Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

Discovering Efficient Periodic Behaviours in Mechanical Systems via Neural Approximators

Yannik P. Wotte[†*1] | Sven Dummer[†*2] | Nicolò Botteghi[†2] | Christoph Brune[2] | Stefano Stramigioli[1] | Federico Califano[1]

[1]Robotics Mechatronics (RaM) group, University of Twente, Enschede, The Netherlands

[2]Mathematics of Imaging and AI (MIA) group, University of Twente, Enschede, The Netherlands

Correspondence
*Yannik Wotte, Email: y.p.wotte@utwente.nl,
Sven Dummer, Email: s.c.dummer@utwente.nl
[†]: Equal contribution author (order determined by coin flip)

Abstract

It is well known that conservative mechanical systems exhibit local oscillatory behaviours due to their elastic and gravitational potentials, which completely characterise these periodic motions together with the inertial properties of the system. The classification of these periodic behaviours and their geometric characterisation are in an on-going secular debate, which recently led to the so-called *eigenmanifold* theory. The eigenmanifold characterises nonlinear oscillations as a generalisation of linear eigenspaces. With the motivation of performing periodic tasks efficiently, we use tools coming from this theory to construct an optimization problem aimed at inducing desired closed-loop oscillations through a state feedback law. We solve the constructed optimization problem via gradient-descent methods involving neural networks. Extensive simulations show the validity of the approach.

KEYWORDS:
Nonlinear Oscillation, Stabilisation, Neural Networks

1 | INTRODUCTION

Mechanical systems, such as industrial robots or bio-inspired ones, often need to perform tasks exhibiting a periodic nature, e.g., pick and place or periodic locomotion. The ubiquity of these tasks, as well as the theoretical appeal of understanding and characterising periodic solutions of dynamical systems, made the study of repetitive motions and their control an immensely important branch in the system theoretic community.

Abstracting from the specific class of mechanical systems and assuming a more general control theoretic perspective, the problem of tracking periodic signals, sometimes referred to as *periodic regulation*, has been intensively tackled with different tools. Without the claim to be exhaustive, we refer to the surveys[1,2] for an overview, to[3,4] (and references therein) for more recent contributions, and to[5] for an application in robotics.

Contrarily to what is pursued in this work, the mentioned approaches are mostly focused on the design of controllers which implement some steady-state cancellation of the plant dynamics to achieve tracking of specific periodic reference signals. As mentioned in[6], these approaches lack a biomimetic perspective in the sense that the design of the periodic regulator is focused on versatility rather than efficiency. In other words, the focus of these approaches is designing a controller that works for a large class of reference signals rather than designing efficient controllers for a smaller class of efficiently stabilisable periodic trajectories. In[6], this efficiency objective is pursued by steering a mechanical system onto natural oscillations of the system itself, which are matched to the mechanical system's physics.

The existence of such periodic oscillations for nonlinear mechanical systems with conservative potentials (usually considered of elastic and gravitational type) is a well-known fact [7,8,9], and the recent theory of *eigenmanifolds* [10] attempts at giving a geometric characterisation of these families of oscillations. These oscillations constitute an invariant of the system, i.e., when no dissipative effects or other disturbances are present, a system initialised on such a nonlinear mode would stay there autonomously, with no need of additional inputs. The control theoretic appeal for such structure is immediate once a nonlinear oscillation is understood as a desired periodic behaviour for the closed-loop system, which can vary from achieving energy efficient forms of locomotion, to industrial-like tasks like e.g., pick and place.

In [6] the authors successfully stabilised these periodic oscillations defined by eigenmanifold theory, claiming an efficient control design. In fact, a controller able to stabilise a specific invariant oscillation of the system only requires a minimal power consumption, as in principle only the energy to compensate for dissipative effects would be injected by the controller. In conclusion, the underlying biomimetic rational drives the designer in exploiting the natural physics (elastic joints, gravity, inertial parameters) to understand and stabilise an efficiently stabilisable behaviour with minimal energy consumption. We refer to the recent paper [11] for further elaborations about the connection between efficiency in robotics and the exploitation of natural physics (referred to as "intrinsic dynamics" in that work) present in mechanical systems.

In [6] the approach was limited to stabilise the open-loop nonlinear modes produced by the conservative elastic and gravitational potentials of the underlying mechanical system. Motivated by the fact that natural modes of the open-loop system might not correspond to desired task-specific oscillations, and that mechanical design of a system achieving specific desired oscillations might be very difficult, we introduce a new scheme, which can be seem as an extension of the one in [6] to account for a broader class of periodic oscillations. In particular, we aim at learning and stabilising a desired oscillation which achieves the fulfillment of some periodic task, which is *close* to the natural mode of the underlying system, but not necessarily coincident. The main contribution of this paper is to present a procedure aimed at finding a potential based state-feedback law which generates desired efficient oscillations in the closed-loop system. In order to do so we cast the control problem into an optimisation framework in which the decision variable is a control potential, approximated by a neural network and updated through gradient descent to minimise a task-dependent performance metric together with a metabolic cost. The learned potential uniquely defines a feedback law which generates a closed-loop system exhibiting the desired oscillations. These are then stabilised using an approach similar to [6,10], where non trivial adaptations have been made to improve the energetic behavior of the control.

Extensive simulations performed on a double pendulum show the validity of the approach.

1.1 | Structure of the paper

The structure of the paper is sketched in Fig. 1. In Sec. 2, we give some background material on the Hamiltonian formulation of controlled mechanical systems and on eigenmanifolds. In Sec. 4.1, the main contribution of this work, the optimisation of the control potential is presented and addressed through gradient descent methods involving neural networks as functional approximators. The section is concluded by defining the controller aimed at stabilising the mechanical system on the learned periodic mode and addressing the energetic behavior (in particular passivity) of the resulting closed-loop system. Sec. 5 contains the simulations and discussions, while Sec. 6 concludes the paper. The extensive appendices B to E show further results of the proposed optimization.

2 | BACKGROUND

2.1 | Hamiltonian formalism for controlled conservative mechanical systems

In this work we deal with conservative mechanical systems, and we use the Hamiltonian formalism to describe their dynamics. Even if not standard in the eigenmanifold literature, this choice will provide technical advantages in formally presenting some properties of interest. In order to keep the focus on the relevant contributions, in this work we will present all the equations in "standard" coordinates with $\mathbb{R}^n$ as the configuration space for an n-dimensional mechanical system [1]. However, all the concepts can be generalised at a manifold level. The Hamiltonian dynamics (with control) of an n-DoF conservative mechanical system with position $\boldsymbol{q}(t) \in \mathbb{R}^n$ and momentum $\boldsymbol{p}(t) \in \mathbb{R}^n$ is described (hiding time dependencies for lightening notation) by

[1] The proper configuration space for a e.g. a double pendulum is the torus T^2, rather than $\mathbb{R}^2$ - in this work, the distinction has a negligible impact.

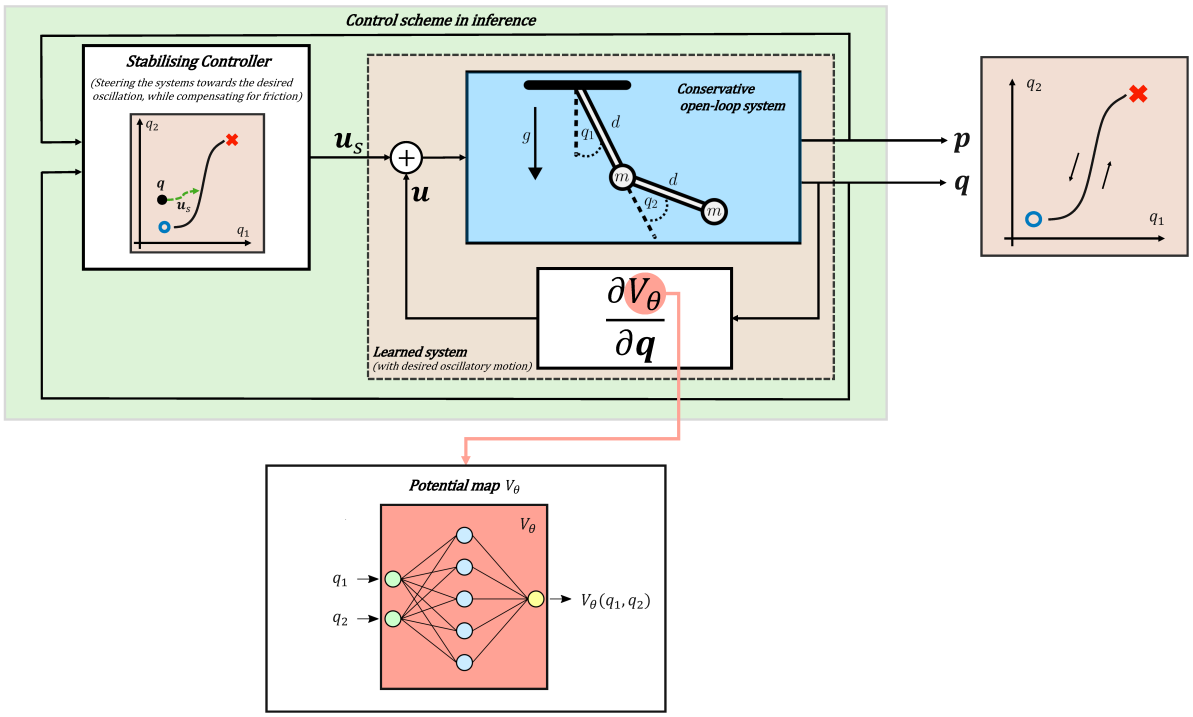


FIGURE 1 General architecture of the control scheme and synopsis of the paper

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{q}\\ \boldsymbol{p}\end{bmatrix} &= \begin{bmatrix}0 & \boldsymbol{I}\\ -\boldsymbol{I} & 0\end{bmatrix}\nabla H(\boldsymbol{p},\boldsymbol{q}) + \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{I}\end{bmatrix}\boldsymbol{u}\\ \begin{bmatrix}\boldsymbol{q}(0)\\ \boldsymbol{p}(0)\end{bmatrix} &= \begin{bmatrix}\boldsymbol{q}_0\\ \boldsymbol{p}_0\end{bmatrix} \end{aligned} \tag{1}$$

where $H(\boldsymbol{q},\boldsymbol{p}) = K(\boldsymbol{q},\boldsymbol{p}) + V(\boldsymbol{q})$ is the Hamiltonian, i.e., the total mechanical energy of the system. The total mechanical energy H is given by the sum of kinetic energy $K(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p}$ (where $M(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the inertia tensor) and the potential energy $V(\boldsymbol{q})$, storing the conservative gravitational and elastic effects. As standard in this formalism, the gradient operator applied to the Hamiltonian is given by $\nabla H(\boldsymbol{p},\boldsymbol{q}) = \left[\frac{\partial}{\partial \boldsymbol{q}}H(\boldsymbol{p},\boldsymbol{q}) \quad \frac{\partial}{\partial \boldsymbol{p}}H(\boldsymbol{p},\boldsymbol{q})\right]^T \in \mathbb{R}^{2n}$, and $\boldsymbol{I}$ and $\boldsymbol{0}$ are the n-dimensional identity and zero matrices respectively. We consider an explicit control input $\boldsymbol{u}$, representing the generalised forces on the mechanical system collocated to the degrees of freedom defining the position coordinates $\boldsymbol{q}$. The usual corollary that the Hamiltonian function is conserved along autonomous evolutions ($\dot{H} = 0$ holds along solutions of (1) with $\boldsymbol{u} = 0$) will be used in the rest of this work.

2.2 | Eigenmanifolds

Eigenmanifold theory [10] generalises the theory of oscillations present in linear mechanical systems to conservative, intrinsically nonlinear mechanical systems. Here, the essentials of this formalism are presented in its Hamiltonian form.

Definition 1. An isolated eigenmode $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^{2n}$ of an autonomous conservative mechanical system (i.e., a system in the form (1) with $\boldsymbol{u} = 0$) is a trajectory $\boldsymbol{x}(t) = (\boldsymbol{q}(t), \boldsymbol{p}(t))$ with the properties:

- $\boldsymbol{x}$ is periodic, i.e. $\exists T > 0 : (\boldsymbol{q}(t), \boldsymbol{p}(t)) = (\boldsymbol{q}(t+T), \boldsymbol{p}(t+T))$.
- within one period there must be two distinct points with zero momentum, i.e. $\exists t_1 \neq t_2, t_2 - t_1 < T : \boldsymbol{p}(t_1) = 0$ and $\boldsymbol{p}(t_2) = 0$.
- the set $\{\boldsymbol{q}(t) | t \in \mathbb{R}\}$ is "line-shaped", i.e., it is homeomorphic to the closed interval $[0, 1] \subset \mathbb{R}$.

An eigenmanifold $E \subseteq \mathbb{R}^{2n}$ is then a collection of such modes $\boldsymbol{x}$, defined with respect to an isolated, stable equilibrium $\boldsymbol{x}_{\text{eq}} = (\bar{\boldsymbol{q}}, 0)$ of the system (1). Such an equilibrium, which represents the "trivial mode" in the eigenmanifold, exists at a minimum of the potential $\boldsymbol{V}(\boldsymbol{q})$. The additional demand is that the collection $\boldsymbol{R} = \boldsymbol{x}_{\text{eq}} \cup \{\boldsymbol{x}(t) | \boldsymbol{p}(t) = 0\}$, called the generator of the eigenmanifold, is a connected, 1-dimensional submanifold [2], see also Fig. 2. The generator represents the collection of points which are the extrema of the oscillations of every mode in the eigenmanifold. These modes, for systems in the form (1), partially characterise the periodic oscillations that a frictionless mechanical system can have.

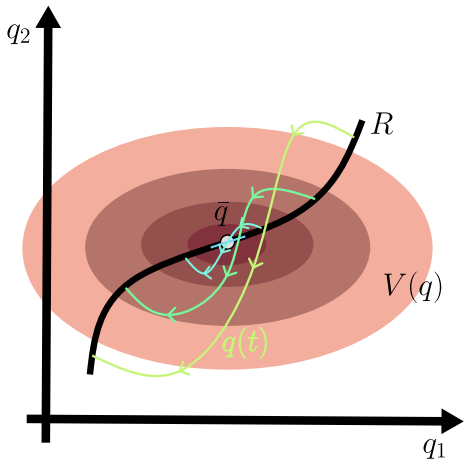


FIGURE 2 In a mechanical system with potential $V(\boldsymbol{q})$ and equilibrium $\bar{\boldsymbol{q}}$, an eigenmanifold is a collection of eigenmodes $x(t) = (\boldsymbol{q}(t), \boldsymbol{p}(t))$, such that the 1-dimensional generator $\boldsymbol{R}$ (which collects particular initial conditions of different eigenmodes) contains the equilibrium $x_{eq} = (\bar{\boldsymbol{q}}, \boldsymbol{0})$ as a limit point. Here, these concepts are depicted after their projection to a 2-dimensional configuration space.

It is instructive to think about these modes as the collection of trajectories of (1) factoring out i) the bounded and non periodic evolutions, whose behavior is commonly referred to as *chaotic* and ii) the periodic evolutions for which no point with $\boldsymbol{p}(t) = 0$ exists, i.e., those which do not qualify as oscillations. Eigenmanifolds are then of particular interest to factor out such trajectories in nonlinear mechanical systems with $n \geq 2$ DoFs (e.g., double pendulum), where chaotic behavior is often present.

Remark 1. Similar to linear oscillations the eigenmodes can often [3] be ordered in the eigenmanifold for increasing levels of energy along a mode (starting with the zero energy level corresponding to the trivial mode which is the equilibrium $\boldsymbol{x}_{\text{eq}}$). Contrarily to the linear case, the eigenmanifold can be bounded (it is not a linear space and it can also not be extended indefinitely for high energy levels), and every mode has in general a different period T (while in the linear case it is constant).

The problem of *existence* and the complete characterisation of eigenmanifolds for conservative mechanical systems is an open problem and is out of the scope of this paper. We refer to [10] for the latest developments in this direction. Nevertheless, as main motivation of this work, both experimental and numerical evidence [12] are confirming that such nonlinear oscillations are structurally present in mechanical systems of any dimension, and can be detected and stabilised, as shown in [13,6].

To formulate the proposed eigenmanifold optimization method in Section 4.1, we need three lemmas involving conservative mechanical systems which can be verified using the Hamiltonian formulation (1). In fact, the latter system (with $\boldsymbol{u} = 0$) is subject to the discrete symmetry $(\boldsymbol{q}, \boldsymbol{p}, t) \to (\boldsymbol{q}, -\boldsymbol{p}, -t)$, i.e., if $(\boldsymbol{q}(t), \boldsymbol{p}(t))$ is a forward in time solution for (1), then $(\boldsymbol{q}(-t), -\boldsymbol{p}(-t))$ is likewise a forward in time solution for (1). The following lemmas, which are proven in [12], act as corollaries.

Lemma 1. Any trajectory with $\boldsymbol{q}(0) = \boldsymbol{q}_0$ arbitrary and initial momentum $\boldsymbol{p}(0) = 0$ will have the property that $\boldsymbol{q}(t) = \boldsymbol{q}(-t)$ and $\boldsymbol{p}(t) = -\boldsymbol{p}(-t)$.

[2] Usually, generators are defined such that only one point of each mode appears on the generator, i.e., each eigenmanifold has two generators. The distinction is not important for the present work.

[3] The energy-level alone does not always induce a unique ordering of eigenmodes on an eigenmanifold.

Lemma 2. Any *periodic* trajectory with $\boldsymbol{p}(0) = 0$ and period T will have the property that $\boldsymbol{p}(T/2) = 0$.

Lemma 3. Any trajectory with two distinct points $\boldsymbol{q}(t_1) \neq \boldsymbol{q}(t_2)$ such that $\boldsymbol{p}(t_1) = 0$ and $\boldsymbol{p}(t_2) = 0$ will be periodic, with period $T = 2|(t_1 - t_2)|$.

Combining the definition of an eigenmode and the previous lemmas, valid for any conservative mechanical system, the following can be concluded. An *eigenmode* with initial conditions[4] ($\boldsymbol{q}(0) = \boldsymbol{q}_0, \boldsymbol{p}(0) = 0$) has necessarily a period $T = 2\bar{t}$, where $\bar{t}$ is the time instant of the other extremum of the oscillation, i.e., $\boldsymbol{p}(\bar{t}) = 0$. Conversely, if a periodic trajectory with properties defined in Lemma 3 presents a line shaped set $\{\boldsymbol{q}(t) | t \in \mathbb{R}\}$, it is necessarily a (possibly isolated) eigenmode. It is worth to note that this condition of line-shapedness was rarely violated in practice.

3 | RELATED WORK: NEURAL NETWORKS IN DYNAMICAL SYSTEMS

The recent developments of artificial intelligence and machine learning has opened the door to new approaches for understanding and controlling dynamical systems by relying on data. In particular, data-driven methods, e.g. neural networks, have often been used as function approximators for learning the dynamics of systems [14,15,16,17] or for representing control strategies [18,19,20] even in high-dimensional optimal control problems [21]. Furthermore, neural networks can be used to approximate the Lyapunov functions in the case of autonomous [22,23,24,25,26] and non-autonomous dynamical systems [27,28,29,30] for stability and control purposes.

However, purely data-driven methods often learn physically-inconsistent models that do not respect physical conservation laws. Therefore, the most recent research trends have shifted towards encoding physical principles into neural networks. Examples are hamiltonian, symplectic, and lagrangian neural networks [31,32,33] and the physics-informed neural networks [34], aiming at exploiting the best of both worlds, namely the expressive power of nonlinear function approximators with grounded physical knowledge.

Another important step in this direction has been the introduction of Neural Ordinary Differential Equations (Neural ODEs) [35]. The neural ODE framework allows the study of a neural network and its training phase as ODE, opening many possibilities for analysis and understanding of black-box methods.

A closely-related approach to our methods is the work of [36], where a neural ODE is used for learning an optimal passive controller in the port-Hamiltonian framework. The learned controller is composed of a learned potential energy term and a learned damping injection term. However, differently from [36] which solve the problem of the stabilization of an inverted pendulum, we focus on a more complex problem, namely the learning of energy-efficient eigenmodes for optimally solving pick and place tasks with a double pendulum. Additionally, instead of learning a damping injection term, we introduce a passive controller injecting only the energy lost by the system due to dissipative elements.

4 | DISCOVERING AND STABILIZING OPTIMAL EIGENMODES

In this work, we want to control trajectories efficiently towards periodic signals that perform some task. To do so, we first need to find a periodic signal that 1) represents the execution of a task and 2) allows for efficient control towards it. We discuss this in Section 4.1. Once such an oscillation is found, we discuss a controller that steers trajectories onto this orbit in Section 4.2

4.1 | Discovering optimal eigenmodes via Neural Approximators

To find an oscillatory motion that allows for efficient control towards it, we consider eigenmodes (see definition 1) of the system in (1) where we restrict the input to the gradient of a control potential $\boldsymbol{u} = \nabla_q V_{\boldsymbol{\theta}}(\boldsymbol{q})$, so that the closed-loop system will have the form of an autonomous mechanical system (1) ($\boldsymbol{u} = 0$) with Hamiltonian $H + V_{\boldsymbol{\theta}}$. The rationale behind this choice is to modify the system dynamics from (1) as little as possible, avoiding potential cancellation approaches and exploiting the natural physics in the most efficient way.

The aim is to find the map $V_{\boldsymbol{\theta}} : \mathbb{R}^n \to \mathbb{R}$, such that for a fixed initial condition the resulting motion is an eigenmode of the closed-loop system and minimizes a task-dependent cost term L_{task}. This yields a constrained optimization problem whose decision variable is the map $V_{\boldsymbol{\theta}}$. To solve this problem with gradient descent methods, a finite-dimensional parametrisation of

[4] Note that due to the definition of eigenmode this choice of initial condition does not induce a loss of generality.

V_θ is necessary. We denote $\boldsymbol{\theta}$ the vector collecting the (finitely many) parameters characterising the map V_θ, which motivates the notation for the latter. In this work $\boldsymbol{\theta}$ will collect the parameters of a neural network, which will be used as functional approximator for $V_\theta(\boldsymbol{q})$. In this perspective, the closed-loop system in optimisation phase becomes a so-called Neural ODE[35].

Summarising the above considerations, the optimisation that we aim to solve is then represented as:

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & L_{\text{task}}(\boldsymbol{x}) \\ \text{s.t.} \quad & \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{q}\\ \boldsymbol{p}\end{bmatrix} = \begin{bmatrix}0 & \boldsymbol{I}\\ -\boldsymbol{I} & 0\end{bmatrix}\nabla(H+V_\theta)(\boldsymbol{p},\boldsymbol{q}) \quad , \begin{bmatrix}\boldsymbol{p}(0)\\ \boldsymbol{q}(0)\end{bmatrix} = \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{q}_0\end{bmatrix} \\ & L_{\text{eigen}}(\boldsymbol{x}) = 0 \end{aligned} \tag{2}$$

where $L_{\text{task}}(\boldsymbol{x})$ is the loss function of the problem, $L_{\text{eigen}}(\boldsymbol{x}) = 0$ represents the constraint forcing the closed-loop trajectory to be an eigenmode, and $\boldsymbol{x} = (\boldsymbol{q}, \boldsymbol{p})$. Notice that the choice $\boldsymbol{p}_0 = \boldsymbol{0}$ happens without loss of generality since we are dealing with periodic orbits, and by Def. 1 an eigenmode is always characterised by $\boldsymbol{p}(t) = \boldsymbol{0}$ for some t. Unless specified otherwise, in this work we assume both the initial position $\boldsymbol{q}_0$ of the eigenmode and its period T to be fixed.

In Section 5, we solve the optimization problem (2) for a pick and place experiment where we move from initial task space position $h(\boldsymbol{q}_0)$ (being $h(\boldsymbol{q})$ the forward kinematic map) to a desired position h^*. In this case we design $L_{\text{task}}(\boldsymbol{x})$ as:

$$L_{\text{task}}(\boldsymbol{x}) = \frac{1}{2}\alpha_{\text{task}}\|h(\boldsymbol{q}\left(\frac{T}{2}\right)) - h^*\|_2^2 + \alpha_{\text{eff}}\int_0^T \|\boldsymbol{u}\|_2^2 \mathrm{d}t, \tag{3}$$

where $\|\cdot\|_2$ is the 2-norm, such that the first term promotes the minimisation of the distance between the end-effector position at time $t = \frac{T}{2}$ and the target position h^*, and the second term is of metabolic nature and penalises high control efforts $\boldsymbol{u}(t) = \nabla V_\theta(\boldsymbol{q}(t))$ [5]. Here, α_{eff} is a positive scalar balancing the contribution of the two terms, whose effect is analysed in Appendix B.1.

The constraint $L_{\text{eigen}}(\boldsymbol{x}) = 0$ in (2) is designed in a way to force the evolution of the closed-loop system to be an oscillation: the construction of the function $L_{\text{eigen}}(\boldsymbol{x})$ is inspired by Lemma's 1, 2, and 3. In particular, given the initial conditions in (2), by Lemma's 2 and 3, it suffices to enforce $\boldsymbol{p}\left(\frac{T}{2}\right) = \boldsymbol{0}$ to get a periodic trajectory of period T. Moreover, given a periodic trajectory with period T, Lemma 1 shows that $\boldsymbol{q}(t) = \boldsymbol{q}(T-t)$ and $\boldsymbol{p}(t) = -\boldsymbol{p}(T-t)$. Finally, by periodicity, the trajectory satisfies $\boldsymbol{q}(T) = \boldsymbol{q}_0$ and $\boldsymbol{p}(T) = \boldsymbol{0}$. Combining all these observations, we chose the following form for L_{eigen}:

$$L_{\text{eigen}}(\boldsymbol{x}(t)) = \lambda_1\left(\|\boldsymbol{q}(t) - \boldsymbol{q}(T-t)\|_{\infty,T} + \alpha_1\|\boldsymbol{p}(t) + \boldsymbol{p}(T-t)\|_{\infty,T}\right) + \frac{\lambda_2}{2}\|\boldsymbol{p}\left(\frac{T}{2}\right)\|_2^2 \tag{4}$$

where $\lambda_i \in \mathbb{R}_+$ ($i = 1,$), $\alpha_1 \in \mathbb{R}_+$, and where $\|\cdot\|_{\infty,T}$ is defined by: $\|\boldsymbol{y}(\cdot)\|_{\infty,T} := \max_{t\in[0,\frac{T}{2}]}(\|\boldsymbol{y}(t)\|_1)$ with $\boldsymbol{y} : [0,\infty) \to \mathbb{R}^n$.

As an alternative to (2), the eigenmode constraint can be relaxed into a soft one by solving the optimisation:

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & L_{\text{task}}(\boldsymbol{x}) + \beta L_{\text{eigen}}(\boldsymbol{x}) \\ \text{s.t.} \quad & \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{q}(t)\\ \boldsymbol{p}(t)\end{bmatrix} = \begin{bmatrix}0 & \boldsymbol{I}\\ -\boldsymbol{I} & 0\end{bmatrix}\nabla(H+V_\theta)(\boldsymbol{p},\boldsymbol{q}) \quad , \begin{bmatrix}\boldsymbol{p}(0)\\ \boldsymbol{q}(0)\end{bmatrix} = \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{q}_0\end{bmatrix} \end{aligned} \tag{5}$$

with $\beta \in \mathbb{R}^+$ a positive constant.

We stress that even though this version of the optimisation does not present $L_{\text{eigen}} = 0$ as a hard constraint, in the moment in which a line shaped periodic trajectory results as a solution of the optimization problem, we are able to assess the learning of an eigenmode with the same confidence as for (2) by considering Definition 1, although we can in principle not ensure that the optimization will result in a periodic orbit.

Remark 2. In the pick and place experiment, we want the end-effector to stop at a specific location $h^* \neq h(\boldsymbol{q}_0)$ at some arbitrary time t. By choosing $t = \frac{T}{2}$ in (3), the constraint $L_{\text{eigen}}(\boldsymbol{x}) = 0$ guarantees that the end-effector will actually stop at h^*.

Remark 3. With the lemmas 1, 2, 3, and the definition of an eigenmode in mind, it is easy to check that the trajectory $\boldsymbol{x}(t)$ will correspond to an eigenmode in the sense of eigenmanifold theory if and only if $\lambda_2 > 0$, $L_{\text{eigen}}(\boldsymbol{x}(t)) = 0$ and $\{\boldsymbol{q}(t) | t \in \mathbb{R}\}$ is line shaped. As a consequence, the term λ_1 is strictly speaking redundant, but was found to improve the convergence (together with the specific choice of norms in (4)) in the optimisation. In conclusion, if the solution of the optimization problem above yields a line-shaped periodic trajectory, we can conclude that the orbit indeed corresponds to an eigenmode. We furthermore stress the

[5] While this cost is well-defined in the given coordinates, we point out the implicit choice of a distance $\|h(\boldsymbol{q}\left(\frac{T}{2}\right)) - h^*\|_2$ on the task space $\mathbb{R}^2$ and a norm $\|\boldsymbol{u}\|_2$ in the vector space $T_q^*\mathbb{R}^2 \cong \mathbb{R}^2$.

practical scarcity of non line-shaped periodic trajectories, which, to the knowledge and the experience of the authors, have been rarely found in the previously studied cases.

4.1.1 | Solving the optimisation

Given the finite-dimensional parametrisation of the map $V_{\boldsymbol{\theta}}(\boldsymbol{q})$, in this work the optimisation is solved through gradient descent methods, i.e., the optimal parameters $\boldsymbol{\theta}$ are found by iterating:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta_k \frac{\partial}{\partial \boldsymbol{\theta}} L(\boldsymbol{x}(\boldsymbol{\theta})) \tag{6}$$

where $L(\boldsymbol{x}) = L_{\text{task}}(\boldsymbol{x}) + \beta L_{\text{eigen}}(\boldsymbol{x})$ is the cost in (5). If η_k, a positive scalar referred to as *learning rate*, is suitably chosen, and $L(\boldsymbol{x})$ is convex, $\boldsymbol{\theta}$ converges to the minimiser of $L(\boldsymbol{x})$ as $k \to \infty$. Although global convergence is no longer guaranteed in the nonconvex case (which is the case of this work), gradient descent techniques are widely used in practical applications, especially among the machine learning community, due to their scalability and computational efficiency.

In order to implement the gradient descent procedure, the sensitivity $\frac{\partial}{\partial \boldsymbol{\theta}} L(\boldsymbol{x}(\boldsymbol{\theta}))$ needs to be computed. This is where the so called neural ODE framework, an extension of the continuous depth framework for recurrent neural networks, is used. In particular, the dynamic constraint in (2) has the structure of a neural ODE, i.e, an ordinary differential equations parametrised by a neural network $V_{\boldsymbol{\theta}}(\boldsymbol{q})$ with parameters $\boldsymbol{\theta}$. The *training* of this continuous network corresponds to solving the optimisation problem (2). The sensitivities $\frac{\partial}{\partial \boldsymbol{\theta}} L(\boldsymbol{x}(\boldsymbol{\theta}))$ are calculated via the backpropagation method, in particular via automatic differentiation[37], that is commonly used for training neural networks. Utilising the adjoint method[35] for computing the exact sensitivies, rather than the approximate ones computed by backpropagation, is an option for future investigation.

4.2 | Stabilising Controller and Analysis of the Closed-Loop System

We formally introduced the optimisation that aims at learning a closed-loop conservative mechanical system exhibiting desired oscillations. In real applications, where dissipative effects and parametric disturbances are present, it is important to design a controller able to robustly stabilise the closed-loop system onto the learned eigenmode. With the motivation of interpreting the learned oscillations as "efficient" (minimizing a certain cost-function), it would furthermore be desirable that the stabilising controller acts in a energetically convenient way (i.e., the control effort is equal to zero on the desired trajectory, and the controller is passive, if no dissipation is present). In other words, the controller should inject the mechanical energy needed to stay on the eigenmode into the system and it should compensate for unavoidable dissipative effects only, resembling a clear biomimetic approach. In[6] such a controller was successfully implemented to stabilise the (open-loop) eigenmodes of a 7-DoF KUKA iiwa robot. Here we propose an alternative stabilising controller that is likewise split into an energy-injecting and an eigenmode stabilizing part. Contrary to[6], the latter is not allowed to inject energy in this work. The effect of this splitting will simplify the analysis of the controller.

The system with stabilizing feedback $\boldsymbol{u}_s : \mathbb{R}^{2n} \to \mathbb{R}^n$ is of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{q}(t)\\ \boldsymbol{p}(t)\end{bmatrix} = \begin{bmatrix}0 & \boldsymbol{I}\\ -\boldsymbol{I} & 0\end{bmatrix} \nabla(H + V_{\boldsymbol{\theta}})(\boldsymbol{p}, \boldsymbol{q}) + \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{I}\end{bmatrix} \boldsymbol{u}_s(\boldsymbol{q}, \boldsymbol{p}) \tag{7}$$

The purpose of this feedback is to stabilize an eigenmode $\bar{\boldsymbol{x}} : \mathbb{R} \to \mathbb{R}^{2n}$ ($\bar{\boldsymbol{x}}(t) = (\bar{\boldsymbol{q}}(t), \bar{\boldsymbol{p}}(t))$), the latter being itself a solution of the learned autonomous system (5). To this end, the desired requirements are

$$\lim_{t\to\infty} \operatorname{dist}(\boldsymbol{q}(t), \bar{\boldsymbol{q}}(\bar{t})) = 0\,, \tag{8}$$

$$\lim_{t\to\infty} (\|\boldsymbol{p}(t) - \sigma \bar{\boldsymbol{p}}(\bar{t})\|) = 0\,, \tag{9}$$

$$\bar{t} = \arg\min_{s} \operatorname{dist}(\boldsymbol{q}(t), \bar{\boldsymbol{q}}(s))\,, \tag{10}$$

$$\sigma = \operatorname{sign}(\boldsymbol{p}^T(t) \boldsymbol{M}^{-1}(\boldsymbol{q}(t)) \bar{\boldsymbol{p}}(\bar{t}))\,. \tag{11}$$

Here $\operatorname{dist}(\boldsymbol{a}, \boldsymbol{b})$ returns the Euclidean distance[6] of points $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$.

[6]In a differential geometric context, $\operatorname{dist}(\boldsymbol{x}, \boldsymbol{y})$ would implement the geodesic distance depending on a choice of metric tensor and connection, while the second requirement would read $\lim_{t\to\infty}(\|\boldsymbol{p}(t) - \rho^* \bar{\boldsymbol{p}}(\bar{t})\|) = 0$, with $\rho^* : T^*_{\bar{\boldsymbol{q}}(\bar{t})}\mathcal{M} \to T^*_{\boldsymbol{q}(t)}\mathcal{M}$ implementing the parallel transport of the momentum $\boldsymbol{p}$ along the geodesic ρ from $\boldsymbol{q}(t)$ to $\bar{\boldsymbol{q}}(\bar{t})$. Here, instead, ρ^* is chosen to be the identity map in the given coordinate system.

Intuitively speaking, $\bar{t}$ in equation (10) is the parameter at which the desired trajectory $\bar{\boldsymbol{q}}$ is closest to the current position $\boldsymbol{q}(t)$. In practice, $\bar{t}$ is implemented as a function $\bar{t} : \mathbb{R}^n \to \mathbb{R}$ that takes as input $\boldsymbol{q} \in \mathbb{R}^n$. Although $\bar{\boldsymbol{q}}(\bar{t})$ is uniquely determined, $\bar{\boldsymbol{p}}(\bar{t})$ is only determined up to a sign for an eigenmode, which is chosen according to the sign function $\sigma : \mathbb{R}^{2n} \to \{-1, 0, 1\}$ in equation (11) to be aligned with the current system momentum $\boldsymbol{p}(t)$.

The choice is made to split the control

$$\boldsymbol{u}_s = \boldsymbol{u}_E + \boldsymbol{u}_M \tag{12}$$

into an energy-controlling feedback $\boldsymbol{u}_E$ (cf.[38]) and an eigenmode stabilizing feedback $\boldsymbol{u}_M$. For an analogous control splitting see[6,13].

4.2.1 | Energy-controlling feedback

The energy-controlling feedback $\boldsymbol{u}_E$ steers the system's energy $E = H + V_\theta$ towards a desired energy $\bar{E} = E(\bar{\boldsymbol{q}}(0), \bar{\boldsymbol{p}}(0))$. The form chosen is

$$\boldsymbol{u}_E = \alpha_E(\bar{E} - E)\hat{\boldsymbol{p}}\,, \tag{13}$$

with $\alpha_E \in \mathbb{R}^+$ a positive control gain and the normalized momentum [7]

$$\hat{\boldsymbol{p}} = \frac{1}{\sqrt{\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p}}}\boldsymbol{p} \tag{14}$$

Since $\dot{\boldsymbol{q}} = M^{-1}(\boldsymbol{q})\boldsymbol{p}$, it holds that the mechanical power $\boldsymbol{u}_E^T\dot{\boldsymbol{q}}$ injected by the energy controller is given by

$$\boldsymbol{u}_E^T\dot{\boldsymbol{q}} = \alpha_E(\bar{E} - E)\sqrt{\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p}}\,. \tag{15}$$

4.2.2 | Eigenmode stabilizing feedback

The eigenmode stabilizing feedback $\boldsymbol{u}_M$ is defined as

$$\boldsymbol{u}_M = \alpha_M \pi_p(\sigma\bar{\boldsymbol{p}}(\bar{t}))\,, \tag{16}$$

where $\alpha_M \in \mathbb{R}^+$ is the positive control gain. Furthermore, $\sigma(\boldsymbol{q}, \boldsymbol{p}) \in \{-1, 0, 1\}$ and $\bar{t}(\boldsymbol{q}) \in \mathbb{R}$ are as defined in (11) and (10), respectively. $\bar{\boldsymbol{p}} : \mathbb{R} \to \mathbb{R}^n$ is the momentum component of the desired eigenmode. Last, π_p is the projection defined by

$$\pi_p(\boldsymbol{X}) := \boldsymbol{X} - \frac{\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{X}}{\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p}}\boldsymbol{p}\,. \tag{17}$$

This projection is such that

$$\boldsymbol{u}_M^T\dot{\boldsymbol{q}} = 0\,, \tag{18}$$

which means that $\boldsymbol{u}_M$ cannot change the energy content of the system, and thus cannot interfere with the control-task of $\boldsymbol{u}_E$.

Remark 4. This is a D-type controller analogous to[6,13], with the only adaptation being that the energy injection is restricted (compare e.g.[39]). The controller of the form (16) follows from

$$\boldsymbol{u}_M = \alpha_M \pi_p(\sigma\bar{\boldsymbol{p}}(\bar{t}) - \boldsymbol{p})\,, \tag{19}$$

by using the property of the projection that $\pi_p(\boldsymbol{p}) = 0$.

4.2.3 | Stability

We first investigate the energetic behavior of the combined controller $\boldsymbol{u}_s = \boldsymbol{u}_E + \boldsymbol{u}_M$, and investigate the stability of the trajectory afterwards. The energy injected by the controller is equal to the mechanical power $\boldsymbol{u}_s^T\dot{q}$:

$$\dot{E} = \frac{\partial E}{\partial \boldsymbol{q}}\dot{\boldsymbol{q}} + \frac{\partial E}{\partial \boldsymbol{p}}\dot{\boldsymbol{p}} = \frac{\partial E}{\partial \boldsymbol{p}}\boldsymbol{u} = \dot{\boldsymbol{q}}^T\boldsymbol{u}_s = \boldsymbol{u}_s^T\dot{\boldsymbol{q}}\,. \tag{20}$$

Here, the second equality holds because the system without feedback $\boldsymbol{u}_s$ conserves E, while the third equality follows from the definition of momentum $M(\boldsymbol{q})^{-1}p = \dot{\boldsymbol{q}}$.

[7] To avoid numerical issues in practice, $\hat{\boldsymbol{p}}$ is chosen as 0 when $\boldsymbol{p}^T M(\boldsymbol{q})\boldsymbol{p} = 0$.

Combining the expressions shows that

$$\dot{E} = \boldsymbol{u}_s^T \dot{\boldsymbol{q}} = \alpha_E(\bar{E} - E)\sqrt{\boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p}}\,. \tag{21}$$

Hence, the energy converges to the desired energy level $\bar{E}$ almost always, i.e. as long as $\boldsymbol{u}_E \neq 0$, and otherwise E is constant. Moreover, as the combined actions $\boldsymbol{u}_E$ and $\boldsymbol{u}_M$ vanish only on the desired mode, we get highly efficient control behavior as highlighted in [13,6].

However, the above does not prove either global or local stability. This work restricts itself to a guarantee of local stability, which can be obtained by evaluating the cycle multipliers of the stabilized periodic orbit. Let $\Psi_t(\boldsymbol{x}(0)) := \boldsymbol{x}(t)$ define the flow of the dynamic system (12), then cycle multipliers can be defined as the ratio of partial derivatives [8]

$$\frac{\frac{\partial}{\partial x^i}\mathrm{dist}(\Psi_T(\boldsymbol{x}), \bar{\boldsymbol{x}}(\bar{t}))_{|\boldsymbol{x}=\boldsymbol{x}_0}}{\frac{\partial}{\partial x^i}\mathrm{dist}(\boldsymbol{x}, \bar{\boldsymbol{x}}(\bar{t}))_{|\boldsymbol{x}=\boldsymbol{x}_0}}\,. \tag{22}$$

Here, x^i denotes the i-th component of $\boldsymbol{x}$ and $\boldsymbol{x}_0$ is a starting point of the stabilized periodic orbit, while $\bar{t}$ and $\bar{\boldsymbol{x}}$ are as defined in and above equation (10). As will be shown along the result section, if these cycle multipliers have absolute values smaller than 1, the periodic orbit is stable.

5 | SIMULATIONS

In this section, we perform numerical experiments for the case of a double pendulum. More precisely, we consider a pick and place experiment where we want the end-effector of the double pendulum to move between two points in an oscillatory fashion. To achieve this, an optimal eigenmode is learned via the optimization strategy in Section 4.1. For our numerical experiments, we solve the optimization problem in (5) with loss functions given in Equations (3) and (4). Subsequently, we stabilize the eigenmode using the control strategy in Section 4.2.

5.1 | Double Pendulum Model

The double pendulum is one of the simplest mechanical systems with non-trivial eigenmanifolds (see also [12]). The presented double pendulum is under the influence of gravity and has a linear spring at the second joint. The equations of motion correspond to the conventions shown in Figure 3. They are fully determined by (1) and the Hamiltonian $H : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ given as in Equations (23) to (25).

$$H(\boldsymbol{q}, \boldsymbol{p}) = \boldsymbol{p}^T M^{-1}(\boldsymbol{q})\boldsymbol{p} + V(\boldsymbol{q})\,, \tag{23}$$

$$M((q_1, q_2)) = md^2 \begin{bmatrix} (3 + 2\cos(q_2)) & \cos(q_2) + 1 \\ \cos(q_2) + 1 & 1 \end{bmatrix}\,, \tag{24}$$

$$V((q_1, q_2)) = V_{\boldsymbol{\theta}}((q_1, q_2)) - mdg(2\cos(q_1) + \cos(q_1 + q_2)) + k(q_2 - \pi/2)^2\,. \tag{25}$$

Here, $V_{\boldsymbol{\theta}}(\boldsymbol{q}) : \mathbb{R}^2 \to \mathbb{R}$ is a potential function, that will be constructed as a neural net with parameters $\boldsymbol{\theta} \in \mathbb{R}^m$. The actual equations of motion are reported for completeness in Appendix A.

5.2 | Results

5.2.1 | Learning Eigenmodes

In Figure 4, we visualise the trajectory of the inner closed-loop conservative system (see Figure 1) at different time instants after the training of the learned potential, via the optimisation procedure described in Section 4.1, for 500 epochs and a given period $T = 1.5\,\mathrm{s}$. The results are obtained with the set of loss function hyperparameters reported in Table 1[9]. The potential $V_{\boldsymbol{\theta}}$ (see Figure 5b) is capable of shaping the systems potential (5a), such that the trajectory of the system is an energy-efficient eigenmode of the desired period T. Additionally in Figure 6, we depict the control inputs $\boldsymbol{u} = \nabla_{\boldsymbol{q}} V_{\boldsymbol{\theta}}(\boldsymbol{q}(t))$ inducing the desired

[8] Typically, cycle multipliers are defined as the eigenvalues of $\frac{\partial}{\partial \boldsymbol{x}}(\Psi_T(\boldsymbol{x}) - \boldsymbol{x}))_{|\boldsymbol{x}=\boldsymbol{x}_0}$. The authors found the alternative definition to be more robust, numerically.
[9] The complete list of hyperparameters is shown in Table E1.

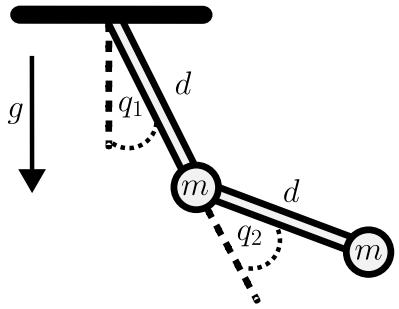


FIGURE 3 Double Pendulum corresponding to Equations (23)-(25).

Hyperparameter	Value
α_{task}	10
α_{eff}	0.0001
α_{task}	10
λ_1	0.05
α_1	0.0005
λ_2	0.95
β	1

TABLE 1 Loss function hyperparameters used in the experiments.

periodic behaviour, and the trajectory in the configuration space (Figure 6c) from which it is possible to notice the line-shaped property of the eigenmode described in Def. 1.

5.2.2 | Stabilization of the Learned Eigenmode

Figure 7 shows the results of applying the control structure introduced in Section 4.2 to the learned trajectory shown in Figure 4, for coefficients $\alpha_M = 10$ and $\alpha_E = 1$.

In the example, we use the starting condition $\boldsymbol{q} = (0.2, 0.2)$, $\boldsymbol{p} = (5, 5)$. In particular, Figure 7d and Figure 7e show the development of $\boldsymbol{q}(t)$ and $\boldsymbol{p}(t)$ over time, which approach the desired $\bar{\boldsymbol{q}}(\bar{t})$ and $\bar{\boldsymbol{p}}(\bar{t})$ (see Section 4.2 for their definition). Figure 7a shows the energy $H + V_\theta$ of the closed loop system, which approaches the constant energy level of the learned mode. Figures 7b and 7c show the distance of the trajectory from the desired trajectory in position and momentum space respectively (i.e. $\|\boldsymbol{q}(t) - \bar{\boldsymbol{q}}(\bar{t})\|_2$ and $\|\boldsymbol{p}(t) - \bar{\boldsymbol{p}}(\bar{t})\|_2$), in both cases approaching 0. The cycle multipliers of the closed loop system are less than 1: for this example, it was found that they are bounded by 0.5, which guarantees that the learned periodic orbit is locally stable.

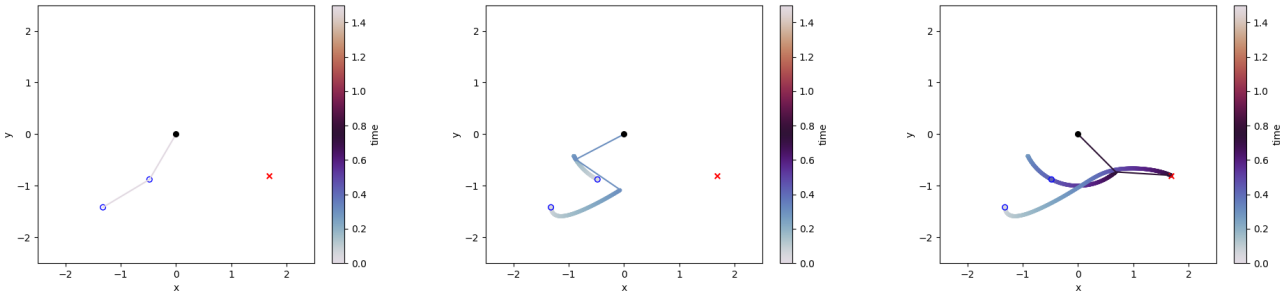
To observe the robustness of the controller in the presence of damping, viscous damping is introduced. With b the damping coefficient, the input $\boldsymbol{u}_s$ in (7) is adapted to read

$$\boldsymbol{u}_s = \boldsymbol{u}_E + \boldsymbol{u}_M - bM^{-1}(q)p\,, \tag{26}$$

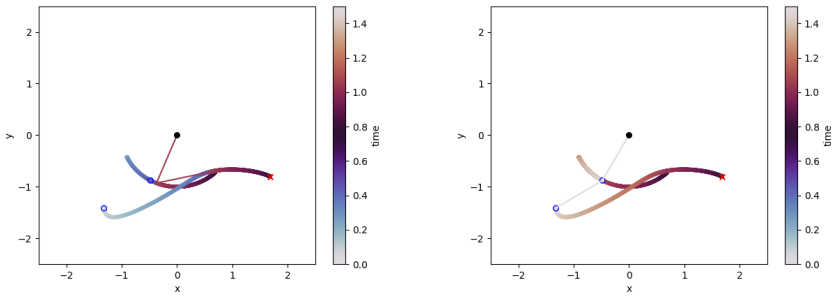
which corresponds to velocity dependent damping. The cases $b = 0.1$ and $b = 1$ are shown in Figures 8f and 9f, respectively. Notably, the damping causes the energy shown in Figures 8a and 9a to continue to fluctuate in the eventual periodic evolution, about a value lower than the desired energy. It is worth noting that the systems remain close to the desired mode, even for such large cases of damping. However, it should be considered to adapt the energy controlling term $\boldsymbol{u}_E$ to compensate for damping more accurately, as was done e.g. in[6] for a particular case of damping that was, among others, linear in velocity.

5.2.3 | Additional Results

To strengthen the numerical contribution, we include additional results and ablation studies in Appendices. In particular, in Appendix A, we show the equations of motion for the double pendulum used in our simulations, while in Appendix B, we

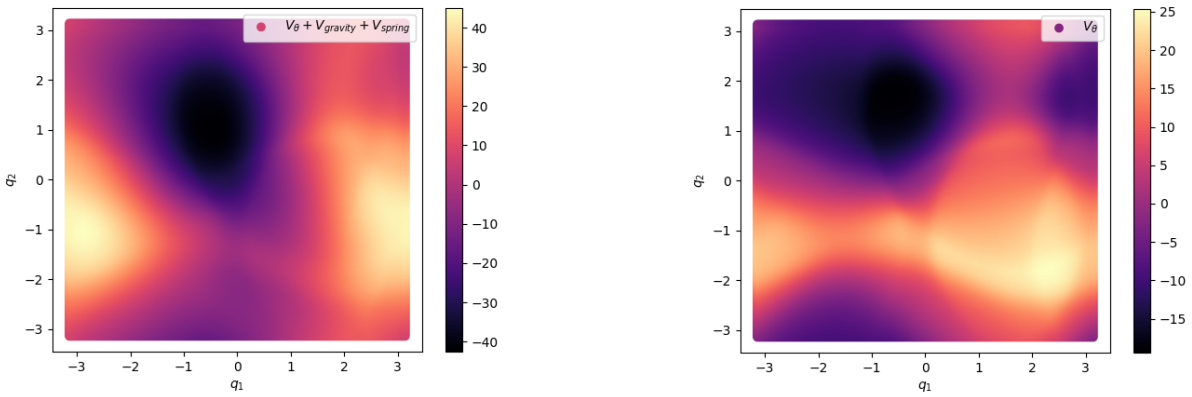


(a) t=0.0 s **(b)** t=0.332 s **(c)** t=0.83 s



(d) t=1.162 s **(e)** T=1.5 s

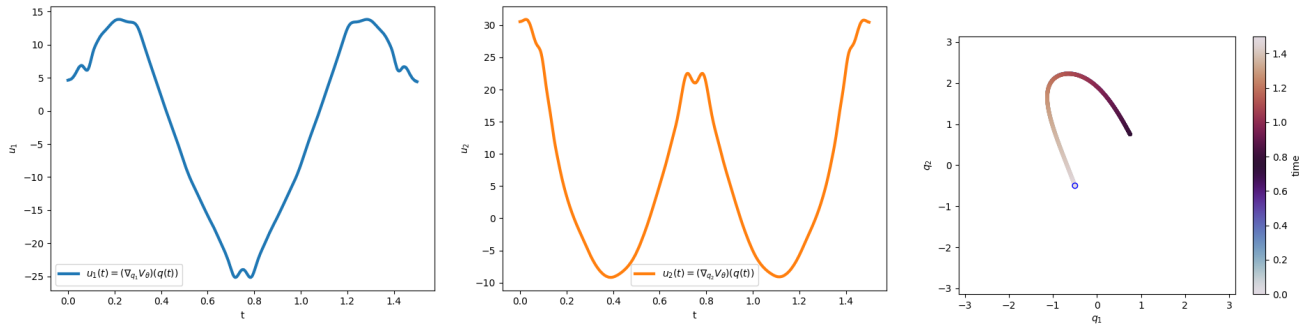
FIGURE 4 Learned eigenmode at different time steps. The blue circles represent the initial position of the joints of the pendulum, while the red cross represents the end-effector target used for computing the first term of $L_{\text{task}}(\boldsymbol{x})$ in (3).



(a) Overall potential $V_{\boldsymbol{\theta}} + V_{\text{spring}} + V_{\text{gravity}}$. **(b)** Learned potential $V_{\boldsymbol{\theta}}$.

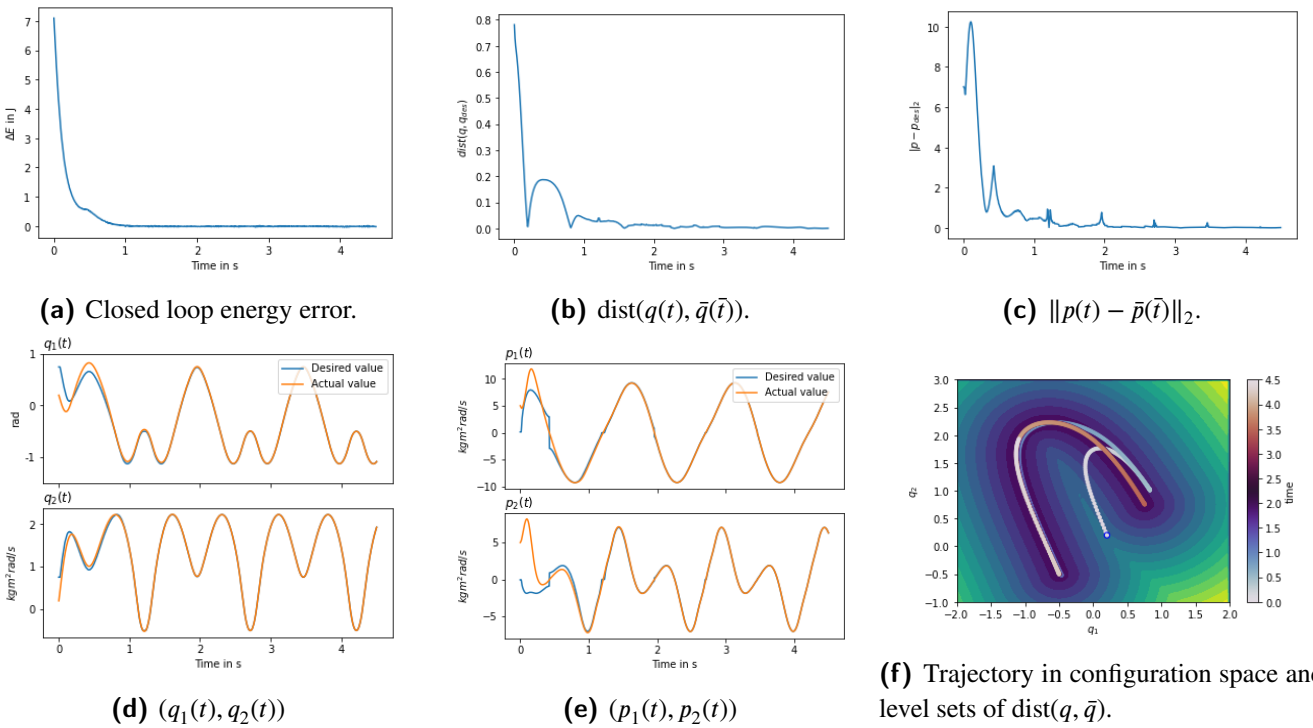
FIGURE 5 Potentials over $\boldsymbol{q} \in [-\pi, \pi]$.

study the effect of varying the regularisation coefficient α_{eff} and the period T on the resulting eigenmode and control inputs. In Appendix C, we apply the method with different initial and target positions, periods, and regularisation coefficients, and in Appedix D, we show a more advanced version of the optimisation problem in (2), where we learn the potential $V_{\boldsymbol{\theta}}$ jointly with the period T. Eventually, Appendix E reports the implementation details for reproducing our experiments.



(a) First component of control $u(t)$ **(b)** Second component of $u(t)$. **(c)** Trajectory in the configuration space.

FIGURE 6 Control inputs over time (Figure 6a and 6b), and trajectory in the configuration space (Figure 6c).



(a) Closed loop energy error. **(b)** $\text{dist}(q(t), \bar{q}(\bar{t}))$. **(c)** $\|p(t) - \bar{p}(\bar{t})\|_2$.

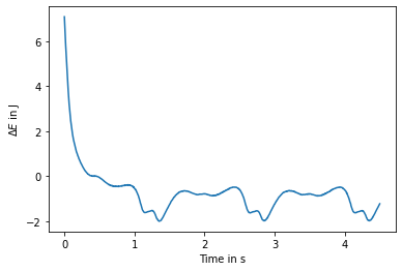
(d) $(q_1(t), q_2(t))$ **(e)** $(p_1(t), p_2(t))$ **(f)** Trajectory in configuration space and level sets of $\text{dist}(q, \bar{q})$.

FIGURE 7 Various features of the stabilised system with learned potential as in Figure 5, stabilizing the mode shown in Figure 4 with gains $\alpha_M = 10$, $\alpha_E = 1$. The starting condition is $\boldsymbol{q}(0) = (0.2, 0.2)$, $\boldsymbol{p}(0) = (5, 5)$, shown here over three periods of oscillation.

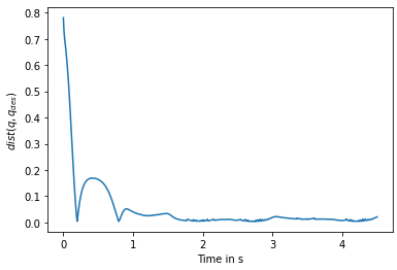
6 | CONCLUSIONS AND FUTURE WORK

In this paper we present a procedure aiming at shaping desired periodic oscillations for mechanical systems. In particular, using tools from eigenmanifold theory and neural networks as function approximators, a state feedback law is learned in such a way to produce a closed-loop system exhibiting a desired periodic motion. This is done by minimising the effort of the learned control law and exploiting at best the natural physical properties of the underlying open-loop systems, characterised by its inertia and its conservative potentials. A stabilising controller able to steer the system along the learned oscillation in presence of parametric disturbances is presented. Extensive simulations show the validity of the approach.

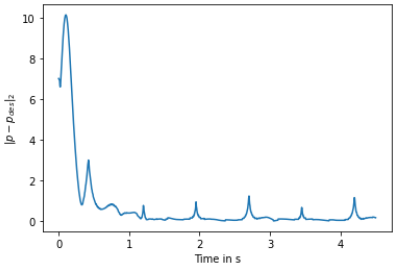
Concerning future developments, besides an experimental validation of the scheme, the proposed approach opens the way to co-design of the mechanical system along the desired periodic task. In fact, by constraining the search space of the learned



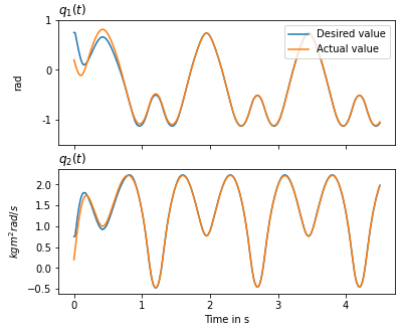
(a) Closed loop energy error.



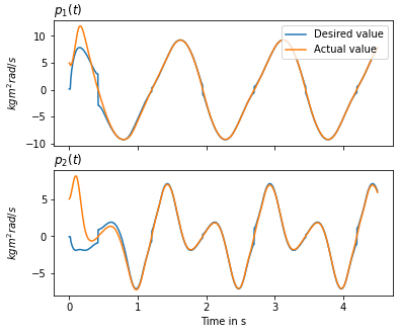
(b) $\mathrm{dist}(q(t), \bar{q}(\bar{t}))$.



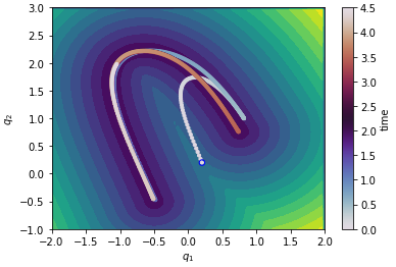
(c) $\|p(t) - \bar{p}(\bar{t})\|_2$.



(d) $(q_1(t), q_2(t))$

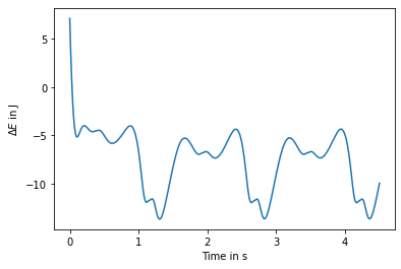


(e) $(p_1(t), p_2(t))$

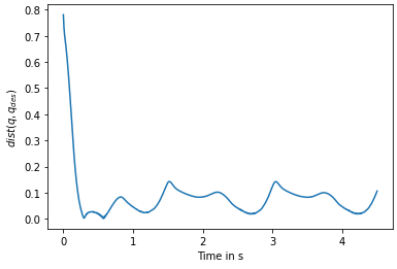


(f) Trajectory in configuration space and level sets of $\mathrm{dist}(q, \bar{q})$.

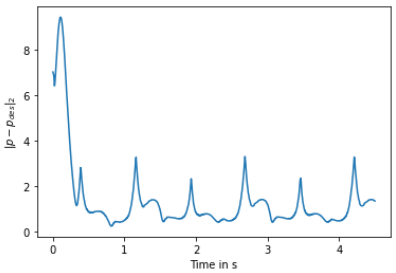
FIGURE 8 Various features of the stabilized system shown in 7, but including damping linear in system velocity with damping coefficient $b = 0.1$



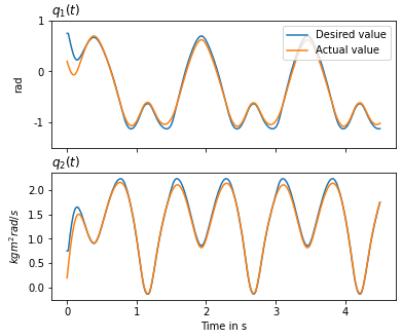
(a) Closed loop energy error.



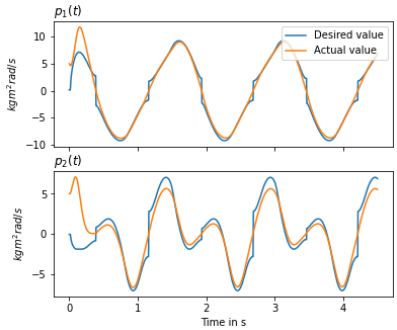
(b) $\mathrm{dist}(q(t), \bar{q}(\bar{t}))$.



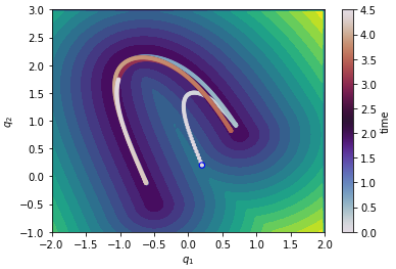
(c) $\|p(t) - \bar{p}(\bar{t})\|_2$.



(d) $(q_1(t), q_2(t))$



(e) $(p_1(t), p_2(t))$



(f) Trajectory in configuration space and level sets of $\mathrm{dist}(q, \bar{q})$.

FIGURE 9 Various features of the stabilized system shown in 7, but including damping linear in system velocity with damping coefficient $b = 1$

potential to a form which can be reproduced mechanically with e.g., nonlinear springs at the joints, it would be possible to exploit the described learning procedure as a preliminary phase for a mechanical design which would produce a mechanical system achieving the desired behaviour in an open-loop fashion.

References

1. Longman RW. Iterative learning control and repetitive control for engineering practice. *International Journal of Control* 2000; 73(10): 930–954. doi: 10.1080/002071700405905

2. Wang Y, Gao F, Doyle FJ. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control* 2009; 19(10): 1589–1600. doi: 10.1016/j.jprocont.2009.09.006

3. Califano F, Bin M, Macchelli A, Melchiorri C. Stability analysis of nonlinear repetitive control schemes. *IEEE control systems letters* 2018; 2(4): 773–778.

4. Astolfi D, Praly L, Marconi L, Paristech M. Nonlinear Robust Periodic Output Regulation of Minimum Phase Systems. 2021.

5. Kasac J, Novakovic B, Majetic D, Brezak D. Passive finite-dimensional repetitive control of robot manipulators. *IEEE Transactions on Control Systems Technology* 2008; 16(3): 570–576. doi: 10.1109/TCST.2007.912235

6. Bjelonic F, Sachtler A, Albu-sch A, Santina CD. Experimental Closed-Loop Excitation of Nonlinear Normal Modes on an Elastic Industrial Robot. 2021: 1–8.

7. Rosenberg RM. On nonlinear vibrations of systems with many degrees of freedom. 1966. doi: 10.1016/S0065

8. Shaw SW, Pierre C. Normal Modes for Non-Linear Vibratory Systems. *Journal of Sound and Vibration* 1993; 164(1): 85–124. doi: 10.1006/jsvi.1993.1198

9. Avramov KV, Mikhlin YV. Review of applications of Nonlinear Normal Modes for Vibrating Mechanical Systems. 2013

10. Albu-Schäffer A, Della Santina C. A review on nonlinear modes in conservative mechanical systems. *Annual Reviews in Control* 2020; 50: 49–71. doi: 10.1016/j.arcontrol.2020.10.002

11. Albu-Schäffer A, Sachtler A. What Can Algebraic Topology and Differential Geometry Teach Us About Intrinsic Dynamics and Global Behavior of Robots?. 2022.

12. Wotte Y, Sachtler A, Albu-Schäffer A, Della Santina C. Sufficient conditions for an eigenmanifold to be of the extended Rosenberg type. 2022. doi: 10.21203/rs.3.rs-1416735/v1

13. Santina CD, Albu-Schaeffer A. Exciting Efficient Oscillations in Nonlinear Mechanical Systems Through Eigenmanifold Stabilization. *IEEE Control Systems Letters* 2021; 5(6): 1916–1921. doi: 10.1109/LCSYS.2020.3048228

14. Brunton SL, Proctor JL, Kutz JN, Bialek W. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America* 2016; 113(15): 3932–3937. doi: 10.1073/PNAS.1517384113/SUPPL_FILE/PNAS.1517384113.SAPP.PDF

15. Steven L Brunton , J Nathan Kutz . *Data-driven science and engineering: Machine learning, dynamical systems, and control* . 2022.

16. Champion K, Lusch B, Nathan Kutz J, Brunton SL. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences of the United States of America* 2019; 116(45): 22445–22451. doi: 10.1073/PNAS.1906995116/SUPPL_FILE/PNAS.1906995116.SAPP.PDF

17. Kalia M, Brunton SL, Meijer HGE, Brune C, Kutz JN. Learning normal form autoencoders for data-driven discovery of universal,parameter-dependent governing equations. 2021.

18. Richard S. Sutton , Andrew G. Barto . Reinforcement Learning: An Introduction. 2018.

19. Lillicrap TP, Hunt JJ, Pritzel A, et al. Continuous control with deep reinforcement learning. 2015.

20. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine* 2017; 34(6): 26–38. doi: 10.1109/MSP.2017.2743240

21. Ruthotto L, Osher SJ, Li W, Nurbekyan L, Fung SW. A machine learning framework for solving high-dimensional mean field game and mean field control problems. *Proceedings of the National Academy of Sciences of the United States of America* 2020; 117(17): 9183–9193. doi: 10.1073/PNAS.1922204117/SUPPL_FILE/PNAS.1922204117.SAPP.PDF

22. Serpen G. Empirical Approximation for Lyapunov
}Functions with Artiicial Neural Nets. 2005.

23. Petridis Vasilios , Petridis Stavros . Construction of Neural Network Based Lyapunov Functions. 2006.

24. Gaby N, Zhang F, Ye X. Lyapunov-Net: A Deep Neural Network Architecture for Lyapunov Function Approximation.

25. Grüne L. Computing Lyapunov functions using deep neural networks. 2020.

26. Richards SM, Berkenkamp F, Krause A. The Lyapunov Neural Network: Adaptive Stability Certification for Safe Learning of Dynamical Systems.

27. Chang YC, Roohi N, Gao S. Neural Lyapunov Control. *Advances in Neural Information Processing Systems* 2020; 32. doi: 10.48550/arxiv.2005.00611

28. Mittal M, Gallieri M, Quaglino A, Salehian SM, Koutník J. Neural Lyapunov Model Predictive Control: Learning Safe Global Controllers from Sub-optimal Examples.

29. Mohammad Khansari-Zadeh S, Billard A. Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions. *Robotics and Autonomous Systems* 2014; 62(6): 752–765. doi: 10.1016/J.ROBOT.2014.03.001

30. Long Y, Bayoumi M.M. . Feedback Stabilization: Control Lyapunov Functions
}Modelled by Neural Networks. 1993.

31. Greydanus Google Brain S, Dzamba PetCube M, Yosinski J. Hamiltonian Neural Networks. *Advances in Neural Information Processing Systems* 2019; 32.

32. Cranmer M, , Greydanus S, et al. Lagrangian Neural Networks.

33. Zhong YD, Dey B, Chakraborty A. Symplectic ODE-Net: Learning Hamiltonian Dynamics with Control. 2019. doi: 10.48550/arxiv.1909.12077

34. Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* 2019; 378: 686–707. doi: 10.1016/J.JCP.2018.10.045

35. Chen RTQ, Rubanova Y, Bettencourt J, Duvenaud DK. Neural Ordinary Differential Equations. *Advances in Neural Information Processing Systems* 2018; 31.

36. Massaroli S, Poli M, Califano F, Park J, Yamashita A, Asama H. Optimal Energy Shaping via Neural Approximators. *SIAM Journal on Applied Dynamical Systems* 2022; 21(3): 2126–2147. doi: 10.1137/21M1414279

37. Paszke A, Gross S, Chintala S, et al. Automatic differentiation in PyTorch.

38. Folkertsma GA, Van Der Schaft AJ, Stramigioli S. Power-continuous synchronisation of oscillators: A novel, energy-free way to synchronise dynamical systems. In: Institute of Electrical and Electronics Engineers Inc.; 2014: 1493–1498

39. Duindam V, Stramigioli S, Scherpen JM. Passive compensation of nonlinear robot dynamics. *IEEE Transactions on Robotics and Automation* 2004; 20(3): 480–487. doi: 10.1109/TRA.2004.824693

40. Poli M, Massaroli S, Yamashita A, Asama H, Ermon S. TorchDyn: Implicit Models and Neural Numerical Methods in PyTorch.

41. Paszke A, Gross S, Massa F, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. 2019.

42. Kingma DP, Lei Ba J. ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION.

APPENDIX

A EQUATIONS OF MOTION FOR DOUBLE PENDULUM

$$\dot{\boldsymbol{q}} = M(\boldsymbol{q})^{-1}\boldsymbol{p} \tag{A1}$$

$$\dot{\boldsymbol{p}} = C(\boldsymbol{q},\boldsymbol{p})\boldsymbol{p} - \frac{\partial}{\partial \boldsymbol{q}} V_{\theta}((q)) - \begin{bmatrix} mdg(2\sin(q_1) + \sin(q_1+q_2)) \\ mdg\sin(q_1+q_2) + k(\pi - 2q_2) \end{bmatrix} \tag{A2}$$

where the inverse mass matrix $M^{-1}(\boldsymbol{q})$ and Coriolis terms $C(\boldsymbol{q},\boldsymbol{p})\boldsymbol{p}$ are given as

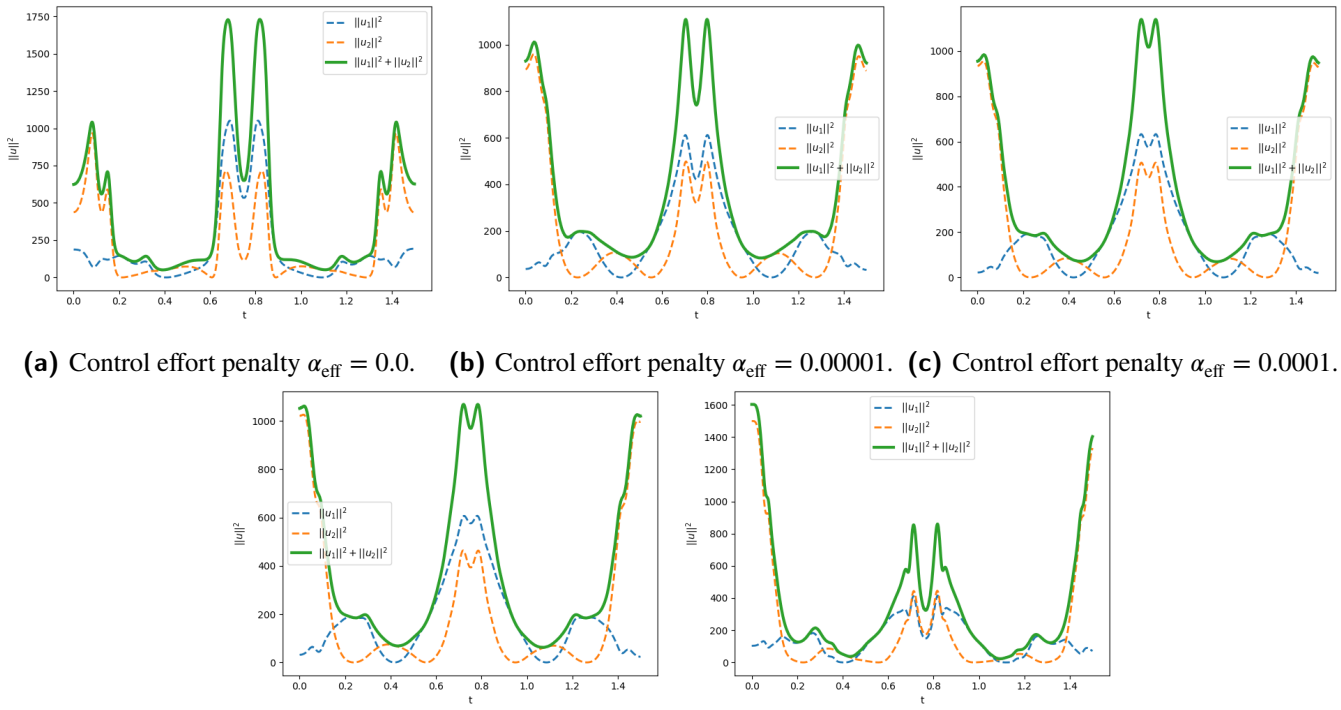
$$M^{-1}((q_1,q_2)) = \frac{1}{md^2(3+2\cos(q_2)-(\cos(q_2)+1)^2)} \begin{bmatrix} 1 & -\cos(q_2)-1 \\ -\cos(q_2)-1 & (3+2\cos(q_2)) \end{bmatrix}, \tag{A3}$$

$$C(\boldsymbol{q},\boldsymbol{p})\boldsymbol{p} = \frac{\sin(q_2)}{2d^2m(1+\sin(q_2)^2)^2} \begin{bmatrix} 0 \\ 2\cos(q_2)p_1^2 - (5+4\cos(q_2)+\cos(2q_2))p_1p_2 + (5+6\cos(q_2)+\cos(2q_2))p_2^2 \end{bmatrix}. \tag{A4}$$

B EIGENMODES FOR DIFFERENT PENALISATION OF THE CONTROL EFFORT AND PERIODS

B.1 Effect of the Control Effort Penalty

In Figure B1, we show the squared control effort $||u_1||^2+||u_2||^2$ derived from the gradient of the learned potential V_θ for different values of the regularization coefficient $\alpha_{\text{eff}} \in \{0.0, 0.00001, 0.0001, 0.001, 0.01\}$. We used this grid-search experiment to find a suitable value for α_{eff}.



(a) Control effort penalty $\alpha_{\text{eff}} = 0.0$. **(b)** Control effort penalty $\alpha_{\text{eff}} = 0.00001$. **(c)** Control effort penalty $\alpha_{\text{eff}} = 0.0001$.

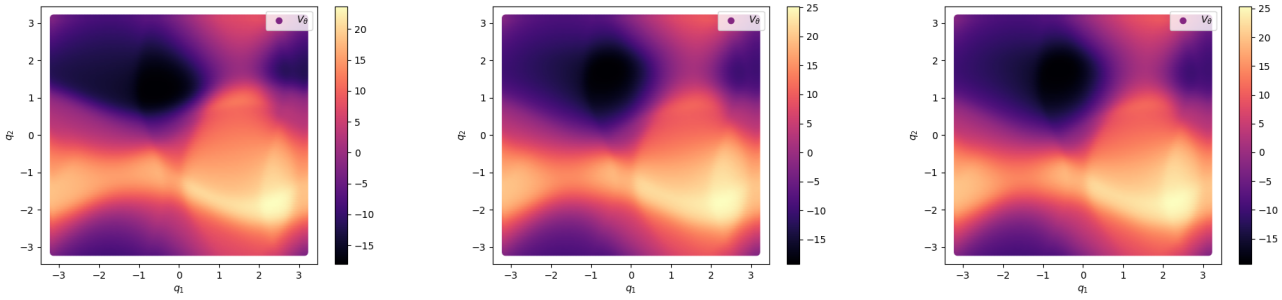
(d) Control effort penalty $\alpha_{\text{eff}} = 0.001$. **(e)** Control effort penalty $\alpha_{\text{eff}} = 0.01$.

FIGURE B1 Control effort squared for different control effort penalty coefficients.

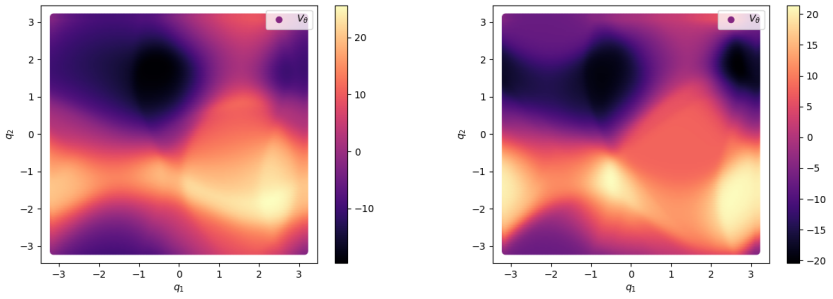
Moreover, for a fixed period $T = 1.5$ s, it is possible to notice from Figure B1 and B2 that the increase of the regularization penalty decreases the control effort (as expected) by improving the smoothness of the potential.

B.2 Learned Eigenmodes for Different Fixed Periods T

In Figure B3, we show the resulting trajectories, learned potentials V_θ, and squared control effort $\boldsymbol{u}$ for different period length T. Our approach is capable of finding eigenmodes for different periods T. It is noticed that the learned potential combines with gravitational and elastic potentials in non trivial ways to steer the system on oscillatory modes with the desired period. In Figure B3q-B3t, the period of oscillation is close to the natural evolution of the system, i.e. when only the gravitation potential is active and no learned potential is present, the learned potential is such that the resulting control effort is extremely small.

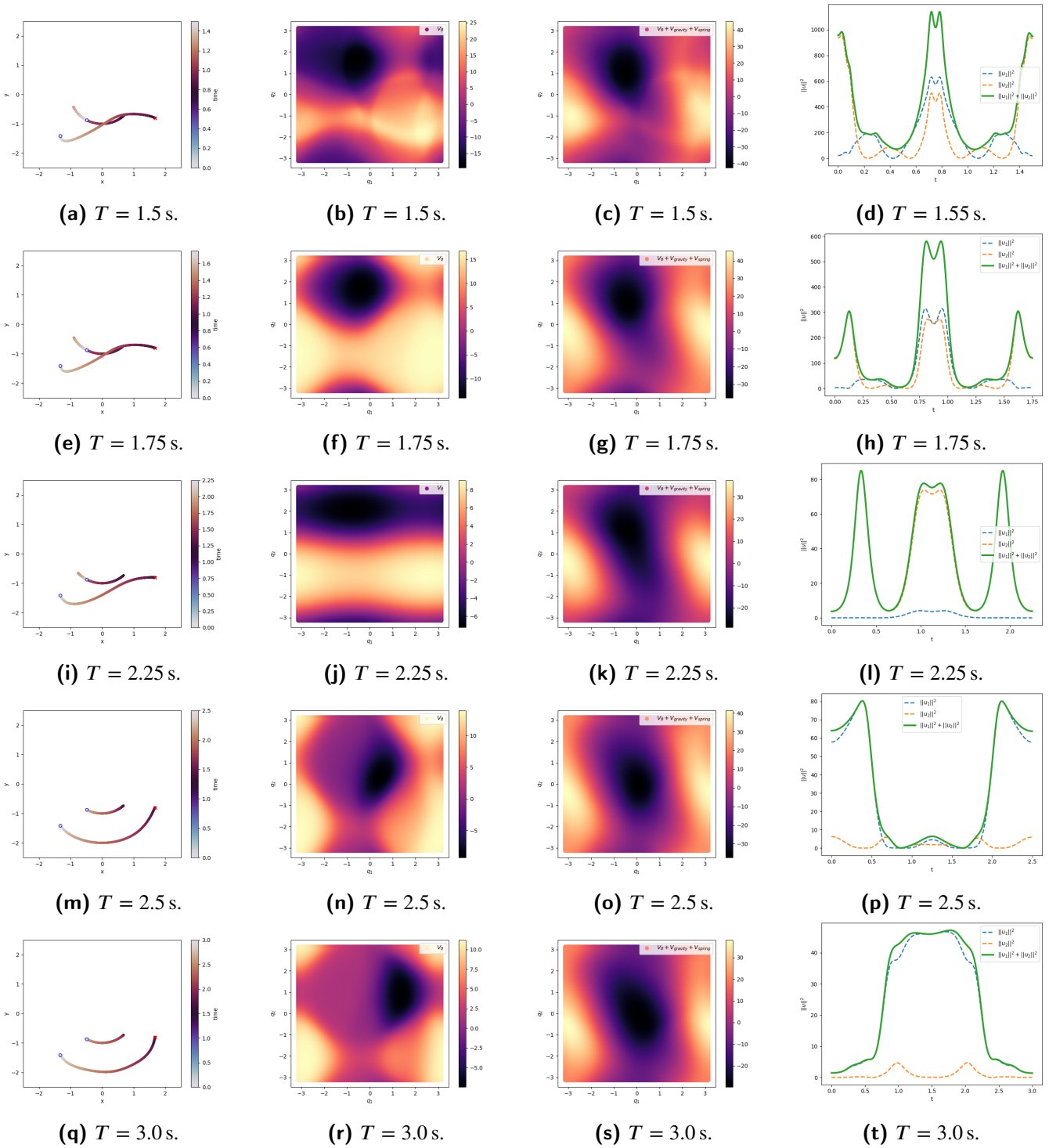


(a) Control effort penalty $\alpha_{\text{eff}} = 0.0$. **(b)** Control effort penalty $\alpha_{\text{eff}} = 0.00001$. **(c)** Control effort penalty $\alpha_{\text{eff}} = 0.0001$.



(d) Control effort penalty $\alpha_{\text{eff}} = 0.001$. **(e)** Control effort penalty $\alpha_{\text{eff}} = 0.01$.

FIGURE B2 Learned potential for different effort penalty coefficients.



(a) $T = 1.5$ s. **(b)** $T = 1.5$ s. **(c)** $T = 1.5$ s. **(d)** $T = 1.55$ s.

(e) $T = 1.75$ s. **(f)** $T = 1.75$ s. **(g)** $T = 1.75$ s. **(h)** $T = 1.75$ s.

(i) $T = 2.25$ s. **(j)** $T = 2.25$ s. **(k)** $T = 2.25$ s. **(l)** $T = 2.25$ s.

(m) $T = 2.5$ s. **(n)** $T = 2.5$ s. **(o)** $T = 2.5$ s. **(p)** $T = 2.5$ s.

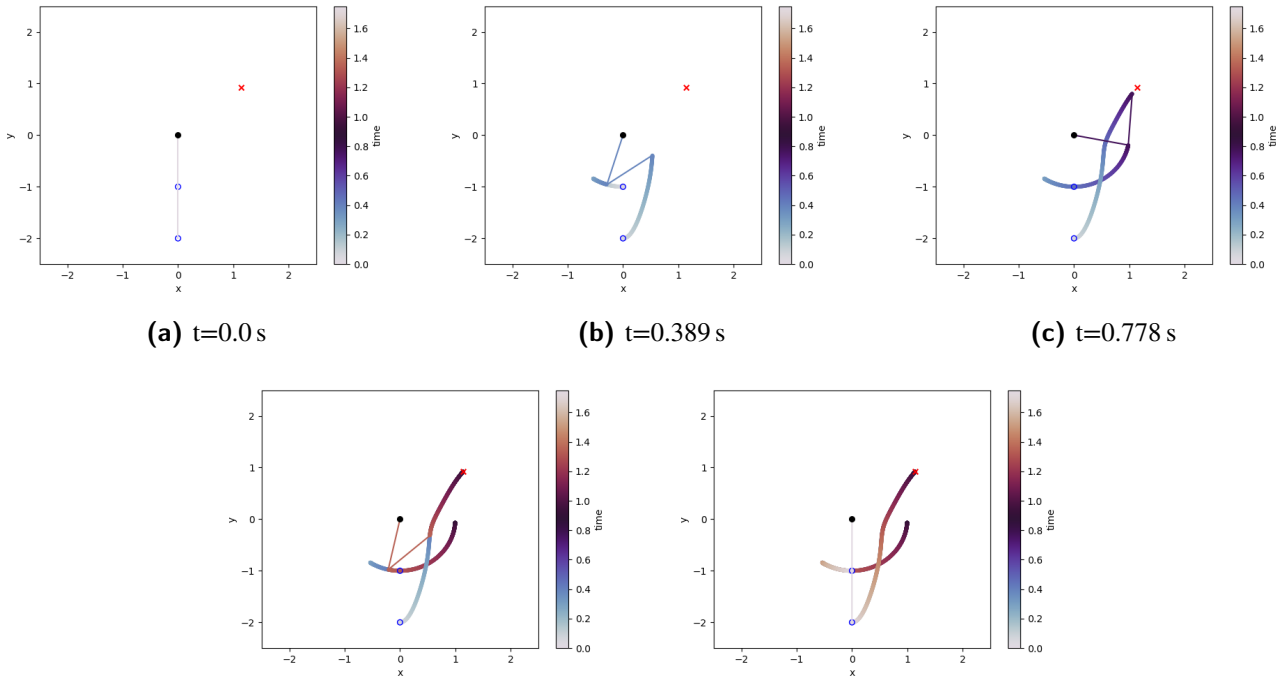
(q) $T = 3.0$ s. **(r)** $T = 3.0$ s. **(s)** $T = 3.0$ s. **(t)** $T = 3.0$ s.

FIGURE B3 Trajectory, learned and total potential, and squared control effort for different period length $T \in \{1.5\,\text{s}, 1.75\,\text{s}, 2.25\,\text{s}, 2.5\,\text{s}, 3.0\,\text{s}\}$.

C EIGENMODES FOR A DIFFERENT TARGET AND INITIAL POSITION

In this appendix, we redo the eigenmode discovery experiment in Section 5 with a different initial position and target. The numerical experiments are done for different values of the fixed period and different values of the control effort regularization $\alpha_{\text{eff}} = 0.0001$.

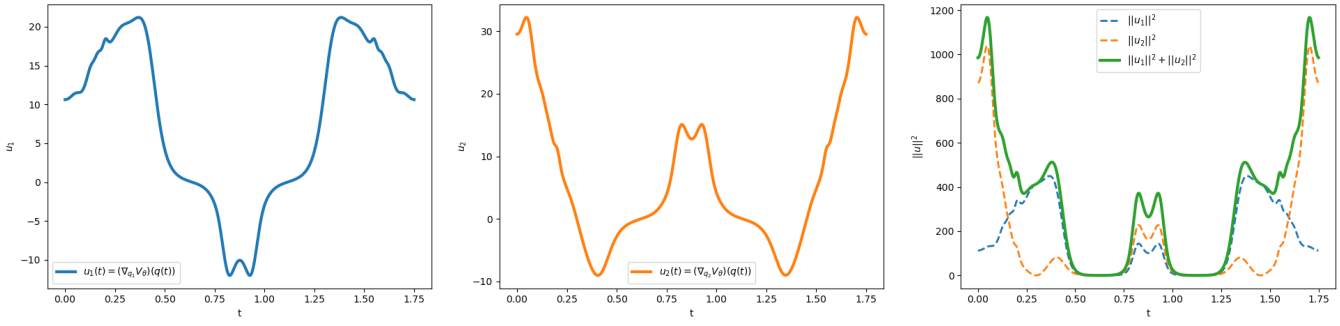
For each value of the period $T \in [1.75, 2.5, 3.0]$s, we show the trajectories of the double pendulum in Figure C4, C8 and C12, the control inputs in Figure C5, C9 and C13, the potentials in Figure C6, C10 and C14, and the state variable over time in Figure C7, C11 and C15, respectively.



(a) t=0.0 s **(b)** t=0.389 s **(c)** t=0.778 s

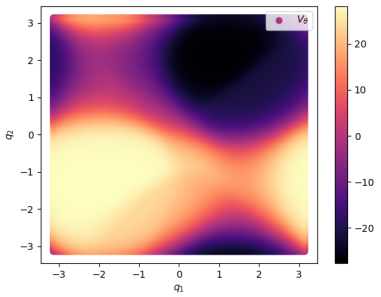
(d) t=1.361 s **(e)** T=1.75 s

FIGURE C4 Eigenmode at different time steps.

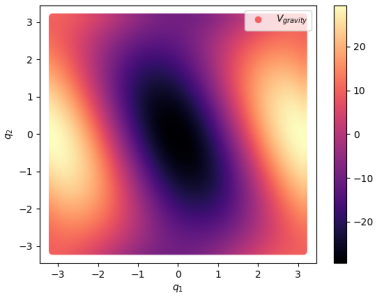


(a) First component of $u(t)$. **(b)** Second component of $u(t)$. **(c)** Squared control effort penalty.

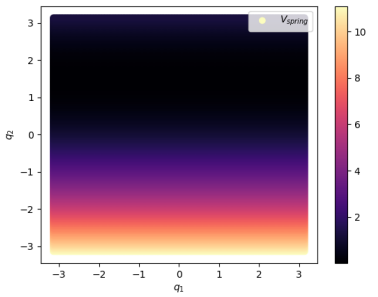
FIGURE C5 Control inputs and control effort.



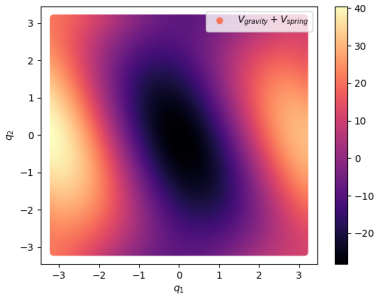
(a) Learned potential V_θ.



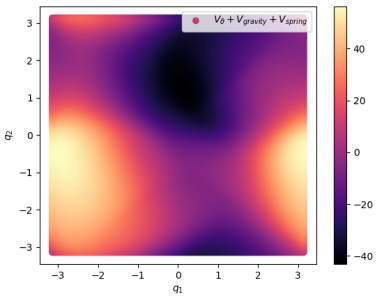
(b) Gravitational potential V_{gravity}.



(c) Spring potential V_{spring}

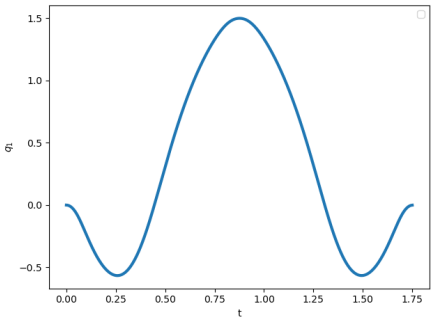


(d) Open-loop potential $V_{\text{spring}} + V_{\text{gravity}}$.

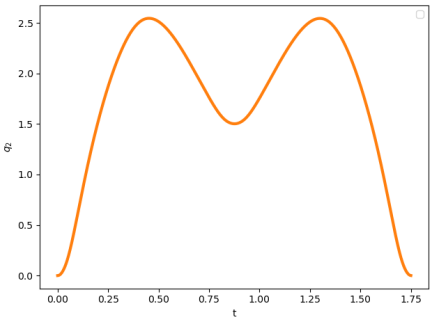


(e) Overall potential $V_\theta + V_{\text{spring}} + V_{\text{gravity}}$.

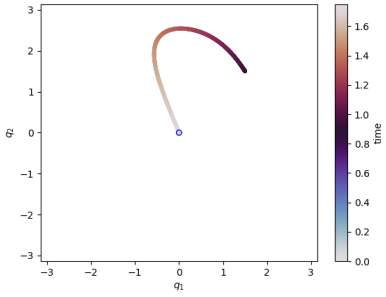
FIGURE C6 Potentials for T=1.75 s over $\boldsymbol{q} \in [-\pi, \pi]$.



(a) q_1 over the period.

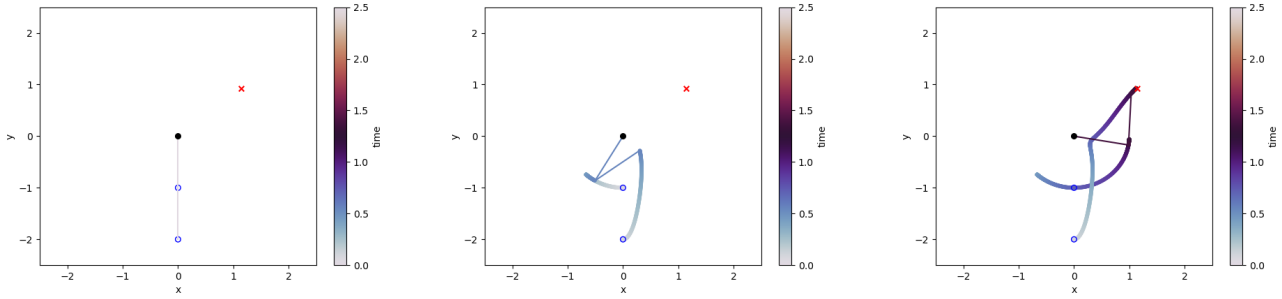


(b) q_2 over the period.

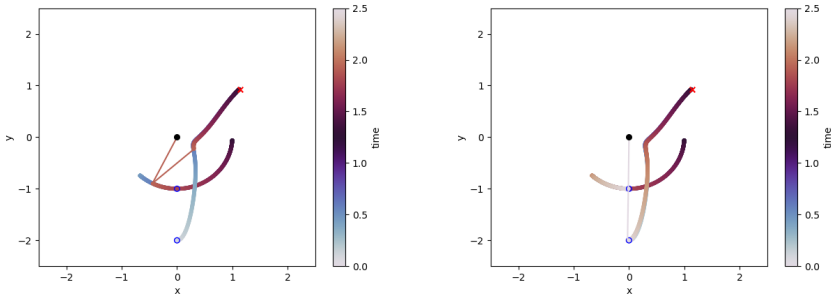


(c) Trajectory in configuration space.

FIGURE C7 The time behavior of the angles q_1 and q_2 over one period.

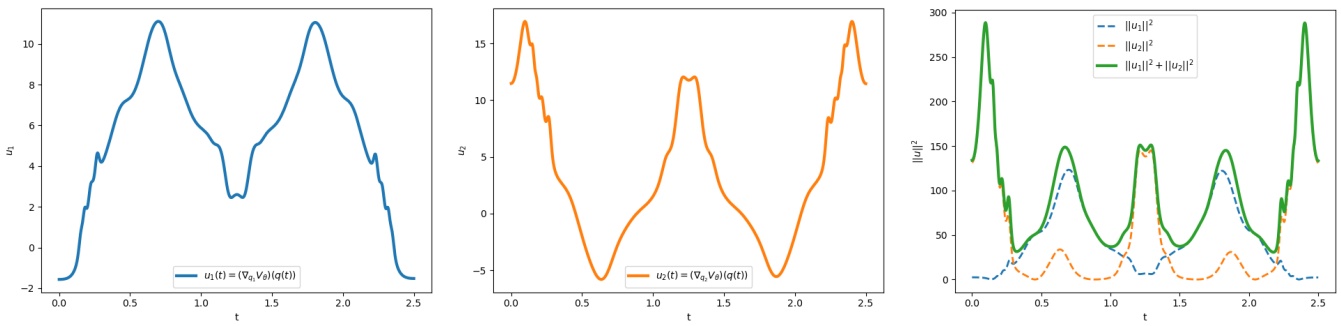


(a) t=0.0 s **(b)** t=0.556 s **(c)** t=1.389 s



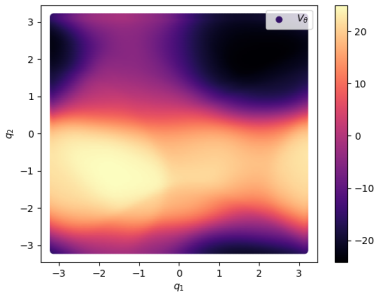
(d) t=1.944 s **(e)** T=2.50 s

FIGURE C8 Eigenmode at different time steps.

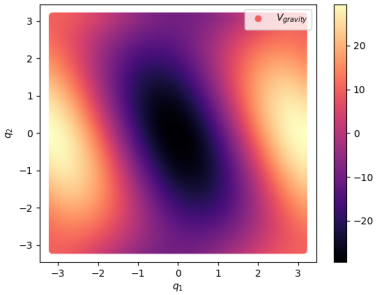


(a) First component of $u(t)$. **(b)** Second component of $u(t)$. **(c)** Squared control effort penalty.

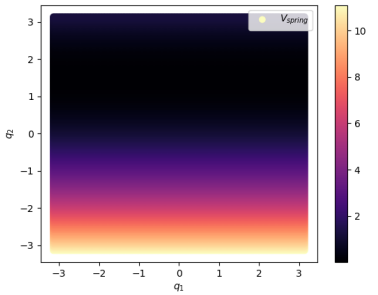
FIGURE C9 Control inputs and control effort.



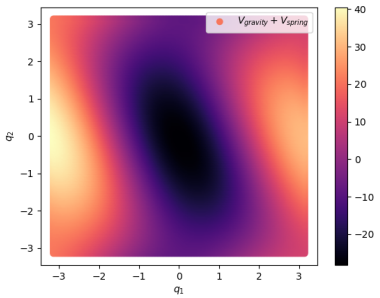
(a) Learned potential V_θ.



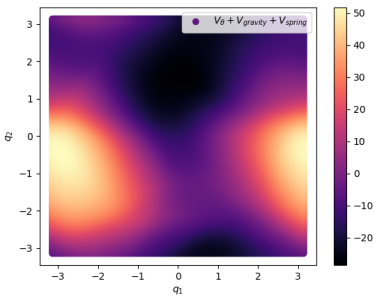
(b) Gravitational potential V_{gravity}.



(c) Spring potential V_{spring}

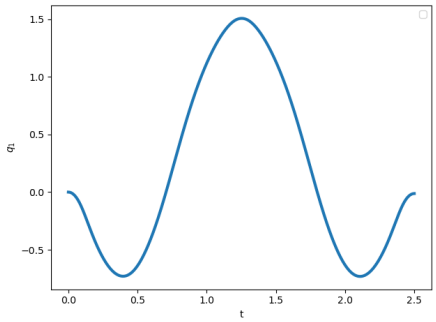


(d) Open-loop potential $V_{\text{spring}} + V_{\text{gravity}}$.

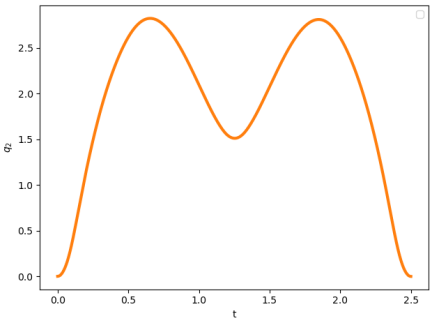


(e) Overall potential $V_\theta + V_{\text{spring}} + V_{\text{gravity}}$.

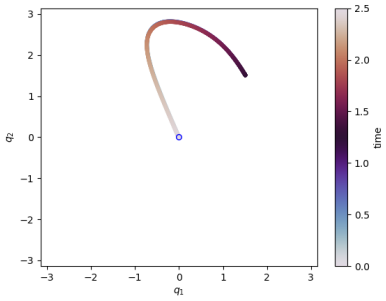
FIGURE C10 Potentials for T=2.50 s over $\boldsymbol{q} \in [-\pi, \pi]$.



(a) q_1 over the period.

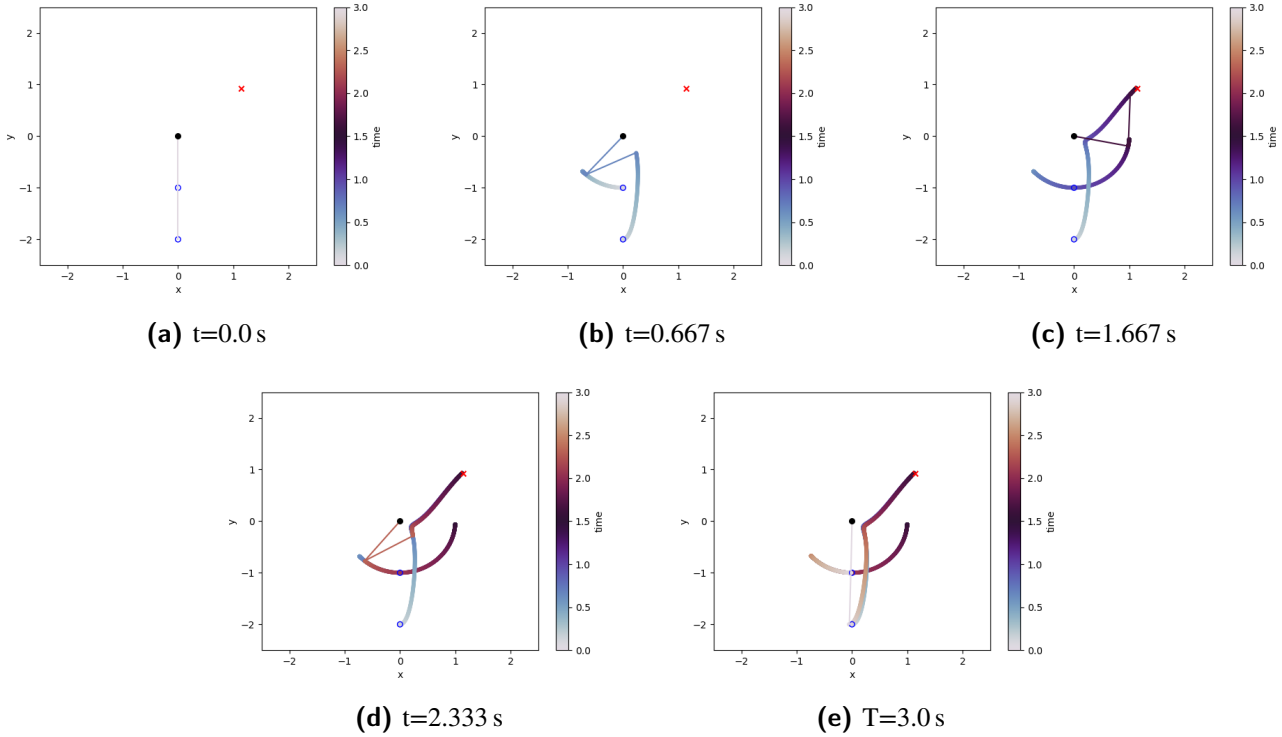


(b) q_2 over the period.



(c) Trajectory in configuration space.

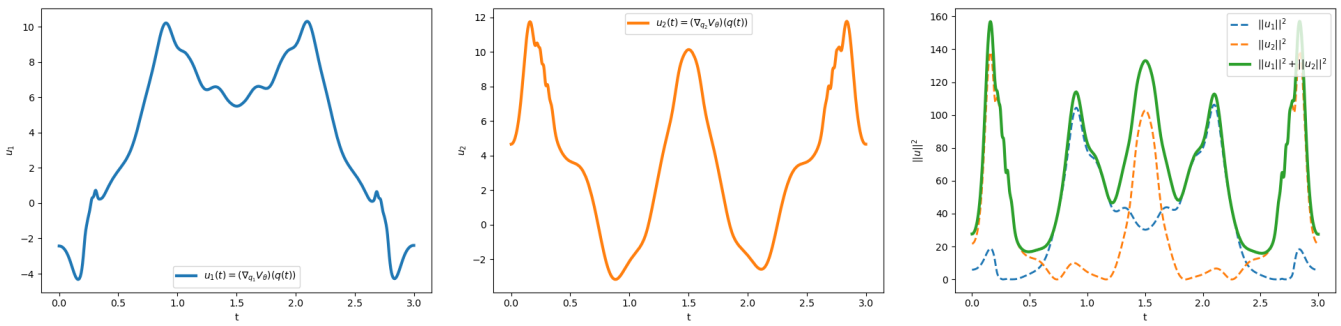
FIGURE C11 The time behavior of the angles q_1 and q_2 over one period.



(a) t=0.0 s **(b)** t=0.667 s **(c)** t=1.667 s

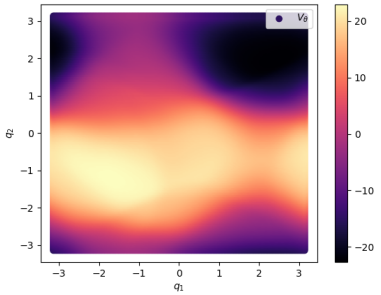
(d) t=2.333 s **(e)** T=3.0 s

FIGURE C12 Eigenmode at different time steps.

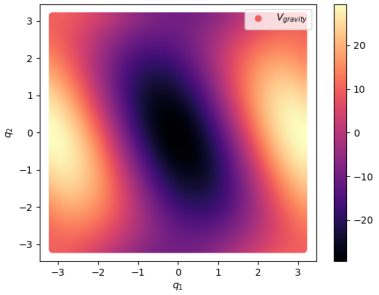


(a) First component of $u(t)$. **(b)** Second component of $u(t)$. **(c)** Squared control effort penalty.

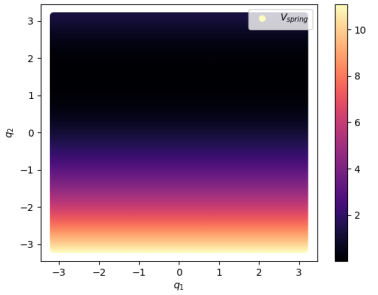
FIGURE C13 Control inputs and control effort.



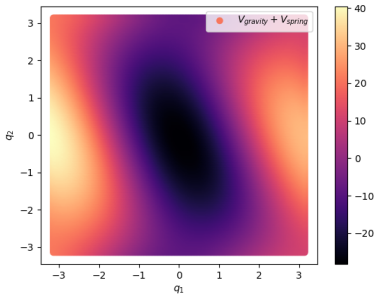
(a) Learned potential V_θ.



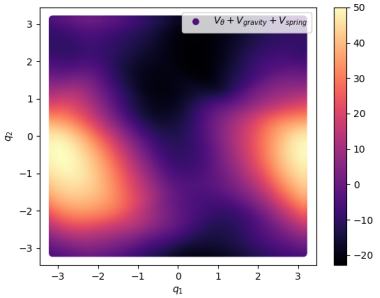
(b) Gravitational potential V_{gravity}.



(c) Spring potential V_{spring}

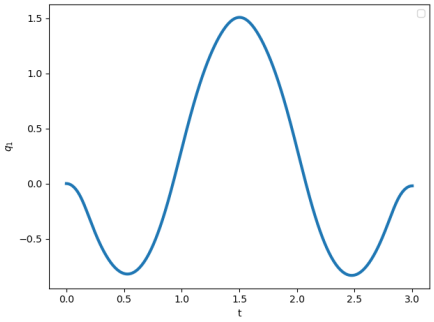


(d) Spring and gravitational potential $V_{\text{spring}} + V_{\text{gravity}}$.

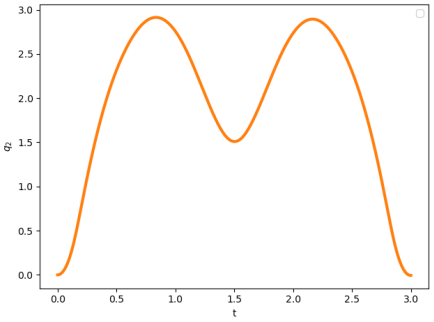


(e) Overall potential $V_\theta + V_{\text{spring}} + V_{\text{gravity}}$.

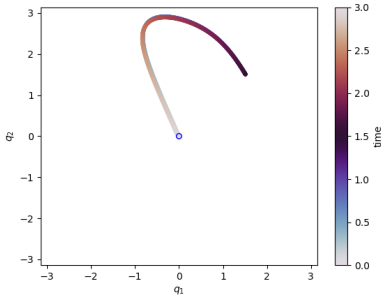
FIGURE C14 Potentials for T=3.0 s over $\boldsymbol{q} \in [-\pi, \pi]$.



(a) q_1 over one period.



(b) q_2 over one period.



(c) Trajectory in configuration space.

FIGURE C15 The time behavior of the angles q_1 and q_2 over one period.

D LEARNING THE PERIOD T JOINTLY WITH THE POTENTIAL

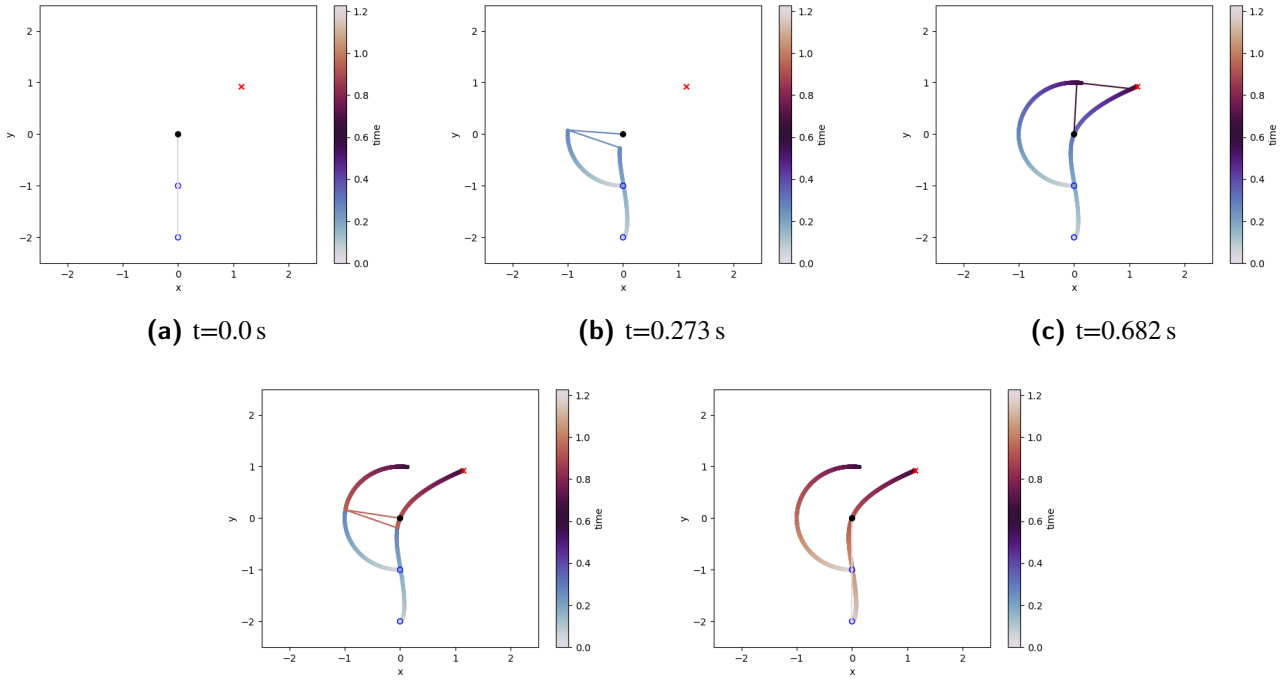
In this appendix, we briefly elaborate on a small extension of the optimization problem in (2). In the optimization problem in (2), the desired period T of the eigenmode is fixed a priori[10]. For the experiments in Section 5, this suffices. However, in some applications it might be necessary to learn a suitable period T of the eigenmode since it might be unknown. As a consequence, we extend the model in (2) to allow for an optimizable period T:

$$\begin{aligned} \min_{\theta,T} \quad & L_{\text{task}}(\boldsymbol{x}) + \beta L_{\text{eigen}}(\boldsymbol{x}) \\ \text{s.t.} \quad & \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{q}(t)\\ \boldsymbol{p}(t)\end{bmatrix} = \begin{bmatrix}0 & \boldsymbol{I}\\ -\boldsymbol{I} & 0\end{bmatrix}\nabla(H+V_\theta)(\boldsymbol{p},\boldsymbol{q}) \quad , \begin{bmatrix}\boldsymbol{p}(0)\\ \boldsymbol{q}(0)\end{bmatrix} = \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{q}_0\end{bmatrix} \end{aligned} \tag{D5}$$

In the remainder of this appendix, we show numerical experiments similar to the experiments in Section 5 but with a learnable period T. We present two such numerical experiments, each with a different pair of initial configuration $\boldsymbol{q}_0$ and target position h^*.

D.1 Results

Similarly to Appendix C, we show the trajectories of the double pendulum in Figure D16 and D20, the control inputs in Figure D17 and D21, the potentials in Figure D18 and D22 and the state variable over time in Figure D19 and D23, for two different initial and final positions.

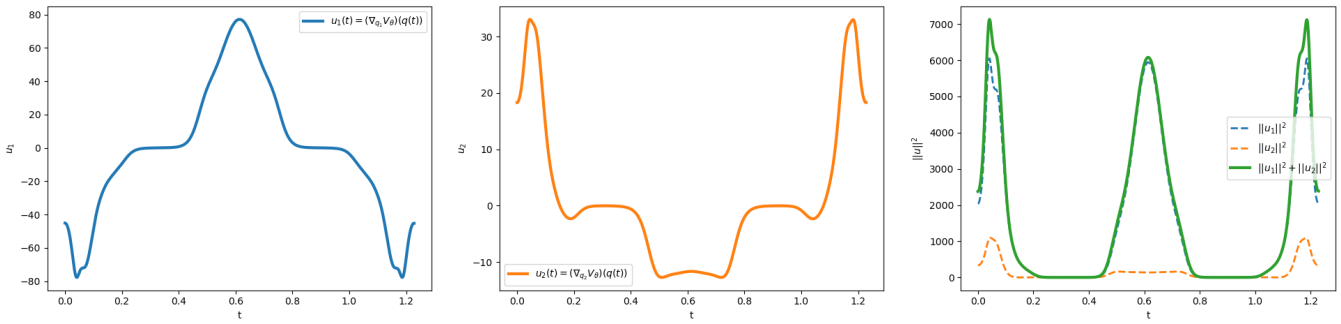


(a) t=0.0 s **(b)** t=0.273 s **(c)** t=0.682 s

(d) t=0.955 s **(e)** T=1.228 s

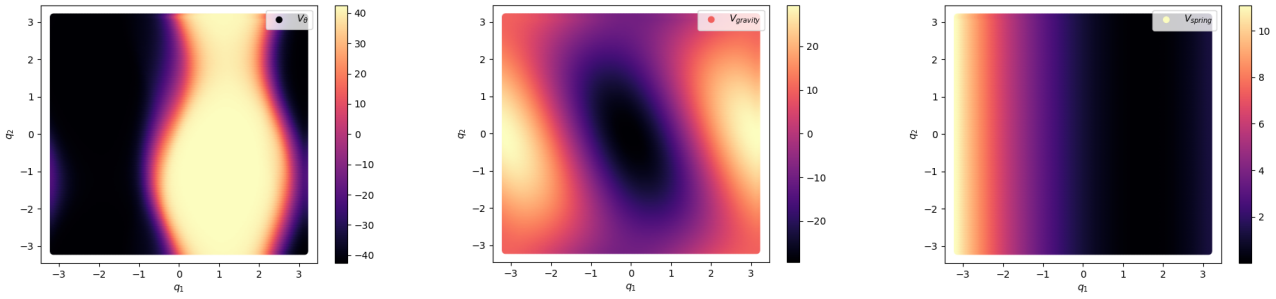
FIGURE D16 Eigenmode at different time steps.

[10] A requirement on the period length is often needed in a pick-and-place task in the context of an automatic machine where timing is crucial.

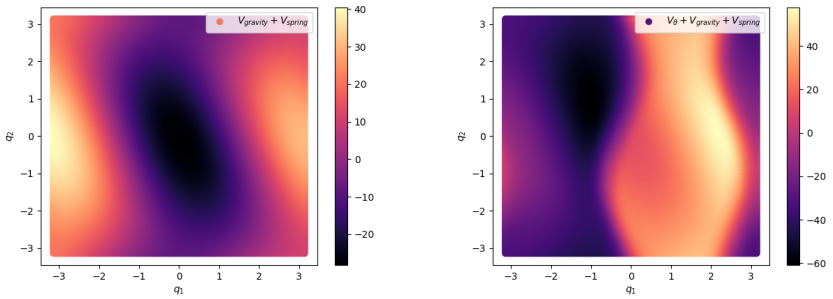


(a) First component of $u(t)$. **(b)** Second component of $u(t)$. **(c)** Squared control effort penalty.

FIGURE D17 Control inputs and control effort.

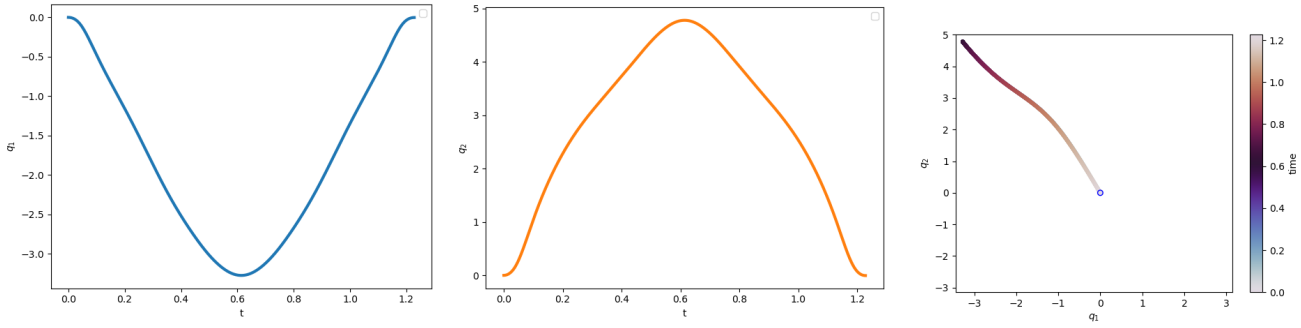


(a) Learned potential V_θ. **(b)** Gravitational potential V_{gravity}. **(c)** Spring potential V_{spring}



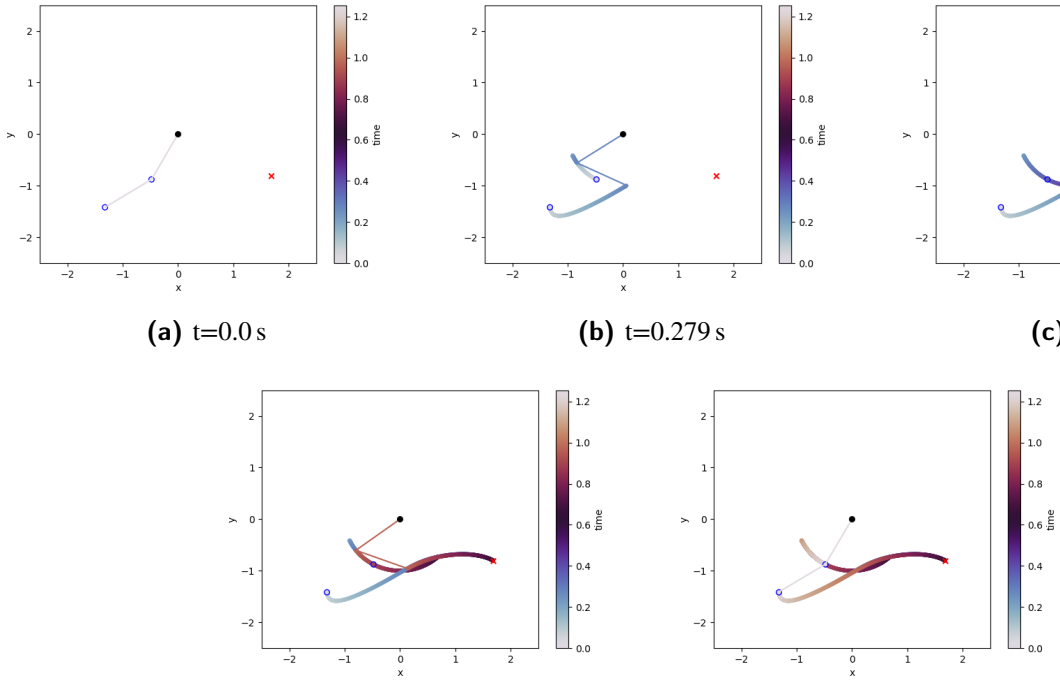
(d) Open-loop potential $V_{\text{spring}} + V_{\text{gravity}}$. **(e)** Overall potential $V_\theta + V_{\text{spring}} + V_{\text{gravity}}$.

FIGURE D18 Potentials for T=1.228 s over $\boldsymbol{q} \in [-\pi, \pi]$.



(a) q_1 over one period. **(b)** q_2 over one period. **(c)** Trajectory in configuration space.

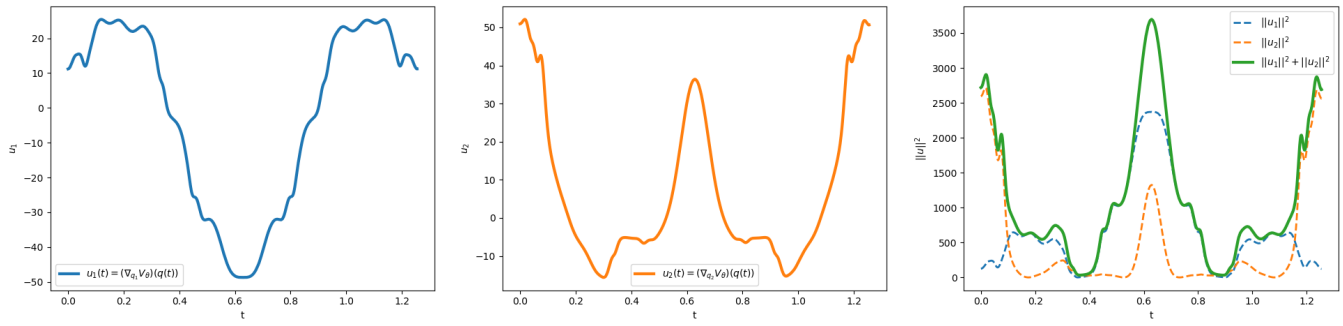
FIGURE D19 The time behavior of the angles q_1 and q_2 over one period.



(a) t=0.0 s **(b)** t=0.279 s **(c)** t=0.697 s

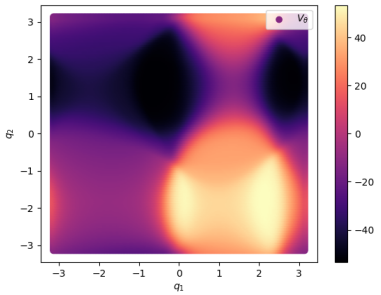
(d) t=0.976 s **(e)** T=1.255 s

FIGURE D20 Eigenmode at different time steps.

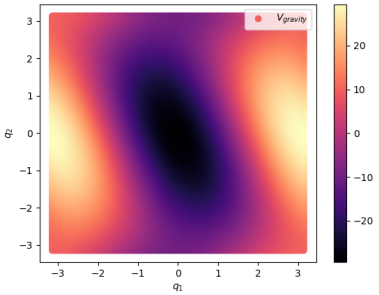


(a) First component of $u(t)$. **(b)** Second component of $u(t)$. **(c)** Squared control effort penalty.

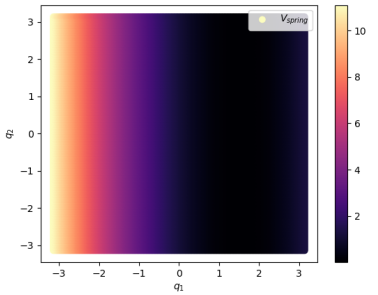
FIGURE D21 Control inputs and control effort.



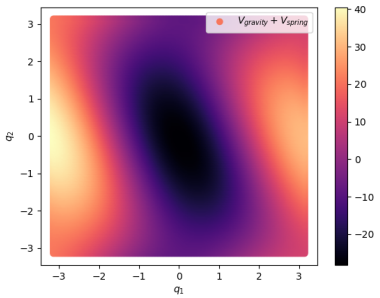
(a) Learned potential V_θ.



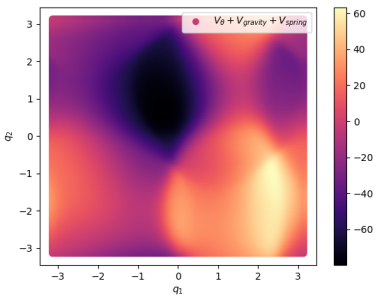
(b) Gravitational potential V_{gravity}.



(c) Spring potential V_{spring}

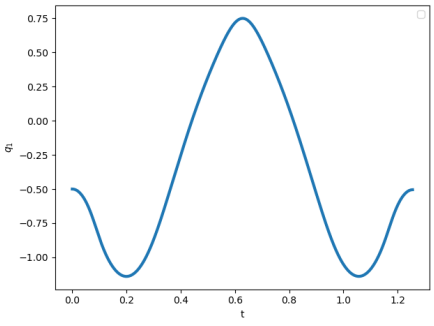


(d) Open-loop potential $V_{\text{spring}} + V_{\text{gravity}}$.

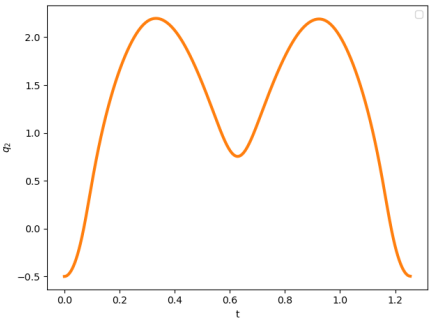


(e) Overall potential $V_\theta + V_{\text{spring}} + V_{\text{gravity}}$.

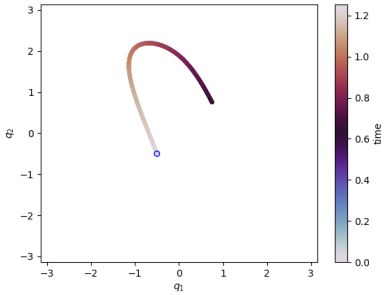
FIGURE D22 Potentials for T=1.255 s over $\boldsymbol{q} \in [-\pi, \pi]$.



(a) q_1 over one period.



(b) q_2 over one period.



(c) Trajectoryin configuration space.

FIGURE D23 The time behavior of the angles q_1 and q_2 over one period.

E IMPLEMENTATION DETAILS

The neural ODE was implemented using the TorchDyn library[40] built on top of PyTorch[41].

The neural network V_θ is composed of an input layer of dimension 2 (in the case of the double pendulum, the two input variables are q_1 and q_2, one hidden layers with 256 neurons, and an output layer of dimension 1 outputting the value of the potential $V_\theta(q_1, q_2)$ for the input pair (q_1, q_2). The choices of hyperparameters are shown in table E1.

Hyperparameter	Value
V_θ input dimension	2
V_θ output dimension	1
V_θ hidden dimension	256
Number of hidden layers	1
Activation function hidden layer	tanh
Activation function output layer	linear
Optimizer	ADAM[42]
Learning rate	1e-3
Training epochs	500
Computation of sensitivity	backpropagation
α_{task}	10
α_{eff}	0.0001
α_{task}	10
λ_1	0.05
α_1	0.0005
λ_2	0.95
β	1
α_M	10
α_E	1
Double pendulum mass link 1	1.0
Double pendulum mass link 2	1.0
Double pendulum length link 1	1.0
Double pendulum length link 2	1.0
Double pendulum spring stiffness joint 1	0.0
Double pendulum spring stiffness joint 2	0.5

TABLE E1 Hyperparameters of the experiments.