Koopman Operator Based Time-Delay Embeddings and State History Augmented LQR for Periodic Hybrid Systems: Bouncing Pendulum and Bipedal Walking

Chun-Ming Yang, Pranav A. Bhounsule

*Department of Mechanical and Industrial Engineering,
University of Illinois at Chicago, 842 W Taylor St, Chicago, IL 60607,
USA. (e-mail: jyang241@uic.edu; pranav@uic.edu)*

Abstract: Time-delay embedding is a technique that uses snapshots of state history over time to build a linear state space model of a nonlinear smooth system. We demonstrate that periodic non-smooth or hybrid system can also be modeled as a linear state space system using this approach as long as its behavior is consistent in modes and timings. We extended time-delay embeddings to generate a linear model of two periodic hybrid systems—the bouncing pendulum and the simplest walker—with control inputs. This leads to a novel state history augmented linear quadratic regulator (LQR) which uses current and past state history for feedback control. Example code can be found at `https://github.com/Chun-MingYang/koopman-timeDelay-lqr.git`

Keywords: Koopman Operator, Hybrid System, LQR Control

1. INTRODUCTION

Periodic hybrid systems are systems that exhibit both continuous dynamics (like moving joints or limbs) and discrete events (like impacts, foot contacts, or switching control modes), and do so in a repeating, cyclic manner. Some examples robotic systems are systems that use legs to walk, run, climb; systems that use hands to brachiate, juggle, and swim; and manipulators that perform periodic pick and place. From a controls synthesis point of view, the hybrid nature of the dynamics are tackled by using a switching controller but it is often difficult to prove the stability of such a controller. In this paper, we use data-driven time delay embedding to identify a smooth model of periodic hybrid systems, which enables the use of provable stable and optimal controllers.

The traditional control approach is to choose two layers of control (Bhounsule et al., 2012; Plooij et al., 2014). First, create a nominal controller that achieves periodic solution. Second, linearize about the periodic solution and use a linear controller to achieve stability. This method does require switching the controller at every discontinuity and is limited to small perturbations around the periodic solution.

The Koopman operator provides a method to create a global linear model for a nonlinear system by projecting the dynamics in the function space (Koopman, 1931). The choice of the functions, also known as observables, is the key to getting the dynamics correctly represented. Given a choice of observables, a computationally efficient method known as Extended Dynamic Mode Decomposition (EDMD) can be used to compute the linear model. One can use neural network-based observables so that the discovery of the observables and the linear model can happen together (Lusch et al., 2018), but this makes EDMD nonlinear and thus, computationally challenging. So far, EDMD has been applied to create linear models of simple smooth systems such as quadcopters (Narayanan et al., 2023), underwater vehicles (Rahmani and Redkar, 2024), autonomous cars (Kim et al., 2001), and soft robot manipulators (Bruder et al., 2020).

This challenge becomes more pronounced when applied to hybrid systems due to their inherent transient features, such as discrete switching dynamics. However, a method known as Direct Encoding (DE) (based on a inner product of observables) was able to compute a global linearization of the hybrid dynamics using sufficiently high number of radial basis functions as observables (Asada, 2023). But the approximation broke down as the number of observables decreased indicating the sensitivity of the method to the choice and size of observables.

Time Delay embeddings use state history as Koopman observables to create a global linear model (Kamb et al., 2020; Brunton et al., 2017). Its basis is the Taken's Embedding theorem (Takens, 1980), which states that the delay-embedded observables captures the structure of the original state space up to a diffeomorphism—meaning it preserves all topological and differential structures (such as smoothness, continuity, and dimensionality). The Time Delay Embedding have been used to model dynamical system with no control such as the van der Pols oscillator and the Lorenz system (Kamb et al., 2020).

Taken's Embedding Theorem, in its original form, does not apply to general non-smooth or hybrid systems. However, under certain specific conditions—such as when the switching behavior is periodic and consistent (i.e., the same sequence of modes and timing)—the delay observ-

ables can preserve the hybrid features by embedding the switching non-smoothness within the periodic structure (Navarrete and Viswanath, 2019). We take this approach to create a smooth model of a periodic hybrid system. Then we use a linear quadratic regulator to stabilize the system. The novelties of the work are as follows. (1) Time delay embedding is used to model a periodic hybrid systems while past works have only modeled smooth systems. (2) Time delay embedding is used to model a system with control inputs while past works have only modeled uncontrolled systems. (3) The development of the state history-augmented linear quadratic regulator (LQR), which uses current and past state for feedback control.

2. METHODS

2.1 Koopman Operator for Systems with Control Inputs

The Koopman operator was originally presented as a method to create a global linear model of an uncontrolled system (Koopman, 1931). The Koopman operator may be extended for a system with control inputs as follows. For a given non-linear system $\mathbf{x}_{i+1} = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i)$, where $\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m$ are the system state and the control inputs respectively. Let $\mathbf{z} = [\mathbf{x}; \mathbf{u}] \in \mathbb{R}^{n+m}$, then there exists the observable function $\mathbf{g}(\mathbf{z}) : \mathbb{R}^{n+m} \to \mathbb{R}^p$, such that the evolution of the system along this coordinate is characterized by a linear dynamics governed by an infinite dimensional Koopman operator $\mathcal{K}$ (Proctor et al., 2018) as:

$$\mathbf{g}(\mathbf{z}_{i+1}) = \mathcal{K}\mathbf{g}(\mathbf{z}_i) \tag{1}$$

A finite-dimensional approximation of the Koopman operator $\mathcal{K}$, denoted as $\mathbf{K}$, can be obtained using EDMD (Williams et al., 2015). Given $M + 1$ snapshot (sample rate dt) pairs of state and input data, where $\mathbf{X}^- = [\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_M], \mathbf{X}^+ = [\mathbf{x}_2, \mathbf{x}_3 \ldots \mathbf{x}_{M+1}], \mathbf{U}^- = [\mathbf{u}_1, \mathbf{u}_2 \ldots \mathbf{u}_M]$ and $\mathbf{U}^+ = [\mathbf{u}_2, \mathbf{u}_3 \ldots \mathbf{u}_{M+1}]$, we can write

$$\mathbf{g}(\mathbf{X}^+, \mathbf{U}^+) = \mathbf{K}\mathbf{g}(\mathbf{X}^-, \mathbf{U}^-)$$
$$\Rightarrow \mathbf{K} = \mathbf{g}(\mathbf{X}^+, \mathbf{U}^+)\mathbf{g}(\mathbf{X}^-, \mathbf{U}^-)^\dagger \tag{2}$$

where $\mathbf{X}^\dagger$ is the pseudo-inverse of $\mathbf{X}$ and linear matrix $\mathbf{K} \in \mathbb{R}^{p \times p}$ is computed via least-squares regression.

2.2 Time-delay Embedding with Koopman Observables

The time-delay embeddings offer another technique for obtaining a linear dynamical system in the function space. We collect the history of states and control inputs and pass these into the observable $\mathbf{g}(\mathbf{z}_i)$ and build the Hankel matrix, $\mathbf{H}_i \in \mathbb{R}^{p(N+1) \times p(M+1)}$

$$\mathbf{H}_i = \begin{bmatrix} \mathbf{g}(\mathbf{z}_i) & \mathbf{g}(\mathbf{z}_{i+1}) & \cdots & \mathbf{g}(\mathbf{z}_{i+M}) \\ \mathbf{g}(\mathbf{z}_{i+1}) & \mathbf{g}(\mathbf{z}_{i+2}) & \cdots & \mathbf{g}(\mathbf{z}_{i+M+1}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{g}(\mathbf{z}_{i+N}) & \mathbf{g}(\mathbf{z}_{i+N+1}) & \cdots & \mathbf{g}(\mathbf{z}_{i+M+N}) \end{bmatrix} \tag{3}$$

where N and M are the time embeddings with sample rate dt. We now use Koopman operator given in Eqn. 2 to get

$$\mathbf{H}_i = \begin{bmatrix} \mathbf{g}(\mathbf{z}_i) & \mathbf{K}\mathbf{g}(\mathbf{z}_i) & \cdots & \mathbf{K}^M\mathbf{g}(\mathbf{z}_i) \\ \mathbf{K}\mathbf{g}(\mathbf{z}_i) & \mathbf{K}^2\mathbf{g}(\mathbf{z}_i) & \cdots & \mathbf{K}^{M+1}\mathbf{g}(\mathbf{z}_i) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}^N\mathbf{g}(\mathbf{z}) & \mathbf{K}^{N+1}\mathbf{g}(\mathbf{z}_i) & \cdots & \mathbf{K}^{M+N}\mathbf{g}(\mathbf{z}_i) \end{bmatrix} \tag{4}$$

Since $\mathbf{K}$ is a linear operator, from Eqn. 4 we can get

$$\mathbf{H}_{i+1} = \mathbf{L}\mathbf{H}_i \Rightarrow \mathbf{L} = \mathbf{H}_{i+1}\mathbf{H}_i^\dagger \tag{5}$$

where $\mathbf{H}^\dagger$ is the pseudo-inverse of $\mathbf{H}$ and Koopman operator is $\mathbf{L} \in \mathbb{R}^{p(N+1) \times p(N+1)}$.

2.3 Time-delay Embedding Observables with State and Control

One common challenge in Koopman operator theory is that the finite-dimensional approximation of the Koopman matrix $\mathbf{K} \in \mathbb{R}^{p \times p}$ often leads to a high-dimensional system where $p \gg m + n$. Additionally, identifying a suitable high-dimensional observable space is both difficult and lacks theoretical guarantees. To address this, a practical approach is to use the identity function $\mathbf{g}(\mathbf{z}) = \mathbf{z} = [\mathbf{x}; \mathbf{u}]$, which has been empirically shown to work well with time-delay embeddings (Kamb et al., 2020). This approach allows for reducing the dimensionality of the Koopman operator $\mathbf{L}$, from $\mathbb{R}^{p(N+1) \times p(N+1)}$ to $\mathbb{R}^{(m+n)(N+1) \times (m+n)(N+1)}$, as shown in Eqn. 5, provided the number of time-delay embeddings N satisfies $(m + n)(N + 1) < p$, and simultaneously avoids the challenge of selecting appropriate observable functions.

We rewrite Eqn. 5 as follows

$$\begin{bmatrix} \mathbf{x}_{i+1} \\ \mathbf{u}_{i+1} \\ \mathbf{x}_{i+2} \\ \mathbf{u}_{i+2} \\ \vdots \\ \mathbf{x}_{i+N+1} \\ \mathbf{u}_{i+N+1} \end{bmatrix} = \mathbf{L} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{u}_i \\ \mathbf{x}_{i+1} \\ \mathbf{u}_{i+1} \\ \vdots \\ \mathbf{x}_{i+N} \\ \mathbf{u}_{i+N} \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{x}_{i+1} \\ \mathbf{x}_{i+2} \\ \vdots \\ \mathbf{x}_{i+N+1} \\ \mathbf{u}_{i+1} \\ \mathbf{u}_{i+2} \\ \vdots \\ \mathbf{u}_{i+N+1} \end{bmatrix} = \overline{\mathbf{L}} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_{i+1} \\ \vdots \\ \mathbf{x}_{i+N} \\ \mathbf{u}_i \\ \mathbf{u}_{i+1} \\ \vdots \\ \mathbf{u}_{i+N} \end{bmatrix} \tag{6}$$

where $\overline{\mathbf{L}} = \mathbf{P}\mathbf{L}\mathbf{P}^T$ and $\mathbf{P} \in \mathbb{R}^{(m+n)(N+1) \times (m+n)(N+1)}$ is a permutation matrix that reorganizes the original basis structure into a form with all the states grouped at the top and the corresponding control inputs grouped at the bottom, and is defined as:

$$P_{i,j} = \begin{cases} 1, & \text{if } j = k(m+n) + r \text{ and } i = k\,n + r, \\ & k = 0, \ldots, N, \ r = 1, \ldots, n, \\ 1, & \text{if } j = k(m+n) + n + s \text{ and } i = (N+1)n + \\ & k\,m + s, \quad k = 0, \ldots, N, \ s = 1, \ldots, m, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

2.4 State History Augmented LQR Control

From Eqn. 6, the state space model can be written as:

$$\begin{bmatrix} \mathbf{X}_{i+1} \\ \mathbf{U}_{i+1} \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{L}}_{11} & \overline{\mathbf{L}}_{12} \\ \overline{\mathbf{L}}_{21} & \overline{\mathbf{L}}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{X}_i \\ \mathbf{U}_i \end{bmatrix} \tag{8}$$

where $\mathbf{X}_i = [\mathbf{x}_i; \mathbf{x}_{i+1} \ldots \mathbf{x}_{i+N}]$, $\mathbf{U}_i = [\mathbf{u}_i; \mathbf{u}_{i+1} \ldots \mathbf{u}_{i+N}]$ and $\overline{\mathbf{L}} = \begin{bmatrix} \overline{\mathbf{L}}_{11} & \overline{\mathbf{L}}_{12} \\ \overline{\mathbf{L}}_{21} & \overline{\mathbf{L}}_{22} \end{bmatrix}$ is a matrix with constants. Given Eqn. 8 and let the reference periodic trajectory be $\mathbf{X}^d$ and the corresponding control profile be $\mathbf{U}^d$, the error dynamics can be formulated as:

$$\begin{bmatrix} \mathbf{X}_{i+1} - \mathbf{X}_{i+1}^d \\ \mathbf{U}_{i+1} - \mathbf{U}_{i+1}^d \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{L}}_{11} & \overline{\mathbf{L}}_{12} \\ \overline{\mathbf{L}}_{21} & \overline{\mathbf{L}}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{X}_i - \mathbf{X}_i^d \\ \mathbf{U}_i - \mathbf{U}_i^d \end{bmatrix}$$
$$\begin{bmatrix} \hat{\mathbf{X}}_{i+1} \\ \hat{\mathbf{U}}_{i+1} \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{L}}_{11} & \overline{\mathbf{L}}_{12} \\ \overline{\mathbf{L}}_{21} & \overline{\mathbf{L}}_{22} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{X}}_i \\ \hat{\mathbf{U}}_i \end{bmatrix} \tag{9}$$
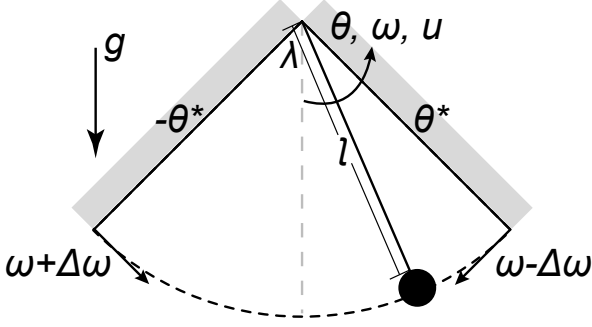
Fig. 1. Bouncing pendulum model (Govindarajan et al., 2016)

where $\hat{\mathbf{X}}_\mathbf{i} = \mathbf{X}_i - \mathbf{X}_i^d$ and $\hat{\mathbf{U}}_\mathbf{i} = \mathbf{U}_i - \mathbf{U}_i^d$, then the first $n \times (N+1)$ rows of left hand side of Eqn. 9 can be written as:

$$\begin{aligned} \hat{\mathbf{X}}_{i+1} &= \overline{\mathbf{L}}_{11}\hat{\mathbf{X}}_i + \overline{\mathbf{L}}_{12}\hat{\mathbf{U}}_i \\ &= \mathbf{A}\hat{\mathbf{X}}_i + \mathbf{B}\hat{\mathbf{U}}_i \end{aligned} \quad (10)$$

where $\mathbf{A} \in \mathbb{R}^{n(N+1) \times n(N+1)}$ and $\mathbf{B} \in \mathbb{R}^{n(N+1) \times m(N+1)}$ are the system matrices. Consider using controller $\mathbf{U}_i = \mathbf{U}_i^d + \hat{\mathbf{U}}_i$, we can now use the linear quadratic regulator (LQR) to compute a gain $\mathbf{K}_{\text{LQR}} \in \mathbb{R}^{m(N+1) \times n(N+1)}$ such that the optimal correction control term $\hat{\mathbf{U}}_i = -\mathbf{K}_{\text{LQR}}\hat{\mathbf{X}}_\mathbf{i}$ can be retrieved. It is important to note that unlike traditional LQR which uses the current state for feedback control, the controller uses the current state and past state history for feedback control. This ensures that the controller is retrospective and less reactive ensuring smoother control.

2.5 Application to Periodic Hybrid Systems

The Takens Embedding Theorem (Takens, 1980) states that for a smooth deterministic, non-noisy dynamical system, the time-delay embeddings map (see Sec. 2.3) is a diffeomorphism (i.e., a smooth, invertible mapping) of the original state space provided rich trajectory data are used for the mapping. In particular, if d is the number of delays and n is the dimension of state space then $d > 2n + 1$.

Although the Takens Embedding Theorem applies only to smooth systems, it can be applied to periodic hybrid systems as long as, d, the number of delays, encompasses one periodic cycle of the system. This can be intuitively understood as follows. When the trajectory data is periodic, it can be represented using a Fourier Series, in other words, it is smooth. Hence the Takens Embedding Theorem could be applied. However, any mode and/or timing changes in the switching will make the system non-smooth and violate the Takens Embedding Theorem.

We present the dynamics of two hybrid systems: a hybrid pendulum bouncing model and a planar model of bipedal walking known as the simplest walker.

A. Hybrid Pendulum Model: The hybrid pendulum model, shown in Fig. 1 , exhibits a periodic hybrid dynamics—It consists of a swing phase represented by a continuous dynamics $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ and impact phases represented by discrete mappings with switch conditions

$\mathbf{x}^+ = \{\mathcal{R}_i(\mathbf{x}) \,|\, \mathcal{S}_i(\mathbf{x}) = 0\}$—with the periodic limit cycle $t \in [0, T]$ as:

$$\mathbf{x} = \mathbf{x}_0 \quad (11)$$
$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, u) \quad (12)$$

$$\mathbf{x}^+ = \{\mathcal{R}_i(\mathbf{x}) \,|\, \mathcal{S}_i(\mathbf{x}) = 0\}, \text{ for } i = 1, 2 \quad (13)$$

where $\mathbf{x}, u$ are the dynamics state and non-dimensional control torque respectively. By defining the state vector $\mathbf{x} = [\theta, \omega]^T$, the continuous dynamics can be written as:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \omega \\ \dfrac{-g}{l} \sin\theta - \lambda\omega + u \end{bmatrix} \quad (14)$$

where g, l, λ are gravity term, pendulum length, and damping constant respectively. The switching conditions $\mathcal{S}_i(\mathbf{x}) = 0$ are defined when pendulum passing through the given angles $\pm\theta^*$ with the discrete reset map $\mathcal{R}_i(\mathbf{x})$—an instantaneous opposite kick—defined as:

$$\begin{aligned} \mathcal{S}_1(\mathbf{x}): &\quad \left\{\theta^- + \theta^* \,\middle|\, \omega^- < 0\right\} = 0, \\ \mathcal{R}_1(\mathbf{x}): &\quad \begin{cases} \theta^+ = -\theta^*, \\ \omega^+ = \omega^- + \Delta\omega, \end{cases} \\ \mathcal{S}_2(\mathbf{x}): &\quad \left\{\theta^- - \theta^* \,\middle|\, \omega^- > 0\right\} = 0, \\ \mathcal{R}_2(\mathbf{x}): &\quad \begin{cases} \theta^+ = \theta^*, \\ \omega^+ = \omega^- - \Delta\omega. \end{cases} \end{aligned} \quad (15)$$

where $-, +$ are the notations representing the instance just before and just after the switch conditions; $\Delta\omega > 0$ is the instantaneous kick modeled as instant augular velocity change. The limit cycle of the system can be obtained by considering $u = \lambda\omega$—which turns the system into an undamped pendulum that is naturally periodic—such that the periodic orbit, given $\mathbf{x}_0 = [\theta_0, \omega_0]^T$, can be obtained manually by selecting $\theta^* = \{\theta(t_s)|t_s \in [0, T]\} < max(\theta(0 : T))$ and $\Delta\omega = 2|\omega(t_s)|$.

B. Bipedal Walking Model: The simplified walking model, shown in Fig. 2, exhibits periodic hybrid dynamics—It consists of a single stance phase represented by continuous dynamics $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ and a support exchange phase represented by discrete mapping $\mathbf{x}^+ = \mathcal{R}(\mathbf{x}(t))$—with the periodic limit cycle $t \in [0, T]$ governed by switch condition $S(\mathbf{x}|t = T) = 0$ as:

$$\begin{aligned} t = 0: &\quad \mathbf{x} = \mathbf{x}_0 &\quad (16) \\ 0 \le t \le T: &\quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) &\quad (17) \\ t = T: &\quad S(\mathbf{x}(t)) = 0 &\quad (18) \\ t = T: &\quad \mathbf{x}^+ = \mathcal{R}(\mathbf{x}(t)) &\quad (19) \end{aligned}$$

where $\mathbf{x} = [\theta, \phi, \dot{\theta}, \dot{\phi}]^T$ is the dynamics state vector with $\theta, \dot{\theta}$ for stance leg angle and angular velocity and $\phi, \dot{\phi}$ for swing foot respectively; $\mathbf{u} = [u_1, u_2]^T$ is the control vector consisted with the stance and swing leg hip torques respectively.

By taking moments about stance foot contact point and hip hinge respectively, and non-dimensionalizaing time with $\sqrt{l/g}$, and applying the limit $m/M \to 0$, and non-dimensionalizing torque by Mgl, we obtain the continuous dynamics during single stance phase as:
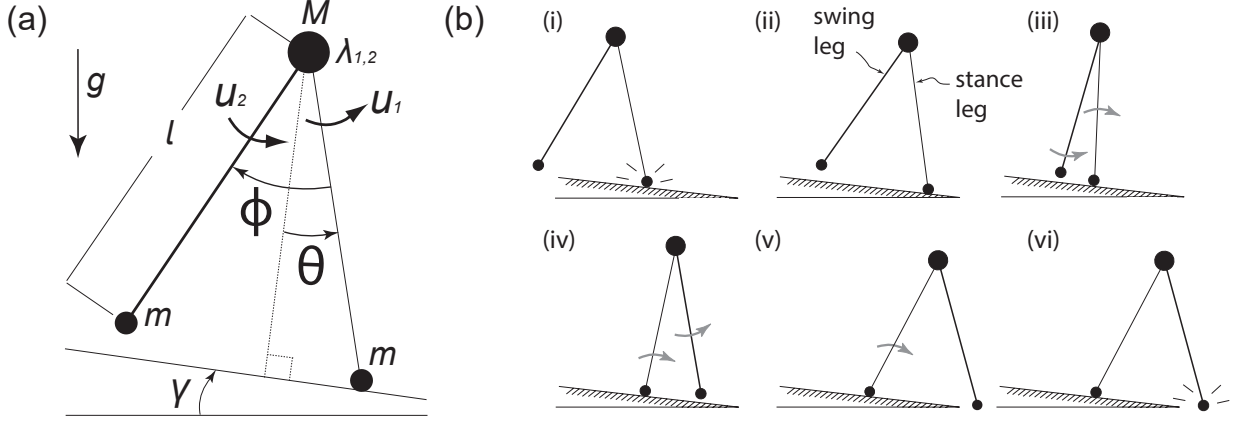
Fig. 2. (a) Simplest walker (Garcia et al., 1998), note that the model has the damping terms and control torques on hips of both stance and swing legs. (b) A typical step of the simplest walker.

$$\ddot{\theta} = \sin(\theta - \gamma) - \lambda_1\dot{\theta} + u_1 \tag{20}$$

$$\ddot{\phi} = \sin(\theta - \gamma) + \left\{\dot{\theta}^2 - \cos(\theta - \gamma)\right\}\sin(\phi) - \lambda_2\dot{\phi} + u_2 \tag{21}$$

where M, m, l are mass of hip, mass and length of the foot respectively; γ is the slope of the ramp; λ_i $(i = 1, 2)$ are the damping coefficient. The switch condition $S(\mathbf{x}|t = T)$ is when the swing foot collides with the ramp but ignoring the condition when the legs are parallel, i.e., $\phi = \theta = 0$, followed by support exchange phase where the legs exchange roles, $\mathbf{x}^+ = \{\mathcal{R}(\mathbf{x})|t = T\}$ as:

$$S(\mathbf{x}|t = T) : \phi^- - 2\theta^- = 0 \tag{22}$$

$$\mathcal{R}(\mathbf{x}|t = T) : \begin{cases} \theta^+ = -\theta^- \\ \phi^+ = -\phi^- = -2\theta^- \\ \dot{\theta}^+ = \cos(2\theta^-)\dot{\theta}^- \\ \dot{\phi}^+ = \left\{1 - \cos(2\theta^-)\right\}\cos(2\theta^-)\dot{\theta}^- \end{cases} \tag{23}$$

where $-, +$ are the notations representing the instance just before and just after the foot strike event. The fixed points can be found by solving the following Poincaré analysis as:

$$\mathcal{P} : \{(\mathbf{x}, t)|t = 0\} \rightarrow \{(\mathbf{x}, t)|t = T^*\} \tag{24}$$

$$\mathbf{x}^* = \mathcal{P}(\mathbf{x}^*, T^*) \tag{25}$$

3. RESULTS

3.1 Bouncing Pendulum

For pendulum with length $l = 1$ m, damping constant $\lambda = 0.1$, the limit cycle shown in Fig. 3 corresponding to the initial position $\theta_0 = 0$ rad, $\omega_0 = -2$ rad/s can be obtained by selecting $\theta^* = 0.5$ rad and $\Delta\omega = 2.538$ rad/s, resulting the period to be $T = 1.144$ sec, such that the reference trajectory and nominal control profile, which also serve as the training dataset, are collected by simulating the model, initialed with Poincaré fixed points, for 6 sec, as shown as dashed lines in Fig. 4.

To obtain the linear model for hybrid pendulum dynamics, the Hankel matrix is formulated with the parameters $N = 110, M = 90$ and $dt = 1e^{-2}$ as the sample rate, then the Koopman operator $\bar{\mathbf{L}}$ is approximated from EDMD as discussed in Sec.2.3. The resulting linear model
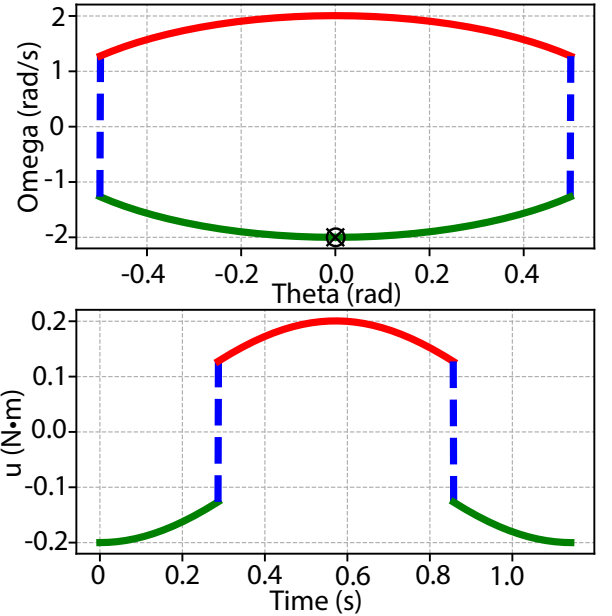

Fig. 3. Bouncing pendulum limit cycle. The red and green lines respectively represent positive and negative rotation direction, and blue lines represent the hybrid switch.

demonstrates accurate prediction, as shown as solid lines shown in Fig. 4, with root mean square errors (RMSE) of $0.008, 0.015$, and 0.017 for θ, ω, and u, respectively.

The LQR controller is used to handle the external disturbance—an impulse $\omega^+(t = 0.35) = \omega^-(t = 0.35) + 0.6$ applied to pendulum at time 0.35 sec. So as to calculate the feedback gain, the system matrices $\mathbf{A}$ and $\mathbf{B}$ are first extracted from $\bar{\mathbf{L}}$ as described in Sec. 2.4. The MATLAB function d2c() is then used to convert them to their continuous-time forms, $\mathbf{A}_c$ and $\mathbf{B}_c$. Finally, the LQR gain $\mathbf{K}_{\text{LQR}}$ is computed using the MATLAB function lqr(). One can see that with the LQR controller, the periodic trajectories can be stabilized, shown in Fig. 5, with the RMSE of $0.035, 0.496$ for tracking errors to θ, ω respectively.
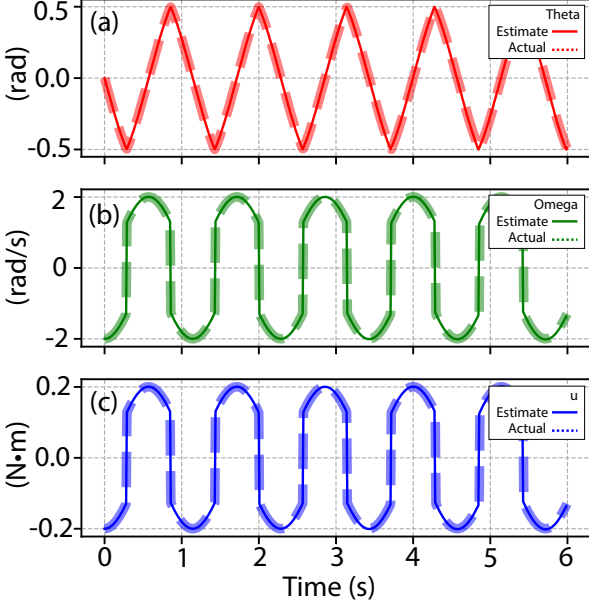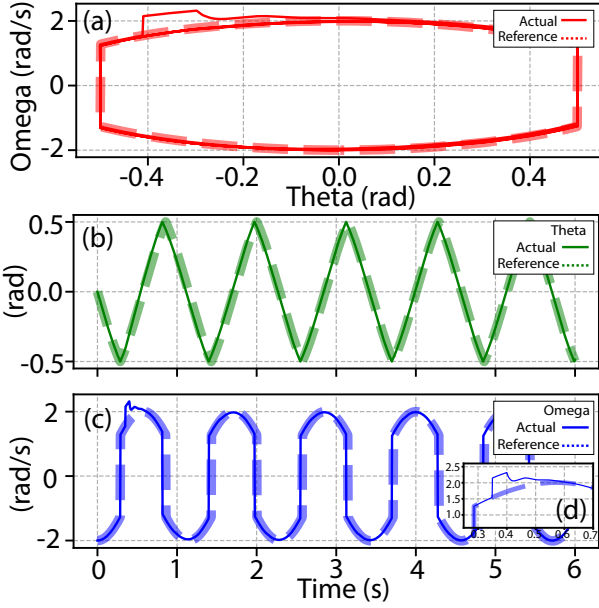
Fig. 4. Bouncing pendulum Koopman prediction.

Fig. 5. Bouncing pendulum LQR control.

3.2 Simplest Walker

Given the initial range $\theta_0 \in [0.1, 0.2], \phi_0 \in [-0.4, -0.3]$ rad and $\dot{\theta}_0 \in [-0.3, -0.2], \dot{\phi}_0 \in [0.03, 0.04]$ rad/s, along with the nominal controller $u_1 = \lambda_1 \dot{\theta}, u_2 = \lambda_2 \dot{\phi}$ ($\lambda_1 = \lambda_2 = 0.5$) to cancel the damping effect, the fixed points of the periodic walking limit cycle—whose Poincaré section is taken right after the foot-strike event—are found through Poincaré analysis to be $[\theta_0, \phi_0, \dot{\theta}_0, \dot{\phi}_0]^* = [0.162, -0.325, -0.231, 0.038]$ rad, such that the reference trajectory and norminal control profile, which also serve as the training dataset, are collected by simulating the model, initialed with Poincaré fixed points, for 8 sec, as shown as dashed lines in Fig. 6.
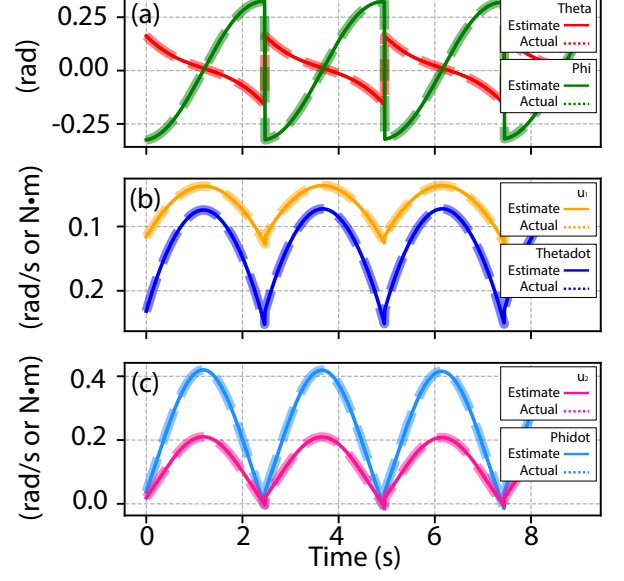

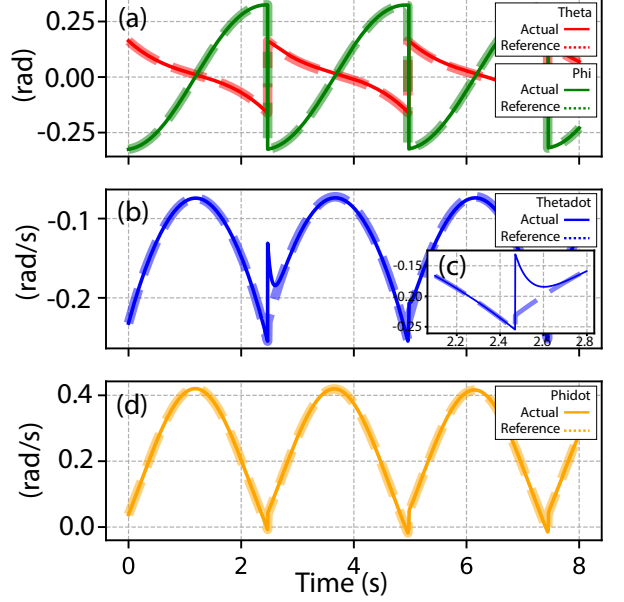Fig. 6. Simplest walker Koopman prediction.

Fig. 7. Simplest walker LQR control.

To obtain the linear model for simplest walker, the Hankel matrix is formulated with the parameters $N = 300, M = 600$ and $dt = 1e^{-2}$ as the sample rate, then the Koopman operator $\bar{\mathbf{L}}$ is approximated from EDMD as discussed in Sec.2.3. The resulting linear model demonstrates accurate prediction, as shown as solid lines shown in Fig. 6, with root mean square errors (RMSE) of $0.001, 0.002, 0.001, 0.003, 0.011$, and 0.006 for θ, ϕ, $\dot{\theta}$, $\dot{\phi}$, u_1, and u_2 respectively.

In order to handle the external disturbance—an impulse applied to the stance foot just after the second foot-strike—an LQR controller is used. So as to calculate the feedback gain, the system matrices $\mathbf{A}$ and $\mathbf{B}$ are first extracted from $\bar{\mathbf{L}}$ as described in Sec. 2.4. The MATLAB function `d2c()` is then used to convert them to their continuous-time forms, $\mathbf{A}_c$ and $\mathbf{B}_c$. Finally, the LQR gain $\mathbf{K}_{\text{LQR}}$ is computed using the MATLAB function `lqr()`.

One can see that with the LQR controller, the periodic trajectories can be stabilized, shown in Fig. 7, with the RMSE of $0.019, 0.038, 0.006, 0.001$ for tracking errors to $\theta, \phi, \dot{\theta}, \dot{\phi}$ respectively.

4. DISCUSSION AND CONCLUSION

This paper demonstrates that a sequence of state and control snapshots over time – known as time-delay embeddings – can be used to compute a linear representation of the dynamics of a periodic hybrid system. The resulting linear dynamics can be used with a standard LQR controller for feedback control. The method is demonstrated on two hybrid systems: a pendulum that bounces on a wall and the simplest walker.

The time delay embedding like the Koopman operator can create a global linear state space model by projecting on high-dimensional function space (Kamb et al., 2020; Brunton et al., 2017). Koopman operator relies on a good choice of observables that span the dynamics of the system. This is often done by trial and error and there is no guarantee of finding such a set of observables. On the other hand, time embedding with system states as observable is able to faithfully reconstruct the dynamics, which is a major advantage of time delay embeddings. Thus Koopman operator lifts using non-linear functions (observers) while time embedding lifts using a set of delay coordinates of the state space thus turning the temporal dynamics into spatial representation.

The LQR controller resulting from time-delay embedding is the state history augmented control which uses the current and past history for feedback control. Unlike traditional LQR which is reactive or MPC which is pro-active, the state history augmented LQR is retrospective. It potentially leads to a smoother control than a purely reactive controller as it takes into account past state history.

Our work has some limitations. In order to apply time embedding to hybrid systems without violating the Takens Embedding Theorem, we had to restrict periodic hybrid systems where the guards, events, and mode changes are unaffected by the disturbance. Consequently, we can only control fully actuated periodic hybrid systems as full actuation provides enough control authority to ensure the system can maintain periodicity in the presence of a disturbance. Moreover, the resulting linear lifting was local to the periodic hybrid system that it modeled. Finally, the resulting linearized system was of high dimension due to use of sufficiently long time history. This can be remediated by using model reduction techniques and is left as a future work.

We conclude that time delay embeddings provides a powerful method for modeling periodic hybrid systems as long as the systems are consistent. That is, the modes and timings remains unaffected by control or disturbance.

REFERENCES

Asada, H.H. (2023). Global, unified representation of heterogenous robot dynamics using composition operators: A koopman direct encoding method. *IEEE/ASME Transactions on Mechatronics*, 28(5), 2633–2644.

Bhounsule, P., Cortell, J., and Ruina, A. (2012). Cornell ranger: Implementing energy-optimal trajectory control using low information, reflex-based control. *Dyn Walk. Pensacola.*

Bruder, D., Fu, X., Gillespie, R.B., Remy, C.D., and Vasudevan, R. (2020). Data-driven control of soft robots using koopman operator theory. *IEEE Transactions on Robotics*, 37(3), 948–961.

Brunton, S.L., Brunton, B.W., Proctor, J.L., Kaiser, E., and Kutz, J.N. (2017). Chaos as an intermittently forced linear system. *Nature communications*, 8(1), 19.

Garcia, M., Chatterjee, A., Ruina, A., and Coleman, M. (1998). The simplest walking model: stability, complexity, and scaling.

Govindarajan, N., Arbabi, H., Van Blargian, L., Matchen, T., Tegling, E., et al. (2016). An operator-theoretic viewpoint to non-smooth dynamical systems: Koopman analysis of a hybrid pendulum. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, 6477–6484. IEEE.

Kamb, M., Kaiser, E., Brunton, S.L., and Kutz, J.N. (2020). Time-delay observables for koopman: Theory and applications. *SIAM Journal on Applied Dynamical Systems*, 19(2), 886–917.

Kim, B., Necsulescu, D., and Sasiadek, J. (2001). Model predictive control of an autonomous vehicle. In *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings (Cat. No. 01TH8556)*, volume 2, 1279–1284. IEEE.

Koopman, B.O. (1931). Hamiltonian systems and transformation in hilbert space. *Proceedings of the National Academy of Sciences*, 17(5), 315–318.

Lusch, B., Kutz, J.N., and Brunton, S.L. (2018). Deep learning for universal linear embeddings of nonlinear dynamics. *Nature communications*, 9(1), 4950.

Narayanan, S.S., Tellez-Castro, D., Sutavani, S., and Vaidya, U. (2023). Se (3) koopman-mpc: Data-driven learning and control of quadrotor uavs. *IFAC-PapersOnLine*, 56(3), 607–612.

Navarrete, R. and Viswanath, D. (2019). Delay embedding of periodic orbits using a fixed observation function. *Physica D: Nonlinear Phenomena*, 388, 1–9.

Plooij, M., Wolfslag, W., and Wisse, M. (2014). Open loop stable control in repetitive manipulation tasks. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 949–956. IEEE.

Proctor, J.L., Brunton, S.L., and Kutz, J.N. (2018). Generalizing koopman theory to allow for inputs and control. *SIAM Journal on Applied Dynamical Systems*, 17(1), 909–930.

Rahmani, M. and Redkar, S. (2024). Enhanced koopman operator-based robust data-driven control for 3 degree of freedom autonomous underwater vehicles: A novel approach. *Ocean Engineering*, 307, 118227.

Takens, F. (1980). Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980: proceedings of a symposium held at the University of Warwick 1979/80*, 366–381. Springer.

Williams, M.O., Kevrekidis, I.G., and Rowley, C.W. (2015). A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25, 1307–1346.