EigenMPC: An Eigenmanifold-Inspired Model-Predictive Control Framework for Exciting Efficient Oscillations in Mechanical Systems

Andre Coelho[1,2,3], Alin Albu-Schaeffer[1,4], Arne Sachtler[1,4], Hrishik Mishra[1,5], Davide Bicego[3], Christian Ott[1,5], Antonio Franchi[3,6]

Abstract— This paper proposes a Nonlinear Model-Predictive Control (NMPC) method capable of finding and converging to energy-efficient regular oscillations, which require no control action to be sustained. The approach builds up on the recently developed Eigenmanifold theory, which defines the sets of line-shaped oscillations of a robot as an invariant two-dimensional submanifold of its state space. By defining the control problem as a nonlinear program (NLP), the controller is able to deal with constraints in the state and control variables and be energy-efficient not only in its final trajectory but also during the convergence phase. An initial implementation of this approach is proposed, analyzed, and tested in simulation.

I. INTRODUCTION

In the last three decades, numerous roboticists have devoted their effort to generating energy-efficient robot motion [1], [2]. The developed approaches are especially useful to render cyclic motions, like walking gaits and pick-and-place trajectories. In most cases, mechanisms and control laws are designed to make use of the shape of the gravitational or elastic potential in order to achieve maneuvers requiring little to no control effort to be sustained.

In that scope, recent works have extended the concept of normal modes to nonlinearly-coupled mechanical systems [3], [4]. In particular, [4] has demonstrated a way of finding sets of invariant line-shaped oscillations, called eigenmodes (see Fig. 1), which belong to a two-dimensional submanifold of the state space: the eigenmanifold. The two properties of eigenmodes, namely invariance and line shape, are of paramount importance for robotic tasks like pick-and-place and locomotion. The former guarantees that no energy injection is needed to sustain a gait, while the latter ensures the oscillations happen between two static poses, which could be for instance the grasping and releasing poses or the leg-lift and ground-contact ones.

Once such regular oscillation modes are found, proper controllers should be designed in order to drive mechanical systems from their initial state onto the eigenmanifold. In [5], a chart to the eigenmanifold is defined and the controller acts along the directions normal to a desired eigenmode to
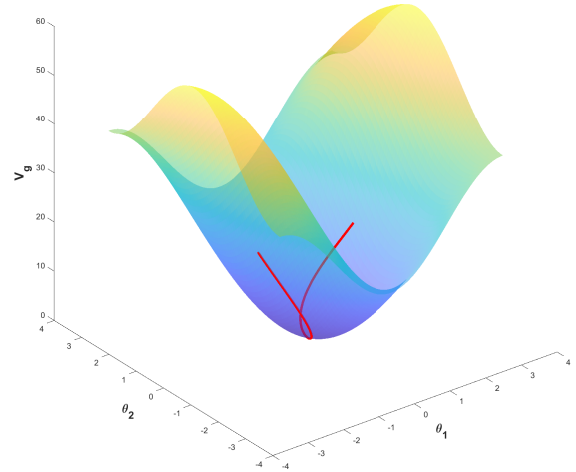

Fig. 1: An eigenmode (red line) on the potential energy surface.

achieve convergence. Although the aforementioned seminal works showed that energy injection is not required to remain on the eigenmanifold, the bounds of the control action needed to converge to them has thus far not been considered. Therefore, such eigenmanifold-stabilization techniques might fail to drive mechanical systems with limited actuation to eigenmodes of arbitrary energy. An example thereof is the DLR Suspended Aerial Manipulator (SAM) [6], which, for safety and energy-efficiency reasons, hangs from a carrier in a pendulum-like setup. There, the mounted propellers are able to stabilize the system around its hanging point (see [7]), but not to bring it to arbitrary poses in a static way. On the other hand, proper control design may allow the SAM to reach high-amplitude oscillations and subsequently perform dynamic grasping or perching maneuvers despite its actuation limitations.

In contrast to state-of-the-art eigenmanifold-stabilization methods, Nonlinear Model Predictive Control (NMPC) [8] – as an optimization-based framework – is able to deal with constraints of different sorts in the control signal, including propeller-thrust limitations (see [9]). In addition, some versions of NMPC (e. g., Economic NMPC [8, Sec. 7.10]) have also achieved periodic orbit stabilization. However, there is no guarantee that the achieved orbits will present the aforementioned properties of eigenmodes, namely invariance (or zero control) and line shape.

In light of that, we propose a novel NMPC framework, which, endowed with knowledge from the eigenmanifold theory, is able to converge to energy-efficient line-shaped

[1]Institute of Robotics and Mechatronics of the German Aerospace Center (DLR), Oberpfaffenhofen, Germany.
[2]Dextrous Robotics Inc., Memphis, TN, USA.
[3]Robotics and Mechatronics Lab, Faculty of Electrical Engineering, Mathematics & Computer Science, University of Twente, Netherlands.
[4]Department of Informatics, Technical University of Munich, Germany.
[5]Automation and Control Institute (ACIN), Technical University of Vienna, Austria.
[6]LAAS-CNRS, University of Toulouse, France.
andre@dextrousrobotics.com

oscillatory trajectories on the eigenmanifolds. The proposed approach can deal with actuation bounds while still being able to converge to regular oscillations in an close-to-optimal manner. Moreover, a heuristic version of the framework which does not require previous computation of the eigenmanifold is also introduced. Rather, an estimate of the direction of the mode – given by its associated eigenvector – is sufficient for the online search and convergence process.

The controller efficacy and performance – both with and without previous knowledge of the eigenmode – are demonstrated through its application to the simple but insightful example of the torque-limited double pendulum. Simulation results confirm that the controller is able to drive the system to trajectories that are almost identical to the eigenmodes found with [4], even when starting from significantly large initial conditions.

II. GEOMETRIC MECHANICS AND EIGENMODES

Recent development in the field of oscillatory normal modes has achieved a generalization of linear modes to non-linear, elastically [3] and inertially-coupled [4] mechanical systems. In particular, by finding invariant line-shaped curves in the state-space, called eigenmodes, the framework presented in [4] is able to realize sustained regular oscillations in otherwise chaotic systems (e. g., robots). In that scope, this section aims at presenting important aspects of [4], upon which the EigenMPC approach is built.

A. Definitions

Definition 1 (Forced simple mechanical system): A **forced simple mechanical system** [10] is a tuple $(\mathsf{T}\mathcal{Q}, \mathbb{G}, \mathsf{V}, \mathsf{F})$, where $\mathsf{T}\mathcal{Q}$ is the tangent-bundle manifold containing all possible trajectories of the system, $\mathbb{G}$ is the Riemannian metric induced by the system's inertia, $\mathsf{V} : \mathcal{Q} \to \mathbb{R}^+ \cup \{0\}$ is a potential function, and $\mathsf{F} : \mathbb{R} \times \mathsf{T}\mathcal{Q} \to \mathsf{T}^*\mathcal{Q}$ is the function of control and external wrenches, where $\mathsf{T}^*\mathcal{Q}$ is the cotangent bundle.

The trajectories of such systems, $\gamma(t) \in \mathcal{Q}$, follow the so-called forced geodesic equation:

$$\overset{\mathbb{G}}{\nabla}_{\dot{\gamma}(t)} \dot{\gamma}(t) = -\mathrm{grad}\mathsf{V}(\gamma(t)) + \mathbb{G}^\sharp \left(\mathsf{F}\left(t, \gamma(t), \dot{\gamma}(t)\right)\right), \quad (1)$$

where $\mathbb{G}^\sharp$ is the sharp map, which transforms a force into an acceleration according to the $\mathbb{G}$ metric. In local coordinates $(x, \dot{x}) \in \mathbb{R}^n \times \mathbb{R}^n$ of the manifold $\mathsf{T}\mathcal{Q}$, locally representing its generalized position and velocity, Eq. 1 translates into the well-known manipulator equation

$$\ddot{x} = M(x)^{-1} \left(-C(x, \dot{x})\dot{x} - \frac{\partial V(x)}{\partial x} + F(t, x, \dot{x}) \right), \quad (2)$$

where $F(t, x, \dot{x})$ are the control and external forces, and $\partial V(x)/\partial x$ are forces induced by the potential V, which are usually either elastic and/or gravitational forces.

Definition 2 (Energy): Given a forced mechanical system with $\mathsf{F} \equiv 0$ we locally define its energy as

$$E(x, \dot{x}) = \frac{1}{2}\dot{x}^\top M(x)\dot{x} + V(x), \quad (3)$$

such that there exists a stable equilibrium point $(x_{eq}, 0) \in \mathbb{R}^n \times \mathbb{R}^n$ which is a minimum for $E(x, \dot{x})$.

Definition 3 (Linear modes): Let $\dot{y} = Ay$ be the linearized version of (2) around a stable equilibrium $(x, \dot{x}) = (x_{eq}, 0)$, $F = 0$ with $y = \begin{bmatrix} x^\top & \dot{x}^\top \end{bmatrix}^\top \in \mathbb{R}^{2n}$. Then, there exists n two-dimensional invariant subspaces $ES \simeq \mathbb{R}^{2n}$, with structure

$$ES = \mathrm{span}\left((c, 0), (0, c)\right), \quad (4)$$

where $c \in \mathbb{R}^n$ are unit vectors in the local chart of $\mathcal{Q}$. Such spaces are called eigenspaces. Moreover, oscillations along the directions c are called **linear modes**.

Definition 4 (Forced geodesic flow): Let $\Phi^S_{t_0, t_f}(q, v_q)$ be the maximal integral curve of a forced simple mechanical system starting at $(q, v_q) \in \mathsf{T}\mathcal{Q}$ at time t_0 and ending at time t_f. The projection $\pi_{\mathcal{Q}} \circ \Phi^S_{t_0, t_f}(q, v_q)$ of the integral curve onto $\mathcal{Q}$ is called the **forced geodesic flow**. In other words, the forced geodesic flow is the path in q the system travels starting at (q, v_q) at $t = t_0$ and ending at $t = t_f$ when $\mathsf{F} \equiv 0$.

Definition 5 (Eigenmodes and eigenmanifold): Let

$$\gamma_{\mathfrak{M}}(q_0, t) \triangleq \Phi^S_{0,t}(q_0, 0), \, q_0 \in \mathcal{Q}, t \in \mathbb{R}, \quad (5)$$

be curves in $\mathsf{T}\mathcal{Q}$ such that:
  (i) $\exists T \in \mathbb{R}^+$, s.t $\gamma_{\mathfrak{M}}(q_0, t) = \gamma_{\mathfrak{M}}(q_0, t + kT)$, $\forall k \in \mathbb{Z}$, and
  (ii) the map $O : \pi_{\mathcal{Q}} \circ \gamma_{\mathfrak{M}}(q_0, t) \to [0, 1]$ exists and is a diffeomorphism.

Then, $\gamma_{\mathfrak{M}}$ are called the **eigenmodes** of $(\mathsf{T}\mathcal{Q}, \mathbb{G}, \mathsf{V}, \mathsf{F})$ and the manifold that collects all $\gamma_{\mathfrak{M}}$ from the same family is called the **eigenmanifold**, denoted $\mathfrak{M}$.

In other words, $\mathfrak{M}$ is an invariant two-dimensional manifold corresponding to the line-shaped periodic trajectories (eigenmodes) of $(\mathsf{T}\mathcal{Q}, \mathbb{G}, \mathsf{V}, \mathsf{F})$ when $\mathsf{F} \equiv 0$.

Finally, as shown in [4], it is possible to find local charts $X : \mathbb{R}^2 \to \mathbb{R}^n$ and $\dot{X} : \mathbb{R}^2 \to \mathbb{R}^n$ with coordinates $(x_m, \dot{x}_m) \in \mathbb{R}^2$ such that $X(x_m, \dot{x}_m) = x$ and $\dot{X}(x_m, \dot{x}_m) = \dot{x}$ for $(x, \dot{x}) \in \mathfrak{M}$. Moreover, for linear systems, $(x_m, \dot{x}_m) = (c^\top x, c^\top \dot{x})$, and $(c^\top X, c^\top \dot{X})$ is the identity function.

B. Energy as similarity function

Due to the periodicity of $\gamma_{\mathfrak{M}}(q_0, t)$, we may define an equivalence relation to drop its dependency on t, calling it $\gamma_{\mathfrak{M}}(q_0)$, i.e., disregarding the number of periods of a trajectory and focusing only on its image in the state space as a set. It is also easy to conclude that due to the absence of energy injecting or dissipating forces F, every point in $\gamma_{\mathfrak{M}}(q_0)$ belongs to the same level set of the energy function (3), which is uniquely defined by the energy of the point $(q_0, 0) \in \mathsf{T}\mathcal{Q}$.

In addition, [4] defines a relationship of continuity between points $(q_0, 0) \in \mathsf{T}\mathcal{Q}$ including $(q_{eq}, 0)$ whose forced geodesic flow generates the eigenmodes, called the generator set $\mathfrak{R}$. This means that, in an arbitrarily small neighborhood of every point in $\mathfrak{R}$, there are other points with different energy levels, which also belong to $\mathfrak{R}$ and, consequently, to an eigenmode of the mechanical system. Through this continuity relationship, the closer a point $(q_0, 0) \in \mathfrak{R}$ is to the equilibrium $(q_{eq}, 0)$, the more the eigenmode associated with

it will resemble one of the straight modes of the linearized version of the system. Thus, we can define the energy of an eigenmode $\gamma_{\mathfrak{M}}(q_0)$ as its degree of similarity with an eigenvector. This property will be used in Section IV-B to endow the EigenMPC framework with online eigenmode-search capabilities. The idea is to penalize the distance to the eigenvector for low energies and gradually relax the penalization as energy increases.

III. NONLINEAR MODEL PREDICTIVE CONTROL

Nonlinear Model-Predictive Control (NMPC) is an optimization-based controller which acts as described below (see [8, Algorithm 3.1]).

Algorithm 1 (NMPC): At each sample $t_n \in [t_0,..,t_{N-1}]$, where N is the length of the prediction horizon, follow the steps:

1) Measure the state $z_n \triangleq z(t_n) \in \mathcal{Z}$.
2) Set $z_0 := z_n$ and solve

$$\min_{u_k} J_N \triangleq \sum_{k=0}^{N-1} \ell(z_u(t_k,z_0),u(t_k)) \qquad (6a)$$

$$\text{s.t. } 0 = z_{k+1} - \phi_k(z_k,u_k), \quad k=0,1,\ldots,N-1, \quad (6b)$$

$$\underline{r}_k \leq r_k(z_k,u_k) \leq \bar{r}_k, \quad k=0,1,\ldots,N-1,, \quad (6c)$$

where $u_k \triangleq u(t_k)$ is the control variable and $z_k \triangleq z_u(t_k,z_0)$ is the open-loop predicted state when u_k is applied. Moreover, ℓ and J_N are called the running and the finite-horizon cost functions, respectively, $\phi_k(z_k,u_k)$ is a one-step state-transition function and (6c) defines lower and upper bounds to a function $r(z_k,u_k)$ of the state and control variables.

3) Denote u^* as the optimal solution and use $u^*(t_0)$ as the control action for the next sampling period.

IV. THE EIGENMPC FRAMEWORK

Based on the eigenmanifold theory and NMPC, we propose a framework that allows constrained mechanical systems to be energy efficient in both converging to and sustaining regular oscillations with different levels of *a priori* knowledge. We call this framework *EigenMPC*: Eigenmanifold-Inspired Model Predictive Control.

The action of EigenMPC is twofold: (a) converging to a pre-computed eigenmode; (b) finding an approximation of the eigenmode and converging to it. Hereafter, we call (a) the curved version and (b) the straight version.

A. Curved EigenMPC

In case the eigenmodes have been pre-computed (e.g., with [4]), the curved version of EigenMPC is able to stabilize the system to it in a close-to-optimal manner while taking possible constraints into account. The controller is defined as follows:

Definition 6 (Curved EigenMPC): Let the chart defined by $X(x_m)$ and $\dot{X}(x_m,\dot{x}_m)$ yield an approximation of a desired eigenmode in $\mathfrak{M}$, with $c(x_m)$ being the unit-norm tangent vector of $X(x_m)$ at x_m. We define the curved version of

EigenMPC as the NMPC algorithm with the following running cost:

$$\ell(z,F) = z^\top W_z z + F^\top W_F F, \qquad (7)$$

$$z = \begin{bmatrix} E_{ref} - E \\ c_\perp(x_m)(x - X(x_m)) \\ c_\perp(x_m)(\dot{x} - \dot{X}(x_m,\dot{x}_m)) \end{bmatrix}, \qquad (8)$$

where $c_\perp(x_m)(x - X(x_m))$ and $c_\perp(x_m)(\dot{x} - \dot{X}(x_m,\dot{x}_m))$ are a measure of the distance of x and $\dot{x}$ to the eigenmode along the normal of its tangent vector, given by the multiplication with $c_\perp(x_m) \triangleq I - c(x_m)c(x_m)^\top$. Moreover, the matrices $W_z = \text{diag}(w_E, w_x, w_x, w_{\dot{x}}, w_{\dot{x}}) \in \mathbb{R}^{2n+1}$ and $W_F \in \mathbb{R}^n$ are positive definite.

To show that the proposed method is able to locally drive the system (2) to the desired eigenmode, we start by noting that (7) is a proper cost function for the desired eigenmode, i.e, $\ell = 0$ when the system moves on the eigenmanifold and $\ell > 0$ otherwise. Moreover, we highlight that the control problem at stake is that of a convergence to a set, where $z = 0$, rather than a trajectory-tracking problem.

In sequence, we use the controller defined in [5] as a baseline and note that its inertia-shaping characteristic removes the need for inertia-weighting in the Lyapunov function. Therefore, for that controller, asymptotic stability can be asserted by using a Lyapunov function with constant diagonal weights like $V(z) = z^\top W_z z$, which is exactly the first term in (7).

Now, we denote ℓ_{PD} and $V_{PD}(z) = z^\top W_z z$ the running cost and its state-related part when the control action from [5], denoted F_{PD}, is applied. We note that asymptotic stability of V_{PD} implies that for some functions $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$,

$$\alpha_1(V_{PD}(z_0)) \leq V_{PD}(z) \leq \alpha_2(V_{PD}(z_0)), \qquad (9)$$

for a connected set of initial conditions $z_0 \in \mathcal{Z}_0 \subseteq \mathcal{Z}$ containing the origin.

In addition, by using the fact that $X, \dot{X}$ and E_{ref} parameterize an eigenmode, which by Def. 5 requires no control force to be sustained, we get that $z \to 0$ implies $F_{PD} \to 0$, which together with (9) implies (local) asymptotic controllability of the system with the small control property (see [8, Def. 4.2]), which states that for each $z \in \mathcal{Z}_0$ there exists F (in this case F_{PD}) and $\beta \in \mathcal{KL}$ such that

$$\ell(z,F) \leq \beta(V_{PD}(z_0),n). \qquad (10)$$

Now, we define

$$\ell^*(z) \triangleq \inf_{F \in \mathbb{R}^n} \ell(z,F). \qquad (11)$$

From the fact that $\ell^*(z) \leq \ell(z,F)$, from its quadratic nature and from the small control property (10), we get

$$\alpha_3(V_{PD}(z)) \leq \ell^*(z) \leq \alpha_4(V_{PD}(z)). \qquad (12)$$

If in addition, there exists a $\beta \in \mathcal{KL}_0$, which is linear in its first argument and summable (see [8, Ass. 6.4]), such that, for each $z \in \mathcal{Z}_0$ and for some F,
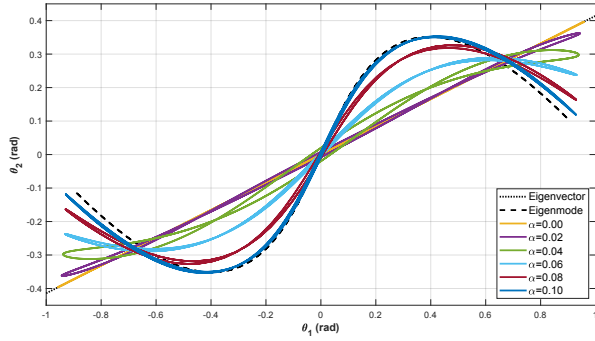
$$\ell(z,F) \leq \beta(\ell^*(z),n). \qquad (13)$$

Fig. 2: Final trajectory of an EigenMPC-controlled double pendulum for different values of α. As α increases, the final trajectory moves from a straight line along the linear mode (dotted line) to a curved one near the actual eigenmode (dashed line).

Then, from the principle of relaxed dynamic programming, we use the result from [8, Th. 6.21] to state that the nominal NMPC closed-loop system with dynamics as (2) and NMPC-feedback is asymptotically stable on $\mathcal{Z}_0$ provided N is sufficiently large.

Although hard to verify in practice, the assumption on the linearity of β seems reasonable for systems like (2), which are feedback linearizable. Further evidence on the convergence capabilities of the approach will be presented in Section V.

Note that by using the controller from [5] as a baseline, the stability analysis is confined to the region of the state-space $\mathcal{Z}_0$ where the that controller is also stable. However, by noting that an infinite-horizon NMPC would find the optimal solution in terms of (6a) and (7) and that the suboptimality number $\alpha \in (0.1]$ of NMPC (see [8, Th. 4.11]) increases with the size of its horizon, we argue that for sufficiently large N, the region of attraction of EigenMPC should be even larger than that of the controller from [5] and its solution closer to the optimal one.

Moreover, while constraints were not directly accounted for in the analysis, their effect is the following: for bounds on the control action, a connected set $\mathcal{Z}_\mathcal{U} \subseteq \mathcal{Z}_0$ around the eigenmanifold will be attractive, given the zero effort needed to sustain an eigenmode and the continuity of the state space. On the other hand, in case the state is constrained with unconstrained control, a set $\bar{\mathcal{Z}} \subseteq \mathcal{Z}_0$ will be attractive given that the eigenmode itself lies inside the viable set. Finally, in case both state and control are constrained, a set of viable initial states $\bar{\mathcal{Z}}_\mathcal{U} \subseteq \mathcal{Z}_0$ can be attractive. However, there might be cases where the control action is not strong enough to drive the system to an eigenmode without exiting the set of viable states. In this case, the NMPC problem is called inviable (see [8, Ass. 3.3]).

B. Straight EigenMPC

While the curved version of EigenMPC has well-defined stability properties, one might benefit from a more heuristic controller, which does not require the pre-computation of the eigenmanifold. This version, called straight EigenMPC is designed as to require only the direction of its associated eigenvector as *a priori* information.

The online-searching capabilities of EigenMPC are built on the fact that the smaller the energy of an eigenmode of a nonlinear system – assuming $E(q_{eq}, 0) = 0$ –, the more it resembles an eigenvector of its linearized counterpart. Therefore, we define the straight EigenMPC program as follows:

Definition 7 (Straight EigenMPC): Let ES be an eigenspace of a linearized version of (2) with c as its direction vector (see (4)). Then, NMPC with the following running cost is expected to locally drive the system to a neighborhood of the eigenmode associated with ES with a desired energy level E_{ref}:

$$\ell(z, F) = z^\top W_z z + F^\top W_F F, \tag{14}$$

$$z = \begin{bmatrix} E_{ref} - E_k \\ c_\perp (x - x_{eq})(1 - \tanh(\alpha E)) \\ c_\perp \dot{x} (1 - \tanh(\alpha E)) \\ F(\tanh(\alpha E) + \beta) \end{bmatrix}, \tag{15}$$

where α and β are scalar tuning factors.

The chosen cost function can be interpreted in natural language as: "Reach a desired energy level while finding a trade-off between remaining close enough to the linear mode and finding the paths that minimize the control action."

Apart from the fixed weights, a gain α is added, which can be understood as an exploration factor. By using $\tanh(\alpha E)$, we ensure a smooth transition between two different behaviors, namely converging to the linear mode and minimizing the control action. The transition occurs as energy increases, at a rate defined by α. With higher α, the controller gets an exploratory behavior, searching for trajectories with minimum power; whereas with lower α, the controller behaves in a conservative way, remaining closer to the linear mode. An example of the controller behavior for different α is shown in Fig. 2. It can be noted that, with $\alpha = 0$ the system converges to the linear mode (dotted line), whereas for increasing α values, the final trajectory approaches the actual eigenmode (dashed line). Moreover, since $\tanh(0) = 0$, the factor β has been added to the control-torque objective in order to ensure that the control action is not excessively high for low energies.

A significant advantage of the straight EigenMPC over its curved counterpart is that it does not require previous offline computation of the modes. Nevertheless, since the minimum cost ℓ is not exactly on the manifold, but a result of the choice of α, we can only guarantee that the controller drives the system to a neighborhood of an eigenmode (see Fig. 2).

V. EXAMPLE: THE TORQUE-LIMITED DOUBLE PENDULUM

In order to validate the EigenMPC approach, we apply it to the double pendulum with point masses at the tip of each link, whose inertia matrix $M(x)$ in local coordinates $x = \begin{bmatrix} \theta_1 & \theta_2 \end{bmatrix}^\top$ is

$$\begin{bmatrix} m_1 l_1^2 + m_2 (l_1 + l_2)^2 - 2m_2 l_1 l_2 (1 - \cos(\theta_2)) & * \\ m_2 (l_2^2 + l_1 l_2 \cos(\theta_2)) & m_2 l_2^2 \end{bmatrix}, \tag{16}$$

(a) In-phase mode, $E_{ref} \approx 2\,\text{J}$. (b) In-phase mode, $E_{ref} \approx 16\,\text{J}$.

(c) Anti-phase mode, $E_{ref} \approx 2\,\text{J}$. (d) Anti-phase mode, $E_{ref} \approx 12\,\text{J}$.
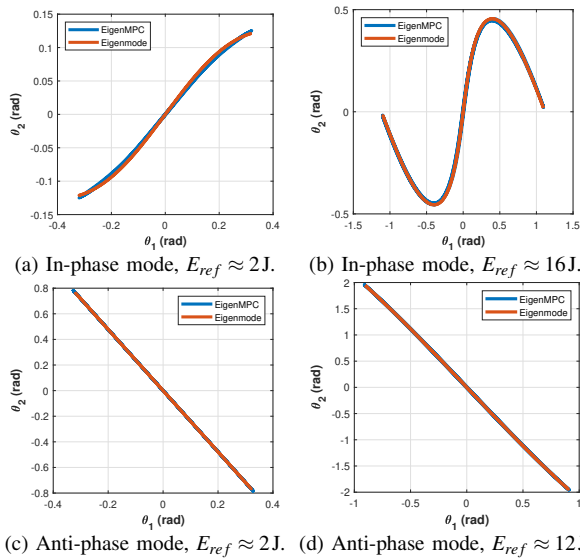
Fig. 3: Oscillations discovered by EigenMPC (in orange) vs. actual eigenmodes (in blue).

and its gravitational potential is $V(x) = V^*(x) - V^*(0)$, where

$$V^*(x) = -m_1 g l_1 \cos(\theta_1) - m_2 g \big(l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)\big), \quad (17)$$

with $(x, \dot{x}) = 0$ being a stable equilibrium configuration where the pendulum is stretched downwards. In addition, for the purpose of this work, its joint torques are limited to $\pm 1\,\text{Nm}$.

As mentioned in [4], the pendulum has two families of eigenmodes evolving from the eigenvectors: the in-phase mode and the anti-phase one. In the next section, both eigenvectors will be used as *a priori* knowledge to find their respective family of eigenmodes.

A. Simulation setup

For the evaluation of the EigenMPC framework, a pendulum with both masses equal to $1\,\text{kg}$ and both link lengths equal to $1\,\text{m}$ was used. Furthermore, MATMPC [11] was applied to solve the NMPC program.

The EigenMPC gains were tuned once for the best performance and used in all simulations. The chosen gains were $w_x = 50$, $w_{\dot{x}} = 2500$ and $w_F = 225$ for both versions, $w_E = 25$ for the curved version, and $w_E = 5$ for the straight version with $\alpha = \beta = 0.1$. The shooting interval for all simulations was $25\,\text{ms}$ and the prediction horizon was $N = 80$ shooting steps. Among all experiments, the worst average computation time was $5.3\,\text{ms}$ when run on a computer equipped with an Intel 3.70GHz Xeon E5-1620 v2 CPU (x8) and 8GB RAM.

The numerical evaluation of the proposed approach is divided into two stages. The first aims at comparing the final trajectories found by the straight version with the eigenmodes computed by the framework presented in [4]. The second analyzes the convergence capabilities and particularities of both versions of EigenMPC. A supplementary video is available at https://doi.org/10.4121/19196765.

B. Eigenmode-finding capabilities

In order to assess the similarity between the trajectories the straight version of EigenMPC converges to and the actual
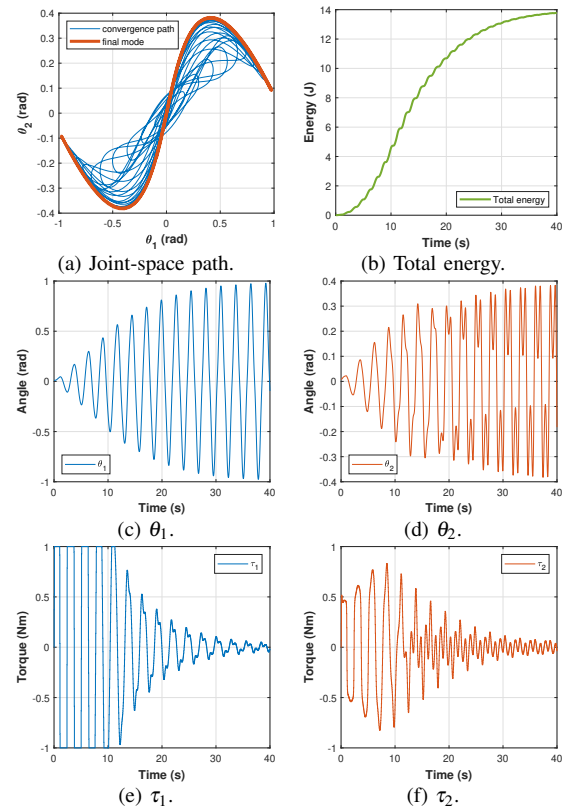

(a) Joint-space path. (b) Total energy.

(c) θ_1. (d) θ_2.

(e) τ_1. (f) τ_2.

Fig. 4: Straight EigenMPC, $E_{ref} = 14\,\text{J}$, saturated controls.

modes, for each of the modes (in-phase and anti-phase) two energy levels were given as reference. The first was $E_{ref} \approx 2\,\text{J}$, the second was $E_{ref} \approx 16\,\text{J}$ for the in-phase mode and $E_{ref} \approx 12\,\text{J}$ for the anti-phase one, which was the highest value for which a line-shaped anti-phase mode could be found by the straight version.

The results are depicted in Fig. 3. The proposed approach performs well in finding both more straight modes, like the anti-phase ones, and more curved ones, most notably the one in Fig. 3b. The most apparent mismatch is in the curvature of the path shown in Fig. 3a, which is slightly more straight than the actual eigenmode. This happens because, for lower energies EigenMPC tends to converge to paths that look more like the original eigenvector.

Despite not having a formal asymptotic stability proof and being able to converge to a neighborhood of the eigenmodes only, the straight EigenMPC produces final results similar to those from the algorithm presented in [4]. Therefore, it serves as an alternative eigenmode-search routine in practical applications.

C. Eigenmode stabilization for constrained systems

After showing that the proposed framework is capable of finding an approximation of the actual eigenmodes, this section aims at assessing its convergence properties in two different scenarios.

1) Nominal case: Initially, both versions of EigenMPC were applied in order to reach an in-phase eigenmode with $E_{ref} = 14\,\text{J}$, starting from the neighborhood of the equilibrium. While for the straight case, the in-phase eigenvector of
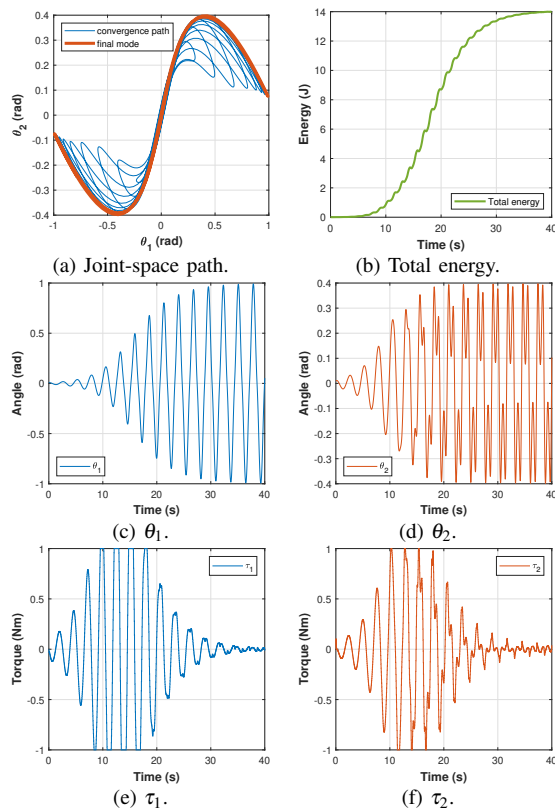
(a) Joint-space path.

(b) Total energy.

(c) θ_1.

(d) θ_2.

(e) τ_1.

(f) τ_2.

Fig. 5: Curved EigenMPC, $E_{ref} = 14$ J, saturated controls.

(a) Straight EigenMPC.

(b) Curved EigenMPC.

Fig. 6: Convergence from large initial conditions.

the linearized system was used, for the curved one, a ninth-order polynomial was used to define X, $\dot{X}$ and c based on the eigenmode found using [4].

The results are shown in Figs. 4 and 5. Figs. 4a and 5a show the joint-space path ($\theta_1 \times \theta_2$) taken by the system until reaching its final trajectory, shown in orange. As expected, for the straight version, the system takes an increasingly curved path as the energy grows. On the other hand, the curved version takes a path whose curvature is similar to the eigenmode. Nevertheless, both versions reach similar final trajectories. It can also be noted that, although the final magnitude of the control action for both controllers is near zero, as expected, the curved version reaches even lower final torques since a chart to the actual eigenmode is used in the cost function. Moreover, due to the initially low cost on the control action, the straight version initially applies higher torques, which reduce as energy builds up. On the other hand, the curved version, having a constant cost on the control action, takes longer to leave the low-energy region. However, both controllers take about the same time to reach the desired energy.

2) Large initial conditions: Subsequently, aiming at assessing a possible weakness of the straight version, i.e., low stiffness (weak position-error feedback) for high energies, we simulated the system starting from larger initial conditions, namely $\theta = (-1.1, 1.1)$ rad. The results are shown in Fig. 6. As expected, due to its constantly high stiffness, the curved EigenMPC recovers quicker than its straight counter part. The straight version, on the other hand, due to its low
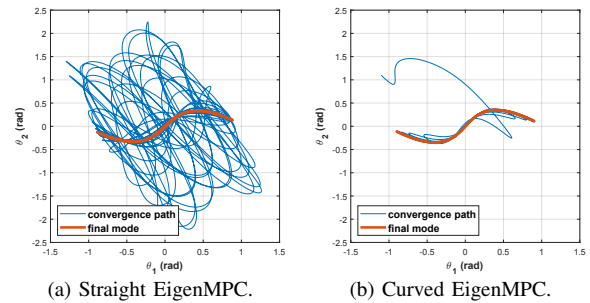
stiffness at high energies, caused by the term $1 - \tanh(\alpha E)$, takes longer to converge, but still reaches the final mode.

VI. CONCLUSIONS AND FUTURE WORK

This paper presented a novel NMPC-based framework, which is capable of finding and converging to sustained nonlinear oscillations in the neighborhood of the eigenmodes or asymptotically converging to an eigenmode in case a pre-computed parametrization is available. Moreover, it guarantees energy efficiency not only in the final trajectory as [5], but also in the convergence phase, due to the suboptimality characteristic it inherits from the NMPC framework. The limitation of the framework lies in the fact that no guarantee of global convergence to a desired mode can be given; however, it has been shown to recover well from significantly large initial conditions. Future work will involve applying EigenMPC to more complex systems, like the DLR SAM, using more realistic propeller constraints, as in [9].

REFERENCES

[1] T. McGeer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.

[2] F. Bjelonic, A. Sachtler, A. Albu-Schäffer, and C. D. Santina, "Experimental closed-loop excitation of nonlinear normal modes on an elastic industrial robot," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1689–1696, 2022.

[3] G. Kerschen, M. Peeters, J.-C. Golinval, and A. F. Vakakis, "Nonlinear normal modes, part i: a useful framework for the structural dynamicist," *Mech. syst. signal process.*, vol. 23, no. 1, pp. 170–194, 2009.

[4] A. Albu-Schaeffer and C. Della Santina, "A review on nonlinear modes in conservative mechanical systems," *Annual Reviews in Control*, vol. 50, pp. 49–71, 2020.

[5] C. D. Santina and A. Albu-Schaeffer, "Exciting efficient oscillations in nonlinear mechanical systems through eigenmanifold stabilization," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 1916–1921, 2021.

[6] Y. S. Sarkisov, M. J. Kim, D. Bicego, D. Tsetserukou, C. Ott, A. Franchi, and K. Kondak, "Development of sam: cable-suspended aerial manipulator," in *2019 Intl. Conf. on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5323–5329.

[7] Y. S. Sarkisov, M. Jun Kim, A. Coelho, D. Tsetserukou, C. Ott, and K. Kondak, "Optimal oscillation damping control of cable-suspended aerial manipulator with a single imu sensor," in *2020 IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2020, pp. 5349–5355.

[8] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. Springer, 2017.

[9] D. Bicego, J. Mazzetto, R. Carli, M. Farina, and A. Franchi, "Nonlinear model predictive control with enhanced actuator model for multi-rotor aerial vehicles with generic designs," *Journal of Intelligent & Robotic Systems*, vol. 100, no. 3, pp. 1213–1247, 2020.

[10] F. Bullo and A. D. Lewis, *Geometric control of mechanical systems*. Springer, 2019.

[11] Y. Chen, M. Bruschetta, E. Picotti, and A. Beghi, "Matmpc - a matlab based toolbox for real-time nonlinear model predictive control," in *2019 18th European Control Conference (ECC)*, 2019, pp. 3365–3370.