Embedding Hybrid Systems into Continuous Latent Vector Fields

Sangli Teng, Hang Liu, Koushil Sreenath
University of California, Berkeley

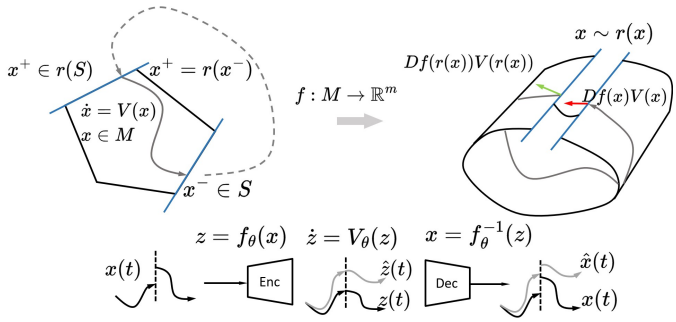

Fig. 1: We proved that the $n-$dimensional *discontinuous* flow of a hybrid system can be embedded into a latent space equipped with an $m-$dimensional *continuous* extrinsic vector field when $m > 2n$. The latent embedding can be learned by the proposed latent ODE framework. The presented work is accepted to ICML 2026. The complete version is available at https://arxiv.org/abs/2606.10596.

***Abstract*—This work proves that an n-dimensional hybrid system can be embedded into an m-dimensional Euclidean space equipped with a continuous vector field on its embedded image whenever $m > 2n$. This result suggests that an *intrinsically* discontinuous hybrid system generically admits a continuous *extrinsic* representation that is well-posed for differentiable optimization. Building on this existence theorem, we show that a latent Neural ODE with consistency loss in both the latent and state space can accurately recover the flow of hybrid systems. Extensive experiments suggest the proposed method outperforms the existing method in learning hybrid systems with varying geometries from only time series data.**

I. Introduction

Hybrid systems (hybrid automata) model a broad class of physical [1, 2] and cyber-physical processes [3, 4] by combining continuous-time vector fields with discrete state resets. Despite its powerful expressiveness, the hybrid system has nonsmooth or discontinuous state evolution that is not well-posed for differentiable optimization [5], especially at state resets.

To learn hybrid systems from data, traditional methods [6, 7] partition the trajectories into different segments and learn the dynamics in each mode. However, these methods require mode selection that is combinatorially complicated. On the other hand, the event function-based method [8] tries to differentiate through the state resets, which, however, suffers from ill-conditioning and bad initializations.

A more recent structural alternative is to represent the hybrid dynamics as continuous flows. The hybrid system theory [9] suggests that the state reset functions induce an equivalence relationship to *glue* the partitioned state space into a continuous latent manifold. Furthermore, the glued manifold can be reconstructed from time-series data in our previous **C**ontinuous **Hy**brid System **L**earning in **L**atent Space (CHyLL) algorithm [10], leveraging the Whitney Embedding Theorem [11].

However, the results in [9, 10] do not suggest that the *latent vector field* is continuous. To further improve the differentiability, we pose the question: ***"Do hybrid systems admit provably continuous latent embedding that induces a continuous vector field?"***. Answering this question will lay a foundation to make differentiable optimization well-posed for hybrid system learning. As in Figure 1, we make the following contributions:

Contribution: The ***theoretical*** contribution is to prove that an n-dimensional hybrid system can be embedded into an m-dimensional Euclidean space equipped with a continuous vector field on its embedded image as long as $m > 2n$. This theorem suggests that an *intrinsically* discontinuous hybrid system admits an *extrinsically* continuous representation that is well-posed for differentiable optimization. Based on this theorem, the ***algorithmic*** contribution is a latent Neural ODE framework for learning hybrid systems from time series data. The experiments and ablations show that the consistency loss in both the latent and the observation space is the key to accurately recovering the flow of hybrid systems with varying topologies. The implementation is included in https://github.com/SangliTeng/Continuous-Hybrid-System-Learning.

II. Preliminary

In this section, we discuss the preliminaries of differential geometry and a geometric description of hybrid systems.

A. Differential Geometry

Consider a finite-dimensional smooth manifold M. The tangent space at a point $x \in M$ is denoted by T_xM. The tangent bundle $TM := \bigcup_{x\in M} T_xM$ is the disjoint union of

tangent spaces. A smooth vector field is a map $V : M \to TM$ such that $V(x) \in T_xM$ for all $x \in M$. The set of all smooth vector fields on M is denoted by $\mathfrak{X}(M)$.

For any smooth curve $c : (-\varepsilon, \varepsilon) \to M$ satisfying $c(0) = x$ and $\dot{c}(0) \in T_xM$, the tangent map $Df(x) : T_xM \to T_{f(x)}N$ of the smooth map $f : M \to N$ is defined by: $Df(x)\,\dot{c}(0) = \frac{d}{dt}\big(f \circ c\big)(t)\big|_{t=0}$. For $f : M \to N$ and $g : G \to M$ with G a smooth manifold, the tangent map of $f \circ g : G \to N$ satisfies the chain rule $D(f \circ g)(x) = Df\big(g(x)\big) \circ Dg(x)$.

We introduce two nondegenerate property for $f : M \to N$.

Definition 1 (Immersion [11]). *f is an immersion if the tangent map $Df : T_xM \to T_{f(x)}N$ is an injective function, or equivalently,* $\operatorname{rank} T_x f = \dim M$.

Definition 2 (Embedding [11]). *f is an embedding if f is an immersion and f maps M homeomorphically*[1] *onto its image.*

For a manifold with boundaries, we can conduct our analysis in the collar coordinates:

Theorem 1 ([11]). *For a manifold M with boundary $S \subset M$, a submanifold of M with co-dimension 1, there exists a collar embedding such that x specifies the position on the boundary and t denotes the inward components of the collar that points to the* interior *of M:*

$$\kappa_S(x, t) : S \times [0, \varepsilon) \to M \tag{1}$$

with $\kappa_S(x, 0) = x, \forall x \in S$.

B. Hybrid Systems

We consider a geometric description of the hybrid system [9, 12] as shown in Figure 1.

Definition 3 (Hybrid System). *A hybrid system is a 4-tuple $\mathcal{H} = (M, S, V, r)$ with the following components:*

(State Space) M is an n-dim smooth manifold.

(Guard) $S \subset \partial M$ is a $(n-1)$-dim smooth submanifold.

(Dynamics) $V : M \to TM$ is a smooth vector field.

(Reset) r: $S \to r(S) \subset \partial M$ a diffeomorphism onto its image and $r(S)$ is an embedded submanifold.

The dynamics of $\mathcal{H}$ can be described as an event ODE:

$$\begin{cases} \dot{x} = V(x), & x \notin S, \\ x^+ = r\big(x^-\big), & x^- \in S. \end{cases} \tag{$\mathcal{H}$-Dynamics}$$

We assume $\mathcal{H}$ has the following properties:

Assumption 1. *M and S are compact.*

Assumption 2. *$V(x)$ points* outward *along the interior of $x \in S$ and* inward *along the interior of $x \in r(S)$.*

Assumption 3. *$S \cap r(S) = \emptyset$ and the intersection of their closure $\overline{S} \cap \overline{r(S)}$ has co-dimension at least 2.*

[1]A function f is a homeomorphism if it is bijective, continuous, and its inverse f^{-1} is also continuous.

Assumption 2 ensures the trajectories will traverse the boundary, and Assumption 3 avoids repeated state resets from happening instantaneously.

Though S and $r(S)$ are different and possibly disconnected, $r(\cdot)$ is a diffeomorphism that defines an equivalence relationship that "glues" S and $r(S)$ to reformulate the state space M as a continuous manifold as indicated in fig. 1:

Theorem 2 (Hybrifold [9]). *Let $\sim$ be the equivalence relation by $x \sim r(x), \forall x \in S$. We collapse the equivalence class by $\sim$ to a point to obtain the piecewise smooth topological manifold, namely, the hybrifold:*

$$M_{\mathcal{H}} = M/\sim, \tag{Hybrifold}$$

C. Generic Property and Transversal Maps

Denote the k-time continuous functions from a finite-dimensional manifold M to N as $C^k(M, N)$. The main goal of this work is to explore if a certain property is held for the majority of elements in $C^k(M, N)$. Thus, we explore the "generic property", and informally, we have:

Remark 1 (Generic property). *A property is generic if it holds in a "large set"* [2] *that contains "most" of the elements in a topological space. For example, the set of irrational numbers $\mathbb{R} \setminus \mathbb{Q}$ contains "most" of the elements in $\mathbb{R}$.*

To study the generic property of $f \in C^k(M, N)$, we leverage the tools related to the transversal map:

Definition 4 (Transversality Condition [11]). *Let M, N be smooth finite-dimensional manifolds and $Z \subset N$ an embedded submanifold. A $C^k (k \geq 1)$ map $f : M \to N$ is* transverse *to Z, written $f \pitchfork Z$, if for every $x \in f^{-1}(Z)$,*

$$\operatorname{Im}(df(x)) + T_{f(x)}Z = T_{f(x)}N. \tag{Transversality}$$

Theorem 3 (Preimage Theorem under Transversality [13]). *If $f \pitchfork Z$, then $f^{-1}(Z)$ is an embedded submanifold of M and with dimension:*

$$\dim f^{-1}(Z) = \dim M - \operatorname{codim}_N Z. \tag{2}$$

In our proof shown in the following sections, we construct the set we wish to avoid as the manifold Z, and we let $\dim f^{-1}(Z) < 0$ to ensure the image of f is empty (bad things never happen).

Theorem 4 (Parametric Transversality Theorem, Section 19.1 [13]). *Let M, N be C^k finite-dimensional manifolds, and let $Z \subset N$ be a closed C^k submanifold. Let $\mathcal{P}$ be a C^k Banach manifold (e.g., $C^k(M, N) : M \to N$). Define the total map $\Gamma : M \times \mathcal{P} \to N$ and the associated slice $\Gamma(\cdot \mid p) : M \to N, \Gamma(x \mid p) := \Gamma(x, p)$. Assume the following conditions hold:*

(i) The total map Γ is of class C^k;

(ii) $\Gamma \pitchfork Z$ (the total map is transversal to Z);

(iii) $r > \max(0, \dim M - \operatorname{codim}_N Z)$.

[2]A residual set, i.e., the complement of a countable union of nowhere dense sets.

Then the set of parameters $\mathcal{P}_Z = \{ p \in \mathcal{P} : \Gamma(\cdot \mid p) \pitchfork Z \}$ is residual (and hence dense) in $\mathcal{P}$.

As a consequence of Theorem 4, one can show that the following embedding exists generically:

Theorem 5 (Whitney Embedding Theorem (weak) [11])**.** *Any C^k-manifold M $(k \geq 1)$ of dimension n can be embedded into $\mathbb{R}^m$ if $m > 2n$.*

III. Problem Formulation

We consider the problem of learning the hybrid system from time series data:

Problem 1. *Consider the flow of $\mathcal{H}$ recorded as dataset $\mathcal{X} := \{(t_0, x_0), (t_1, x_1), (t_2, x_2), \cdots, (t_T, x_T)\}$. Our goal is to recover the flow of $\mathcal{H}$ from $\mathcal{X}$.*

We note that due to the reset map r on the guard surface S, the flow of $\mathcal{H}$ is discontinuous and extremely hard to differentiate. Although Theorem 5 suggests that (Hybrifold) employs a continuous representation to make the flow of $\mathcal{H}$ continuous in a learned latent space [10], the vector field on (Hybrifold) is not guaranteed to be continuous. Thus, we ask:

Problem 2. *Can we embed $\mathcal{H}$ in a latent space and characterize the flow of $\mathcal{H}$ by a continuous latent vector field?*

By doing so, we can convert $\mathcal{H}$ into a globally continuous structure, which will significantly improve its differentiability for machine learning. In this work, we prove that such a latent representation can be constructed by elements that **generically exist** in the space of functions.

IV. Main Result

In this section, we leverage Theorem 3 and 4 to show that whenever $m > 2n$, there generically exists a continuous function (encoder) $f : M \to \mathbb{R}^m$ to make the latent manifold and its induced vector field both continuous on the entire M. Formally, we have:

Theorem 6 (Continuous Extrinsic Representation of Hybrid Systems)**.** *For a hybrid system $\mathcal{H}$ satisfying Assumption 1, 2 and 3, whenever $m > 2n$, there exists an $f \in C^k(M, \mathbb{R}^m)$ that satisfies:*

$$f(x) = f(r(x)), \forall x \in S \tag{C-1}$$

$$Df(x)V(x) = Df(r(x))V(r(x)), \forall x \in S \tag{C-2}$$

$$f \text{ induces an embedding of } M_{\mathcal{H}} \text{ into } \mathbb{R}^m. \tag{C-3}$$

Moreover, f can be obtained by a generic choice within the admissible construction space.

Proof: We here briefly introduce the procedure. The proof is to show that a construction of f exists *generically* when $m > 2n$ leveraging Theorem 4. The construction of f is done on the collar coordinates ensured by Theorem 1.

Denote $R := r(S)$. To enforce (C-1), we choose an embedding $g : S \to \mathbb{R}^m$ (guaranteed by Theorem 5) to embed $S-$side as $f_S = g$ and $R-$side as $f_R = g \circ r^{-1}$ which ensures $f_R(r(x)) = f_S(x), \forall x \in S$. Then we apply the first order Taylor extension in the collar coordinate on $S-$side as $\bar{f}_S(x,t) = f_S(x) + t g_S(x)$ with $\bar{f}_S(x,t) : S \times [\epsilon, 0) \to M$ defined on the collar coordinates with $f_S(x)$ the components on the boundary S and $g_S : S \to \mathbb{R}^m$ the component inward S (similarly on R-side). By choosing g_S and enforcing (C-2), we can uniquely determine g_R on R side. Given this construction in the collar neighborhood $\mathrm{Img}(\kappa_S) \cup \mathrm{Img}(\kappa_R)$, we show that whenever $m > 2n$, a generic choice of g and g_S ensures the first-order extension is injective and without rank deficiency. Finally, we extend the local property to the entire M.

The geometric interpretation of the theorem is illustrated in Figure 2 in the collar coordinates in M and $f(M)$. ■

Based on Theorem 6, we can confirm that the answer to Problem 2 is positive.

Corollary 7 (Differentiability of latent flow)**.** *Let $z(t) = f(x(t))$ be the latent trajectory given a generic choice of f and $x(t)$ the hybrid execution of $\mathcal{H}$. By Theorem 6, the induced latent vector field $\dot{z} = Df(x)V(x)$ is C^0 on $f(M)$ and the the latent trajectory $z(t)$ is C^1 in time.*

Proof: This is an immediate result by (C-1) and (C-2) for the differentiability at any $x \in S \cup r(S)$, and (C-3) for the differentiability elsewhere. ■

Though we have this *extrinsic* continuous representation f, we note that the *intrinsic* discontinuity of the hybrid system $\mathcal{H}$ is irrelevant to the representation.

Remark 2 (Intrinsic and Extrinsic Representation of $\mathcal{H}$)**.** *By [9], the* intrinsic *vector field of $\mathcal{H}$ is not guaranteed to be continuous as it is possible that $\exists x \in S$*

$$Dr(x)V(x) \neq V(r(x)).$$

On the contrary, $f : M \to \mathbb{R}^m$ is the extrinsic *representation that ensures* (C-1) *and* (C-2) *are satisfied on $S \cup r(S)$ by additional degree of freedom.*

In this work, we seek such an extrinsic representation that makes Problem 1 well-posed for differentiable optimization, despite the intrinsic non-smooth structures. Thus, we have the following learnable components of our latent ODE:

Remark 3. *Whenever $m > 2n$, a hybrid system $\mathcal{H}$ with hybrid dynamics* ($\mathcal{H}$-Dynamics) *can be embedded into a latent space by an encoder*

$$f_\theta : M \to \mathbb{R}^m, \tag{Encoder}$$

that satisfies $f_\theta(x) = f_\theta(r(x)), \forall x \in S$, and the associated latent vector field

$$V_\theta : \mathrm{Img}(f_\theta) \to \mathbb{R}^m, \tag{Latent Vector Field}$$

that relates to $V(\cdot)$ by $V_\theta(f_\theta(x)) = Df_\theta V(x), \forall x \in M$. To recover x from $z \in \mathrm{Img}(f_\theta)$, we have the the decoder:

$$f_\theta^{-1} : \mathrm{Img}(f_\theta) \to M. \tag{Decoder}$$

For machine learning implementations, we can optimize the

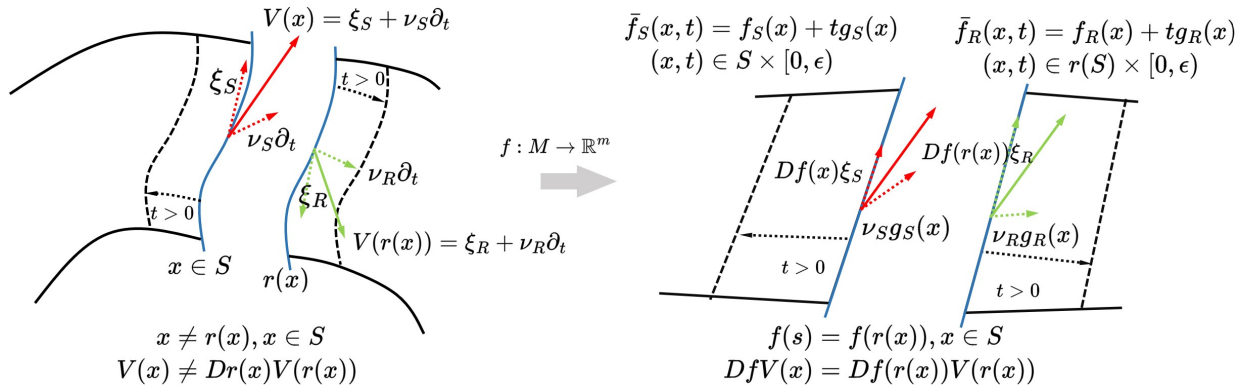

Fig. 2: **Left:** the state and vector field at pre-impact state $x \in S$ and the post-impact state $r(x) \in r(S)$ are possibly mismatched as an *intrinsic* property of $\mathcal{H}$. **Right:** by designing the embedding $f : M \to \mathbb{R}^m$, the *extrinsic* representation satisfy (C-1) and (C-2). The additional degree of freedom by f is the key for the extrinsic representation to eliminate the discontinuity.

parameter θ to obtain f_θ, V_θ, and f_θ^{-1} given $\mathcal{X}$, the time-series observation from the flow of $\mathcal{H}$.

V. Learning $\mathcal{H}$ by Latent Neural ODE

Based on Remark 3, we propose a latent ODE framework that recovers the flow of $\mathcal{H}$ from time series data observation $\mathcal{D}$ by designing the encoder $f_\theta : M \to \mathbb{R}^m$, latent vector field $V_\theta : \mathrm{Img}(f_\theta) \to \mathbb{R}^m$, and the decoder $f_\theta^{-1} : \mathrm{Img}(f_\theta) \to \mathbb{R}^m$. Each of these components can be represented by an MLP.

The latent flow is obtained by integrating V_θ from the initial state $z_0 = f_\theta(x_0)$ by Neural ODE [14]:

$$\hat{z}_k = \int_{t_0}^{t_k} V_\theta(z(t))dt + f_\theta(x_0). \tag{3}$$

Then we enforce the consistency loss both in the state space:

$$\mathcal{L}_x = \mathrm{MSE}(f_\theta^{-1}(\hat{z}_k), x_k), \tag{4}$$

and in the latent space

$$\mathcal{L}_z = \mathrm{MSE}(\hat{z}_k, f_\theta(x_k)). \tag{5}$$

Though representing the latent vector field $V_\theta(x)$ as an MLP (with a finite Lipschitz constant [15]) automatically ensures the learned latent flow is unique [16] and continuous, we consider additional inductive bias as in [10] to explicitly enforce the conditions in Theorem 6 and see if they can further improve the performance.

To enforce (C-1), we have the gluing loss [10]:

$$\mathcal{L}_g = \mathrm{MSE}(f_\theta(x_k), f_\theta(x_{k+1})), \forall k \in \mathcal{I}, \tag{6}$$

with $\mathcal{I}$ the index set that labels all the pre- and post-reset states by thresholding the empirical Lipchitz constant of the data point in $\mathcal{D}$, i.e., $\|\frac{x_{k+1}-x_k}{t_{k+1}-t_k}\|$.

Then we explore if we need to enforce (C-2) that is not used in [10]. Thus, we design the velocity compatibility loss:

$$\mathcal{L}_v = \mathrm{MSE}(\dot{z}_k^-, \dot{z}_k^+), \forall k \in \mathcal{I}, \tag{7}$$

with the pre-reset velocity approximated by $\hat{\dot{z}}_k^- = \frac{f_\theta(x_k) - f_\theta(x_{k-1})}{t_k - t_{k-1}}$ the backward finite difference and the post-reset one by $\hat{\dot{z}}_k^+ = \frac{f_\theta(x_{k+1}) - f_\theta(x_k)}{t_{k+1} - t_k}$ the forward version.

Algorithm 1 CHyLL++

Require: Trajectory dataset $\mathcal{D}$; gluing index set $\mathcal{I}$; curriculum $\{T_1 < \cdots < T_\ell\}$; steps per curriculum S.

```
for ℓ = 1, ..., L do
    for step = 1, ..., S do
        x_{0:T_ℓ} ~ D                                  ▷ Sample mini-trajectories
        ẑ_{0:T_ℓ} <(3)- x_0                            ▷ Encode and rollout
        x̂_{0:T_ℓ} <f_θ^{-1}(·)- ẑ_{0:T_ℓ}              ▷ Decode
        L(θ) <(4-9)- {x̂_k, ẑ_k}_{k=0}^{T_l}            ▷ Compute loss
        θ ← θ − η ∇_θ L(θ)                             ▷ Update parameters
    end for
end for
    return f_θ, V_θ and f_θ^{-1}.
```

Finally, to avoid the latent space from collapsing, we enforce the latent space to have positive covariance by

$$\mathcal{L}_c = \sum_{i=1}^{m} \mathrm{ReLU}(\Lambda - \mathrm{Cov}(f_\theta(x_k)_i)), \tag{8}$$

with Λ the threshold for the latent covariance [10]. Finally, we have the loss function as:

$$\mathcal{L}(\theta) = w_x\mathcal{L}_x + w_z\mathcal{L}_z + w_g\mathcal{L}_g + w_v\mathcal{L}_v + w_c\mathcal{L}_c, \tag{9}$$

with the weights $w_{(\cdot)}$. Finally, we consider a rollout curriculum to learn the trajectories from short to long segments, and we conclude our method in Algorithm 1.

VI. Numerical Experiments

A. Analytical Example

We provide an analytical example to illustrate how an inherently discontinuous hybrid systems admits a continuous representation in the embedded space.

Consider a vector field on a 1-dimensional disconnected manifold with different vector fields on each partition:

$$V(x) = \begin{cases} 1, x \in [0,1) \\ 2, x \in [2,3) \end{cases} \quad r(x) = \begin{cases} x+1, x = 1 \\ x-3, x = 3 \end{cases} \tag{10}$$

TABLE I: We conduct the experiments five times for each case to compute the MSE (mean $\pm$ std.dev). For methods that exhibit numerical instability, divergence, or obvious failure in capturing the pattern, we report the qualitative results. The notation * indicates the results reported from [10]. Though [10] works for the first four cases, it overfits to one modality of the input signal for the last case.

	Proposed (ReLu, $\mathcal{L}_{x,z,c}$)	[10]	[14]	[17]	[18]	[8]
Bouncing Ball	$\mathbf{0.158} \pm 0.0376$	$0.237 \pm 0.0629^*$	Large Penetration	Diverge	Diverge	Ill-conditioned
Torus	$\mathbf{0.00367} \pm 0.00166$	$0.0164 \pm 0.00989^*$	Pattern Incorrect	Pattern Incorrect	Diverge	Ill-conditioned
Klein Bottle	$\mathbf{0.00587} \pm 0.00461$	$0.0220 \pm 0.00584^*$	Pattern Incorrect	Pattern Incorrect	Diverge	Ill-conditioned
Three-Link Walker	$\mathbf{0.0952} \pm 0.00424$	$0.234 \pm 0.0140^*$	$0.275 \pm 0.00610^*$	$0.253 \pm 0.0770^*$	Diverge	Ill-conditioned
3D Bouncing Ball	$\mathbf{0.162} \pm 0.0150$	Collapse to z-direction	0.5244 ± 0.0707	Diverge	Diverge	Ill-conditioned

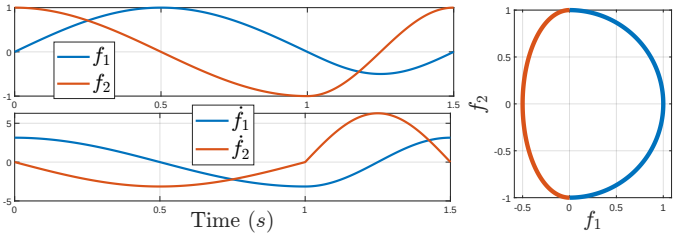

Fig. 3: The 2-D embedding of a 1-D hybrid system. We find that the embedded manifold is continuous and admits a global C^0 extrinsic vector field $\dot{f}$. On the right-hand side, we see that the manifold is glued by hemispheres from two different ellipses.

We can verify that the two vector fields are not intrinsically continuous by the fact that the pre- and post-reset velocity $(Dr(x^-)V(x^-), V(r(x^-)))$ at $x^- = 1, 2$ are $(1, 2)$ and $(2, 1)$ respectively, which is not identical. As the $r(x)$ glues the state space as a circle, we construct f by sinusoidal functions. Thus we consider the equivalence relationship $1 \sim 2$ and $3 \sim 0$ and analytically construct f as:

$$f(x) = \begin{cases} A_1 y_1(x), x \in (0,1) \\ A_2 y_2(x), x \in (2,3) \end{cases}, \tag{11}$$

with $y_1(x) = [\cos(\pi x), \sin(\pi x)]^\top$, $y_2(x) = [\cos(\pi(x-1)), \sin(\pi(x-1))]$ and $A_i \in \mathbb{R}^{2\times 2}$ the linear coefficient matrices. Then we enforce (C-1) by letting $f(1) = f(2), f(3) = f(0)$, and (C-2) by $Df(x^-) \circ V(x^-) = Df(r(x^-)) \circ V(r(x^-))$ with $x^- = 1, 3$. Thus, we have

$$A_1[-1, 0]^\top = A_2[-1, 0]^\top, \; A_1[1, 0]^\top = A_2[1, 0]^\top, \tag{12}$$

for (C-1) and the following equations for (C-2):

$$A_1[0, 1]^\top \pi = A_2[0, 1]^\top 2\pi, \; A_1[0, 1]^\top 2\pi = A_2[0, 1]^\top \pi. \tag{13}$$

Thus, we choose a solution that also makes f an embedding:

$$A_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & -0.5 \\ 1 & 0 \end{bmatrix}.$$

Finally, we have the time evolutions of f and $\dot{f}$ shown in Figure 3, which is a 1D continuous manifold embedded in 2D space Euclidean space equipped with a continuous vector field on it. Then, for each segment of the manifold, we can design a decoder by $\mathrm{atan2}(\cdot, \cdot)$ with proper scaling of f_1 and f_2 as the inverse of the encoder f.

B. Learning $\mathcal{H}$ from Time-Series Data

In this section, we consider three 2-dimensional examples with varying geometry and two 6-dimensional physical systems. In the 2D experiments, we have dynamics evolve on a manifold with boundary $M = [0, 1]^2$ with linear dynamics $\dot{x} = Cx, \forall x \in [0, 1]^2$.

We consider the reset map that gives the quotient manifold $M_{\mathcal{H}}$ a complicated topology by gluing two neighboring boundaries of M, i.e., $S = S_1 \cup S_2$ with $S_1 = \{x | x_2 \in [0, 1], x_1 = 1\}$ and $S_2 = \{x | x_1 \in [0, 1], x_2 = 1\}$ to their opposite sides. We consider the following reset function that makes $M_{\mathcal{H}}$ diffeomorphic to ***Torus*** and ***Klein Bottle*** by $r_{\mathrm{t}}(x) = \begin{cases} (0, x_2) & x \in S_1 \\ (x_1, 0) & x \in S_2 \end{cases}$ and $r_{\mathrm{k}}(x) = \begin{cases} (0, x_2) & x \in S_1 \\ (1 - x_1, 0) & x \in S_2 \end{cases}$.

The ***Bouncing Ball*** system satisfies the linear dynamics with gravity g by $\dot{x}_1 = x_2, \dot{x}_2 = -g$. The guard surface $S_{\mathrm{b}} := \{(x_1, x_2) | x_1 = 0, x_2 \leq 0\}$ indicates the ground where elastic collisions happen and is represented by the reset map $r_{\mathrm{b}}(x) = [x_1, -\alpha x_2]^\top, [x_1, x_2]^\top \in S_{\mathrm{b}}$.

The first 6-dimensional system is the ***Three-Link Walker*** governed by the rigid body dynamics in the continuous-time $D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \pi(q, \dot{q})$ with $q, \dot{q} \in \mathbb{R}^3$, $D(q)$ the mass matrix, $C(q, \dot{q})$ the Coriolis matrix, $G(q)$ the gravity vector and $\pi(q, \dot{q})$ an nonlinear feedback controller to enforce a periodic gait. The state reset happens when changing the stance and swing foot by the contact dynamics Δ [19] and the relabling of stance and swing foot by a permutation matrix P [1] $\dot{q}^+ = \Delta(q^-, \dot{q}^+), q^+ = Pq^-$.

Then we consider a ***3D-Bouncing Ball*** thrown into a sink. The dynamics in the vertical direction are identical to the Bouncing Ball case, while the horizontal dynamics are not subject to any external force when in the air: $\dot{x} = v_x, \dot{v_x} = 0, \dot{y} = v_y, \dot{v_y} = 0$, but subject to the impact $r([x, v_x]^\top) = [x-, -\alpha v_x^-]^\top$, $\forall [x, v_x]^\top \in S_1 \cup S_2$, when reaching the wall defined by $S_{\mathrm{b}1} := \{[x, v_x] : x = 0.8, v_x > 0\}$ and $S_{\mathrm{b}2} := \{[x, v_x] : x = -0.8, v_x < 0\}$. The dynamics in y-direction are identical to x.

C. Comparison with Existing Methods

We compare Algorithm 1 with the existing methods for learning dynamical systems. The most relevant method is [10] inspired by the Whitney Embedding Theorem to bias the latent space towards a continuous high-dimensional manifold by the gluing loss and latent consistency loss. We note that [10] admits a two-stage training strategy that learns the encoder

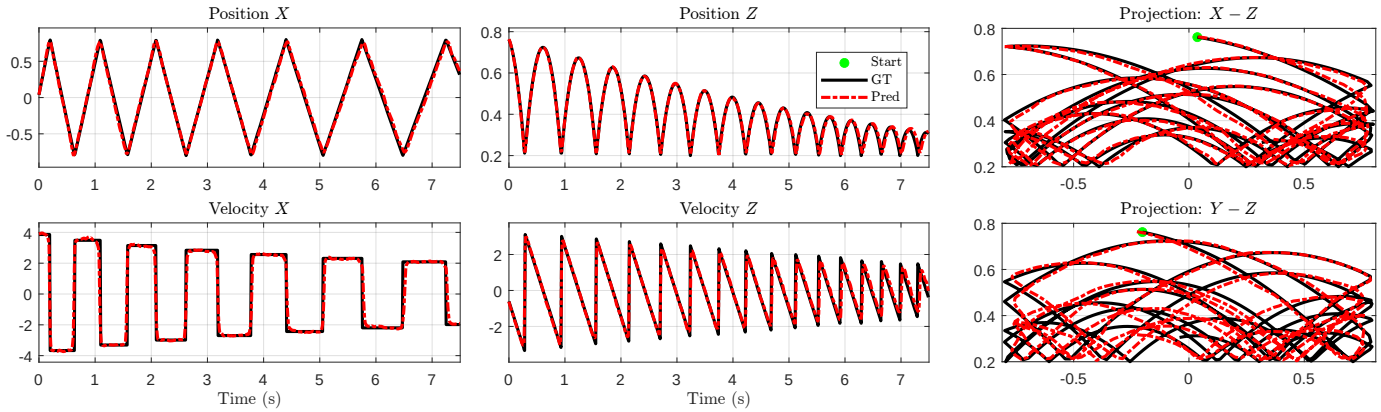

Fig. 4: Trajectories of the 3D Bouncing Ball in a sink. The maximal training horizon in the curriculum is $2s$ while the testing set is $7.5s$. We see that the ball is colliding with the walls and the floor, but without obvious penetrations. The square wave in the horizontal velocity is extremely challenging for differentiable optimization that makes the baselines fail.

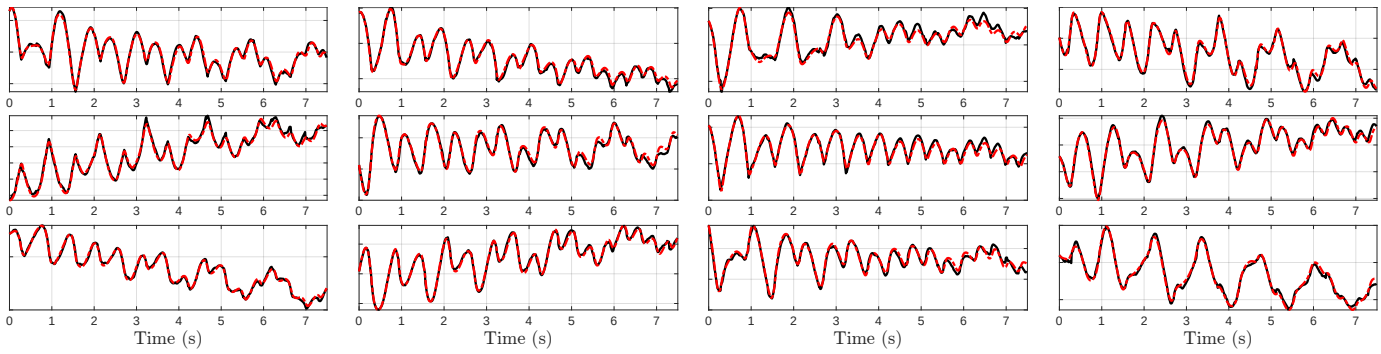

Fig. 5: The continuous latent trajectories ($m = 12$) of the 3D Bouncing Ball.

and vector first and then the decoder. The second baseline is the classical Neural ODE [14] in the state space. The third baseline is the latent Neural ODE with RNN encoders [17]. We also consider the deep Koopman operator [18] that is designed for general dynamical systems. We note that all the baselines do not admit mode selections that are combinatorially hard.

We conduct the experiments five times for each example to compute the MSE (mean $\pm$ std.dev). We train the system for at most 200 steps in Neural ODE and rollout for more than 500 steps. For the Bouncing Ball, Klein Bottle, Torus, and Three-Link Walker, we consider the identical experimental settings in [10]. For the 3D Bouncing Ball, we consider a larger neural network. For fairness, we consider that all activations in the hidden layers of the proposed method are ReLU, though in the ablations, we find that the $\sin$ activation function lead to better results. The statistics of the experiments is shown in Table I. We find that the proposed method greatly outperforms [10] in all the cases, while the other methods without the geometry-based inductive bias have different kinds of failures.

We plotted the trajectories predicted by Algorithm 1 of the 3D Bouncing Ball in Figure 4 and Figure 5. As the dynamics are decoupled in three directions, we present the trajectories in only $x-$ and $z-$directions.

VII. Conclusions

In this work, we theoretically proved that an $n-$dimensional hybrid system can be embedded into m-dimensional Euclidean space with a continuous vector field in its embedded image whenever $m > 2n$. Building on this existence theorem, we show that the proposed latent Neural ODE is well-posed to learn hybrid systems from time-series data with superior accuracy. The ablation study suggested that the consistency loss in both the state and the latent space is the key to successful implementations.

References

[1] Eric R Westervelt, Jessy W Grizzle, and Daniel E Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE transactions on automatic control*, 48(1):42–56, 2003.

[2] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *International Journal of Robotics Research*, 33(1):69–81, 2014.

[3] Paulo Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic control*, 52(9):1680–1685, 2007.

[4] Aaron D Ames, Kevin Galloway, Koushil Sreenath, and Jessy W Grizzle. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *IEEE Transactions on Automatic Control*, 59(4):876–891, 2014.

[5] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison,

Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[6] Michael Poli, Stefano Massaroli, Luca Scimeca, Sanghyuk Chun, Seong Joon Oh, Atsushi Yamashita, Hajime Asama, Jinkyoo Park, and Animesh Garg. Neural hybrid automata: Learning dynamics with multiple modes and stochastic transitions. *Advances in Neural Information Processing Systems*, 34:9977–9989, 2021.

[7] Hang Liu, Sangli Teng, Ben Liu, Wei Zhang, and Maani Ghaffari. Discrete-time hybrid automata learning: Legged locomotion meets skateboarding. *arXiv preprint arXiv:2503.01842*, 2025.

[8] Ricky TQ Chen, Brandon Amos, and Maximilian Nickel. Learning neural event functions for ordinary differential equations. *arXiv preprint arXiv:2011.03902*, 2020.

[9] Slobodan N Simic, Karl Henrik Johansson, John Lygeros, and Shankar Sastry. Towards a geometric theory of hybrid systems. *Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications and Algorithms*, 12(5-6):649–687, 2005.

[10] Sangli Teng, Hang Liu, Jingyu Song, and Koushil Sreenath. Chyll: Learning continuous neural representations of hybrid systems. *arXiv preprint arXiv:2512.10117*, 2025.

[11] Morris W Hirsch. *Differential topology*, volume 33. Springer Science & Business Media, 2012.

[12] William Clark and Anthony Bloch. Invariant forms in hybrid and impact systems and a taming of zeno. *Archive for Rational Mechanics and Analysis*, 247(2):13, 2023.

[13] Ralph Abraham and Joel Robbin. *Transversal mappings and flows*. WA Benjamin New York, 1967.

[14] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[15] Aladin Virmaux and Kevin Scaman. Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems*, 31, 2018.

[16] Hassan K. Khalil. Nonlinear systems third edition. 2008. URL https://api.semanticscholar.org/CorpusID:59790953.

[17] Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

[18] Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature communications*, 9(1):4950, 2018.

[19] Mike Featherstone. Body, image and affect in consumer culture. *Body & society*, 16(1):193–221, 2010.