Proceedings of Machine Learning Research vol 144:1–14, 2022

Safe Autonomous Navigation for Systems with Learned SE(3) Hamiltonian Dynamics

Zhichao Li* ZHICHAOLI@UCSD.EDU
Thai Duong* TDUONG@UCSD.EDU
Nikolay Atanasov NATANASOV@UCSD.EDU
Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093

Abstract

Safe autonomous navigation in unknown environments is an important problem for ground, aerial, and underwater robots. This paper proposes techniques to learn the dynamics models of a mobile robot from trajectory data and synthesize a tracking controller with safety and stability guarantees. The state of a mobile robot usually contains its position, orientation, and generalized velocity and satisfies Hamilton's equations of motion. Instead of a hand-derived dynamics model, we use a dataset of state-control trajectories to train a translation-equivariant nonlinear Hamiltonian model represented as a neural ordinary differential equation (ODE) network. The learned Hamiltonian model is used to synthesize an energy-shaping passivity-based controller and derive conditions which guarantee safe regulation to a desired reference pose. Finally, we enable adaptive tracking of a desired path, subject to safety constraints obtained from obstacle distance measurements. The trade-off between the system's energy level and the distance to safety constraint violation is used to adaptively govern the reference pose along the desired path. Our safe adaptive controller is demonstrated on a simulated hexarotor robot navigating in unknown complex environments.

Keywords: dynamics learning, neural ODE network, reference governor, safe tracking control

1. Introduction

Designing controllers that handle safety constraints and guarantee system stability is an important problem in safety-critical applications, such as autonomous driving (Ames et al., 2014b; Shalev-Shwartz et al., 2016), walking robots (Ames et al., 2014a) or medical robots (Yip and Camarillo, 2014). Safety depends on the system states, governed by the system dynamics, and the environment constraints. This leads to two requirements for designing provably safe controllers: 1) the availability of an accurate dynamics model of the system and 2) the satisfaction of time-varying safety constraints that are only known at runtime.

The first requirement has motivated many data-driven dynamics learning approaches recently, where machine learning techniques are used to learn dynamical systems, e.g. based on Gaussian processes (Deisenroth et al., 2015; Kabzan et al., 2019) or neural networks (Raissi et al., 2018; Chua et al., 2018). For physical systems, recent works (Lutter et al., 2019; Zhong et al., 2019; Duong and Atanasov, 2021b) design the model architecture to encode Lagrangian or Hamiltonian formulation of robot dynamics (Lurie, 2013; Holm, 2008), which a black-box model might struggle to infer. For Hamiltonian formulation, Zhong et al. (2019) use a differentiable neural ODE solver (Chen et al., 2018) to generate predicted state trajectory. A loss function is back-propagated through the ODE

* These authors contributed equally.

solver to update its parameters. Duong and Atanasov (2021b) extend this approach by imposing both Hamiltonian formulation and the $SE(3)$ constraints on the ODE structure. A Hamiltonian-based model architecture also simplifies the design of a stable regulation or tracking controller by energy shaping (Zhong et al., 2019; Duong and Atanasov, 2021a,b). The key idea of energy-based controller, e.g. interconnection and damping assignment passivity-based control design (IDA-PBC) (Van Der Schaft and Jeltsema, 2014), is to inject additional energy via the control input into the system to achieve a desired total energy, minimized at a desired set point.

The second requirement has gained significant attention in planning and control. Model predictive control (MPC) methods (Borrelli et al., 2017; Grüne and Pannek, 2017; Bravo et al., 2006; Mayne et al., 2000) include safety constraints in an optimization problem, which is typically solved by discretizing time on linearized dynamics. Meanwhile, reachability-based techniques (Herbert et al., 2017; Majumdar and Tedrake, 2017; Kousik et al., 2020) directly work for nonlinear systems and offer strong safety guarantees but have high computation cost and scalability issues for high-dimensional systems. Control barrier function (CBF) with quadratic programming (QP) (Ames et al., 2014b, 2017, 2019) offers an elegant and efficient framework for realtime safe control synthesis. However, it is challenging to construct a valid CBF (Ames et al., 2019) that guarantees the feasibility of the QP problem (Xu, 2018). Given a stabilizing regulation controller, reference governor techniques (Bemporad, 1998; Kolmanovsky et al., 2014; Garone and Nicotra, 2016) maintain a virtual system to adaptively generate the regulation point so that the stabilized system can follow reference command safely. Recent work (Arslan and Koditschek, 2017; Li et al., 2020) can achieve safe navigation in unknown environments, but it is hard to apply them for nonlinear system unless it is feedback linearizable.

In this paper, we consider both requirements for rigid-body systems, such as unmanned ground, aerial and underwater vehicles, whose states are described by their $SE(3)$ pose and generalized velocity. We assume that the robot dynamics are unknown but, as a physical system, satisfy Hamilton's equations of motion over the $SE(3)$ manifold. Instead, we are given a set of state-control trajectories, from past experiments or collected by a human operator, and seek to safely track a desired path of robot positions with safety constraints obtained online from sensor measurements, e.g. distances to obstacles from a depth sensor. We first learn a Hamiltonian model of the system dynamics using a physics-guided neural ODE networks (Duong and Atanasov, 2021b). As the robot dynamics are equivariant to translation, we offset the trajectories to start from the origin and train a translation-equivariant Hamiltonian neural ODE model. The Hamiltonian structure of the learned model offers an energy-based regulation controller with the total energy of the system viewed as a Lyapunov function. This, in turn, enables us to enforce safety constraints using reference governor techniques without the needs to linearize the system dynamics. Inspired by constraint embedding techniques using a Lyapunov function (Garone and Nicotra, 2016; Arslan and Koditschek, 2017), we impose the safety constraints, based on sensor measurements, on the Lyapunov function. We use the trade-off between safety (distance from constraint violation) and system activeness (measured by the Lyapunov function) to regulate the reference governor and achieve safe and stable position tracking in an unknown environment.

Contributions. In summary, the contributions of this paper are 1) a neural ODE network approach for learning translation-equivariant $SE(3)$ Hamiltonian system dynamics from state-control trajectory data and 2) a tracking control design for $SE(3)$ Hamiltonian systems with stability and safety guarantees. Our dynamics learning and tracking control techniques are demonstrated on a simulated hexarotor robot using a depth sensor to navigate in unknown complex environments.

2. Problem Statement

Consider a robot modeled as a rigid body with position $\mathbf{p} \in \mathbb{R}^3$, orientation $\mathbf{R} \in SO(3)$, body-frame linear velocity $\mathbf{v} \in \mathbb{R}^3$, and body-frame angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$. Let $\mathfrak{q} = [\mathbf{p}^\top \;\; \mathbf{r}_1^\top \;\; \mathbf{r}_2^\top \;\; \mathbf{r}_3^\top]^\top \in \mathbb{R}^{12}$ denote the robot's generalized coordinates, where $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3 \in \mathbb{R}^3$ are the rows of the rotation matrix $\mathbf{R}$. Let $\boldsymbol{\zeta} = [\mathbf{v}^\top \;\; \boldsymbol{\omega}^\top]^\top \in \mathbb{R}^6$ denote the robot's generalized velocity. The generalized momentum $\mathfrak{p}$ of the system is defined as:

$$\mathfrak{p} = \mathbf{M}(\mathfrak{q})\boldsymbol{\zeta} \in \mathbb{R}^6, \tag{1}$$

where $\mathbf{M}(\mathfrak{q}) \succ 0$ denotes a positive-definite 6×6 generalized mass matrix. Let $\mathbf{x} = (\mathfrak{q}, \mathfrak{p}) \in \mathbb{R}^{18}$ be the robot state. The Hamiltonian, $\mathcal{H}(\mathfrak{q}, \mathfrak{p})$, captures the total energy of the system as the sum of the kinetic energy $\mathcal{T}(\mathfrak{q}, \mathfrak{p}) = \frac{1}{2}\mathfrak{p}^\top \mathbf{M}(\mathfrak{q})^{-1}\mathfrak{p}$ and the potential energy $\mathcal{U}(\mathfrak{q})$:

$$\mathcal{H}(\mathfrak{q}, \mathfrak{p}) = \mathcal{T}(\mathfrak{q}, \mathfrak{p}) + \mathcal{U}(\mathfrak{q}) = \frac{1}{2}\mathfrak{p}^\top \mathbf{M}(\mathfrak{q})^{-1}\mathfrak{p} + \mathcal{U}(\mathfrak{q}). \tag{2}$$

As a mechanical system, the time evolution of the state $\mathbf{x}$ is governed by Hamilton's equations of motion (Lee et al., 2017):

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u}, \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{3}$$

where $\mathbf{u} \in \mathbb{R}^p$ is the control input, $\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{0} & \mathfrak{q}^\times \\ -\mathfrak{q}^{\times\top} & \mathfrak{p}^\times \end{bmatrix} \begin{bmatrix} \frac{\partial \mathcal{H}(\mathfrak{q},\mathfrak{p})}{\partial \mathfrak{q}} \\ \frac{\partial \mathcal{H}(\mathfrak{q},\mathfrak{p})}{\partial \mathfrak{p}} \end{bmatrix}$, $\mathbf{G}(\mathbf{x}) = \begin{bmatrix} \mathbf{0} \\ \mathbf{B}(\mathfrak{q}) \end{bmatrix}$, and $\mathbf{B}(\mathfrak{q}) \in \mathbb{R}^{6 \times p}$ is an input gain matrix. The operators $\mathfrak{q}^\times$, $\mathfrak{p}^\times$ and the hat map $\hat{\mathbf{w}}$ for $\mathbf{w} \in \mathbb{R}^3$ are defined as:

$$\mathfrak{q}^\times = \begin{bmatrix} \mathbf{R}^\top & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{r}}_1^\top & \hat{\mathbf{r}}_2^\top & \hat{\mathbf{r}}_3^\top \end{bmatrix}^\top, \quad \mathfrak{p}^\times = \begin{bmatrix} \mathfrak{p}_{\mathbf{v}} \\ \mathfrak{p}_{\boldsymbol{\omega}} \end{bmatrix}^\times = \begin{bmatrix} \mathbf{0} & \hat{\mathfrak{p}}_{\mathbf{v}} \\ \hat{\mathfrak{p}}_{\mathbf{v}} & \hat{\mathfrak{p}}_{\boldsymbol{\omega}} \end{bmatrix}, \quad \hat{\mathbf{w}} = \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix}.$$

We consider the case that the parameters of the Hamiltonian dynamics model in (3), including the mass $\mathbf{M}(\mathfrak{q})$, potential energy $\mathcal{U}(\mathfrak{q})$, and input matrix $\mathbf{g}(\mathfrak{q})$, are unknown. Instead, we are given a trajectory dataset $\mathcal{D} = \{t_{0:N}^{(i)}, \mathfrak{q}_{0:N}^{(i)}, \boldsymbol{\zeta}_{0:N}^{(i)}, \mathbf{u}^{(i)}\}_{i=1}^D$ consisting of D sequences of generalized coordinates and velocities $(\mathfrak{q}_{0:N}^{(i)}, \boldsymbol{\zeta}_{0:N}^{(i)})$ at times $t_0^{(i)} < t_1^{(i)} < \ldots < t_N^{(i)}$, collected by applying a constant control input $\mathbf{u}^{(i)}$ to the system with initial condition $(\mathfrak{q}_0^{(i)}, \boldsymbol{\zeta}_0^{(i)})$. We aim to learn the dynamics from the data set $\mathcal{D}$ and design a control policy $\mathbf{u} = \boldsymbol{\pi}(\mathbf{x})$ such that the robot follows a desired reference path without violating safety constraints in an unknown environment. Let $\mathcal{O}$ and $\mathcal{F} := \mathbb{R}^3 \setminus \mathcal{O}$ denote the unsafe (e.g., obstacle) set and the safe (obstacle free) set, respectively. Denote the interior of $\mathcal{F}$ as $\mathrm{int}(\mathcal{F})$. We assume that $\mathcal{O}$ is not known a priori but the robot can sense the distance $\bar{d}(\mathbf{p}, \mathcal{O})$ from its position $\mathbf{p}$ to $\mathcal{O}$ locally with a limited sensing range $\beta > 0$:

$$\bar{d}(\mathbf{p}, \mathcal{O}) := \min\{d(\mathbf{p}, \mathcal{O}), \beta\}, \tag{4}$$

where $d(\mathbf{p}, \mathcal{O}) := \inf_{\mathbf{a} \in \mathcal{O}} \|\mathbf{p} - \mathbf{a}\|$ denotes the Euclidean distance from $\mathbf{p}$ to the set $\mathcal{O}$. The safe autonomous navigation problem considered in this paper is summarized below.

Problem *Let $\mathcal{D} = \{t_{0:N}^{(i)}, \mathfrak{q}_{0:N}^{(i)}, \boldsymbol{\zeta}_{0:N}^{(i)}, \mathbf{u}^{(i)}\}_{i=1}^D$ be a training dataset of state-control trajectories obtained from a robot with unknown Hamiltonian dynamics in (3). Let $\mathbf{r} : [0,1] \mapsto Int(\mathcal{F})$ be a piecewise-continuous function specifying a desired position reference path for the robot. Assume* *that the reference path starts at the initial robot position at time t_0, i.e., $\mathbf{r}(0) = \mathbf{p}(t_0) \in Int(\mathcal{F})$. Using local distance observations $\bar{d}(\mathbf{p}(t), \mathcal{O})$ of the unsafe set $\mathcal{O}$ in an unknown environment, design a control policy $\boldsymbol{\pi}(\mathbf{x})$ so that the position $\mathbf{p}(t)$ of the closed-loop system converges asymptotically to $\mathbf{r}(1)$, while remaining safe, i.e., $\mathbf{p}(t) \in \mathcal{F}$ for all $t \geq t_0$.*

3. Learning Hamiltonian Dynamics on the $SE(3)$ Manifold

3.1. Training a translation-equivariant $SE(3)$ Hamiltonian dynamics model

We obtain the training dataset $\mathcal{D} = \{t^{(i)}_{0:N}, \mathfrak{q}^{(i)}_{0:N}, \boldsymbol{\zeta}^{(i)}_{0:N}, \mathbf{u}^{(i)}\}_{i=1}^{D}$ by applying a constant input $\mathbf{u}^{(i)}$ to the system and sampling the generalized coordinates and velocities at times $t^{(i)}_0 < t^{(i)}_1 < \ldots < t^{(i)}_N$ using an odometry algorithm (Delmerico and Scaramuzza (2018); Mohamed et al. (2019)) or a motion capture system. The control input $\mathbf{u}^{(i)}$ can be generated by manually driving the robot or using an existing controller. Since the system dynamics does not change if we shift the position $\mathbf{p}$ to any points in the world frame, we offset the trajectories in the dataset $\mathcal{D}$ so that they start from the position $\mathbf{0}$ and learn the system dynamics well around the origin. This is sufficient for control purposes, e.g. using the controller design in Sec. 4.1, because the control input driving the system from state $\mathbf{x}$ with position $\mathbf{p}$ to the desired state $\mathbf{x}^*$ with position $\mathbf{p}^*$ is the same as the one driving the system from the state $\mathbf{x}$ with position $\mathbf{0}$ to the desired state $\mathbf{x}^*$ with the offset position $\mathbf{p}^* - \mathbf{p}$.

Since the mometum $\mathfrak{p}$ is not directly available from the dataset $\mathcal{D}$, we use the time derivative of the generalized velocity, derived from (1):

$$\dot{\boldsymbol{\zeta}} = \left(\frac{d}{dt}\mathbf{M}^{-1}(\mathfrak{q})\right)\mathfrak{p} + \mathbf{M}^{-1}(\mathfrak{q})\dot{\mathfrak{p}}. \tag{5}$$

Eq. (3) and (5) describe the Hamiltonian dynamics of the generalized coordinates and velocities with unknown inverse generalized mass matrix $\mathbf{M}(\mathfrak{q})^{-1}$, input matrix $\mathbf{B}(\mathfrak{q})$, and potential energy $\mathcal{U}(\mathfrak{q})$, for which we aim to approximate by three neural networks $\mathbf{M}_{\boldsymbol{\theta}}(\mathfrak{q})^{-1}, \mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ and $\mathcal{U}_{\boldsymbol{\theta}}(\mathfrak{q})$, respectively, with parameters $\boldsymbol{\theta}$.

To optimize for the parameters $\boldsymbol{\theta}$, we use the Hamiltonian-based neural ODE framework that encodes the Hamiltonian dynamics (3) and (5) with $\mathbf{M}_{\boldsymbol{\theta}}(\mathfrak{q}), \mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ and $\mathcal{U}_{\boldsymbol{\theta}}(\mathfrak{q})$ in the network structure (Fig. 1(a)). The forward pass rolls out the Hamiltonian dynamics (3) and (5) with the neural networks $\mathbf{M}_{\boldsymbol{\theta}}(\mathfrak{q}), \mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ and $\mathcal{U}_{\boldsymbol{\theta}}(\mathfrak{q})$ using a neural ODE solver (Chen et al. (2018)) to obtain an predicted sequence $(\bar{\mathfrak{q}}^{(i)}_{0:N}, \bar{\boldsymbol{\zeta}}^{(i)}_{0:N})$ at times $t^{(i)}_0 < t^{(i)}_1 < \ldots < t^{(i)}_N$ for each $i = 1, \ldots, D$. The loss function is defined as $\mathcal{L} = \sum_{i=1}^{D}\sum_{n=1}^{N} c(\mathfrak{q}^{(i)}_{0:N}, \boldsymbol{\zeta}^{(i)}_{0:N}, \bar{\mathfrak{q}}^{(i)}_{0:N}, \bar{\boldsymbol{\zeta}}^{(i)}_{0:N})$ where the distance metric c is defined as the sum of position, orientation, and velocity errors on the tangent bundle $TSE(3)$ of the pose manifold $SE(3)$:

$$c\left(\mathfrak{q}^{(i)}_{0:N}, \boldsymbol{\zeta}^{(i)}_{0:N}, \bar{\mathfrak{q}}^{(i)}_{0:N}, \bar{\boldsymbol{\zeta}}^{(i)}_{0:N}\right) = c_{\mathbf{p}}(\mathbf{p}, \bar{\mathbf{p}}) + c_{\mathbf{R}}(\mathbf{R}, \bar{\mathbf{R}}) + c_{\boldsymbol{\zeta}}(\boldsymbol{\zeta}, \bar{\boldsymbol{\zeta}}), \tag{6}$$

with the position error $c_{\mathbf{p}}(\mathbf{p}, \bar{\mathbf{p}}) = \|\mathbf{p} - \bar{\mathbf{p}}\|_2^2$, the velocity error $c_{\boldsymbol{\zeta}}(\boldsymbol{\zeta}, \bar{\boldsymbol{\zeta}}) = \|\boldsymbol{\zeta} - \bar{\boldsymbol{\zeta}}\|_2^2$, and the rotation error $c_{\mathbf{R}}(\mathbf{R}, \bar{\mathbf{R}}) = \|\left(\log(\bar{\mathbf{R}}\mathbf{R}^\top)\right)^\vee\|_2^2$. The log-map $\log(\cdot) : SE(3) \mapsto \mathfrak{so}(3)$ is the inverse of the exponential map, returning a skew-symmetric matrix in $\mathfrak{so}(3)$ from a rotation matrix in $SE(3)$, and the $\vee$-map $(\cdot)^\vee : \mathfrak{so}(3) \mapsto \mathbb{R}^3$ is the inverse of the hat map $\hat{(\cdot)}$ in Sec. 2. The network parameters $\boldsymbol{\theta}$ are optimized using gradient descent by back-propagating the loss through the neural ODE solver. This is done efficiently using adjoint method, where the gradient $\partial\mathcal{L}/\partial\boldsymbol{\theta}$ is calculated by a single call to a reverse-time ODE starting from $t = t_N$ at $(\mathfrak{q}^{(i)}_N, \boldsymbol{\zeta}^{(i)}_N)$.

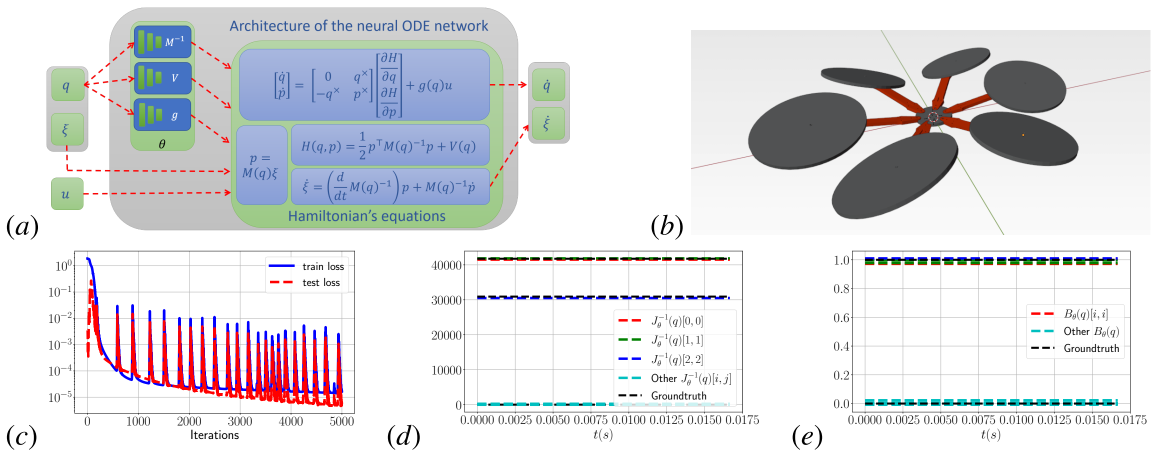

Figure 1: $SE(3)$ Hamiltonian neural ODE network: (a) network architecture, (b) simulated hexarotor for evaluation, (c) training loss, (d) learned inverse inertia $\mathbf{J}_{\boldsymbol{\theta}}(\mathfrak{q})^{-1}$, and (e) learned input matrix $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ along a test trajectory, evaluated on the simulated hexarotor.

3.2. Evaluation of the $SE(3)$ Hamiltonian dynamics model of a simulated hexarotor

We consider a simulated hexarotor unmanned aerial vehicle (UAV) (Fig. 1(b)) with fixed-tilt rotors pointing in different directions (Rajappa et al. (2015)) modeled as a fully-actuated rigid body with mass $m = 0.027$ and inertia matrix $\mathbf{J} = 10^{-5}\text{diag}([2.4, 2.4, 3.2])$. The robot's ground-truth dynamics satisfy Hamilton's equations in (3) with generalized mass $\mathbf{M}(\mathfrak{q}) = \text{diag}(m\mathbf{I}, \mathbf{J})$, potential energy $\mathcal{U}(\mathfrak{q}) = mgz$, and the input matrix $\mathbf{B}(\mathfrak{q}) = \mathbf{I}$. The control input $\mathbf{u}$ is a 6-dimensional wrench, including a 3-dimensional force and a 3-dimensional torque. Since the mass m of the hexarotor can be easily measured, we assume the mass m is known, leading to a known potential energy $\mathcal{U}(\mathfrak{q}) = mg\begin{bmatrix}0 & 0 & 1\end{bmatrix}\mathbf{p}$, where $\mathbf{p}$ is the UAV position and $g \approx 9.8ms^{-2}$ is the gravitational acceleration. We approximate the inverse generalized mass matrix by $\mathbf{M}_{\boldsymbol{\theta}}(\mathfrak{q})^{-1} = \text{diag}(m^{-1}\mathbf{I}, \mathbf{J}_{\boldsymbol{\theta}}^{-1}(\mathfrak{q}))$ and learn the inverse inertia matrix $\mathbf{J}_{\boldsymbol{\theta}}(\mathfrak{q})^{-1}$ and the input matrix $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ from data.

We mimic manual flights in an area free of obstacles using a PID controller and drive the hexarotor from a random initial pose to a desired poses, generating 18 1-second trajectories. We shift the trajectories to start from the origin and create a dataset $\mathcal{D} = \{t^{(i)}_{0:N}, \mathfrak{q}^{(i)}_{0:N}, \boldsymbol{\zeta}^{(i)}_{0:N}, \mathbf{u}^{(i)})\}_{i=1}^{D}$ with $N = 5$ and $D = 432$. The Hamiltonian-based neural ODE network[1] is trained with the dataset $\mathcal{D}$, as described in Sec. 3, for 5000 iterations and learning rate 10^{-3}. Fig. 1(c) shows the loss function during training. Note that if we scale $\mathbf{M}_{\boldsymbol{\theta}}(\mathfrak{q})$ and the input matrix $\mathbf{B}(\mathfrak{q})$ by a constant γ, the dynamics of $(\mathfrak{q}, \boldsymbol{\zeta})$ in (3) and (5) does not change. Fig. 1(d) and 1(e) plot the scaled version of the learned inverse mass $\mathbf{J}_{\boldsymbol{\theta}}(\mathfrak{q})^{-1}$ and the input matrix $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$, converging to the constant ground truth values.

4. Safe Tracking using a Reference Governor

In this section, we first describe a passivity-based regulation controller for arbitrary pose stabilization in Sec. 4.1. We derive sufficient conditions for safety based on an invariant level set of the

1. Code: https://thaipduong.github.io/SE3HamDL/

closed-loop system's Hamiltonian. Finally, in Sec. 4.2, we propose a reference governor control policy to adaptively generate a regulation pose along the desired path and achieve safe navigation.

4.1. Passivity-based control for learned Hamiltonian dynamics

Given the learned model of the system dynamics:

$$\dot{\mathbf{x}} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}) + \mathbf{G}_{\boldsymbol{\theta}}(\mathbf{x})\mathbf{u}, \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{7}$$

we want to find a control policy that stabilizes the system to a desired equilibrium $\mathbf{x}^* := (\mathfrak{q}^*, \mathbf{0})$ with the desired generalized coordinates $\mathfrak{q}^* = (\mathbf{p}^*, \mathbf{R}^*)$ and zero momentum $\mathfrak{p}^* = \mathbf{0}$, i.e. zero generalized velocity. Specifically, we design a control policy $\mathbf{u} = \boldsymbol{\pi}(\mathbf{x}, \mathbf{x}^*)$ to shape the total energy (Hamiltonian) of the closed-loop system so that it achieves a minimum at the desired state $\mathbf{x}^* = (\mathfrak{q}^*, \mathbf{0})$. By injecting energy into the system through the controller $\mathbf{u} = \boldsymbol{\pi}(\mathbf{x}, \mathbf{x}^*)$, we aim to achieve the following desired Hamiltonian:

$$\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*) = \frac{1}{2}k_{\mathbf{p}}(\mathbf{p} - \mathbf{p}^*)^\top(\mathbf{p} - \mathbf{p}^*) + \frac{1}{2}k_{\mathbf{R}}\,\mathrm{tr}(\mathbf{I} - \mathbf{R}^{*\top}\mathbf{R}) + \frac{1}{2}(\mathfrak{p} - \mathfrak{p}^*)^\top \mathbf{M}_{\boldsymbol{\theta}}^{-1}(\mathfrak{q})(\mathfrak{p} - \mathfrak{p}^*), \tag{8}$$

where $k_{\mathbf{p}}$ and $k_{\mathbf{R}}$ are positive gains. We use the interconnection and damping assignment passivity-based control (IDA-PBC) approach (Van Der Schaft and Jeltsema, 2014) to obtain matching conditions relating the dynamics in (7) to error state dynamics associated with the desired Hamiltonian. Solving the matching conditions, as described in Duong and Atanasov (2021b), leads to a controller, consisting of an energy-shaping term $\mathbf{u}_{ES}$ and a damping-injection term $\mathbf{u}_{DI}$:

$$\mathbf{u} = \underbrace{\mathbf{B}_{\boldsymbol{\theta}}^{\dagger}(\mathfrak{q}) \left(\mathfrak{q}^{\times\top}\partial V/\partial\mathfrak{q} - \mathfrak{p}^{\times}\mathbf{M}_{\boldsymbol{\theta}}^{-1}(\mathfrak{q})\mathfrak{p} - \mathbf{e}(\mathfrak{q}, \mathfrak{q}^*)\right)}_{\mathbf{u}_{ES}} + \underbrace{\left(-\mathbf{B}_{\boldsymbol{\theta}}^{\dagger}(\mathfrak{q})\mathbf{K}_{\mathbf{d}}\,\mathbf{M}_{\boldsymbol{\theta}}^{-1}(\mathfrak{q})\mathfrak{p}\right)}_{\mathbf{u}_{DI}}, \tag{9}$$

where $\mathbf{B}_{\boldsymbol{\theta}}^{\dagger}(\mathfrak{q}) = \left(\mathbf{B}_{\boldsymbol{\theta}}^{\top}(\mathfrak{q})\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})\right)^{-1}\mathbf{B}_{\boldsymbol{\theta}}^{\top}(\mathfrak{q})$ is the pseudo-inverse of $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$, $\mathbf{K}_{\mathbf{d}} = \mathrm{diag}(k_{\mathbf{v}}\mathbf{I}, k_{\boldsymbol{\omega}}\mathbf{I})$ is a damping gain with positive terms $k_{\mathbf{v}}$, $k_{\boldsymbol{\omega}}$, and $\mathbf{e}(\mathfrak{q}, \mathfrak{q}^*)$ is the error between $\mathfrak{q}$ and $\mathfrak{q}^*$:

$$\mathbf{e}(\mathfrak{q}, \mathfrak{q}^*) = \begin{bmatrix} \mathbf{e}_{\mathbf{p}}(\mathfrak{q}, \mathfrak{q}^*) \\ \mathbf{e}_{\mathbf{R}}(\mathfrak{q}, \mathfrak{q}^*) \end{bmatrix} = \begin{bmatrix} k_{\mathbf{p}}\mathbf{R}^\top(\mathbf{p} - \mathbf{p}^*) \\ \frac{1}{2}k_{\mathbf{R}}\left(\mathbf{R}^{*\top}\mathbf{R} - \mathbf{R}^\top\mathbf{R}^*\right)^\vee \end{bmatrix}. \tag{10}$$

Lemma 1 *If the input gain matrix $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ of the system in (7) is invertible, the control policy $\mathbf{u} = \boldsymbol{\pi}(\mathbf{x}, \mathbf{x}^*)$ in (9) always exists and asymptotically stabilizes the system to an arbitrary reference $\mathbf{x}^* = (\mathfrak{q}^*, \mathbf{0})$ with Lyapunov function given by the desired Hamiltonian $\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*)$ in (8).*

Proof See Appendix A in the extended version (Li et al., 2021). ■

Next, given the closed-loop system:

$$\dot{\mathbf{x}} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}) + \mathbf{G}_{\boldsymbol{\theta}}(\mathbf{x})\boldsymbol{\pi}\left(\mathbf{x}, \mathbf{x}^*\right), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \tag{11}$$

we derive conditions on the initial state $\mathbf{x}_0$ under which the position $\mathbf{p}$ converges to $\mathbf{p}^*$ safely, remaining in the safe set $\mathcal{F}$. We first define a dynamic safety margin (DSM) $\Delta E(\mathbf{x}, \mathbf{x}^*)$ for the Hamiltonian dynamics (11):

$$\Delta E(\mathbf{x}, \mathbf{x}^*) := \bar{d}^{\,2}(\mathbf{p}^*, \mathcal{O}) - \frac{2}{k_{\mathbf{p}}}\mathcal{H}_d\left(\mathbf{x}, \mathbf{x}^*\right), \tag{12}$$

where $\bar{d}^{\,2}(\mathbf{p}^*, \mathcal{O})$ is the truncated distance to the unsafe set $\mathcal{O}$ in (4). Given a fixed desired point $\mathbf{x}^*$, the DSM function measures the the trade-off between safety, measured by $\bar{d}^{\,2}(\mathbf{p}^*, \mathcal{O})$, and system activeness, measured by the Lyapunov function $\mathcal{H}_d\,(\mathbf{x}, \mathbf{x}^*)$. and allows us to find a positively forward invariant set $\mathcal{S}(\mathbf{x}, \mathbf{x}^*)$ in Prop. 2 such that for any $\mathbf{x}_0 \in \mathcal{S}(\mathbf{x}, \mathbf{x}^*)$, the position $\mathbf{p}$ converges to $\mathbf{p}^*$ while remaining in the safe set $\mathcal{F}$.

Proposition 2 *Controller* (9) *renders* $\mathcal{S}(\mathbf{x}, \mathbf{x}^*) := \{\mathbf{x} \mid \Delta E(\mathbf{x}, \mathbf{x}^*) \geq 0\}$ *a positively forward invariant set for system* (11)*. Furthermore, if* $\mathbf{x}_0 \in \mathcal{S}(\mathbf{x}, \mathbf{x}^*)$*, then the state* $\mathbf{p}(t)$ *converges to* $\mathbf{p}^*$ *asymptotically without violating the collision, i.e.,* $d(\mathbf{p}(t), \mathcal{O}) \geq 0$ *for all* $t \geq t_0$*.*

Proof See Appendix B in the extended version (Li et al., 2021). ■

Based on the results in this section for a fixed equibrium point $\mathbf{x}^*$, we develop a reference governor in Sec. 4.2 to adaptive change $\mathbf{x}^*$ over time so that the robot can safely track a desired path of robot positions.

4.2. Reference governor design

We introduce a virtual system, called a *reference governor* (Bemporad, 1998), which will adaptively track the path $\mathbf{r}$ defined in Sec. 2 and provide a time-varying reference $\mathbf{x}^*(t)$ for the actual system in (11). The motion of the governor system needs to be regulated to balance the energy of the Hamiltonian system with the distance to the unsafe set $\mathcal{O}$, keeping the safety margin in (12) positive. Define the governor as a first-order linear time-invariant system with state $\mathbf{g}(t) \in \mathbb{R}^3$ and dynamics:

$$\dot{\mathbf{g}} = -k_g\,(\mathbf{g} - \mathbf{u}_\mathbf{g}) \tag{13}$$

where k_g is a positive gain and $\mathbf{u}_\mathbf{g}$ is the governor control input. The input $\mathbf{u}_\mathbf{g}$ will be chosen to move the governor system along the reference path without violating the safety condition $\Delta E(\mathbf{x}, \mathbf{x}^*) \geq 0$ obtained in Proposition 2. Define a *local safe set* $\mathcal{LS}(\mathbf{x}, \mathbf{g})$ as a region around the governor state $\mathbf{g}$ that does not violate safety:

$$\mathcal{LS}(\mathbf{x}, \mathbf{g}) := \left\{\mathbf{q} \in \mathbb{R}^m \mid \|\mathbf{q} - \mathbf{g}\|^2 \leq (1+\epsilon)^{-1} \Delta E(\mathbf{x}, \mathbf{x}^*)\right\},$$

where $\epsilon > 0$ is arbitrarily, ensuring that $\mathcal{LS}(\mathbf{x}, \mathbf{g}) \subseteq \text{int}(\mathcal{F})$. The size of the local safe set determines how fast the governor can move along the reference path without endangering safety.

Definition 3 *A* local projected goal *at system-governor state* $(\mathbf{x}, \mathbf{g})$ *is a point* $\bar{\mathbf{g}} \in \mathcal{LS}(\mathbf{x}, \mathbf{g})$ *that is furthest along the reference path* $\mathbf{r}$*:*

$$\bar{\mathbf{g}} = \mathbf{r}(\sigma^*), \quad \sigma^* = \underset{\sigma \in [0,1]}{\text{argmax}} \{\sigma \mid \mathbf{r}(\sigma) \in \mathcal{LS}(\mathbf{x}, \mathbf{g})\}. \tag{14}$$

Choosing the governor input as $\mathbf{u}_\mathbf{g} = \bar{\mathbf{g}}$ forces the governor to track the reference path adaptively, taking the safety condition $\Delta E(\mathbf{x}, \mathbf{x}^*) \geq 0$ into account. Given the local projected goal $\bar{\mathbf{g}}(t) \in \mathbb{R}^3$ and the governor state $\mathbf{g}(t) \in \mathbb{R}^3$, we also generate a desired reference state $\mathbf{x}^*(t)$ for the system in (11) by lifting $\mathbf{g}(t)$ to $\mathbb{R}^{18}$. We may choose $\mathbf{g}(t)$ as the desired position with zero desired velocity but, to provide guidance on the $SE(3)$ manifold, we need to also generate a desired orientation $\mathbf{R}^*(t)$. We construct a lifting function $\boldsymbol{\ell} : \mathcal{F} \times \text{int}(\mathcal{F}) \mapsto \mathbb{R}^{18}$ to obtain $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$ as:

$$\boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}}) = \begin{bmatrix}\mathbf{p}^{*\top} & \mathbf{r}_1^{*\top} & \mathbf{r}_2^{*\top} & \mathbf{r}_3^{*\top}, \mathbf{0}^\top, \mathbf{0}^\top\end{bmatrix}^\top, \tag{15}$$

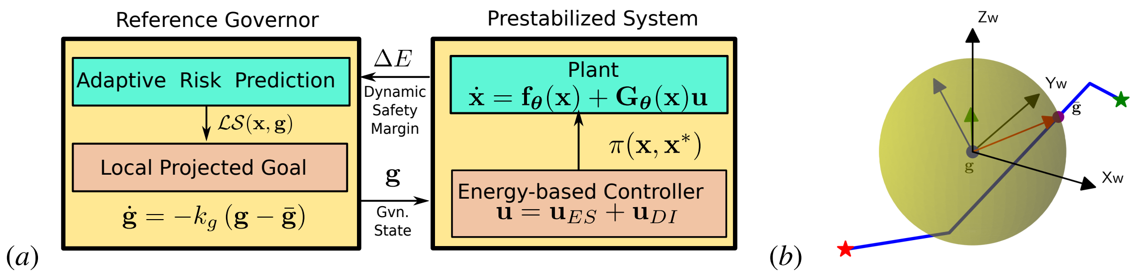

Figure 2: (a): structure of the reference-governor tracking controller: a reference governor with state $\mathbf{g}$ adaptively tracks a point $\bar{\mathbf{g}}$ along the desired path $\mathbf{r}$ and generates a time-varying equilibrium $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$ for the closed-loop Hamiltonian system. (b): a local projecte goal $\bar{\mathbf{g}}$ (purple dot) is generated as the furthest intersection between the local safe set $\mathcal{LS}(\mathbf{x}, \mathbf{g})$ (yellow sphere) and the path $\mathbf{r}$ (blue curve). Given $\mathbf{g}$ and $\bar{\mathbf{g}}$, a desired equilibrium $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$ is generated for the Hamiltonian system with orientation indicated by the red, green, and blue arrows, respectively.

where $\mathbf{p}^* = \mathbf{g}$ and $\mathbf{r}_1^*$, $\mathbf{r}_2^*$, $\mathbf{r}_3^*$ are the rows of the matrix:

$$\mathbf{R}^*(\mathbf{g}, \bar{\mathbf{g}}) = \begin{cases} \mathbf{I} & \text{if } \mathbf{e}_3 \times \mathbf{c}_1 = 0 \\ [\mathbf{c}_1 \;\; \mathbf{c}_2 \;\; \mathbf{c}_3] & \text{otherwise,} \end{cases} \tag{16}$$

with $\mathbf{e}_3 = [0, 0, 1]^\top$, $\mathbf{c}_1 = (\bar{\mathbf{g}} - \mathbf{g})/\|\bar{\mathbf{g}} - \mathbf{g}\|$, $\mathbf{c}_2 = (\mathbf{e}_3 \times \mathbf{c}_1)/\|\mathbf{e}_3 \times \mathbf{c}_1\|$, and $\mathbf{c}_3 = (\mathbf{c}_1 \times \mathbf{c}_2)/\|\mathbf{c}_1 \times \mathbf{c}_2\|$. If $\mathbf{g} = \bar{\mathbf{g}}$, the most recent backup of $\mathbf{R}^*$ or $\mathbf{I}$ may be used.

Our safe tracking control design is visualized in Fig. 2. It consists of two parts: 1) a first-order reference governor system with state $\mathbf{g}$ adaptively following the local projected goal $\bar{\mathbf{g}}$ along the path $\mathbf{r}$ and 2) a closed-loop Hamiltonian system tracking the reference signal $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$. Our main result is summarized in the following theorem.

Theorem 4 *Given a reference path $\mathbf{r}$, consider the closed-loop Hamiltonian system in (11) and the closed-loop reference governor system, $\dot{\mathbf{g}} = -k_g (\mathbf{g} - \bar{\mathbf{g}})$, with local projected goal $\bar{\mathbf{g}}$ provided in Definition 3. Suppose that the initial state $(\mathbf{x}_0, \mathbf{g}_0)$ satisfies:*

$$\Delta E \left(\mathbf{x}_0, \boldsymbol{\ell}(\mathbf{g}_0, \bar{\mathbf{g}}_0)\right) > 0, \quad \mathbf{g}_0 = \mathbf{r}(0) = \mathbf{p}(t_0) \in \mathit{int}(\mathcal{F}), \tag{17}$$

where $\Delta E(\mathbf{x}, \mathbf{x}^*)$ is a dynamic safety margin defined in (12). Then, the joint system state $(\mathbf{x}, \mathbf{g})$ converges to $(\boldsymbol{\ell}(\mathbf{r}(1), \mathbf{r}(1)), \mathbf{r}(1))$ without violating the constraints, i.e., $\mathbf{p}(t) \in \mathcal{F}$, $\forall t \geq t_0$.

Proof See Appendix C in the extended version (Li et al., 2021). ■

5. Evaluation

This section evaluates our safe tracking controller on a simulated hexarotor UAV using the learned Hamiltonian dynamics in Sec. 3.2. The task is to navigate from a start postion to a goal in an

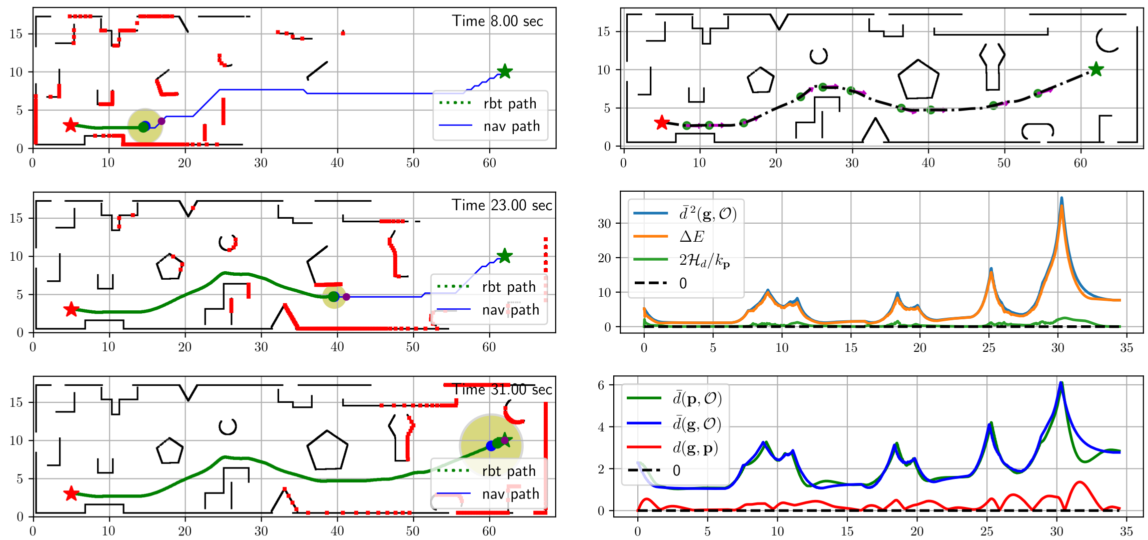

Figure 3: Safe tracking control for a hexarotor aerial robot in an unknown cluttered environment. The hexarotor (green dot) needs to navigate from a start location (red star) to a goal location (green star) while avoiding obstacles. The obstacles are sensed by a simulated LiDAR sensor (red points). On the left are snapshots of the environment showing the robot (green dot), lidar scans (red dots), and the unsafe set $\mathcal{O}$ (black surfaces) at different times. The reference path (blue curve) is re-planned online from the governor position (blue dot) to the goal (green star) using an A^* algorithm. The local projected goal $\bar{\mathbf{g}}$ (purple dot) is computed based on the obstacle distance (gray ball) and the local safe set (yellow ball). On the upper right, projections of the robot's heading (purple arrow) are plotted along the path. The middle right figure shows the dynamic safety margin ΔE in (12) and the scaled desired Hamiltonian $2\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*)/k_\mathbf{p}$ while the lower right figure shows the distance to the obstacles $d(\mathbf{p}(t), \mathcal{O})$ (green curve), indicating that the safety constraints are never violated.

environment without colliding with the obstacles. The following control gains were used with the regulation controller in Sec. 4.1: $k_p = 0.25$, $k_\mathbf{R} = 125\mathbf{J}$, $k_v = 0.125$, $k_{\boldsymbol{\omega}} = 10\mathbf{J}$ in (8) and $k_\mathbf{g} = 1.0$ with the governor control in (13). We test the controller in a challenging unknown 3D environment with complex obstacles $\mathcal{O}$. A simulated LiDAR scanner provides point cloud measurements $\mathcal{P}(t) := \{\mathbf{y}_i(t)\}$ of the surface of the unsafe set $\mathcal{O}$, depending on the system pose, with a maximum sensing range of $\beta = 30$. The distance from the governor $\mathbf{g}(t)$ to the unsafe set $\mathcal{O}$ is approximated via $\bar{d}(\mathbf{g}(t); \mathcal{O}) \approx \min_{\mathbf{y} \in \mathcal{P}(t)} \|\mathbf{g}(t) - \mathbf{y}\|$. The point clouds $\mathcal{P}(t)$ are used to construct an occupancy grid map online and a reference path $\mathbf{r}$ is replanned periodically using the A* algorithm to ensure that $\mathbf{r}(\sigma) \in \text{int}(\mathcal{F})$.

Fig. 3 and Fig. 4 show the behavior of the closed-loop hexarotor system in an two different environments. The reference governor follows the projected goal $\bar{\mathbf{g}}$ and generates a time-varying equilibrium $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$ for the hexarotor. The dynamic safety margin $\Delta E(\mathbf{x}, \mathbf{x}^*)$ fluctuates during this process but, as can be seen in the figures, it never becomes negative. The augmented

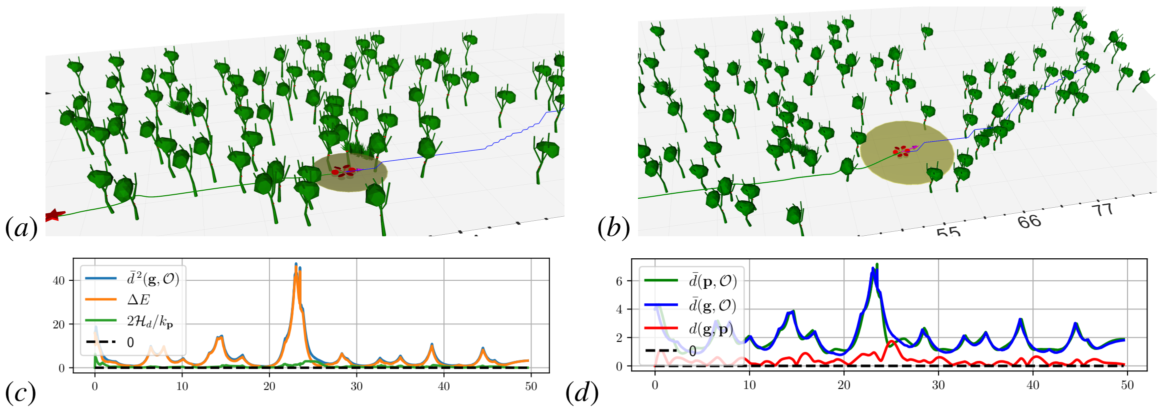

Figure 4: Safe navigation of a hexarotor robot in a forest is demonstrated in (a) and (b). The bottom plots (c) and (d) show the dynamic safety margin ΔE, the scaled desired Hamiltonian $2\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*)/k_\mathbf{p}$, and the distance to the obstacles $d(\mathbf{p}(t), \mathcal{O})$, indicating that the safety constraints are never violated.

system $(\mathbf{x}, \mathbf{g})$ is controlled adaptively, slowing down when the dynamic safety margin decreases (e.g., when the robot is close to an obstacle and has large total energy $\mathcal{H}_d$) and speeding up otherwise (e.g., when the robot is far away from the obstacles or has small total energy $\mathcal{H}_d$). The simulations show that our control policy successfully drives the system from the start to the end of the reference path while avoiding sensed obstacle online, i.e., $d(\mathbf{p}, \mathcal{O})$ remains positive throughout the motion.

6. Conclusion

This paper developed a neural ODE network for learning a dynamics model of an $SE(3)$ Hamiltonian system and a tracking controller that enables safe autonomous navigation in unknown environments. Given only a training set of system state-control trajectories, our approach estimates the system dynamics and synthesizes a controller that avoids obstacles based on run-time distance measurements. Our method was demonstrated on a simulated hexarotor aerial robot navigating in complex 3D environments. Future work will focus on capturing model uncertainty and external disturbances in the design and deploying it on a hardware platform.

Acknowledgments

We gratefully acknowledge support from NSF RI IIS-2007141 and NSF CCF-2112665 (TILOS).

Appendix A.

Proof Let $\mathbf{p}_e = \mathbf{p} - \mathbf{p}^*$ be the position error and $\mathbf{R}_e := \mathbf{R}^{*\top}\mathbf{R} = \begin{bmatrix}\mathbf{r}_{e1} & \mathbf{r}_{e2} & \mathbf{r}_{e3}\end{bmatrix}^\top$ be the rotation error, then the error state can be expressed as $\mathbf{x}_e := (\mathfrak{q}_e, \mathfrak{p}_e)$, where

$$\mathfrak{q}_e = \begin{bmatrix}(\mathbf{p} - \mathbf{p}^*)^\top & \mathbf{r}_{e1}^\top & \mathbf{r}_{e2}^\top & \mathbf{r}_{e3}^\top\end{bmatrix}^\top, \quad \mathfrak{p}_e = \mathfrak{p}.$$

Since the system is fully-actuated, i.e. the input matrix $\mathbf{B}_{\boldsymbol{\theta}}(\mathfrak{q})$ is invertible. The controller in (9) exists and the resulting closed-loop error dynamics becomes (Duong and Atanasov, 2021b):

$$\begin{bmatrix}\dot{\mathfrak{q}}_e \\ \dot{\mathfrak{p}}_e\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{J} \\ -\mathbf{J}^\top & -\mathbf{K}_d\end{bmatrix}\begin{bmatrix}\frac{\partial \mathcal{H}_d}{\partial \mathfrak{q}_e} \\ \frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e}\end{bmatrix}, \qquad \mathbf{J} = \begin{bmatrix}\mathbf{R}^\top & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{r}}_{e1}^\top & \hat{\mathbf{r}}_{e2}^\top & \hat{\mathbf{r}}_{e3}^\top\end{bmatrix}^\top. \tag{18}$$

Using group property, rotation error matrix $\mathbf{R}_e = \mathbf{R}^{*\top}\mathbf{R} \in SO(3)$, therefore, $\mathbf{R}_e$ is a orthogonal matrix. All columns of $\mathbf{R}_e$ are orthonormal and all elements in $\mathbf{R}_e$ are less than 1, hence, $\operatorname{tr}\left(\mathbf{I} - \mathbf{R}^{*\top}\mathbf{R}\right) \geq 0$. Since $\mathbf{M}$ is a positive definite matrix, it is easy to see that $\mathcal{H}_d$ is positive definite, and 0 minimum value is achieved only at $\mathbf{x}_e^* = (\mathfrak{q}_e, \mathbf{0})$ with $\mathfrak{q}_e = [\mathbf{0}^\top, \mathbf{e}_1^\top, \mathbf{e}_2^\top, \mathbf{e}_3^\top]^\top$. The time derivative can be computed as:

$$\begin{aligned}\dot{\mathcal{H}}_d(\mathbf{x}, \mathbf{x}^*) &= \frac{\partial \mathcal{H}_d}{\partial \mathfrak{q}_e}^\top \dot{\mathfrak{q}}_e + \frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e}^\top \dot{\mathfrak{p}}_e \\ &= \frac{\partial \mathcal{H}_d}{\partial \mathfrak{q}_e}^\top \mathbf{J}\frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e} - \frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e}^\top \mathbf{J}^\top \frac{\partial \mathcal{H}_d}{\partial \mathfrak{q}_e} - \frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e}^\top \mathbf{K}_d \frac{\partial \mathcal{H}_d}{\partial \mathfrak{p}_e} \\ &= -\mathfrak{p}_e^\top \mathbf{M}^{-1}(\mathfrak{q})\mathbf{K}_d\mathbf{M}^{-1}(\mathfrak{q})\mathfrak{p}_e.\end{aligned} \tag{19}$$

Hence, $\dot{\mathcal{H}}_d(\mathbf{x}, \mathbf{x}^*) \leq 0$ for all $\mathbf{x}_e$. It is not hard to show that the only point can stay within set $\left\{\dot{\mathcal{H}}_d = 0\right\}$ is at origin. By the LaSalle's invariance principle (Khalil, 2002), the system (18) asymptotically stabilizes to desired equilibrium $\mathbf{x}_e^*$, i.e. $\mathbf{x}^* = (\mathfrak{q}^*, \mathbf{0})$.

■

Appendix B.

Proof From Lemma 1, we know that for any constant $\mathbf{x}^* = (\mathfrak{q}^*, \mathbf{0})$, $\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*)$ is a Lyapunov function to certificate stability of $\mathbf{x}^*$. Hence,

$$\mathcal{S}(\mathbf{x}, \mathbf{x}^*) = \{\mathbf{x} \mid \Delta E(\mathbf{x}, \mathbf{x}^*) \geq 0\} = \left\{\mathbf{x} \mid H_d(\mathbf{x}, \mathbf{x}^*) \leq \frac{2\bar{d}^2(\mathbf{p}^*, \mathcal{O})}{k_{\mathbf{p}}}\right\}$$

is forward invariant and control signal $\boldsymbol{\pi}(\mathbf{x}, \mathbf{x}^*)$ can steer the state $\mathbf{p}$ of the system towards $\mathbf{p}^*$. It remains to show that during the convergence, constraints over $\mathbf{p}(t)$ is never violated. In the proof of Lemma 1, we have shown that the second term of desired Hamiltonian (8), $\frac{1}{2}k_{\mathbf{R}} \operatorname{tr}(\mathbf{I} - \mathbf{R}^{*\top}\mathbf{R})$ is non-negative for all $\mathbf{R}^* \in SO(3)$, therefore, $\frac{2}{k_{\mathbf{p}}}\mathcal{H}_d(\mathbf{x}, \mathbf{x}^*) \geq (\mathbf{p} - \mathbf{p}^*)^\top(\mathbf{p} - \mathbf{p}^*)$. From (4) and above inequality, we know that $\mathbf{p}(t) \in \mathcal{F}$ for all $t \geq t_0$, since

$$d^2(\mathbf{p}^*, \mathcal{O}) \geq \bar{d}^2(\mathbf{p}^*, \mathcal{O}) \geq \frac{2}{k_{\mathbf{p}}}\mathcal{H}_d(\mathbf{x}_0, \mathbf{x}^*) \geq \frac{2}{k_{\mathbf{p}}}\mathcal{H}_d(\mathbf{x}(t), \mathbf{x}^*) \geq d^2(\mathbf{p}^*, \mathbf{p}(t)).$$

■

Appendix C.

Proof For conciseness, $\Delta E(t) = \Delta E\left(\mathbf{x}(t), \boldsymbol{\ell}(\mathbf{g}(t), \bar{\mathbf{g}}(t))\right)$. Initially, $\mathbf{g}_0 = \mathbf{p}_0 = \mathbf{r}(0) \in \mathcal{LS}(\mathbf{x}_0, \mathbf{g}_0)$ and $\Delta E(t_0) > 0$, the local projected goal $\bar{\mathbf{g}}$ and associated rotation matrix $\mathbf{R}^*(\mathbf{g}, \bar{\mathbf{g}})$ is well defined. As $\bar{\mathbf{g}}$ moves along the reference path $\mathbf{r}$, i.e., path parameter σ in (14) increases. While $\mathbf{g}$ chasing $\bar{\mathbf{g}}$ by (13), system state $\mathbf{x}$ tracks $\mathbf{x}^* = \boldsymbol{\ell}(\mathbf{g}, \bar{\mathbf{g}})$ using controller $\boldsymbol{\pi}(\mathbf{x}, \mathbf{x}^*)$ in (9). During this process, the safety margin $\Delta E(t)$ is fluctuating which regulates behavior of $\mathbf{g}$ through $\bar{\mathbf{g}}$. Since the system dynamics are continuous, $\Delta E(t)$ cannot become negative without crossing 0 from above at some time T_0. As $\Delta E(t) \downarrow 0$, the local safe zone will shrink to a point, i.e., $\mathcal{LS}(\mathbf{x}, \mathbf{g}) \downarrow \{\mathbf{g}\}$. This immediately stops the movement of governor because $\bar{\mathbf{g}} = \mathbf{g}(T_0)$ and $\dot{\mathbf{g}}(T_0) = 0$. From Proposition 2, we know that $\mathbf{x}(t)$ will stay within $\mathcal{S}(\mathbf{x}, \mathbf{x}^*(T_0))$ for $t \geq T_0$ and position constraints will not be violated. As $\mathbf{x} \to \mathbf{x}^*(T_0)$ because $\dot{H}_d(\mathbf{x}, \mathbf{x}^*(T_0)) < 0$ when $\mathbf{g}$ is static and $\mathbf{x} \neq \mathbf{x}^*(T_0)$, there exists $h > 0$ such that $\Delta E(T_0 + h)$ becomes strictly positive. Hence, the governor is able to move again towards new $\bar{\mathbf{g}}$ getting further along the path as discussed previously. This process continues until the augmented system stabilized at $\left(\boldsymbol{\ell}\left(\mathbf{r}(1), \mathbf{r}(1)\right), \mathbf{r}(1)\right)$ where $\bar{\mathbf{g}}$ stops changing. ■

References

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 (TAC)*, 59(4):876–891, 2014a.

Aaron D Ames, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *IEEE Conference on Decision and Control (CDC)*, pages 6271–6278, 2014b.

Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control (TAC)*, 62(8):3861–3876, 2017.

Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *IEEE European Control Conference (ECC)*, pages 3420–3431, 2019.

Omur Arslan and Daniel E Koditschek. Smooth extensions of feedback motion planners via reference governors. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

Alberto Bemporad. Reference governor for constrained nonlinear systems. *IEEE Transactions on Automatic Control (TAC)*, 43(3):415–419, 1998.

Francesco Borrelli, Alberto Bemporad, and Manfred Morari. *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

José Manuel Bravo, Teodoro Alamo, and Eduardo F Camacho. Robust mpc of constrained discrete-time nonlinear systems based on approximated reachable sets. *Automatica*, 42(10):1745–1751, 2006.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Marc Peter Deisenroth, Dieter Fox, and Carl Edward Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):408–423, 2015. doi: 10.1109/TPAMI.2013.218.

J. Delmerico and D. Scaramuzza. A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

Thai Duong and Nikolay Atanasov. Learning adaptive control for se(3) hamiltonian dynamics. *arXiv preprint arXiv:2109.09974*, 2021a.

Thai Duong and Nikolay Atanasov. Hamiltonian-based Neural ODE Networks on the SE(3) Manifold For Dynamics Learning and Control. In *Proceedings of Robotics: Science and Systems*, Virtual, July 2021b. doi: 10.15607/RSS.2021.XVII.086.

Emanuele Garone and Marco M Nicotra. Explicit reference governor for constrained nonlinear systems. *IEEE Transactions on Automatic Control (TAC)*, 61(5):1379–1384, 2016.

Lars Grüne and Jürgen Pannek. *Nonlinear model predictive control*. Springer, 2017.

Sylvia L Herbert, Mo Chen, SooJean Han, Somil Bansal, Jaime F Fisac, and Claire J Tomlin. FaSTrack: A modular framework for fast and guaranteed safe motion planning. In *IEEE Conference on Decision and Control (CDC)*, pages 1517–1522, 2017.

Darryl D Holm. *Geometric Mechanics*. World Scientific Publishing Company, 2008.

Juraj Kabzan, Lukas Hewing, Alexander Liniger, and Melanie N. Zeilinger. Learning-based model predictive control for autonomous racing. *IEEE Robotics and Automation Letters*, 4(4):3363–3370, 2019. doi: 10.1109/LRA.2019.2926677.

Hassan Khalil. *Nonlinear systems*. Prentice Hall, 2002.

Ilya Kolmanovsky, Emanuele Garone, and Stefano Di Cairano. Reference and command governors: A tutorial on their theory and automotive applications. In *IEEE American Control Conference (ACC)*, pages 226–241, 2014.

Shreyas Kousik, Sean Vaskov, Fan Bu, Matthew Johnson-Roberson, and Ram Vasudevan. Bridging the gap between safety and real-time performance in receding-horizon trajectory design for mobile robots. *The International Journal of Robotics Research (IJRR)*, 2020.

Taeyoung Lee, Melvin Leok, and N Harris McClamroch. *Global formulations of Lagrangian and Hamiltonian dynamics on manifolds*. Springer, 2017.

Zhichao Li, Omur Arslan, and Nikolay Atanasov. Fast and Safe Path-Following Control using a State-Dependent Directional Metric. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

Zhichao Li, Thai Duong, and Nikolay Atanasov. Safe Autonomous Navigation for Systems with Learned SE(3) Hamiltonian Dynamics. Technical report, University of California, San Diego, 2021. URL https://thaipduong.github.io/refgovham/.

Anatolii Isakovich Lurie. *Analytical mechanics*. Springer Science & Business Media, 2013.

M Lutter, K Listmann, and J Peters. Deep Lagrangian Networks for end-to-end learning of energy-based control for under-actuated systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

Anirudha Majumdar and Russ Tedrake. Funnel libraries for real-time robust feedback motion planning. *The International Journal of Robotics Research (IJRR)*, 36(8):947–982, 2017.

David Q Mayne, James B Rawlings, Christopher V Rao, and Pierre OM Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

S. A. S. Mohamed, M. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila. A survey on odometry for autonomous navigation systems. *IEEE Access*, 7, 2019.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Multistep neural networks for data-driven discovery of nonlinear dynamical systems. *arXiv preprint arXiv:1801.01236*, 2018.

S. Rajappa, M. Ryll, H. H. Bülthoff, and A. Franchi. Modeling, control and design optimization for a fully-actuated hexarotor aerial vehicle with tilted propellers. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

Arjan Van Der Schaft and Dimitri Jeltsema. Port-Hamiltonian systems theory: An introductory overview. *Foundations and Trends in Systems and Control*, 1(2-3), 2014.

Xiangru Xu. Constrained control of input–output linearizable systems using control sharing barrier functions. *Automatica*, 87:195–201, 2018.

Michael C Yip and David B Camarillo. Model-less feedback control of continuum manipulators in constrained environments. *IEEE Transactions on Robotics*, 30(4):880–889, 2014.

Yaofeng Desmond Zhong, Biswadip Dey, and Amit Chakraborty. Symplectic ODE-Net: learning Hamiltonian dynamics with control. In *International Conference on Learning Representations (ICLR)*, 2019.