Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 525–532

Learning Switching Port-Hamiltonian Systems with Uncertainty Quantification

Thomas Beckers * **Tom Z. Jiahao** ** **George J. Pappas** ***

* *Department of Computer Science, Vanderbilt University, Nashville, TN 37212, USA, (e-mail: thomas.beckers@vanderbilt.edu)*

** *Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104, USA, (e-mail: zjh@seas.upenn.edu)*

*** *Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA, (e-mail: pappasg@seas.upenn.edu)*

Abstract: Switching physical systems are ubiquitous in modern control applications, for instance, locomotion behavior of robots and animals, power converters with switches and diodes. The dynamics and switching conditions are often hard to obtain or even inaccessible in case of a-priori unknown environments and nonlinear components. Black-box neural networks can learn to approximately represent switching dynamics, but typically require a large amount of data, neglect the underlying axioms of physics, and lack of uncertainty quantification. We propose a Gaussian process based learning approach enhanced by switching Port-Hamiltonian systems (GP-SPHS) to learn physical plausible system dynamics and identify the switching condition. The Bayesian nature of Gaussian processes uses collected data to form a distribution over all possible switching policies and dynamics that allows for uncertainty quantification. Furthermore, the proposed approach preserves the compositional nature of Port-Hamiltonian systems. A simulation with a hopping robot validates the effectiveness of the proposed approach.

Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)

Keywords: Bayesian methods, Nonparametric methods, Grey box modelling, Mechatronic systems, Uncertainty quantification

1. INTRODUCTION

The modeling and identification of switching dynamical systems is a crucial task in a wide range of domains, such as robotics and power systems (Anderson et al., 2020; Brogliato and Brogliato, 1999; Wu et al., 2018). System identification has a long history in control as many control strategies are derived based on a precise model of the plant. Whereas classical models of physical systems are typically based on first principles, there is a recent interest in data-driven modeling of switching systems to capture more details with reduced engineering effort.

However, this paradigm shift poses new questions regarding the efficiency, interpretability, and physical correctness of these models (Hou and Wang, 2013). By physical correctness, we mean that the learned model respects physical principles such as the conservation of energy and passivity. Including these physical principles in a data-driven approach is beneficial in several ways: The models i) are more meaningful as they respect the postulates of physics, ii) come with increased interpretability, and iii) can be more data-efficient, since the satisfaction of physical axioms results in a meaningful inductive bias of the model (Karniadakis et al., 2021). A large class of switching physical systems can be described by Port-Hamiltonian systems. Port-Hamiltonian systems (PHS) leverage network approaches from electrical engineering and constitute a cornerstone of mathematical systems

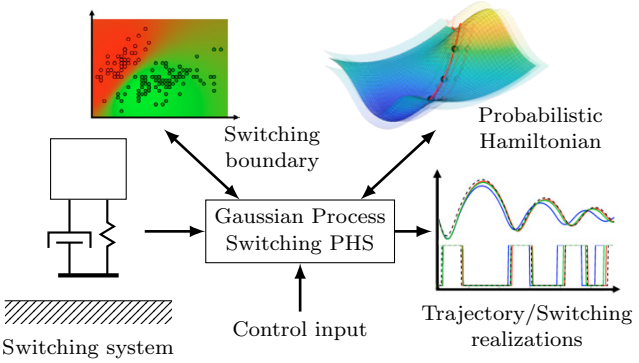


Fig. 1. Gaussian Process switching Port-Hamiltonian systems (GP-SPHS) allow to learn a probabilistic Hamiltonian and switching policy based on collected data. All sample trajectories of GP-SPHS are physically correct in terms of the evolution of the energy.

theory. While most of the analysis of physical systems has been performed within the Lagrangian and Hamiltonian framework, the network point of view is attractive for the modeling and simulation of complex physical systems with many connected components. The main idea of switching Port-Hamiltonian systems (SPHS) is that the total energy in the system is described by the Hamiltonian, a smooth scalar function of the system's generalized coordinates which is not affected by any switching (Van Der Schaft and Jeltsema, 2014). Instead, the switching is modeled

2405-8963 Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2023.10.1621

by a rerouting of the energy flow that may be internally induced (state dependent) or externally triggered. It has been shown that SPHS are able to model a range of multi-domain complex physical systems, in particular, mechanical and electrical systems (Van der Schaft, 2000). However, the Hamiltonian and the switching policy are often highly nonlinear and, thus, challenging for modeling and identification. The purpose of this paper is to combine the expressiveness of Gaussian processes with the physical plausibility of switching Port-Hamiltonian systems to model physical switching systems with unknown dynamics and switching condition based on data.

There is a rapidly growing body of literature on data-driven modeling of dynamical systems using deep learning. In particular, switching dynamics arises in stiff systems or systems with frictional contact (Anderson et al., 2020). Notably, a family of neural networks, neural ordinary differential equations, was proposed to represent the vector field of dynamical systems using neural networks (Chen et al., 2018), but the successes have been limited to low-stiff systems (Jiahao et al., 2021). Efforts to build models that conserve the total energy of a system led to a body of work on Hamiltonian neural networks (HNN) (Greydanus et al., 2019). However, since many physical systems are dissipative, the assumption of HNNs limits its applicability to real-world systems. Results for data-driven learning of PHS with neural networks have been established in (Desai et al., 2021; Nageshrao et al., 2015). However, standard neural network architectures can have difficulty generalizing from sparse data and have no uncertainty quantification of the outputs. In contrast, Bayesian models such as Gaussian processes (GP) can generalize well even for small training datasets and have a built-in notion of uncertainty quantification (Rasmussen and Williams, 2006). GPs have been recently used for learning generating functions for Hamiltonian systems (Rath et al., 2021), dynamics of structured systems (Ridderbusch et al., 2021; Bhouri and Perdikaris, 2022) and PHS (Beckers et al., 2022), but they have not been applied to the class of switching PHS.

Contribution: We propose a Bayesian learning approach using Gaussian processes and SPHS to learn switching physical systems with partially unknown dynamics based on data. In contrast to existing modeling methods for SPHS, the probabilistic model includes all possible realizations of the SPHS under the GP prior, which guarantees physical plausibility, high flexibility, and allows for uncertainty quantification. We prove that the passivity and interconnection properties of SPHS are preserved in GP-SPHS, which is a beneficial characteristic for the composition and control of physical system.

The remainder of the paper is structured as follows. We start with the introduction of SPHS in Section 2 and the problem setting in Section 3 followed by proposing GP-SPHS in Section 4. Finally, a simulation highlights the benefits of GP-SPHS in Section 5.

2. SWITCHING PORT-HAMILTONIAN SYSTEMS

Composing Hamiltonian systems with input/output ports leads to a Port-Hamiltonian system, which is a dynamical system with ports that specify the interactions of its components. The switching of the dynamics can occur due to an internal or external trigger which affects the routing of the energy (Van Der Schaft and Camlibel, 2009). The dynamics of a SPHS is described by [1]

$$\begin{aligned} \dot{\boldsymbol{x}} &= [J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]\nabla_{\boldsymbol{x}} H(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u} \\ \boldsymbol{y} &= G(\boldsymbol{x})^\top \nabla_{\boldsymbol{x}} H(\boldsymbol{x}), \end{aligned} \tag{1}$$

with the state $\boldsymbol{x}(t) \in \mathbb{R}^n$ (also called energy variable) at time $t \in \mathbb{R}_{\geq 0}$, the total energy represented by a smooth function $H\colon \mathbb{R}^n \to \mathbb{R}$ called the Hamiltonian, and the I/O ports $\boldsymbol{u}, \boldsymbol{y} \in \mathbb{R}^m$. The dynamics depends on a mode $s \in \{1, \dots, n_s\}$, where $n_s \in \mathbb{N}$ is the total number of system modes.

For each mode s, the matrix $J_s\colon \mathbb{R}^n \to \mathbb{R}^{n\times n}$ is skew-symmetric and specifies the interconnection structure and $R_s\colon \mathbb{R}^n \to \mathbb{R}^{n\times n}, R = R^\top \succeq 0$ specifies the dissipation in the system. The interchange with the environment is defined by the matrix $G\colon \mathbb{R}^n \to \mathbb{R}^{n\times m}$. The structure of the interconnection matrix J_s is typically derived from kinematic constraints in mechanical systems, Kirchhoff's law, power transformers, gyrators, etc. Loosely speaking, the interconnection of the elements in the SPHS is defined by J_s, whereas the Hamiltonian H characterizes their dynamical behavior. These are typically imposed by nonlinear elements, physical coupling effects, or highly nonlinear electrical and magnetic fields. The port variables $\boldsymbol{u}$ and $\boldsymbol{y}$ are conjugate variables in the sense that their duality product defines the power flows exchanged with the environment of the system, for instance, forces and velocities in mechanical systems.

3. PROBLEM SETTING

We consider the problem of learning the dynamics of a physical switching system with the following assumptions.

Assumption 1. We have access to potentially noisy observations $\tilde{\boldsymbol{x}}(t) \in \mathbb{R}^n$ of the system state $\boldsymbol{x}(t) \in \mathbb{R}^n$ whose evolution over time $t \in \mathbb{R}_{\geq 0}$ follows the *unknown* dynamics

$$\dot{\boldsymbol{x}} = [J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]\nabla_{\boldsymbol{x}} H(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u}, \tag{2}$$

where a continuous solution $\boldsymbol{x}(t)$ exists and is unique for all $x_0 \in \mathcal{X}_0 \subseteq \mathbb{R}^n$ and $t \in \mathbb{R}_{\geq 0}$. The measurements $\tilde{\boldsymbol{x}}(t_i)$ are generated by $\tilde{\boldsymbol{x}}(t_i) = \boldsymbol{x}(t_i) + \boldsymbol{\eta}$ where $\boldsymbol{\eta} \in \mathbb{R}^n$ is distributed according to a zero-mean Gaussian $\eta \sim \mathcal{N}(\boldsymbol{0}, \mathrm{diag}[\sigma_1^2, \dots, \sigma_n^2])$ with unknown variances $\sigma_1^2, \dots, \sigma_n^2 \in \mathbb{R}_{\geq 0}$.

Assumption 2. The interconnection matrix $J_s\colon \mathbb{R}^n \to \mathbb{R}^{n\times n}$, dissipation matrix $R_s\colon \mathbb{R}^n \to \mathbb{R}^{n\times n}$ and the I/O matrix $G\colon \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are known for all $s \in \{1, \dots, n_s\}$.

Assumption 3. The mode s of (2) is measurable.

Assumption 1 ensures that we have access to Gaussian noise corrupted state measurements that can be provided by either real sensors or an state observer. Additionally, it is assumed that the real system can be described by an SPHS where the switch configuration s does not entail

[1] Vectors $\boldsymbol{a}$ and vector-valued functions $\boldsymbol{f}(\cdot)$ are denoted with bold characters. Matrices are described with capital letters. I_n is the n-dimensional identity matrix and 0_n the zero matrix. The expression $A_{:,i}$ denotes the i-th column of A. For a positive semidefinite matrix Λ, $\|x-y\|_\Lambda^2 = (x-y)^\top \Lambda (x-y)$. $\mathbb{R}_{>0}$ denotes the set of positive real number whereas $\mathbb{R}_{\geq 0}$ is the set of non-negative real numbers. $\mathcal{C}^i$ denotes the class of i-th times differentiable functions. The operator $\nabla_{\boldsymbol{x}}$ with $\boldsymbol{x} \in \mathbb{R}^n$ denotes $[\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n}]^\top$.

algebraic constraints on the state variables $\boldsymbol{x}$, which holds for many switching systems, including contact situations with finite stiffness, see (Van Der Schaft and Camlibel, 2009). Assumption 2 is necessary but not restrictive, as the Hamiltonian H typically captures the unstructured, nonlinear uncertainties of the system, while the structure of J, R and G is rather simple, and accurate estimates can often be easily obtained, see (Van der Schaft, 2000). Finally, Assumption 3 allows to measure the mode of the physical switching system, for instance, a hopping robot can be equipped with a contact sensor to distinguish the flying phase and the ground contact phase.

Problem 1. Given Assumptions 1 to 3, a dataset of time-stamps $\{t_i\}_{i=1}^N$ and noisy state observations with input and mode $\{\tilde{\boldsymbol{x}}(t_i), \boldsymbol{u}(t_i), s(t_i)\}_{i=1}^N$ of the system (2), we aim to learn a probabilistic model

$$\dot{\boldsymbol{x}} = [J_{\hat{s}}(\boldsymbol{x}) - R_{\hat{s}}(\boldsymbol{x})]\nabla_{\boldsymbol{x}}\hat{H}(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u}, \tag{3}$$

with an estimated Hamiltonian $\hat{H}$ and a switching policy $\hat{s}\colon \boldsymbol{x} \mapsto \{1, \dots, n_s\}$ that allows for uncertainty quantification.

4. GAUSSIAN PROCESS SWITCHING PORT-HAMILTONIAN SYSTEMS

In this section, we propose Gaussian process switching Port-Hamiltonian systems (GP-SPHS) whose structure is visualized in Fig. 2. Starting with data of a physical switching system (dashed rectangle), we use a surrogate GP to estimate the state's derivative, which is fed into another GP to model the Hamiltonian as a nonparametric, probabilistic function. A separate GP classifier is trained on the measured modes of the system. After training, samples of the GP regressor and classifier are drawn. Then, an ODE solver can produce sample trajectories based on the drawn Hamiltonian and the switching policy that allows for uncertainty quantification of the model.

In contrast to parametric models, our approach is beneficial in several ways. First, we do not rely on prior knowledge about the parametric structure of the Hamiltonian and switching policy. For instance, instead of assuming a linear spring model and fitting Hooke's spring constant based on data, the nonparametric structure of the GP allows to learn complex, nonlinear spring models. The switching policy is typically challenging to model as it is affected by the nonlinearities in the system. For instance, switching in an electronic system might be triggered by a threshold voltage of some nonlinear element.

Second, the Bayesian nature of the GP enables the model to represent *all possible* SPHS under the Bayesian prior based on a finite number of data points. This is not only interesting from a model identification perspective, but this uncertainty quantification can also be explicitly useful for robust control approaches.

In the following section, we present the general structure of GP-SPHS starting with the prior model followed by the learning and prediction procedure.

4.1 Modeling

First, we start with the modeling of the Hamiltonian H using a GP. Let $(\Omega, \mathcal{F}, P)$ be a probability triple with the

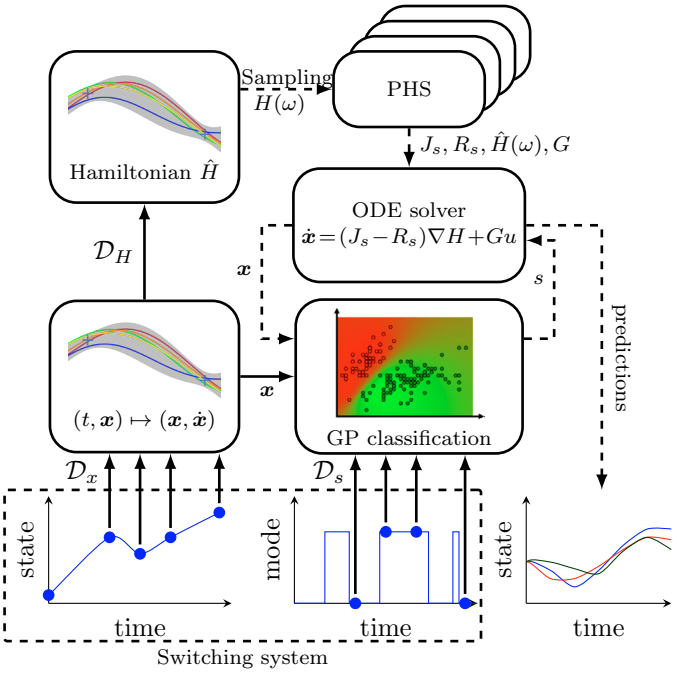


Fig. 2. Block diagram of a GP-PHS. The training phase is indicated by solid lines and the prediction phase by dashed lines.

probability space Ω, the corresponding σ-algebra $\mathcal{F}$ and the probability measure P. Then, a GP $\mathcal{GP}(m_{\text{GP}}, k)$ with mean $m_{\text{GP}}(x)$ and kernel $k(\boldsymbol{x}, \boldsymbol{x}')$ is a stochastic process on a set $\mathcal{X} \subseteq \mathbb{R}^n$ where any finite set of points follows a normal distribution. The kernel k is a measure for the correlation of two inputs. Combined with Bayes' Theorem, GPs can provide tractable statistical inference for regression and classification (Rasmussen and Williams, 2006). We place an independent GP prior on each dimension of the derivative of the estimated Hamiltonian

$$\frac{\partial \hat{H}}{\partial x_i} \sim \mathcal{GP}(0, k_H(\boldsymbol{x}, \boldsymbol{x}')), \quad \forall i \in \{1, \dots, n\}, \tag{4}$$

where k_H is the squared exponential kernel $k_H(\boldsymbol{x}, \boldsymbol{x}') = \sigma_f^2 \exp(-\|\boldsymbol{x} - \boldsymbol{x}'\|_\Lambda^2)$. This choice of the kernel results in sampled Hamiltonians which are smooth and allows to approximate any continuous function arbitrarily exactly, see (Rasmussen and Williams, 2006). The hyperparameters of the squared exponential kernel are the signal noise $\sigma_f \in \mathbb{R}_{>0}$ and the lengthscales $\Lambda = \text{diag}(l_1^2, \dots, l_n^2) \in \mathbb{R}_{>0}^n$, which can be optimized based on data. Due to the finite number of system's modes, the switching policy is handled as GP multi-class classification problem. Thus, we put a GP prior on the switching policy $\hat{s}$ such that

$$\hat{s} \sim \mathcal{GP}(0, k_s(\boldsymbol{x}, \boldsymbol{x}')). \tag{5}$$

By design, the GP approaches to classification and regression problems are similar, except that the Gaussian distributed error model in the regression case is replaced by a Bernoulli distribution over a logistic function, see (Williams and Barber, 1998) for more details.

The choice of the kernel function k_s allows to incorporate prior knowledge about the switching policy, e.g., by using linear or polynomial kernels. However, without prior knowledge, the squared exponential kernel is a common choice as a starting point due to its universality property.

4.2 GP-SPHS Training

As indicated in the problem formulation, we consider an observed system trajectory

$$\mathcal{D}_x = \{(t_1, \tilde{\boldsymbol{x}}(t_1), \boldsymbol{u}(t_1)), \dots, (t_N, \tilde{\boldsymbol{x}}(t_N), \boldsymbol{u}(t_N))\} \quad (6)$$

of the unknown dynamics (2) corresponding to measured inputs $\{\boldsymbol{u}(t_1), \dots, \boldsymbol{u}(t_N)\}$, where N is the total number of training pairs consisting of a time t_i and a noisy state measurement $\tilde{\boldsymbol{x}}(t_i)$. The first challenge we address is how to extract data of the form $\{\boldsymbol{x}, \dot{\boldsymbol{x}}\}$ from (6) so we may apply GP regression to learn the derivative of the Hamiltonian of the system (2). The advantages over classical filtering techniques are the inherent noise handling and the uncertainty quantification, which we include in the learning of the Hamiltonian. To obtain the derivative $\dot{\boldsymbol{x}}$, we exploit that GPs are closed under affine operations (Adler, 2010). We learn n separated GPs on the training sets $\mathcal{D}_x$, one GP for each dimension j of the state $\boldsymbol{x} \in \mathbb{R}^n$. We rearrange the training set as input and output matrix written as

$$\begin{aligned} T &= [t_1, \dots, t_N] \in \mathbb{R}^{1\times N} \\ \tilde{X} &= [\tilde{\boldsymbol{x}}(t_1), \dots, \tilde{\boldsymbol{x}}(t_N)]^\top \in \mathbb{R}^{N\times n}. \end{aligned} \quad (7)$$

Using a differentiable kernel k, we obtain the distribution for each element of the *state derivative* $\dot{x}_j \in \mathbb{R}$ by

$$\begin{aligned} \mu(\dot{x}_j \mid t, \mathcal{D}_x) &= \boldsymbol{k}^{(1)}(t, T)^\top K^{-1} \tilde{X}_{:,j} \quad (8) \\ \mathrm{var}(\dot{x}_j \mid t, \mathcal{D}_x) &= \boldsymbol{k}^{(1,2)}(t,t) - \boldsymbol{k}^{(1)}(t, T)^\top K^{-1} \boldsymbol{k}^{(1)}(t, T), \end{aligned}$$

where (8) is the closed-form posterior distribution of the GP, see (Rasmussen and Williams, 2006) for more details. The function $K\colon \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^{N\times N}$ is called the Gram matrix whose elements are given by $K_{i,i'} = k(t_i, t_{i'}) + \delta(i,i')\sigma_j^2$ for all $i', i \in \{1, \dots, N\}$ with the delta function $\delta(i,i') = 1$ for $i = i'$ and zero, otherwise. The vector-valued function $\boldsymbol{k}\colon \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N$, with the elements $k_i = k(t, T_i)$ for all $i \in \{1, \dots, N\}$, expresses the covariance between an input t and the input training data T.

The term $k^{(l)}$ denotes the derivative of the kernel function k with respect to the l-th argument, i.e., $k^{(1)}(t,T)_i = \frac{\partial}{\partial z} k(z, T_i)\big|_{z=t}$, $k^{(2)}(t,T)_i = \frac{\partial}{\partial z} k(t, z)\big|_{z=T_i}$, and $k^{(1,2)}(t,t) = \frac{\partial^2}{\partial z \partial z'} k(z, z')\big|_{z=t, z'=t}$ for $i = 1, \dots, N$. Further, an estimate of the state $\boldsymbol{x}(t_i)$ based on the noise state measurement $\tilde{\boldsymbol{x}}(t_i)$ is obtained by standard GP regression $\mu(x_j \mid t, \mathcal{D}_x) = \boldsymbol{k}(t, T)^\top K^{-1} \tilde{X}_{:,j}$. With the Gaussian prior $\tilde{X} | \boldsymbol{\varphi}, T \sim \mathcal{N}(\boldsymbol{0}, K)$ we can compute the negative log marginal likelihood (NLML) to learn the unknown hyperparameters $\boldsymbol{\varphi} = [\sigma_f, l_1, \dots, l_n, \sigma_1^2, \dots, \sigma_n^2]^\top \in \mathbb{R}^{2n+1}$ of (8). At this point, we have access to the estimated state $\mu(\boldsymbol{x} \mid t, \mathcal{D}_x)$ and its derivative $\mu(\dot{\boldsymbol{x}} \mid t, \mathcal{D}_x)$.

Remark 1. For the sake of simplicity, we focus here on a single trajectory. However, the above procedure can be repeated for multiple trajectories in the same manner.

Next, the learning of the derivatives of the Hamiltonian H is presented. Analogously to (8), each dimension of $\nabla_{\boldsymbol{x}} H(\boldsymbol{x})$ is learned by an independent GP. For this purpose, we need a suitable set of input and output training data. The input data consists of the state estimates $\mu(\boldsymbol{x} \mid t, \mathcal{D}_x)$ as $H(\boldsymbol{x})$ is state dependent. For the output data, we can rewrite the system dynamics (2) as

$$[J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]^{-1}(\dot{\boldsymbol{x}} - G(\boldsymbol{x})\boldsymbol{u}) = \nabla_{\boldsymbol{x}} H(\boldsymbol{x}), \quad (9)$$

to isolate $\nabla_{\boldsymbol{x}} H$ on the right-hand site if $[J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]$ is invertible. However, that might not hold for all states $\boldsymbol{x}$ and system modes s. In this case, the invertable subspace of $[J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]$ is used such that the number of available training data can differ between the dimensions of $\nabla_{\boldsymbol{x}} H$. The output data for the j-dimension $\dot{X}^j$ is given by

$$\dot{X}^j = [\boldsymbol{p}^j_{s_i}(\bar{\boldsymbol{x}}_i)(\bar{\dot{\boldsymbol{x}}}_i - G(\bar{\boldsymbol{x}}_i)\boldsymbol{u}(t_i))]_{\forall i \in \mathcal{I}^j}, \quad (10)$$

where $\bar{\boldsymbol{x}}_i = \mu(\boldsymbol{x} \mid t_i, \mathcal{D})$, $\bar{\dot{\boldsymbol{x}}}_i = \mu(\dot{\boldsymbol{x}} \mid t_i, \mathcal{D})$ and $s_i = s(t_i)$. The vector $\boldsymbol{p}^j_{s_i}(\bar{\boldsymbol{x}}_i) \in \mathbb{R}^{1\times n}$ is defined such that it is a unique solution of $\nabla_{x_j} H(\bar{\boldsymbol{x}}_i) = [\boldsymbol{p}^j_{s_i}(\bar{\boldsymbol{x}}_i)(\bar{\dot{\boldsymbol{x}}}_i - G(\bar{\boldsymbol{x}}_i)\boldsymbol{u}(t_i))]$ for all i in an index set $\mathcal{I}^j \subseteq \{1, \dots, N\}$. Loosely speaking, the pair $(\bar{\boldsymbol{x}}_i, \bar{\dot{\boldsymbol{x}}}_i)$ is exploited as a training point for the j-dimension of $\nabla_x H$ if there exists a bijective mapping between $(\bar{\boldsymbol{x}}_i, \bar{\dot{\boldsymbol{x}}}_i)$ and $\nabla_{x_j} H$. The vector $\boldsymbol{p}^j_{s_i}(\bar{\boldsymbol{x}}_i)$ will equal the j-th dimension of the inverse of $[J_s(\bar{\boldsymbol{x}}_i) - R_s(\bar{\boldsymbol{x}}_i)]$ if it exists. In this case, each index set $\mathcal{I}^j$ contains all training points, i.e., $\mathcal{I}^j = \{1, \dots, N\}$. Then, the corresponding input data is defined by

$$X^j = [\mu(\boldsymbol{x} \mid t_i, \mathcal{D})]_{\forall i \in \mathcal{I}^j}. \quad (11)$$

Then, we can create a new dataset $\mathcal{D}_H = \{X^j, \dot{X}^j\}_{j=1,\dots,n}$ that allows to learn an estimated $\nabla_{\boldsymbol{x}} \hat{H}$.

Analogously to (8), the Gram matrix $K^j_H \in \mathbb{R}^{N^j \times N^j}$, where N_j is the cardinality of $\mathcal{I}^j$, is defined by

$$\begin{aligned} K^j_H = &\begin{bmatrix} k_H(X^j_{:,1}, X^j_{:,1}) & \dots & k_H(X^j_{:,1}, X^j_{:,N^j}) \\ \vdots & \ddots & \vdots \\ k_H(X^j_{:,N^j}, X^j_{:,1}) & \dots & k_H(X_{:,N^j}, X_{:,N^j}) \end{bmatrix} \\ &+ \begin{bmatrix} \mathrm{var}(\dot{x}_j \mid t_1, \mathcal{D}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathrm{var}(\dot{x}_j \mid t_{N^j}, \mathcal{D}) \end{bmatrix}. \end{aligned} \quad (12)$$

Here, we use the posterior variance (8) of the estimated state derivative data $\dot{X}$ as noise in the covariance matrix (12). This allows us to consider the uncertainty of the estimation in the modeling of the SPHS. Finally, the unknown hyperparameters $\boldsymbol{\varphi}_H$ can be computed by minimization of the NLML

$$-\log \mathrm{p}(\dot{X}^j | \boldsymbol{\varphi}_H, X^j) \sim [\dot{X}^j]^\top [K^j_H]^{-1} \dot{X}^j + \log |K^j_H|, \quad (13)$$

for each $j = \{1, \dots, n\}$ via, e.g., a gradient-based method as the gradient is analytically tractable.

As a last step, the state-dependent switching policy $\hat{s}$ needs to be learned. For this purpose, we create a training set based on the estimated state and the measured mode of the system, i.e., $\mathcal{D}_s = \{(\bar{\boldsymbol{x}}_1, s_1), \dots, (\bar{\boldsymbol{x}}_N, s_N)\}$. Analogously to the GP model for the estimated Hamiltonian $\hat{H}$, the hyperparameters of the kernel k_s can be optimized by means of the NLML. However, in case of GP classification, there exists no analytically tractable solution for the NLML but, instead, a numerical approximation can be used, see (Rasmussen and Williams, 2006, Algorithm 3.1).

In Algorithm 1, we summarize the steps to train the GP-SPHS. The complexity of the algorithm is dominated by the cost of training the GPs that is $\mathcal{O}(N^3 + N^2 n)$ with respect to the number N of training points and the dimension n of the state $\boldsymbol{x}$. The complexity can be reduced by inducing points methods, for instance, as presented in (Wilson and Nickisch, 2015).

Algorithm 1 Training of GP-SPHS

Input: Trajectory $\mathcal{D} \leftarrow \{(t_i, \tilde{\boldsymbol{x}}(t_i))\}_{i=1,\dots,N}$
Input: Control inputs $\{\boldsymbol{u}(t_i)\}_{i=1,\dots,N}$
Input: System modes $\{s(t_i)\}_{i=1,\dots,N}$
Obtain $(\boldsymbol{x}, \dot{\boldsymbol{x}})$ **pairs**
Train n independent GPs with $\mathcal{D}_x$ (7)
Create training set $\mathcal{D}_H = \{X^j, \dot{X}^j\}_{j=1,\dots,n}$ (10) and (11)
Create training set $\mathcal{D}_s$
Obtain GP-SPHS model:
Learn $\nabla_x \hat{H}$ with $\mathcal{D}_H$ and $\{\boldsymbol{u}(t_i)\}_{i=1,\dots,N}$
Compute posterior variance $\mathrm{var}(\dot{\boldsymbol{x}} \mid T, \mathcal{D}_x)$ (8)
Minimize NLML (13) to estimate $\boldsymbol{\varphi}_H$
Learn $\hat{s}$ with $\mathcal{D}_s$, minimize NLML

4.3 Prediction

Once the GP-SPHS is trained, an ODE solver is used to compute trajectories from the estimated dynamics

$$\dot{\boldsymbol{x}} = [J_{\hat{s}(\boldsymbol{x})}(\boldsymbol{x}) - R_{\hat{s}(\boldsymbol{x})}(\boldsymbol{x})]\nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u}. \tag{14}$$

For this purpose, samples from the stochastic Hamiltonian $\hat{H}$ and the switching policy $\hat{s}$ are required. For $\hat{H}$, we can draw samples [2] $\nabla_x \hat{H}(\boldsymbol{x}^*, \omega), \forall \omega \in \Omega$ from the posterior distribution using the joint distribution at $\boldsymbol{x}^* \in \mathbb{R}^n$

$$\begin{bmatrix} \dot{X}^j \\ \nabla_{x_j} \hat{H}(\boldsymbol{x}^*) \end{bmatrix} = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K_H^j & k_H(X^j, \boldsymbol{x}^*) \\ k_H(X^j, \boldsymbol{x}^*)^\top & k_H(\boldsymbol{x}^*, \boldsymbol{x}^*) \end{bmatrix}\right), \tag{15}$$

to obtain an estimate of the Hamiltonian's derivative $\nabla_x H$. For using a numerical integrator to generate trajectories, we will need to be able to access the same sample at an arbitrary number of points at arbitrary locations. However, the joint distribution (15) only allows to sample from a finite number of a-priori known test points $\boldsymbol{x}^* \in X^*$. Thus, we are looking for an analytically tractable function to approximate a sample $\nabla_x \hat{H}(\boldsymbol{x}^*, \omega)$ which can be achieved via the Matheron's rule, see (Wilson et al., 2020). The error of the interpolation does not affect the SPHS properties as shown later.

In addition to a sampled Hamiltonian, the switching policy $\hat{s}$ is required to solve the estimated dynamics (14). As predictions with a GP classifier are not analytically tractable, a Laplacian approximation is used to obtain an approximated GP posterior, see (Williams and Barber, 1998). Then, analogously to (15), we can draw from the posterior distribution of the approximated GP to obtain a sample $\hat{s}(\boldsymbol{x}, \omega), \omega \in \Omega$, see (Rasmussen and Williams, 2006, Algorithm 3.2).

In Algorithm 2, we summarize the steps to achieve sampled trajectories of a GP-SPHS starting at $\boldsymbol{x}_0 \in \mathcal{X}_0$. As data-driven method, the accuracy of GP-SPHS typically increases with the number of training points N. The complexity scales with $\mathcal{O}(N^2)$. Thus, the choice of N is a trade-off between computational complexity and accuracy of the prediction. Next, we prove that a GP-SPHS generates valid samples of a SPHS with probability 1.

4.4 Theoretical analysis

Theorem 1. Consider a Hamiltonian $\hat{H}$ and a switching policy $\hat{s}$ sampled from a GP-SPHS. For all realizations ω in the sample space Ω, the dynamics

$$\begin{aligned} \dot{\boldsymbol{x}} &= [J_{\hat{s}(\boldsymbol{x},\omega)}(\boldsymbol{x}) - R_{\hat{s}(\boldsymbol{x},\omega)}(\boldsymbol{x})]\nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}, \omega) + G(\boldsymbol{x})\boldsymbol{u} \\ \boldsymbol{y} &= G(\boldsymbol{x})^\top \nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}, \omega), \end{aligned} \tag{16}$$

on a compact space $\mathcal{X} \subseteq \mathbb{R}^n$, describes a switching Port-Hamiltonian system that is almost surely passive with respect to the supply rate $\boldsymbol{u}^\top \boldsymbol{y}$.

Proof. See appendix A.

As a consequence, the GP-SPHS model allows us to build physically correct models in terms of conversation or dissipation of energy. Next, we show that even for approximations of $\hat{H}$ and $\hat{s}$, the resulting SPHS remains passive.

Corollary 1. Consider a GP-SPHS trained on a dataset (6) with sampled Hamiltonian $\hat{H}(\cdot, \omega)$ and switching policy $\hat{s}(\cdot, \omega)$ with $\omega \in \Omega$. Let $\hat{H}^* \colon \mathbb{R}^n \to \mathbb{R}$ be a smooth and bounded function approximator of $\hat{H}(\cdot, \omega)$. Then, $\dot{\boldsymbol{x}} = [J_{\hat{s}^*(\boldsymbol{x})}(\boldsymbol{x}) - R_{\hat{s}^*(\boldsymbol{x})}(\boldsymbol{x})]\nabla_{\boldsymbol{x}} \hat{H}^*(\boldsymbol{x}) + G(\boldsymbol{x})\boldsymbol{u}$ describes a Port-Hamiltonian system that is passive with respect to the supply rate $\boldsymbol{u}^\top \boldsymbol{y}$.

Proof. The proof results from Theorem 1.

As a consequence, we can use Matheron's rule, see (Wilson et al., 2020), that generates smooth and bounded functions to achieve analytically tractable approximations of $\nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}^*, \omega)$ and $\hat{s}(\boldsymbol{x}^*, \omega)$. To model more complex systems, we often wish to simplify the modeling process by separating the system into connected subsystems. The class of PHS are closed under such interconnections, see (Cervera et al., 2007). We will show that GP-SPHS share the same characteristic which allows to learn separated subsystems of complex systems.

Proposition 2. Consider two GP-SPHS (16) described by $\{J_{\hat{s},1}, R_{\hat{s},1}, \hat{H}_1, G_1\}$ with input dimension $m_1 \in \mathbb{N}$ and $\{J_{\hat{s},2}, R_{\hat{s},2}, \hat{H}_2, G_2\}$ with input dimension $m_2 \in \mathbb{N}$, respectively. Let $(\boldsymbol{u}_1^{\mathrm{c}}, \boldsymbol{y}_1^{\mathrm{c}}) \in \mathbb{R}^{m_{\mathrm{c}} \times m_{\mathrm{c}}}$ and $(\boldsymbol{u}_2^{\mathrm{c}}, \boldsymbol{y}_2^{\mathrm{c}}) \in \mathbb{R}^{m_{\mathrm{c}} \times m_{\mathrm{c}}}$ be the corresponding input / output pairs of dimension m_{c} with $m_{\mathrm{c}} \leq \min\{m_1, m_2\}$ for the connection of the two GP-SPHS. Then, the interconnection $\boldsymbol{u}_1^{\mathrm{c}} = -\boldsymbol{y}_2^{\mathrm{c}}$ and $\boldsymbol{u}_2^{\mathrm{c}} = \boldsymbol{y}_1^{\mathrm{c}}$ yields a GP-SPHS.

Proof. See appendix A.

Theorem 2 shows that the negative feedback interconnection of two GP-SPHS leads again to a GP-SPHS. This is in particular interesting for passivity-based control approaches, e.g., (Ortega and Garcia-Canseco, 2004).

Algorithm 2 Prediction with GP-SPHS

Input: Number of desired sample trajectories
Input: Initial state $\boldsymbol{x}_0$
for each sample **do**
Sample posterior Hamiltonian $\nabla_x \hat{H} \sim \mathcal{GP}$
Sample posterior switching policy $\hat{s} \sim \mathcal{GP}$
ODE solver for GP-SPHS (14) with
$\nabla_x \hat{H}(\boldsymbol{x}, \omega)$ and $\hat{s}(\boldsymbol{x}, \omega)$
end for

[2] We use the notation $\hat{H}(\boldsymbol{x}^*, \omega)$ to distinguish between a sample and the stochastic process $\hat{H}(\boldsymbol{x}^*)$

5. NUMERICAL EVALUATION

For the evaluation of the proposed GP-SPHS, we aim to learn the dynamics of a hopper robot, see Fig. 3, adapted from (Winkler, 2017). For simplification, we assume that the robot is stabilized such that the pose of the body can be fully described by its position on the vertical axis. In this setting, the robot acts as a mechanical system with dissipation. The mass of the robot is known to be $m = 1\,\mathrm{kg}$ and is concentrated mainly in the body of the robot. The stiffness and damping of the two joints are unknown. Figure 4 shows one trajectory of a hopper robot that starts above the ground such that the foot is not in contact. During ground contact, the distance between the robot's body to the ground and the robot's body to the foot are identical. For the modeling with a GP-SPHS, we consider the following SPHS from (Van der Schaft, 2000)

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \frac{\hat{s}-1}{d} & 0 & \hat{s} \\ 0 & 0 & 1 \\ -\hat{s} & -1 & -\hat{s}d \end{bmatrix} \frac{\partial \hat{H}}{\partial \boldsymbol{x}}(\boldsymbol{x}) \tag{17}$$

with the length x_1 of a (nonlinear) spring, the position of the body's center of mass x_2 and its momentum x_3 as depicted in Fig. 5. By comparing the trajectory in Fig. 4 to a linear mass-spring-damper system, we estimate the damping coefficient in (17) to $d = 2$. For simplicity of demonstration, no external inputs are considered. The contact situation is described by a variable $s = \{0, 1\}$ with values $s = 0$ (no contact) and $s = 1$ (contact). For

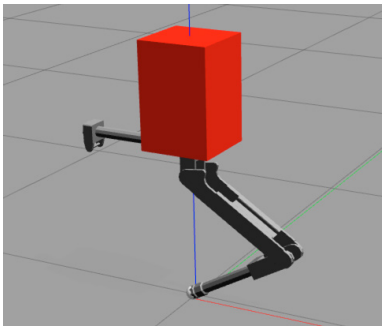


Fig. 3. Simulation of a hopper robot in Gazebo (adapted from (Winkler, 2017)).

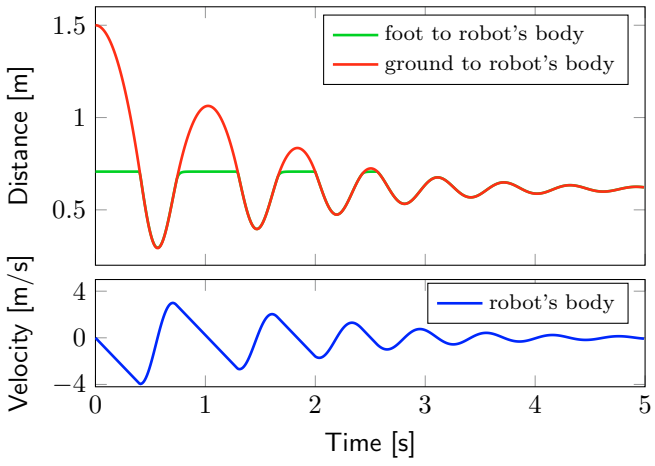


Fig. 4. One trajectory of the hopper robot. During ground contact, the distance between the robot's body to the ground and robot's body to the foot are identical.

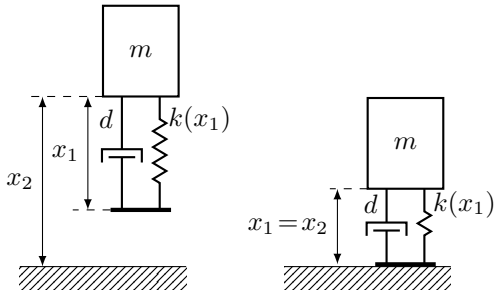


Fig. 5. Model of hopper robot without ground contact $s = 0$ (left) and with ground contact $s = 1$ (right).

GP-SPHS training, we performed 20 simulations between $t = [0, 5]$s with random initial points $\boldsymbol{x}_0$ drawn from a uniform distribution over $[0, 1] \times [0, 2] \times [-1, 1]$ such that the robot always starts in a non-contact situation. Within each run, every 0.1 s a data point, which includes the time, state and contact situation of the robot, is collected, which leads to a total of 1000 points. The measurements are corrupted by Gaussian noise with a signal-to-noise ratio of 39dB, 34dB, and 18dB for the three states, respectively.

Following Algorithm 1, a GP-SPHS is trained on a MacBook M1 Pro implemented in Python with a runtime of 22 s. Figure 6 shows the result of the GP classifier for the switching policy $\hat{s}$ based on the training data (circles). For each state $\boldsymbol{x}$, the classifier is able to predict the probability for each mode of the system. The mean accuracy on the given training data is 0.98, which indicates that almost all points can be correctly classified. For visualization, two slices in the 3d-space display the predicted probabilities. The green area indicates the subspace where $s = 1$ (contact) and red indicates $s = 0$ (no contact). After training, three sample trajectories for an unseen initial point $\boldsymbol{x}_0$ are drawn based on Algorithm 2 with Euler's method as an ODE solver and a step size of 1 ms. Figure 7 depicts the resulting trajectories (solid lines) and the trajectory of the actual system (dashed). The mean squared error between the actual system trajectory and the samples of the GP-PHS is 0.193. It can be observed that the uncertainty of

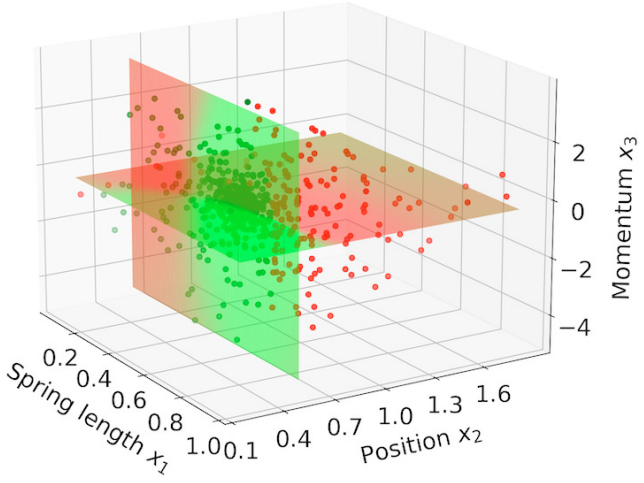


Fig. 6. GP classifier for the mode of the system. Based on the training data (points), the classifier estimates for each state the probability of being in mode $s = 0$ (no contact, red) and $s = 1$ (contact, green).

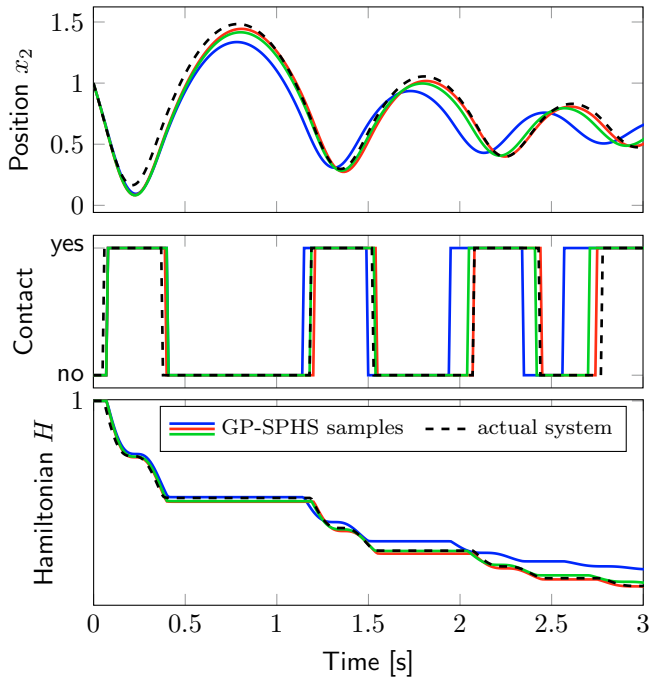


Fig. 7. Three realizations of the GP-SPHS and the resulting system trajectories. Real system in dashed black. Top: Position of mass always converges to a steady state. Middle: The contact time varies with the realization. Bottom: Hamiltonian is always decreasing over time such that all realizations represent physical plausible systems.

the prediction is increasing over time, but all realizations represent physical plausible systems for all time as the energy is always decreasing.

6. CONCLUSION

We propose Gaussian process switching Port-Hamiltonian systems to learn the dynamics of physical switching systems with partially unknown dynamics. The GP-SPHS model guarantees physical plausibility of its sampled trajectories, high flexibility for the learning of nonlinearities and allows for uncertainty quantification. Furthermore, it is shown that GP-SPHS preserve the passivity and interconnection properties of Port-Hamiltonian systems. In future work, we will extend GP-SPHS to a broader class of switching systems and perform experimental validation.

REFERENCES

Adler, R.J. (2010). *The geometry of random fields*. SIAM.

Anderson, R.B., Marshall, J.A., L'Afflitto, A., and Dotterweich, J.M. (2020). Model reference adaptive control of switched dynamical systems with applications to aerial robotics. *Journal of Intelligent & Robotic Systems*, 100(3), 1265–1281.

Beckers, T. and Hirche, S. (2016). Equilibrium distributions and stability analysis of Gaussian process state space models. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 6355–6361.

Beckers, T., Seidman, J., Perdikaris, P., and Pappas, G.J. (2022). Gaussian process port-Hamiltonian systems: Bayesian learning with physics prior. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, 1447–1453.

Bhouri, M.A. and Perdikaris, P. (2022). Gaussian processes meet NeuralODEs: a Bayesian framework for learning the dynamics of partially observed systems from scarce and noisy data. *Philosophical Transactions of the Royal Society A*, 380(2229), 20210201.

Brogliato, B. and Brogliato, B. (1999). *Nonsmooth mechanics*, volume 3. Springer.

Cervera, J., van der Schaft, A.J., and Baños, A. (2007). Interconnection of port-Hamiltonian systems and composition of dirac structures. *Automatica*, 43(2), 212–225.

Chen, R.T.Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D.K. (2018). Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31.

Desai, S.A., Mattheakis, M., Sondak, D., Protopapas, P., and Roberts, S.J. (2021). Port-Hamiltonian neural networks for learning explicit time-dependent dynamical systems. *Physical Review E*, 104(3), 034312.

Greydanus, S., Dzamba, M., and Yosinski, J. (2019). Hamiltonian neural networks. *Advances in Neural Information Processing Systems*, 32.

Hou, Z.S. and Wang, Z. (2013). From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, 235, 3–35.

Jiahao, T.Z., Hsieh, M.A., and Forgoston, E. (2021). Knowledge-based learning of nonlinear dynamics and chaos. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(11), 111101.

Karniadakis, G., Kevrekidis, I., Lu, L., Perdikaris, P., Wang, S., and Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, 3(6), 422–440.

Maschke, B.M. and van der Schaft, A.J. (1993). Port-controlled Hamiltonian systems: modelling origins and systemtheoretic properties. In *Nonlinear Control Systems Design*, 359–365. Elsevier.

Nageshrao, S.P., Lopes, G.A., Jeltsema, D., and Babuška, R. (2015). Port-Hamiltonian systems in adaptive and learning control: A survey. *IEEE Transactions on Automatic Control*, 61(5), 1223–1238.

Ortega, R. and Garcia-Canseco, E. (2004). Interconnection and damping assignment passivity-based control: A survey. *European Journal of Control*, 10(5), 432–450.

Rasmussen, C.E. and Williams, C.K. (2006). *Gaussian processes for machine learning*. MIT press Cambridge.

Rath, K., Albert, C.G., Bischl, B., and von Toussaint, U. (2021). Symplectic Gaussian process regression of maps in Hamiltonian systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(5), 053121.

Ridderbusch, S., Offen, C., Ober-Blöbaum, S., and Goulart, P. (2021). Learning ODE models with qualitative structure using Gaussian processes. In *Proceedings of the IEEE Conference on Decision and Control*, 2896–2896.

Van Der Schaft, A. and Camlibel, M.K. (2009). A state transfer principle for switching port-Hamiltonian systems. In *Proceedings of the IEEE Conference on Decision and Control (CDC)*, 45–50.

Van der Schaft, A. (2000). *L2-gain and passivity techniques in nonlinear control*. Springer.

Van Der Schaft, A. and Jeltsema, D. (2014). Port-Hamiltonian systems theory: An introductory overview.

Foundations and Trends in Systems and Control, 1(2-3), 173–378.

Williams, C.K. and Barber, D. (1998). Bayesian classification with Gaussian processes. *IEEE Transactions on pattern analysis and machine intelligence*, 20(12), 1342–1351.

Wilson, A. and Nickisch, H. (2015). Kernel interpolation for scalable structured Gaussian processes (KISS-GP). In *International conference on machine learning*, 1775–1784. PMLR.

Wilson, J., Borovitskiy, V., Terenin, A., Mostowsky, P., and Deisenroth, M. (2020). Efficiently sampling functions from Gaussian process posteriors. In *International Conference on Machine Learning*, 10292–10302. PMLR.

Winkler, A.W. (2017). Xpp - A collection of ROS packages for the visualization of legged robots. URL `https://doi.org/10.5281/zenodo.1037901`.

Wu, X., Zhang, K., Cheng, M., and Xin, X. (2018). A switched dynamical system approach towards the economic dispatch of renewable hybrid power systems. *International Journal of Electrical Power & Energy Systems*, 103, 440–457.

Appendix A

Proof of Theorem 1 As we place a GP with squared exponential kernel on $\nabla_x \hat{H}$, all realizations $\nabla_x \hat{H}(\boldsymbol{x}, \omega)$ with $\omega \in \Omega$ are smooth functions in $\boldsymbol{x}$, see (Rasmussen and Williams, 2006, Section 4.2). Using the fact that every smooth function $\hat{H} : \mathbb{R}^n \to \mathbb{R}$ defines a Port-Hamiltonian vector field under the affine transformation $\hat{H} \mapsto (J_s - R_s)\nabla \hat{H} + Gu$, see (Maschke and van der Schaft, 1993), all realizations $\nabla_x \hat{H}(\boldsymbol{x}, \omega)$ define PHS vector fields.

To show passivity, we must first show that there exists a $c_0 \in \mathbb{R}$ such that for almost all $\omega \in \Omega$, $\hat{H}(\boldsymbol{x}, \omega) \geq c_0$, see (Beckers et al., 2022). This guarantees that infinite energy cannot be extracted from the system. As the mean and covariance of a GP with squared exponential kernel are bounded, see (Beckers and Hirche, 2016), the standard deviation metric $d(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{var}(\hat{H}(\boldsymbol{x}) - \hat{H}(\boldsymbol{x}'))^{1/2}$ for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$ is totally bounded, and we get $d(\boldsymbol{x}, \boldsymbol{x}') < c_1$ for some $c_1 \in \mathbb{R}_{>0}$. This, together with the sample path continuity resulting from the squared exponential kernel allows us to conclude that $\hat{H}$ is almost surely bounded on $\mathcal{X}$ as the GP is closed under linear transformations, i.e, $\mathrm{P}(\sup_{\boldsymbol{x} \in \mathbb{R}^n} |\hat{H}| < \infty) = 1$. Thus, we have shown that there exists a $c_0 \in \mathbb{R}$ such that almost surely $\hat{H} > c_0$. Further, we can now consider the time-derivative of the sampled Hamiltonian for a $\omega \in \Omega$ during a mode $s \in \{1, \dots, n_s\}$

$$\begin{aligned}\dot{\hat{H}}(\boldsymbol{x}, \omega) &= \nabla_{\boldsymbol{x}}^\top \hat{H}(\boldsymbol{x}, \omega)[J_s(\boldsymbol{x}) - R_s(\boldsymbol{x})]\nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}, \omega) \\ &\quad + \nabla_{\boldsymbol{x}}^\top \hat{H}(\boldsymbol{x}, \omega) G(\boldsymbol{x})\boldsymbol{u} \qquad \text{(A.1)} \\ &= -\nabla_{\boldsymbol{x}}^\top \hat{H}(\boldsymbol{x}, \omega) R_s(\boldsymbol{x}) \nabla_{\boldsymbol{x}} \hat{H}(\boldsymbol{x}, \omega) + \boldsymbol{u}^\top \boldsymbol{y} \leq \boldsymbol{u}^\top \boldsymbol{y},\end{aligned}$$

where $\boldsymbol{u}^\top \boldsymbol{y}$ is the supply rate. As the dissipation matrix R_s is positive semi-definite by definition, equation (A.1) can be simplified to $\dot{\hat{H}}(\boldsymbol{x}, \omega) \leq \boldsymbol{u}^\top \boldsymbol{y}$. Thus, the change in the total energy of the system $\hat{H}$ in the mode s is less than the supply rate with the difference of the dissipation energy.

Finally, consider the state $\boldsymbol{x}^- \in \mathbb{R}^n$ of an SPHS at a switching time where the switch configuration of the system changes to $s^+ \in \{1, \dots, n_s\}$. Using the continuity property in Assumption 1 and the smoothness of $\hat{H}$, the new state $\boldsymbol{x}^+ \in \mathbb{R}^n$ just after the switching time satisfies $\boldsymbol{x}^- = \boldsymbol{x}^+$ such that $\hat{H}(\boldsymbol{x}^-, \omega) = \hat{H}(\boldsymbol{x}^+, \omega)$. As a consequence, the passivity condition $\dot{\hat{H}}(\boldsymbol{x}, \omega) \leq \boldsymbol{u}^\top \boldsymbol{y}$ holds for all time.

Proof of Theorem 2. Analogous to (Beckers et al., 2022), we start with the definition of two GP-PHS. The first system is given by

$$\begin{aligned}\dot{\boldsymbol{x}} &= \hat{J}_{R1}(\boldsymbol{x})\nabla_{\boldsymbol{x}} \hat{H}_1(\boldsymbol{x}, \omega_1) + G_1(\boldsymbol{x})\boldsymbol{u}_1 \\ \boldsymbol{y}_1 &= G_1(\boldsymbol{x})^\top \nabla_{\boldsymbol{x}} \hat{H}_1(\boldsymbol{x}, \omega_1),\end{aligned}$$

with Hamiltonian $\hat{H}_1 \sim \mathcal{GP}(0, k_H(\boldsymbol{x}, \boldsymbol{x}'))$, state $\boldsymbol{x} \in \mathbb{R}^{n_1}$, sample $\omega_1 \in \Omega$, modes $\hat{s}_1(\boldsymbol{x}, \omega_1) \in \{1, \dots, n_{s,1}\}$, $\hat{J}_{R1}(\boldsymbol{x}) = J_{\hat{s}_1(\boldsymbol{x},\omega_1),1}(\boldsymbol{x}) - R_{\hat{s}_1(\boldsymbol{x},\omega_1),1}(\boldsymbol{x})$, and input/output $\boldsymbol{u}_1, \boldsymbol{y}_1 \in \mathbb{R}^{m_1}$. We separate the I/O matrix G_1 into $G_1^{\mathrm{c}}(\boldsymbol{x}) \in \mathbb{R}^{n_1 \times m_c}$ and $G_1^{\mathrm{ex}}(\boldsymbol{x}) \in \mathbb{R}^{n_1 \times m_1 - m_c}$ such that $G(\boldsymbol{x})\boldsymbol{u}_1 = \hat{G}_1^{\mathrm{c}}\boldsymbol{u}_1^{\mathrm{c}} + G_1^{\mathrm{ex}}\boldsymbol{u}_1^{\mathrm{ex}}$ with external input $\boldsymbol{u}_1^{\mathrm{ex}} \in \mathbb{R}^{m_1 - m_c}$. The output for connection $\boldsymbol{y}_1^{\mathrm{c}}$ is given by $\boldsymbol{y}_1^{\mathrm{c}} = G_1^{\mathrm{c}}(\boldsymbol{x})^\top \nabla_{\boldsymbol{x}} \hat{H}_1(\boldsymbol{x}, \omega_1)$. Analogously, the second system is defined by

$$\begin{aligned}\dot{\boldsymbol{\xi}} &= \hat{J}_{R2}(\boldsymbol{\xi})\nabla_{\boldsymbol{\xi}} \hat{H}_2(\boldsymbol{\xi}, \omega_2) + G_2(\boldsymbol{\xi})\boldsymbol{u}_2 \\ \boldsymbol{y}_2 &= G_2(\boldsymbol{\xi})^\top \nabla_{\boldsymbol{\xi}} \hat{H}_2(\boldsymbol{\xi}, \omega_2),\end{aligned}$$

with Hamiltonian $\hat{H}_2 \sim \mathcal{GP}(0, k)_H(\boldsymbol{\xi}, \boldsymbol{\xi}'))$, state $\boldsymbol{\xi} \in \mathbb{R}^{n_2}$, sample $\omega_2 \in \Omega$, modes $\hat{s}_2(\boldsymbol{x}, \omega_2) \in \{1, \dots, n_{s,2}\}$, $\hat{J}_{R2}(\boldsymbol{x}) = J_{\hat{s}_2(\boldsymbol{x},\omega_2),2}(\boldsymbol{x}) - R_{\hat{s}_2(\boldsymbol{x},\omega_2),2}(\boldsymbol{x})$, and input/output $\boldsymbol{u}_2, \boldsymbol{y}_2 \in \mathbb{R}^{m_2}$. The I/O matrix G_2 is separated into $G_2^{\mathrm{c}}(\boldsymbol{\xi}) \in \mathbb{R}^{n_2 \times m_c}$ and $\hat{G}_2^{\mathrm{ex}}(x) \in \mathbb{R}^{n_2 \times m_2 - m_c}$ such that $G(\boldsymbol{\xi})\boldsymbol{u}_2 = G_2^{\mathrm{c}}\boldsymbol{u}_2^{\mathrm{c}} + G_2^{\mathrm{ex}}\boldsymbol{u}_2^{\mathrm{ex}}$ with $\boldsymbol{u}_2^{\mathrm{ex}} \in \mathbb{R}^{m_2 - m_c}$. The output of the connection $\boldsymbol{y}_2^{\mathrm{c}}$ is given by $\boldsymbol{y}_2^{\mathrm{c}} = G_2^{\mathrm{c}}(\boldsymbol{\xi})^\top \nabla_{\boldsymbol{\xi}} \hat{H}_2(\boldsymbol{\xi}, \omega_2)$.
For the interconnection $\boldsymbol{u}_1^{\mathrm{c}} = -\boldsymbol{y}_2^{\mathrm{c}}$ and $\boldsymbol{u}_2^{\mathrm{c}} = \boldsymbol{y}_1^{\mathrm{c}}$, we get

$$\begin{aligned}\underbrace{\begin{bmatrix}\dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\xi}}\end{bmatrix}}_{\boldsymbol{z}} &= \underbrace{\begin{bmatrix}\hat{J}_{R1}(\boldsymbol{x}) & -G_1^{\mathrm{c}}(\boldsymbol{x})[G_2^{\mathrm{c}}(\boldsymbol{\xi})]^\top \\ G_2^{\mathrm{c}}(\boldsymbol{\xi})[G_1^{\mathrm{c}}(\boldsymbol{x})]^\top & \hat{J}_{R2}(\boldsymbol{\xi})\end{bmatrix}}_{J_{\hat{s}(\boldsymbol{z},\boldsymbol{\omega})}(\boldsymbol{z}) - R_{\hat{s}(\boldsymbol{z},\boldsymbol{\omega})}(\boldsymbol{z})} \begin{bmatrix}\nabla_{\boldsymbol{x}} \\ \nabla_{\boldsymbol{\xi}}\end{bmatrix} \hat{H}(\boldsymbol{z}, \boldsymbol{\omega}) \\ &\quad + \underbrace{[\hat{G}_1^{\mathrm{ex}}(\boldsymbol{x}), \hat{G}_2^{\mathrm{ex}}(\boldsymbol{\xi})]}_{G(\boldsymbol{z})} \begin{bmatrix}\boldsymbol{u}_1^{\mathrm{ex}} \\ \boldsymbol{u}_2^{\mathrm{ex}}\end{bmatrix} \\ \boldsymbol{y} &= [\hat{G}_1^{\mathrm{ex}}(\boldsymbol{x}), \hat{G}_2^{\mathrm{ex}}(\boldsymbol{\xi})]^\top \begin{bmatrix}\nabla_{\boldsymbol{x}} \\ \nabla_{\boldsymbol{\xi}}\end{bmatrix} \hat{H}(\boldsymbol{z}, \boldsymbol{\omega})\end{aligned}$$

with $\boldsymbol{z} = [\boldsymbol{x}^\top, \boldsymbol{\xi}^\top]^\top$, $\boldsymbol{\omega} = [\omega_1, \omega_2]^\top$, $J_{\hat{s}(\boldsymbol{z},\boldsymbol{\omega})}(\boldsymbol{z}), R_{\hat{s}(\boldsymbol{z},\boldsymbol{\omega})}(\boldsymbol{z}) \in \mathbb{R}^{n \times n}$, $n = n_1 + n_2$, and $G(\boldsymbol{z}) \in \mathbb{R}^{n \times m}$, $m = m_1 + m_2 - 2m_c$ and output $\boldsymbol{y} \in \mathbb{R}^n$. Then, we can write the derivative of the Hamiltonian of the overall system as $\nabla_{\boldsymbol{z}} H(\boldsymbol{z}, \boldsymbol{\omega}) = [\nabla_{\boldsymbol{x}}^\top \hat{H}_1(\boldsymbol{x}, \omega_1), \nabla_{\boldsymbol{\xi}}^\top \hat{H}_2(\boldsymbol{\xi}, \omega_2)]^\top$ and the switching variable as $\hat{s}(\boldsymbol{z}) \in \{1, \dots, n_{s,1} n_{s,2}\}$ that completes the proof.