Learning Nonholonomic Dynamics with Constraint Discovery

Baiyue Wang BAIYUEW@UMICH.EDU
Ann Arbor, MI 48104

Anthony Bloch ABLOCH@UMICH.EDU
Department of Mathmatics, University of Michigan, Ann Arbor, MI 48109

Editors: G. Sukhatme, L. Lindemann, S. Tu, A. Wierman, N. Atanasov

Abstract

We consider learning nonholonomic dynamical systems while discovering the constraints, and describe in detail the case of the rolling disk. A nonholonomic system is a system subject to nonholonomic constraints. Unlike holonomic constraints, nonholonomic constraints do not define a sub-manifold on the configuration space. Therefore, the inverse problem of finding the constraints has to involve the tangent bundle. This paper discusses a general procedure for learning the dynamics of a nonholonomic system through Hamel's formalism, while discovering the system constraints by parameterizing them, given the data set of discrete trajectories on the tangent bundle TQ. We prove that there is a local minimum for convergence of the network. We also preserve symmetry of the system by reducing the Lagrangian to the Lie algebra of the selected group.

Keywords: Nonholonomic Systems, Machine Learning, Systems with Symmetry

1. Introduction

The theory of symmetries has been generalized to describe the invariant quantities in a dynamical system under transformations. It has caught the attention of researchers in the field of machine learning due to the increasing need for understanding the structure of neural networks. Although it seems that neural networks are far from symmetric, there can be some innate symmetries Bloch et al. (2024), and a network can be designed to be symmetric Huang et al. (2024); Varghese et al. (2024).

Some research has studied how to discover the symmetries of a system through machine learning. Most of the them are based on symplecticity and Hamiltonian systems. Examples include: learning the symplectic map Chen and Tao (2021), learning the symplectic form Chen et al. (2021), learning through the Lie-Poisson brackets Eldred et al. (2023, 2024); Gruber et al. (2023), learning with a symplectic network design Varghese et al. (2024), or learning through the G-invariant Lagrangian submanifold Vaquero et al. (2023). These works were conducted based on the assumption that the system is symmetric. While we can make that assumption, there are ways to discover the symmetry automatically through an equivariant network design such as a Lie algebra convolutional network Dehmamy et al. (2021), or dictionary learning Ghosh et al. (2023).

A rather less studied topic is to learn a symmetric system with constraints. Constraints can be holonomic (defining a submanifold of the configuration space) or nonholonomic (constraints on the velocities which are not integrable). The difficulties of this problem lie in that one needs to learn the constraints implicitly from the trajectory of the system. Learning a holonomic constraint (a constraint on the system configurations) can be seen as learning the null space of the constraint equation Lin et al. (2015) or solving an integer feasibility problem Chou et al. (2020), in which case constraints on velocities are not considered and the system symmetry is ignored.

In this work, we propose a general method for learning nonholonomic systems, and at the same time not ignoring the system symmetry, by leveraging Hamel's formalism of nonholonomic dynamics with symmetry Bloch et al. (2009). Hamel's formalism treats the constraints intrinsically as a distribution in the tangent bundle rather than imposing the constraints via multipliers; therefore, the dynamics is interpreted as an ordinary differential equation without an additional constraint. In this formalism we train the network in a similar manner to a Neural ODE Chen et al. (2019).

2. Problem Statement

Let us first define a nonholonomic constraint.

Definition 1 (Bloch (2015)) *Consider a mechanical system, subject to linear velocity constraints that in generalized coordinates can be expressed as*

$$a(q)\dot{q} = [a_1(q), \ldots, a_n(q)]\dot{q} = 0, \tag{1}$$

*where $\dot{q}$ is regarded as a column vector, and $a(q)$ is a matrix of n columns. A constraint is said to be **nonholonomic** or **non-integrable** if there is no function h of q such that the constraint can be written as $h(q) = \text{constant}$, or $(\partial h/\partial q^i)\dot{q}^i = 0$.*

In the rolling disk dynamics, which will be introduced in more detail in Section 5, the constraints have the form

$$\begin{bmatrix} -R\cos\varphi & 0 & 1 & 0 \\ -R\sin\varphi & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\varphi} \\ \dot{x} \\ \dot{y} \end{bmatrix} = 0. \tag{2}$$

One can see that the constraints are non-integrable.

Suppose that we are given a set of trajectory data of the nonholonomic system $(q^i, \dot{q}^i) \in TQ$, and the Lagrangian $L : TQ \to \mathbb{R}$, but we are not given the constraints in Definition 1. The goal is to find the constraint distribution, or the collection of all horizontal space $\mathcal{D} = \sqcup_q H_q$ (explained in Section 3.1) of the system, and the full dynamics of the system.

3. Background

We will discuss some mathematical preliminaries for nonholonomic systems, symmetry, Lagrangian reduction, and the Hamel's formalism for mechanical systems on a moving basis.

3.1. Connections and the Horizontal Lift

A connection is a mathematical object that helps one understand a nonholonomic constraint as a map between spaces, which leads to the following discussion of the distribution under constraints, and the horizontal lift as an action of mapping any velocity to the constraint distribution. We introduce the concept of fiber bundle first to give a general geometric structure for nonholonomic systems.

Definition 2 (Lee (2000) Page 268) *Let Q_s and Q_r be topological spaces. A **fiber bundle** over Q_s with model fiber Q_r is a topological space Q with a surjective continuous map $\pi : Q \to Q_s$ with the property that for each $s \in Q_s$, there exist a neighborhood U of s in Q_s and a homeomorphism $\Phi : \pi^{-1}(U) \to U \times Q_r$, called a local trivialization of Q over U.*

Considering the tangent map $T_q\pi$, if we split T_qQ into separate parts $T_qQ = H_q \times V_q$, then with $T_q\pi : T_qQ \to H_q$, TQ is a fiber bundle, where the fiber V_q lies in the kernel of the map $T_q\pi$. Usually we denote H_q the **horizontal space**, V_q the **vertical space**, and $T_q\pi$ the **horizontal lift** hor.

Definition 3 (Bloch (2015)) *An **Ehresmann connection** A is a vector-valued one-form on Q that satisfies: (1) A is vertical valued: $A_q : T_qQ \to V_q$ is a linear map for each point $q \in Q$. (2) A is a projection: $A(v_q) = v_q$ for all $v_q \in V_q$.*

One can see that H_q is the kernel of A_q. Suppose the configuration space Q, or $Q_r \times Q_s$ has coordinates (r^α, s^a) [1]. Then we represent the connection as

$$A = \omega^a \frac{\partial}{\partial s^a}, \text{ where } \omega^a(q) = ds^a + \mathcal{A}^a_\alpha(r,s)dr^\alpha, \tag{3}$$

Then for a given $v_q = \dot{r}^\alpha \frac{\partial}{\partial r^\alpha} + \dot{s}^a \frac{\partial}{\partial s^a}$ in T_qQ,

$$A_q(v_q) = (\dot{s}^a + \mathcal{A}^a_\alpha \dot{r}^\alpha)\frac{\partial}{\partial s^a}. \tag{4}$$

We can see that A is a tangent projection map and gives a fiber bundle T_qQ. Correspondingly, the **horizontal lift** can be defined by subtracting the vertical component from the vector v_q,

$$\text{hor}\, v_q = v_q - A_q(v_q) = \dot{r}^\alpha \frac{\partial}{\partial r^\alpha} - \mathcal{A}^a_\alpha \dot{r}^\alpha \frac{\partial}{\partial s^a}. \tag{5}$$

3.2. Matrix Representation of Connections and the Horizontal Lift

One way to represent the connection A is through matrix operators[2], by considering each differential form ω^a as an operator in the row space, mapping a vector in the column space to a real value, and then the value is assigned to $\frac{\partial}{\partial s^a}$. More specifically, if we let s and r also represent the dimensionality of Q_s and Q_r,

$$A = \begin{bmatrix} I^{s\times s} & \mathcal{A}^{s\times r} \\ \mathbf{0}^{r\times s} & \mathbf{0}^{r\times r} \end{bmatrix}, \tag{6}$$

where the previous notation $\mathcal{A}^i_j$ in Section 3.1 simply represents the element of $\mathcal{A}$ at the i^{th} row and j^{th} column, and A is the **reduced row echelon form** of matrix a in Definition 1. For a random velocity $v_q = \begin{bmatrix}\dot{s} & \dot{r}\end{bmatrix}^T$ in T_qQ,

$$A_q(v_q) = \begin{bmatrix} \dot{s} + \mathcal{A}(r,s)\dot{r} \\ \mathbf{0}^{r\times 1} \end{bmatrix} \in V_q\,. \tag{7}$$

Similarly, we can represent the horizontal map by subtracting A from I,

$$\text{hor} = I - A = \begin{bmatrix} \mathbf{0}^{s\times s} & -\mathcal{A}^{s\times r} \\ \mathbf{0}^{r\times s} & I^{r\times r} \end{bmatrix}, \tag{8}$$

1. In this paper we assume Einstein summation convention for simplicity.
2. This is also what we write in computer programs.

which can act on v_q and result in

$$\mathrm{hor}_q\, v_q = \begin{bmatrix} -\mathcal{A}(r,s)\dot{r} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} -\mathcal{A}(r,s) \\ I^{r\times r} \end{bmatrix} \dot{r} \in H_q\,. \tag{9}$$

Since $\dot{r}$ represents any vector of r dimension, we have

$$H_q = \mathrm{Col}\Big(\begin{bmatrix} -\mathcal{A}(r,s) \\ I^{r\times r} \end{bmatrix} \Big). \tag{10}$$

Proposition 4 *(1) Both A and hor are projections. (2) They lie in the kernel of each other. (3) Both V_q and H_q are unique given a nonholonomic constraint* (1).

Proof *It is easy to check that (1) $A^2 = A$ and* $\mathrm{hor} \circ \mathrm{hor} = \mathrm{hor}$, *and (2)* $A\,\mathrm{hor} = \mathrm{hor}\, A = 0$. *For (3), consider that for any given constraint $a(q)$, its reduced row echelon form A_q is unique, so* hor *is unique. Therefore, V_q and H_q are unique.* ■

3.3. Reduction to the Lie Algebra

Now suppose the system is partially evolving on a Lie group (refer to Appendix A for a more rigorous definition of vector fields, Lie groups, Lie algebras), and that can split the configuration space as $Q/G \times G$ with points (r, g) (either globally or in a local trivialization) in contrast to $Q_r \times Q_s$ with points (r, s). Suppose that the basis $e_a \in \mathfrak{g}$, $a = 1, \ldots, k$ spans the entire Lie algebra, so all the vectors on G generated by left translation $L_g^* e_a$, $g \in G$ form a set of left-invariant Lie algebra vector fields. Rewriting the horizontal space (10) in the coordinates corresponding to $\frac{\partial}{\partial r^\alpha}$ and $L_g^* e_a$, we get

$$u_\alpha = \frac{\partial}{\partial r^\alpha} - \mathcal{A}(r,g) L_g^* e_a, \quad \alpha = 1, \ldots, \sigma. \tag{11}$$

which can be seen as a new set of coordinates that span either partially or the entire constraint distribution $\mathcal{D}$, and $\mathcal{A}$ becomes a $\mathfrak{g}$-valued function. For the part of vector field that is G-invariant, we can still represent it by the vector fields

$$u_{\sigma+a} = L_g^* e_a, \quad a = 1, \ldots, k. \tag{12}$$

With u_i for $i = 1, \ldots, \sigma + k$, we can represent any vector on $TQ/G \times \mathfrak{g}$. Therefore, these vector fields are sufficient to represent the entire vector field of a nonholonomic system with connection A and with a Lagrangian that is invariant under the group action L_g.

The components of a velocity vector relative to the basis $(u_\alpha, u_{\sigma+a})$ can be represented as $(\dot{r}^\alpha, \Omega^a)$, where $\Omega^a = \xi^a + \mathcal{A}^a_\alpha \dot{r^\alpha}$ and $\xi = L^*_{g^{-1}} \dot{g}$ are body velocities with and without connections. Here we can understand $(\dot{r}^\alpha, \Omega^a)$ as the horizontal and vertical velocities (recall (4), (5)). The G-invariant Lagrangian $L(r, \dot{r}, g, \dot{g})$ can be reduced to $l(r, \dot{r}, \Omega)$ or $l(r, \dot{r}, \xi)$ by pulling back the global velocities in TG to the body velocities in $\mathfrak{g}$. Furthermore, the system must satisfy the Hamel equations:

3.4. The Hamel Equations

The original Hamel equations were derived in a simpler setting, in particular to deal with the Euler-Lagrange equations on a moving basis. While the above description of the reduced nonholonomic systems can be interpreted in the moving basis (11) and (12), there are some differences in coordinate selections compared to the original Hamel equations. For readability of this paper, we introduce the original Hamel equations and let the reader refer to Bloch et al. (2009) for a more detailed derivation for the nonholonomic version.

Let $v = (v^1, \dots, v^n) \in \mathbb{R}$ be components of the velocity vector $\dot{q} \in TQ$ relative to the basis $u_1, \dots, u_n$, i.e., $\dot{q} = v^i u_i(q)$, so that the Lagrangian $L(q, \dot{q}) = L(q, v^i u_i(q))$ can be reduced to $l(q, v)$.

Theorem 5 (Hamel Equations) *The evolution of the variables (q, v) satisfying Hamilton's principle is governed by the Hamel equations*

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial l}{\partial v} = \left[v, \frac{\partial l}{\partial v}\right]_q^* + u[l] \tag{13}$$

In the coordinate form,

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial l}{\partial v^i} = c_{ji}^m \frac{\partial l}{\partial v^m} v^j + u_i[l], \tag{14}$$

where the Lie algebra structure constants c_{ji}^m are defined by

$$[u_j, u_i] = c_{ji}^m u_m. \tag{15}$$

The velocities v^i in our case are $(\dot{r}, \xi^a)$ or $(\dot{r}, \Omega^a)$. While it is non-trivial to obtain the same formalism for nonholonomic systems with symmetry, one can in general apply the coordinates defined in (11) and (12) to Theorem 5 to generate the full dynamics of momentum $(\frac{\partial l}{\partial \dot{r}}, \frac{\partial l}{\partial \xi})$ or $(\frac{\partial l}{\partial \dot{r}}, \frac{\partial l}{\partial \Omega})$. In the rest of the paper, we also denote them $p_{\dot{r}}$, p_ξ and p_Ω. Notice that the difference between the dynamics of $(p_{\dot{r}}, p_\xi)$ and $(p_{\dot{r}}, p_\Omega)$ is that the former gives a reduced system with symmetry while the latter gives a reduced system with symmetry and also with the connection.

4. Learning Nonholonomic Dynamics

Here we state our main contribution of this paper: a general procedure to learn nonholonomic dynamics with known Lagrangian $L(q, \dot{q})$ and unknown constraint a. From now on, we represent any element parameterized by a neural network with a tilde symbol $\tilde{\cdot}$.

4.1. Step One: Data Reduction to Lie Algebra

For a general nonholonomic systems on TQ with data set $(q, \dot{q})^i$, we first split the system into the product of a known Lie group G and the quotient of TQ by this group TQ/G, so the data is represented as $(r, \dot{r}, g, \dot{g})^i$ (explained in Section 3.3). For each data point, we can pull back the group elements by $\xi^i = L^*_{(g^i)^{-1}} g^i$, and the data set becomes $(r, \dot{r}, \xi)^i$.

4.2. Step Two: Generate Moving Basis and Quasi-velocities

Since we already chose the Lie group G, we can generate the Lie algebra vector field by push forward of a frame e_a defined in Section 3.3, $L_g^* e_a$. This also corresponds to the vector fields $u_{\sigma+a}$ defined in (12). It can be seen that $u_{\sigma+a}$ is known once a group G is chosen.

Now to have u_α defined in (11), we parameterize $\mathcal{A}(r,g)$ with a neural network, so that

$$\tilde{u}_\alpha = \frac{\partial}{\partial r^\alpha} - \tilde{\mathcal{A}}(r,g) L_g^* e_a. \tag{16}$$

Once we have the parameterized $\tilde{u}_\alpha$ and $u_{\sigma+a}$, we can compute the parameterized quasi-velocities $\tilde{\Omega}^a = \xi^a + \tilde{\mathcal{A}}_\alpha^a \dot{r}^\alpha$.

4.3. Step Three: Compute Momentum and Generate Dynamics

Knowing the Lagrangian $L(q,\dot{q})$, we can first reduce it to $l(r,\dot{r},\xi) = L_{g^{-1}}^* L(q,\dot{q})$. We check by the chain rule that $\frac{\partial l}{\partial \xi} = \frac{\partial l}{\partial(\xi+\tilde{\mathcal{A}}\dot{r})}\frac{\partial(\xi+\tilde{\mathcal{A}}\dot{r})}{\partial \xi} = \frac{\partial l}{\partial \tilde{\Omega}}$. So the momentum $\{p_\Omega^i\}$ can be directly calculated from the known $l(r,\dot{r},\xi)$ and ξ^i. The momentum $\{p_r^i\}$ can be calculated by the chain rule

$$\tilde{p}_{\dot{r}} = \frac{\partial l}{\partial \dot{r}} = \frac{\partial l_{\dot{r}}}{\partial \dot{r}} + \left\langle \frac{\partial l}{\partial \xi}, -\tilde{\mathcal{A}} \right\rangle, \tag{17}$$

where $l_{\dot{r}}$ represents the part of function in l that does not contain ξ. An interesting observation here is that if one takes the time derivative of the second part of the equation $\left\langle \frac{\partial l}{\partial \xi}, -\tilde{\mathcal{A}} \right\rangle$, it is equivalent to the nonholonomic momentum equation. However, this discussion is beyond the scope of this paper.
.

4.4. Step Four: Training with Neural ODE

With the data points $\{p^i\} = \{(p_{\dot{r}}, p_\Omega)^i\}$, and the dynamics calculated by substituting u_α, $u_{\sigma+a}$ into Theorem 5, we can train the dynamics as a neural ODE, with loss

$$\mathcal{L} = \sum_i \left\| \int_{t_i}^{t_{i+1}} \frac{\mathrm{d}}{\mathrm{d}t} \tilde{p}^i \mathrm{d}t - p^{i+1} \right\|_2^2. \tag{18}$$

In this formulation, the dynamics is generated through parameterizing only the nonholonomic constraint, and the constraint is fully represented in an unconstrained ODE. After training, the constraint is therefore implicitly discovered as the output of the network $\tilde{A}(r,g)$. To guarantee local convergence of this training process, we prove the following theorem.

Theorem 6 (Unique Dynamics Generated by $\mathcal{A}$) *Consider a nonholonomic system with left invariant Lagrangian $L(q,\dot{q})$, and nonholonomic constraint $a(q)$, evolving on $TQ/G \times G$. Define its coordinates as in (11) and (12), then its dynamics generated by (13) is defined locally uniquely by the connection.*

We leave the proof of the theorem in Appendix B for readers who are interested. This theorem guarantees that there is a minimum for the loss function (18).

5. Learning the Rolling Disk Dynamics

To demonstrate our approach in a minimal and intuitive example, we discuss our approach applied to the rolling disk dynamics Bloch (2015). The rolling disk dynamics, by definition, is the dynamics of a disk-like object rolling on a flat surface without slipping. In our paper, we also assume the disk does not fall. The configuration space is denoted by $(\theta, \varphi, x, y, \dot{\theta}, \dot{\varphi}, \dot{x}, \dot{y})$, where θ is the angle of one fixed point on the disk relative to the horizontal line, φ is the rolling direction of the disk relative to the horizontal coordinate of the plane, and x, y denotes the location of the disk on the plane. The Lagrangian is given by the kinetic energy

$$L(q, \dot{q}) = \frac{1}{2} I \dot{\theta}^2 + \frac{1}{2} J \dot{\varphi}^2 + \frac{1}{2} m(\dot{x}^2 + \dot{y}^2), \tag{19}$$

where I is the inertia of the disk about its center of rotation, J is the inertia of the disk about φ, and m is the mass of the disk. The non-slip condition gives the constraint (2), which is non-integrable.

5.1. The $\mathbb{R} \times SE(2)$ version

For demonstration purposes, we derive the learning process of the system on the configuration space $\mathbb{R} \times SE(2)$, namely $r = \theta$ and $g = (\varphi, x, y)$. The group action of $SE(2)$ gives the left-invariant Lie algebra vector field

$$u_2 = \frac{\partial}{\partial \varphi} - y \frac{\partial}{\partial x} + x \frac{\partial}{\partial y}, \tag{20}$$

$$u_3 = \frac{\partial}{\partial x}, \tag{21}$$

$$u_4 = \frac{\partial}{\partial y}, \tag{22}$$

corresponding to (12). So for a given velocity at time t_i, $(g, \dot{g})^i$, we compute the Lie algebra element as $(\xi^1, \xi^2, \xi^3)^i = (\dot{\varphi}, \dot{x} + y\dot{\varphi}, \dot{y} - x\dot{\varphi})^i$.

Now we come to Step Two, where we parameterize $\mathcal{A}(r, g)$ and define u_1.

$$\tilde{u}_1 = \frac{\partial}{\partial r} - \tilde{\mathcal{A}}^a(r, g) u_{1+a}, \quad a = 1, 2, 3. \tag{23}$$

Together with (ξ_1, ξ_2, ξ_3), we can compute the parameterized quasi velocities $\Omega^a = \xi^a + \tilde{\mathcal{A}}^a \dot{\theta}$.

Then we compute the dynamics as in Step Three. Substituting ξ^a to the Lagrangian, we get the reduced Lagrangian,

$$l(\theta, \dot{\theta}, \xi^1, \xi^2, \xi^3) = \frac{1}{2} I \dot{\theta}^2 + \frac{1}{2} J (\xi^1)^2 + \frac{1}{2} m\big((-y\xi^1 + \xi^2)^2 + (x\xi^1 + \xi^3)^2\big), \tag{24}$$

so we can calculate the momentum by

$$p_2 = \frac{\partial l}{\partial \Omega^1} = \frac{\partial l}{\partial \xi^1} = J\xi^1 - my(-y\xi^1 + \xi^2) + mx(x\xi^2 + \xi^3), \tag{25}$$

$$p_3 = \frac{\partial l}{\partial \Omega^2} = \frac{\partial l}{\partial \xi^2} = m(-y\xi^1 + \xi^2), \tag{26}$$

$$p_4 = \frac{\partial l}{\partial \Omega^3} = \frac{\partial l}{\partial \xi^3} = m(x\xi^1 + \xi^3), \tag{27}$$

$$p_1 = \frac{\partial l}{\partial \dot{\theta}} = I\dot{\theta}. \tag{28}$$

Note that these are the ground truth momenta. In the parameterized momenta, p_2, p_3, p_4 stay the same, while we use (17) to compute p_1. Then we compute the Lie bracket coefficients with the coordinates u_i.

$$[u_2, \tilde{u}_1] = -(\frac{\partial \tilde{\mathcal{A}}^a}{\partial \varphi} - y\frac{\partial \tilde{\mathcal{A}}^a}{\partial x} + x\frac{\partial \tilde{\mathcal{A}}^a}{\partial y})u_{1+a} + \tilde{\mathcal{A}}^a u_{1+a}[u_2] = \tilde{c}^m_{21}u_m, \tag{29}$$

$$[u_2, u_3] = -\frac{\partial}{\partial y} = -1 \cdot u_4 = c^4_{23}u_4, \tag{30}$$

$$[u_2, u_4] = \frac{\partial}{\partial x} = 1 \cdot u_3 = c^3_{24}u_3, \tag{31}$$

$$\dots$$

We omit the entire calculation for simplicity. The general idea is to calculate c^m_{ji}, where some of them are parameterized and some of them are not. Then we generate the dynamics based on (13), where we see v^1 as $\dot{r}$ and v^2, v^3, v^4 as $\Omega^1, \Omega^2, \Omega^3$. In Appendix C, we derive the dynamics with known constraints to familiarize the readers with the Hamel equations.

It is worth mentioning that this approach is agnostic to the choice of group G. That is, one will still recover the momentum dynamics by choosing the configuration $\mathbb{R} \times S^1 \times \mathbb{R}^2$ with coordinate $(\varphi, \dot{\varphi}, g, \dot{g})$ and $g = (\theta, x, y)$. However, you are evaluating a different momentum about a different moving basis, and the map $\mathcal{A}$ will appear to be different. By changing coordinates, the system will still follow the same trajectory.

5.2. Results

The neural network chosen for this task is a Feed-forward Neural Network[3] with 3 hidden layers and a hidden dimension of 20. Each hidden layer has a linear layer and a nonlinear layer of the *Sine()* function, and the last nonlinear layer is chosen to be the *ELU()* function that allows near zero output. The training set is calculated as described in Section 5 with random system parameters I, J, m, R and initial conditions, in the time ranging from 0 to 20 sec with a step size of 0.01 sec. The entire training process involves 2000 epochs. The program was carried out with *PyTorch* and *torchdiffeq* Chen et al. (2019).

5.2.1. LEARNING ON $\mathbb{R} \times SE(2)$

The results of our training process carried out on $\mathbb{R} \times SE(2)$ are shown in Figure 1. The ground truth $\mathcal{A}^a(r, g)$ is defined by $\mathcal{A}^1(\theta, \varphi, x, y) = 0$, $\mathcal{A}^2(\theta, \varphi, x, y) = -R\cos\varphi$, and $\mathcal{A}^3(\theta, \varphi, x, y) = R\sin\varphi$. We can see that the learned momentum converges to the ground truth, and the learned constraint $\tilde{\mathcal{A}}^a(r, g)$ converges to $\mathcal{A}^a(r, g)$ by evaluation. Notice that these constraints were learned completely from the data set $\{(r, \dot{r}, g, \dot{g})^i\}$ without knowing the ground truth $\mathcal{A}^a(r, g)$. Therefore, we call this result "constraint discovery". Any baseline methods that parameterize the dynamics directly do not imply any information about the constraint or the symmetry of the system.

Notice that the constraint does not perfectly match the ground truth (this is further discussed in our conclusions). We think that this is due to the following: the numerical integration method introduces some error; the training process only includes one trajectory with one initial condition, so

3. Code can be found on the GitHub page.

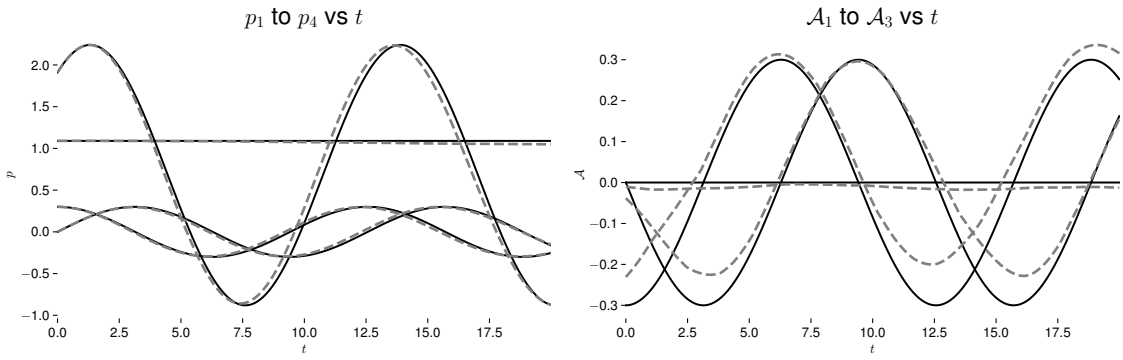

Figure 1: The above figure shows the training result after 2000 epochs. The sub plot on the left shows the ground truth momenta (solid lines) and the learned dynamics (dashed lines) generated by a numerical integrator. The sub plot on the right shows the ground truth $\mathcal{A}(r, g)$ (solid lines) and the parameterized $\tilde{\mathcal{A}}(r, g)$ (dashed lines) evaluated after training.

in some cases the momentum trajectory does not indicate enough about the constraint. Actually, although the entire vector field is unique according to Theorem 6, a single trajectory can be generated by different vector fields. However, based on the uniqueness of the vector field, the convergence can be improved to reach a sufficiently small error by training on different trajectories.

6. Conclusions and Future Work

In this paper we proposed a general learning approach for learning the dynamics of a nonholonomic system subject to unknown constraints, where the results imply that the parameterized constraint is implicitly learned by training the system in a neural ODE fashion represented by Hamel equations. This is one of the first approaches to learning a nonholonomic system with symmetry and there are some open problems that the authors seek to improve in future work.

Unknown Lagrangian In our proposed method, we assumed that the Lagrangian is known while the nonholonomic constraint is unknown. This restricts us to learn nonholonomic systems with known Lagrangian. However, in real world applications, generally one needs to identify the Lagrangian when the provided model is inaccurate. Finding an approach for learning the Lagrangian together with the constraint would therefore be an interesting direction.

Impact of Training Set The proposed method in this paper aims at learning the constraints based on a single trajectory of many points starting from the same initial condition. Therefore, although the chosen initial condition in the example is random, a badly chosen initial condition may lead to results that are not perfectly converging. We need to find a way to learn from different initial conditions, without relying on a single trajectory, to address this problem. Furthermore, we find it interesting to study the global convergence of the network.

Numerical Integration error Since the entire dynamics of the system is calculated by numerical ODE integration, the error caused by the integrator cannot be avoided. We expect to improve future results by considering discrete nonholonomic dynamics Cortés and Martínez (2001); Colombo et al. (2015) with a different integration approach that preserves symmetry properties, or the momentum

equation. There are some approaches for discrete Lagrangian systems with external forces Hansen et al. (2025).

Application to Real-world Systems Nonholonomic systems exist broadly in the real world, including robotics, vehicle dynamics, control theory, aerospace, marine, and computer graphics. Having some observed data from those systems and discovering the constraint would be an interesting application of the proposed learning process.

Acknowledgments

Partially supported by NSF grant DMS-2103026, and AFOSR grants FA 9550-22-1-0215 and FA 9550-23-1-0400.

References

Anthony M. Bloch. *Nonholonomic Mechanics*, pages 235–313. Springer New York, New York, NY, 2015. ISBN 978-1-4939-3017-3. doi: 10.1007/978-1-4939-3017-3_5. URL https://doi.org/10.1007/978-1-4939-3017-3_5.

Anthony M. Bloch, Jerrold E. Marsden, and Dmitry V. Zenkov. Quasivelocities and symmetries in non-holonomic systems. *Dynamical Systems*, 24(2):187–222, 2009. doi: 10.1080/14689360802609344. URL https://doi.org/10.1080/14689360802609344.

Anthony M. Bloch, Peter E. Crouch, and Tudor S. Ratiu. Symmetric discrete optimal control and deep learning, 2024. URL https://arxiv.org/abs/2404.06556.

Renyi Chen and Molei Tao. Data-driven prediction of general hamiltonian dynamics via learning exactly-symplectic maps. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1717–1727. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/chen21r.html.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations, 2019. URL https://arxiv.org/abs/1806.07366.

Yuhan Chen, Takashi Matsubara, and Takaharu Yaguchi. Neural symplectic form: Learning hamiltonian equations on general coordinate systems. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 16659–16670. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/8b519f198dd26772e3e82874826b04aa-Paper.pdf.

Glen Chou, Dmitry Berenson, and Necmiye Ozay. Learning constraints from demonstrations. In Marco Morales, Lydia Tapia, Gildardo Sánchez-Ante, and Seth Hutchinson, editors, *Algorithmic Foundations of Robotics XIII*, pages 228–245, Cham, 2020. Springer International Publishing. ISBN 978-3-030-44051-0.

Leonardo Colombo, Rohit Gupta, Anthony Bloch, and David Martín de Diego. Variational discretization for optimal control problems of nonholonomic mechanical systems. In *2015 54th*

IEEE Conference on Decision and Control (CDC), pages 4047–4052, 2015. doi: 10.1109/CDC.2015.7402849.

J. Cortés and S. Martínez. Non-holonomic integrators. *Nonlinearity*, 14(5):1365, aug 2001. doi: 10.1088/0951-7715/14/5/322. URL https://doi.org/10.1088/0951-7715/14/5/322.

Nima Dehmamy, Robin Walters, Yanchen Liu, Dashun Wang, and Rose Yu. Automatic symmetry discovery with lie algebra convolutional network. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 2503–2515. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/148148d62be67e0916a833931bd32b26-Paper.pdf.

Christopher Eldred, François Gay-Balmaz, Sofiia Huraka, and Vakhtang Putkaradze. Lie-poisson neural networks (lpnets): Data-based computing of hamiltonian systems with symmetries, 2023. URL https://arxiv.org/abs/2308.15349.

Christopher Eldred, François Gay-Balmaz, and Vakhtang Putkaradze. Clpnets: Coupled lie-poisson neural networks for multi-part hamiltonian systems with symmetries, 2024. URL https://arxiv.org/abs/2408.16160.

Subhroshekhar Ghosh, Aaron Y. R. Low, Yong Sheng Soh, Zhuohang Feng, and Brendan K. Y. Tan. Dictionary learning under symmetries via group representations, 2023. URL https://arxiv.org/abs/2305.19557.

Anthony Gruber, Kookjin Lee, and Nathaniel Trask. Reversible and irreversible bracket-based dynamics for deep graph neural networks, 2023. URL https://arxiv.org/abs/2305.15616.

Martine Dyring Hansen, Elena Celledoni, and Benjamin Kwanen Tapley. Learning mechanical systems from real-world data using discrete forced lagrangian dynamics, 2025. URL https://arxiv.org/abs/2505.20370.

Haojie Huang, Dian Wang, Arsh Tangri, Robin Walters, and Robert Platt. Leveraging symmetries in pick and place. *The International Journal of Robotics Research*, 43(4):550–571, 2024. doi: 10.1177/02783649231225775. URL https://doi.org/10.1177/02783649231225775.

John M. Lee. Introduction to smooth manifolds. 2000.

Hsiu-Chin Lin, Matthew Howard, and Sethu Vijayakumar. Learning null space projections. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2613–2619, 2015. doi: 10.1109/ICRA.2015.7139551.

Miguel Vaquero, Jorge Cortés, and David Martín de Diego. Symmetry preservation in hamiltonian systems: Simulation and learning, 2023. URL https://arxiv.org/abs/2308.16331.

Alan John Varghese, Zhen Zhang, and George Em Karniadakis. Sympgnns: Symplectic graph neural networks for identifiying high-dimensional hamiltonian systems and node classification, 2024. URL https://arxiv.org/abs/2408.16698.

Appendix A. Vector Fields, Lie Groups, Lie Algebras

Let Q be a smooth n-manifold and let $q \in Q$. Then T_qQ is an n-dimensional vector space. Let the coordinate vectors $\frac{\partial}{\partial q^1}, \ldots, \frac{\partial}{\partial q^n}$ form a basis for T_qQ. A tangent vector $v \in T_qQ$ can be written uniquely as a linear combination (see Lee (2000) page 61)

$$v = v^i \frac{\partial}{\partial q^i}\bigg|_q, \tag{32}$$

where the coeffcients $(v^1, \ldots, v^n)$ are called the components of v. A vector field on Q is a continuous map $u : Q \to TQ$ with the property that $u(q) \in T_qQ$ for each $q \in Q$. Let u_i, $i = 1, \ldots, n$ be smooth independent local vector fields defined by (see Bloch et al. (2009))

$$u_i(q) = \psi_i^j(q) \frac{\partial}{\partial q^j}, \quad i, j = 1, \ldots, n. \tag{33}$$

Then the u_i form a basis for TQ. With the u_i as a basis, and again using v^i as components as in (32), a velocity vector $\dot{q} \in T_qQ$ can be written as

$$\dot{q} = v^i u_i(q), \tag{34}$$

then for a function $L : TQ \to \mathbb{R}$, we can define the reduced Lagrangian $l : TQ \to \mathbb{R}$

$$l(q, v) := L(q, v^i u_i(q)). \tag{35}$$

We can view the vector fields u_i as operators and define the directional derivatives $u_i[l]$ as

$$u_i[l] = \psi_i^j \frac{\partial l}{\partial q^j}. \tag{36}$$

The evolution of the variables (q, v) satisfying Hamilton's principle is governed by the Hamel equations (13) (Bloch et al. (2009)).

Now we consider systems on Lie groups. A Lie group is a smooth manifold G that is also a group in the algebraic sense (Lee (2000) Chapter 7). If G is a Lie group, any element $g \in G$ defines maps $L_g : G \to G$ called left translation and $L_g^* : TG \to TG$ called pushforward,

$$L_g(h) = gh, \quad h \in G, \tag{37}$$

$$L_g^*(v_h) = (dL_g)_h(v_h), \quad v_h \in T_hG. \tag{38}$$

Let X, Y be smooth vector fields on Q, and $X = X^i \frac{\partial}{\partial q^i}$, $Y = Y^i \frac{\partial}{\partial q^j}$. Then a Lie bracket is defined by (Lee (2000) Proposition 8.26)

$$[X, Y] = \left(X^i \frac{\partial Y^j}{\partial q^i} - Y^i \frac{\partial X^j}{\partial q^i} \right) \frac{\partial}{\partial x^j}. \tag{39}$$

A Lie algebra is a real vector space $\mathfrak{g}$ with a Lie bracket $\mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ that satisfies bilinearity, antisymmetry, and the Jacobi identity (Lee (2000) Page 190). The Lie algebra of all smooth left-invariant vector field is called the Lie algebra of G, which is also isomorphic to T_eG.

Appendix B. Proof of Theorem 6

To prove this theorem, it suffices to say that under some small perturbation of $\mathcal{A}(r, g)$, the generated dynamics is different. To make the proof cleaner, we proceed in a more general setting on the configuration space Q. Define a basis for Q

$$u_\alpha(q) = \psi^j_\alpha(q)\frac{\partial}{\partial q^j}, \quad j = 1, \dots, n, \quad \alpha = 1, \dots, \sigma \tag{40}$$

$$u_{\sigma+a}(q) = \psi^j_{\sigma+a}(q)\frac{\partial}{\partial q^j}, \quad j = m, \dots, n \quad a = 1, \dots, n-\sigma \tag{41}$$

and the velocity is

$$\dot{q} = v^\alpha u_\alpha + v^{\sigma+a}u_{\sigma+a}. \tag{42}$$

Assume that the reduced Lagrangian is $l(q, v)$ and the nominal dynamics is generated by

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial l}{\partial v^i} = c^m_{ji}\frac{\partial l}{\partial v^m}v^j + u_i[l]. \tag{43}$$

Then, we assume that if we perturb the coordinates by a function δ of q

$$u'_\alpha(q) = \psi^j_\alpha\frac{\partial}{\partial q^j} - \delta^{\sigma+a}_\alpha u_{\sigma+a}, \tag{44}$$

the generated dynamics stays the same. The velocity after this modification is

$$\dot{q} = v^\alpha u'_\alpha + w^{\sigma+a}u_{\sigma+a} := v^\alpha u'_\alpha + (v^{\sigma+a} + \delta^{\sigma+a})u_{\sigma+a}, \tag{45}$$

where $\delta^{\sigma+a} = \sum_\alpha \delta^{\sigma+a}_\alpha$. By the chain rule, $\frac{\partial l}{\partial v^{\sigma+a}} = \frac{\partial l}{\partial w^{\sigma+a}}$. Now considering the case $i = \alpha, j = \sigma + a$ in Hamel equations,

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial l}{\partial v^\alpha} = c'^m_{\sigma+a,\alpha}\frac{\partial l}{\partial v^m}w^{\sigma+a} + c'^m_{\beta,\alpha}\frac{\partial l}{\partial v^m}v^\beta + u'_\alpha[l], \tag{46}$$

one can see that if $u_\alpha[l] = u'_\alpha[l]$, then $\delta^{\sigma+a}_\alpha u_{\sigma+a}[l] = 0$. So if l is a function of $q^i, i = m, \dots, n$, then $\delta^{\sigma+a}_\alpha = 0$. In the case where l is not a function of $q^i, i = m, \dots, n$, one needs to check the rest of the equation. Now we check $c'^m_{\sigma+a,\alpha}$

$$[u_{\sigma+a}, u'_\alpha] = [u_{\sigma+a}, u_\alpha - \delta^{\sigma+a}_\alpha u_{\sigma+a}] \tag{47}$$

$$= [u_{\sigma+a}, u_\alpha] + [u_{\sigma+a}, -\delta^{\sigma+a}_\alpha u_{\sigma+a}] \tag{48}$$

$$:= c^m_{\sigma+a,\alpha}u_m + b_{\sigma+a,\alpha}u_{\sigma+a}; \tag{49}$$

compared to the nominal case, there is an extra term $b_{\sigma+a,\alpha}u_{\sigma+a}$. We examine this term

$$[u_{\sigma+a}, -\delta^{\sigma+a}_\alpha u_{\sigma+a}] = -\psi^j_{\sigma+a}\frac{\partial\delta^{\sigma+a}_\alpha\psi^l_{\sigma+a}}{\partial q^j}\frac{\partial}{\partial q^l} + \delta^{\sigma+a}_\alpha\psi^j_{\sigma+a}\frac{\partial\psi^l_{\sigma+a}}{\partial q^j}\frac{\partial}{\partial q^l} \tag{50}$$

$$= -\psi^j_{\sigma+a}\Big(\frac{\partial\delta^{\sigma+a}_\alpha}{\partial q^j}\psi^l_{\sigma+a} + \delta^{\sigma+a}_\alpha\frac{\partial\psi^l_{\sigma+a}}{\partial q^j}\Big)\frac{\partial}{\partial q^l} + \delta^{\sigma+a}_\alpha\psi^j_{\sigma+a}\frac{\partial\psi^l_{\sigma+a}}{\partial q^j}\frac{\partial}{\partial q^l} \tag{51}$$

$$= -\psi^j_{\sigma+a}\frac{\partial\delta^{\sigma+a}_\alpha}{\partial q^j}\psi^l_{\sigma+a}\frac{\partial}{\partial q^l} \tag{52}$$

$$= -\psi^j_{\sigma+a}\frac{\partial\delta^{\sigma+a}_\alpha}{\partial q^j}u_{\sigma+a} \tag{53}$$

$$= b_{\sigma+a,\alpha}u_{\sigma+a}. \tag{54}$$

For the term $c'^{m}_{\sigma+a,\alpha} p_m w^{\sigma+a}$ to equal the nominal $c^{m}_{\sigma+a,\alpha} p_m v^{\sigma+a}$, we subtract them and get

$$c'^{m}_{\sigma+a,\alpha} \frac{\partial l}{\partial v^m} \delta^{\sigma+a} + b_{\sigma+a,\alpha} \frac{\partial l}{\partial v^{\sigma+a}} (v^{\sigma+a} + \delta^{\sigma+a}) = 0, \tag{55}$$

so for a small $\delta^{\sigma+a}$,

$$b_{\sigma+a,\alpha} \frac{\partial l}{\partial v^{\sigma+a}} v^{\sigma+a} = 0, \tag{56}$$

meaning

$$b_{\sigma+a,\alpha} = 0 \tag{57}$$

$$\frac{\partial \delta^{\sigma+a}_{\alpha}}{\partial q^j} = 0, \tag{58}$$

so $\delta^{\sigma+a}_{\alpha}$ is constant relative to q, which contradicts our assumption. Therefore, the perturbed coordinate must generate a different dynamics unless $\delta = 0$.

As a general proof, this applies to our case in (11), (12). Together with Proposition 4 (3), a nonholonomic system with constraint (1) generates a locally unique dynamics.

Appendix C. The Rolling Disk Dynamics with Known Constraint

In Section 5.1, we already derived the Lie algebra vector field u_2, u_3, u_4 and the parameterized horizontal vector field $\tilde{u}_1$.

With known constraint (2), we have

$$u_1 = \frac{\partial}{\partial \theta} + R\cos\varphi \frac{\partial}{\partial x} + R\sin\varphi \frac{\partial}{\partial y}, \tag{59}$$

so a vector that lies in the constraint distribution can be written

$$\dot{q} = \dot{\theta} u_1 + \Omega^1 u_2 + \Omega^2 u_3 + \Omega^3 u_4, \tag{60}$$

or

$$\dot{q} = \dot{\theta} \frac{\partial}{\partial \theta} + \xi^1 u_2 + \xi^2 u_3 + \xi^3 u_4. \tag{61}$$

Note that ξ^1 to ξ^3 can be calculated directly from data and from u_2 to u_4. Choose

$$\Omega^1 = \dot{\varphi} = \xi^1 + 0, \tag{62}$$

$$\Omega^2 = y\dot{\varphi} = \xi^2 + (-R\cos\varphi\dot{\theta}), \tag{63}$$

$$\Omega^3 = -x\dot{\varphi} = \xi^3 + (-R\sin\varphi\dot{\theta}) \tag{64}$$

so that $\dot{\theta} u_1$ and $\Omega^1 u_2 + \Omega^2 u_3 + \Omega^3 u_4$ span the constraint distribution

$$\mathcal{D} = \{ \frac{\partial}{\partial \theta} + R\cos\varphi \frac{\partial}{\partial x} + R\sin\varphi \frac{\partial}{\partial y}, \frac{\partial}{\partial \varphi} \}. \tag{65}$$

Now we compute Lie brackets.

$$[u_2, u_1] = -R\sin\varphi\frac{\partial}{\partial x} + R\cos\varphi\frac{\partial}{\partial y} - R\cos\varphi\frac{\partial}{\partial y} + R\sin\varphi\frac{\partial}{\partial x} = 0, \tag{66}$$

$$[u_3, u_1] = 0, \tag{67}$$

$$[u_4, u_1] = 0, \tag{68}$$

$$[u_3, u_2] = \frac{\partial}{\partial y} = 1 \cdot u_4 = c_{32}^4 u_4, \tag{69}$$

$$[u_4, u_2] = -\frac{\partial}{\partial x} = -1 \cdot u_3 = c_{42}^3 u_3, \tag{70}$$

$$[u_4, u_3] = 0. \tag{71}$$

The anti-symmetric property of Lie brackets also gives $c_{ij}^m = -c_{ji}^m$. The values of p_1 to p_4 was given in (25). Now using the Hamel equations,

$$\frac{\mathrm{d}}{\mathrm{d}t}p_1 = 0, \tag{72}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}p_2 = c_{32}^4 p_4 \Omega^2 + c_{42}^3 p_3 \Omega^3 = y\dot{\varphi}p_4 + x\dot{\varphi}p_3, \tag{73}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}p_3 = c_{23}^4 p_4 \Omega^1 = -\dot{\varphi}p_4, \tag{74}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}p_4 = c_{24}^3 p_3 \Omega^1 = \dot{\varphi}p_3, \tag{75}$$

and we obtain the dynamics of p_1 to p_4.