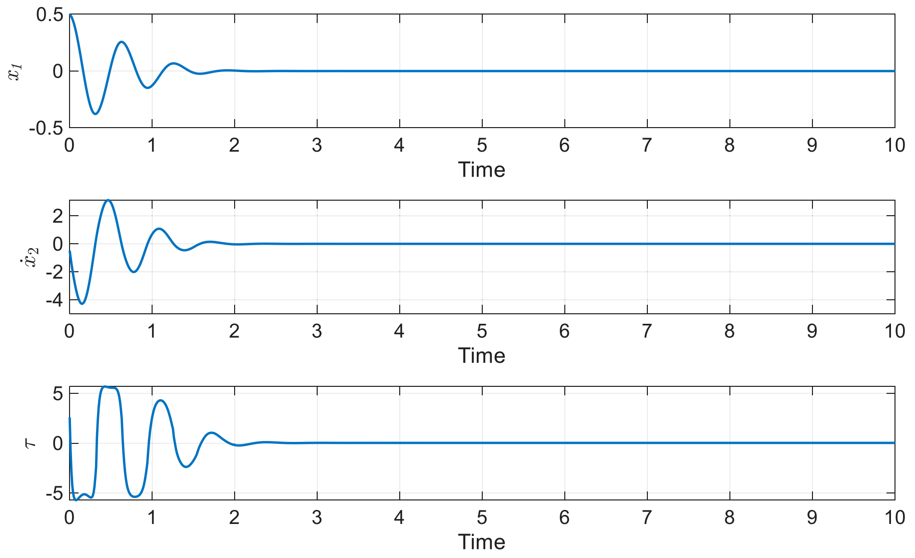

FIGURE 4
Time history of states and control input of the system while $[x_1(0), x_2(0)] = [0.5, 0]$.

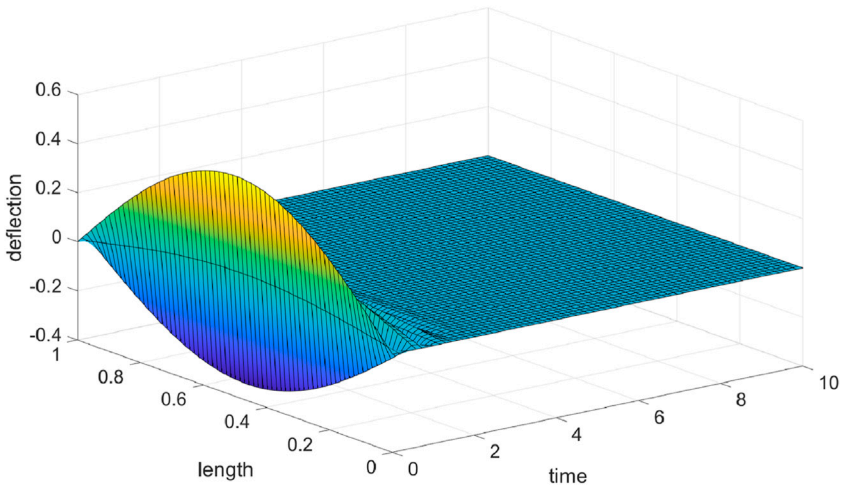

FIGURE 5
The deflection of NSG nano-beam while $[x_1(0), x_2(0)] = [0.5, 0]$.

ensuring that the neural network captures the essential dynamics of the system accurately.

In what follows, two distinct situations have been taken into account, and the proposed controller has been implemented for each. The reasoning behind having chosen two different initial conditions — $[x_1(0), x_2(0)] = [[0.5, 0]$ and $[x_1(0), x_2(0)] = [-0.5, 0.1]$ — has been to accommodate a wide range of situations. By selecting these initial conditions, we have been able to comprehensively explore both positive and negative initial positions. Furthermore, these conditions have also allowed us to investigate the impact of zero and non-zero speeds at the starting point.

Figure 4 display the outcomes of the stabilization process for the nanobeam, employing the suggested control technique with the initial states of the system set as $[x_1(0), x_2(0)] = [0.5, 0]$. These figures vividly exhibit the remarkable capability of the proposed robust adaptive controller to effectively counteract and completely reject disturbances. Additionally, Figure 5 showcases the temporal evolution of the nanobeam's deflection when utilizing the proposed control scheme. The figures demonstrate that the controller, which integrates a neural state-space model equipped with a robust term estimator, adeptly addresses control problems of the unknown system. This particular ability to handle uncertainties plays a crucial role in controlling nano systems.

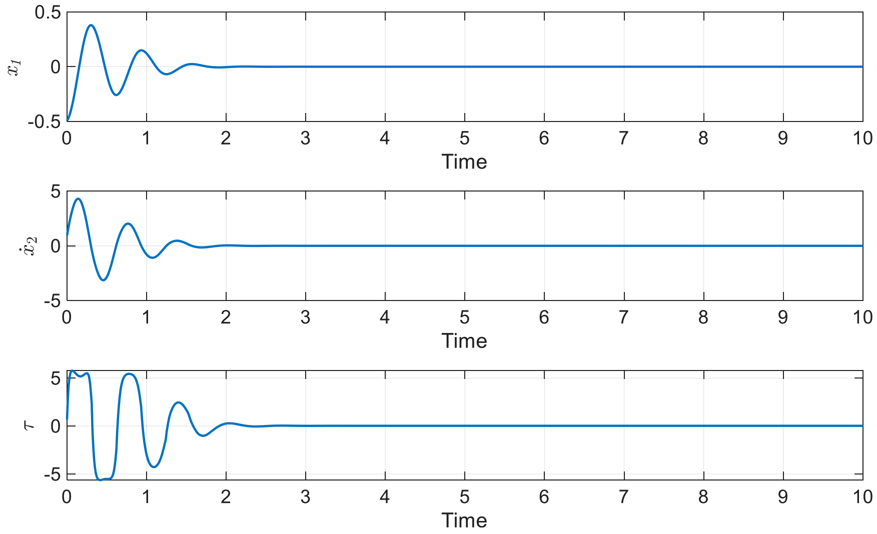

FIGURE 6
Time history of states and control input of the system while $[x_1(0), x_2(0)] = [-0.5, 0.1]$.

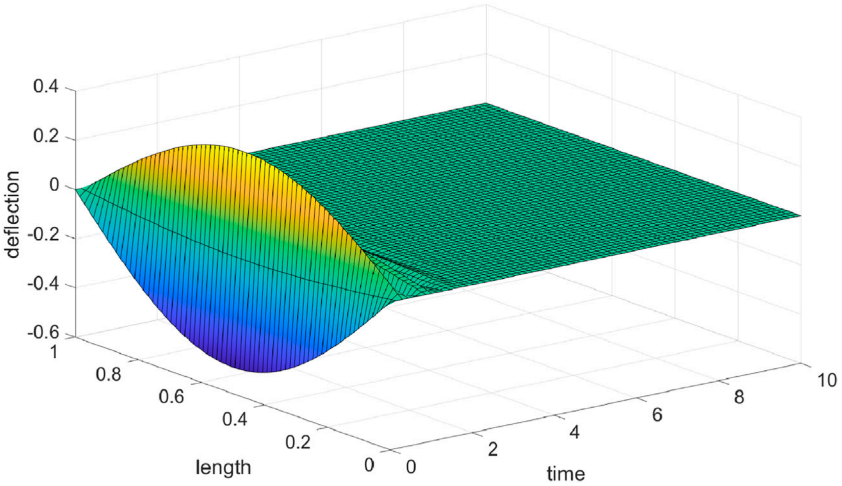

FIGURE 7
The deflection of NSG nano-beam while $[x_1(0), x_2(0)] = [-0.5, 0.1]$.

To evaluate the effectiveness of our proposed controller, we performed an additional test by varying the initial values of the system's states. Specifically, we selected $[x_1, x_2] = [-0.5, 0.1]$. Figure 6 illustrates the controller's performance in achieving system stabilization under these modified conditions. The figure demonstrates that the proposed controller operates within an acceptable range, ensuring feasible control signal values. Notably, despite the system's dynamics being completely unknown, the proposed controller exhibits outstanding performance, achieving state stabilization in less than 2 time units. Figure 7 illustrates the deflection of the system, clearly indicating that the suggested controller facilitates rapid vibration suppression in the nano beam.

To facilitate a more in-depth assessment of the proposed controller's efficacy, Table 1 outlines the settling time, as well as the maximum and norm of control signals for both numerical instances illustrated in this section.

In summary, the simulations and numerical results presented in Table 1 clearly demonstrate that the proposed control scheme excels in vibration suppression in the nanobeam with completely unknown dynamics, ensuring the stability and robustness of the system. Compared to conventional MPC and robust controls [42, 43], our method provides significant advantages in handling nano-beam vibrations, especially when full knowledge of the system's dynamics is not readily available. By combining DNNs with MPC, we develop an intelligent control framework that effectively mitigates MPC's limitations and reduces vibrations in NSG nano-

TABLE 1 Settling time and control signal properties.

Example	Settling time	Maximum of the control signal	Norm of the control signal
Example 1, $[x_1(0), x_2(0)] = [0.5, 0]$	1.350	5.07	75.06
Example 2, $[x_1(0), x_2(0)] = [-0.5, 0.1]$	1.551	5.25	76.57

beams. We further enhance the control strategy by introducing an extra control term for robustness and improved system convergence. However, it is important to note that our method requires pre-processing and data collection for training the model before real-world deployment, unlike traditional approaches.

5 Conclusion

The present study introduced a neural state-space-based MPC framework with guaranteed convergence. The framework entailed a systematic identification of system dynamics and the learning of the MPC policy through function approximations. Specifically, the system dynamics were captured utilizing DNN, and the MPC policy was established based on the identified model. Additionally, the robustness and convergence of the closed-loop system were guaranteed by incorporating an additional control term. Subsequently, the governing equation of motion for NSG nano-beams was presented and derived. Then, the proposed control technique was validated by applying it to NSG nano-beams. The obtained results exhibited exceptional performance, confirming the efficacy of the proposed method. In this study, the robust control term has been consistently applied in conjunction with the optimal control term at all stages. Nevertheless, there are ways to further streamline the system without compromising accuracy. Incorporating event-triggered approaches could be beneficial in this regard. These strategies would enable the controller to be deployed only as required and then deactivated afterward, creating a more optimal control. Therefore, a potential area for future research in this domain would be to enhance the proposed controller's efficiency through the integration of event-trigger mechanisms.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

HA: Writing–original draft.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Zhu Q, Liu Y, Wen G. Adaptive neural network control for time-varying state constrained nonlinear stochastic systems with input saturation. *Inf Sci* (2020) 527: 191–209. doi:10.1016/j.ins.2020.03.055

2. Ge SS, Hang CC, Lee TH, Zhang T. *Stable adaptive neural network control*. Springer Science and Business Media (2013).

3. Wu Q, Wang Y, Wang H. An extended linearized neural state space based modeling and control. *IFAC Proc Volumes* (2014) 35:975–80. doi:10.3182/20020721-6-ES-1901.00977

4. Sun W, Wang X, Cheng Y. Reinforcement learning method for continuous state space based on dynamic neural network. 2008 7th World Congress on Intelligent Control and Automation. IEEE (2008). p. 750–4.

5. Afram A, Janabi-Sharifi F. Theory and applications of HVAC control systems–A review of model predictive control (MPC). *Building Environ* (2014) 72:343–55. doi:10.1016/j.buildenv.2013.11.016

6. Camacho E, Alba C. Model predictive control [sl]: Springer science and business media. *Citado* (2013) 2:79. doi:10.1007/978-0-85729-398-5

7. Guo N, Lenzo B, Zhang X, Zou Y, Zhai R, Zhang T. A real-time nonlinear model predictive controller for yaw motion optimization of distributed drive electric vehicles. *IEEE Trans Vehicular Technol* (2020) 69:4935–46. doi:10.1109/tvt.2020.2980169

8. Shen C, Shi Y. Distributed implementation of nonlinear model predictive control for AUV trajectory tracking. *Automatica* (2020) 115:108863. doi:10.1016/j.automatica.2020.108863

9. Woo D-O, Junghans L. Framework for model predictive control (MPC)-based surface condensation prevention for thermo-active building systems (TABS). *Energy and Buildings* (2020) 215:109898. doi:10.1016/j.enbuild.2020.109898

10. Karamanakos P, Liegmann E, Geyer T, Kennel R. Model predictive control of power electronic systems: Methods, results, and challenges. *IEEE Open J Industry Appl* (2020) 1:95–114. doi:10.1109/ojia.2020.3020184

11. Arroyo J, Manna C, Spiessens F, Helsen L. Reinforced model predictive control (RL-MPC) for building energy management. *Appl Energ* (2022) 309:118346. doi:10.1016/j.apenergy.2021.118346

12. Schwenzer M, Ay M, Bergs T, Abel D. Review on model predictive control: An engineering perspective. *Int J Adv Manufacturing Technol* (2021) 117:1327–49. doi:10.1007/s00170-021-07682-3

13. Ohashi H, Yamazaki H, Yumoto H, Koyama T, Senba Y, Takeuchi T, et al. Stable delivery of nano-beams for advanced nano-scale analyses. *J Phys Conf Ser* (2013) 425: 052018. doi:10.1088/1742-6596/425/5/052018

14. Arefi M, Pourjamshidian M, Arani AG. Free vibration analysis of a piezoelectric curved sandwich nano-beam with FG-CNTRCs face-sheets based on various high-order

shear deformation and nonlocal elasticity theories. *The Eur Phys J Plus* (2018) 133:193. doi:10.1140/epjp/i2018-12015-1

15. Sheykhi M, Eskandari A, Ghafari D, Arpanahi RA, Mohammadi B, Hashemi SH. Investigation of fluid viscosity and density on vibration of nano beam submerged in fluid considering nonlocal elasticity theory. *Alexandria Eng J* (2023) 65:607–14. doi:10.1016/j.aej.2022.10.016

16. Bhatt M, Shende P. Surface patterning techniques for proteins on nano-and micro-systems: A modulated aspect in hierarchical structures. *J Mater Chem B* (2022) 10:1176–95. doi:10.1039/d1tb02455h

17. Lyshevski SE. *Nano-and micro-electromechanical systems: Fundamentals of nano-and microengineering*. CRC Press (2018).

18. Sabarianand D, Karthikeyan P, Muthuramalingam T. A review on control strategies for compensation of hysteresis and creep on piezoelectric actuators based micro systems. *Mech Syst Signal Process* (2020) 140:106634. doi:10.1016/j.ymssp.2020.106634

19. Roudbari MA, Jorshari TD, Lü C, Ansari R, Kouzani AZ, Amabili M. A review of size-dependent continuum mechanics models for micro-and nano-structures. *Thin-Walled Structures* (2022) 170:108562. doi:10.1016/j.tws.2021.108562

20. Miandoab EM, Yousefi-Koma A, Pishkenari HN. Nonlocal and strain gradient based model for electrostatically actuated silicon nano-beams. *Microsystem Tech* (2015) 21:457–64. doi:10.1007/s00542-014-2110-2

21. Cao Q, Wei DQ. Dynamic surface sliding mode control of chaos in the fourth-order power system. *Chaos, Solitons and Fractals* (2023) 170:113420. doi:10.1016/j.chaos.2023.113420

22. Jahanshahi H, Rajagopal K, Akgul A, Sari NN, Namazi H, Jafari S. Complete analysis and engineering applications of a megastable nonlinear oscillator. *Int J Non-Linear Mech* (2018) 107:126–36. doi:10.1016/j.ijnonlinmec.2018.08.020

23. Vagia M. How to extend the travel range of a nanobeam with a robust adaptive control scheme: A dynamic surface design approach. *ISA Trans* (2013) 52:78–87. doi:10.1016/j.isatra.2012.09.001

24. Jahanshahi H, Zambrano-Serrano E, Bekiros S, Wei Z, Volos C, Castillo O, et al. On the dynamical investigation and synchronization of variable-order fractional neural networks: The hopfield-like neural network model. *Eur Phys J Spec Top* (2022) 231:1757–69. doi:10.1140/epjs/s11734-022-00450-8

25. Jahanshahi H, Yao Q, Khan MI, Moroz I. Unified neural output-constrained control for space manipulator using tan-type barrier Lyapunov function. *Adv Space Res* (2023) 71:3712–22. doi:10.1016/j.asr.2022.11.015

26. Keighobadi J, Faraji J, Rafatnia S. Chaos control of atomic force microscope system using nonlinear model predictive control. *J Mech* (2017) 33:405–15. doi:10.1017/jmech.2016.89

27. Long X, He Z, Wang Z. Online optimal control of robotic systems with single critic NN-based reinforcement learning. *Complexity* (2021) 2021:1–7. doi:10.1155/2021/8839391

28. Chen Y, Tong Z, Zheng Y, Samuelson H, Norford L. Transfer learning with deep neural networks for model predictive control of HVAC and natural ventilation in smart buildings. *J Clean Prod* (2020) 254:119866. doi:10.1016/j.jclepro.2019.119866

29. Carlet PG, Favato A, Bolognani S, Dörfler F. Data-driven continuous-set predictive current control for synchronous motor drives. *IEEE Trans Power Electron* (2022) 37:6637–46. doi:10.1109/tpel.2022.3142244

30. Li Y, Tong Z. Model predictive control strategy using encoder-decoder recurrent neural networks for smart control of thermal environment. *J Building Eng* (2021) 42:103017. doi:10.1016/j.jobe.2021.103017

31. Bonassi F, Farina M, Xie J, Scattolini R. On recurrent neural networks for learning-based control: Recent results and ideas for future developments. *J Process Control* (2022) 114:92–104. doi:10.1016/j.jprocont.2022.04.011

32. Khan A, Sohail A, Zahoora U, Qureshi AS. A survey of the recent architectures of deep convolutional neural networks. *Artif intelligence Rev* (2020) 53:5455–516. doi:10.1007/s10462-020-09825-6

33. Stanley KO, Clune J, Lehman J, Miikkulainen R. Designing neural networks through neuroevolution. *Nat Machine Intelligence* (2019) 1:24–35. doi:10.1038/s42256-018-0006-z

34. Jahanshahi H, Shahriari-Kahkeshi M, Alcaraz R, Wang X, Singh VP, Pham V-T. Entropy analysis and neural network-based adaptive control of a non-equilibrium four-dimensional chaotic system with hidden attractors. *Entropy* (2019) 21:156. doi:10.3390/e21020156

35. Hutchinson J, Fleck N. Strain gradient plasticity. *Adv Appl Mech* (1997) 33:295–361.

36. Fleck N, Muller G, Ashby MF, Hutchinson JW. Strain gradient plasticity: Theory and experiment. *Acta Metallurgica et materialia* (1994) 42:475–87. doi:10.1016/0956-7151(94)90502-9

37. Li X, Li L, Hu Y, Ding Z, Deng W. Bending, buckling and vibration of axially functionally graded beams based on nonlocal strain gradient theory. *Compos Structures* (2017) 165:250–65. doi:10.1016/j.compstruct.2017.01.032

38. Ebrahimi F, Barati MR, Dabbagh A. A nonlocal strain gradient theory for wave propagation analysis in temperature-dependent inhomogeneous nanoplates. *Int J Eng Sci* (2016) 107:169–82. doi:10.1016/j.ijengsci.2016.07.008

39. Lim CW, Zhang G, Reddy JN. A higher-order nonlocal elasticity and strain gradient theory and its applications in wave propagation. *J Mech Phys Sol* (2015) 78:298–313. doi:10.1016/j.jmps.2015.02.001

40. Alsubaie H, Yousefpour A, Alotaibi A, Alotaibi ND, Jahanshahi H. Fault-tolerant terminal sliding mode control with disturbance observer for vibration suppression in non-local strain gradient nano-beams. *Mathematics* (2023) 11:789. doi:10.3390/math11030789

41. Rhoads JF, Shaw SW, Turner KL. The nonlinear response of resonant microbeam systems with purely-parametric electrostatic actuation. *J Micromechanics Microengineering* (2006) 16:890–9. doi:10.1088/0960-1317/16/5/003

42. Do L, Korda M, Hurák Z. Controlled synchronization of coupled pendulums by koopman model predictive control. *Control Eng Pract* (2023) 139:105629. doi:10.1016/j.conengprac.2023.105629

43. Wang Y, Liu X. Model predictive position control of permanent magnet synchronous motor servo system with sliding mode observer. *Asian J Control* (2023) 25:443–61. doi:10.1002/asjc.2817