Received 1 December 2022; revised 19 March 2023 and 4 May 2023; accepted 6 May 2023. Date of publication 17 May 2023; date of current version 21 June 2023. Recommended by Senior Editor Lacra Pavel.

Digital Object Identifier 10.1109/OJCSYS.2023.3277308

Model-Based Reinforcement Learning via Stochastic Hybrid Models

HANY ABDULSAMAD [1] AND JAN PETERS [2] (Fellow, IEEE)

(Intersection of Machine Learning With Control)

[1]Department of Electrical Engineering and Automation, Aalto University, 02150 Espoo, Finland
[2]Department of Computer Science, Technical University of Darmstadt, 64289 Darmstadt, Germany

CORRESPONDING AUTHOR: HANY ABDULSAMAD (e-mail: hany.abdulsamad@aalto.fi).

ABSTRACT Optimal control of general nonlinear systems is a central challenge in automation. Enabled by powerful function approximators, data-driven approaches to control have recently successfully tackled challenging applications. However, such methods often obscure the structure of dynamics and control behind black-box over-parameterized representations, thus limiting our ability to understand closed-loop behavior. This article adopts a hybrid-system view of nonlinear modeling and control that lends an explicit hierarchical structure to the problem and breaks down complex dynamics into simpler localized units. We consider a sequence modeling paradigm that captures the temporal structure of the data and derive an expectation-maximization (EM) algorithm that automatically decomposes nonlinear dynamics into stochastic piecewise affine models with nonlinear transition boundaries. Furthermore, we show that these time-series models naturally admit a closed-loop extension that we use to extract local polynomial feedback controllers from nonlinear experts via behavioral cloning. Finally, we introduce a novel hybrid relative entropy policy search (Hb-REPS) technique that incorporates the hierarchical nature of hybrid models and optimizes a set of time-invariant piecewise feedback controllers derived from a piecewise polynomial approximation of a global state-value function.

INDEX TERMS Bayesian inference, behavioral cloning, expectation-maximization, hidden Markov models, hybrid models, piecewise feedback control, reinforcement learning, system identification.

I. INTRODUCTION

The class of nonlinear dynamical systems governs a vast range of real-world applications and underpins the most challenging problems in classical control, and reinforcement learning (RL) [1], [2]. Recent developments in learning-for-control have pushed towards deploying more complex and highly sophisticated representations, e.g., (deep) neural networks and Gaussian processes, to capture the structure of both dynamics and controllers. This trend led to unprecedented success in the domain of RL [3] and can be observed in both approximate optimal control [4], [5], [6], and approximate value and policy iteration algorithms [7], [8], [9].

However, before the latest revival of neural networks, research has focused on different paradigms for solving complex control tasks. One interesting concept relied on decomposing nonlinear structures of dynamics and control into simpler piecewise (affine) components, each responsible for an area of the state-action space. Instances of this abstraction can be found in the control literature under the labels of hybrid systems or switched models [10], [11], [12], [13], while in the machine and reinforcement learning communities, the terminology of switching dynamical systems and hybrid state-space models is more widely used [14], [15], [16], [17].

While the hybrid-state paradigm is a natural choice for studying jump processes, it also provides a surrogate piecewise approximation of general nonlinear dynamical behavior. Despite being less flexible than generic black-box approximators, hybrid models can regularize functional complexity and contribute to improved interpretability by imposing a structured representation.

Adopting this perspective in this article, we present techniques for data-driven automatic system identification and closed-loop control of general nonlinear systems using piecewise polynomial hybrid surrogate models. More concretely, we focus on dynamic Bayesian graphical models as hybrid representations due to their favorable properties. These models have an inherent time-recurrent structure that captures correlations over extended horizons and carry over the advantages of well-established recursive Bayesian inference techniques for dynamical time series data.

In prior work [18], we presented a maximum likelihood approach for hierarchical piecewise system identification and behavioral cloning. Here, we robustify that approach by introducing suitable priors over all parameters. However, the central contribution of this article is the introduction of an infinite horizon reinforcement learning framework that integrates the structured representation of stochastic hybrid models. The resulting algorithm interactively synthesizes nonlinear feedback controllers and value functions via a hierarchical piecewise polynomial architecture.

This article is structured as follows. In Section II, we start by reviewing and comparing prominent paradigms of system modeling and optimal control of hybrid systems. Using that context in Section III, we highlight the advantages of our contributions in comparison with the literature. In Section IV, we cast the control problem as an infinite horizon Markov decision process and extended it to accommodate a hybrid structure. Next, in Section V, we introduce our notation of stochastic switching models in the form of hybrid dynamic Bayesian networks, as previously established in [18]. In Section VI, we recap our approach from [18] and improve it to derive a maximum a posteriori expectation-maximization (EM) algorithm for inferring the parameters of probabilistic hybrid models from data. This inference method is helpful for automatically decomposing nonlinear open-loop dynamics into switching affine regimes with arbitrary boundaries and deconstructing state-of-the-art nonlinear expert controllers into piecewise polynomial policies. Furthermore, in Section VII, we formulate hybrid optimal control as a stochastic optimization problem and derive a trust-region reinforcement learning algorithm that incorporates an explicit hierarchical model of the nonlinear dynamics. We use this approach to iteratively learn piecewise approximations of the global nonlinear value function and stationary feedback controller. Finally, in Section VIII, we empirically evaluate our approaches on examples of stochastic nonlinear systems, including results from [18] that contribute to the overall picture.

Our empirical evaluation indicates that hybrid models can provide an alternative to generic black-box representations for system identification, behavioral cloning, and learning-based control. Hybrid models are able to reach comparable performance and deliver simpler, easily identifiable switching patterns of dynamics and control while requiring a fraction of the number of parameters of other functional forms. However, the results also reveal certain drawbacks, mainly in poor scalability and increased algorithmic complexity. We address these issues in a final outlook in Section IX.

II. RELATED WORK

This section reviews work related to the modeling and control of hybrid systems and highlights connections and parallels between approaches stemming from the control and machine and reinforcement learning literature.

Hybrid systems have been extensively studied in the control community and are widely used in real-world applications [19], [20]. For research on hybrid system identification, we refer to survey work in [21] and [22]. There, the authors focus on piecewise affine (PWA) systems and introduce taxonomies of different representations and procedures commonly used for identifying sub-regimes of dynamics, ranging from algebraic approaches [23] to mixed-integer optimization [24], and Bayesian methods [25]. Furthermore, identification techniques for piecewise nonlinear systems have been developed based on sparse optimization [26] and kernel methods [27]. Finally, it is worth noting that the majority of literature considers deterministic regime-switching events with exceptions in [28], [29].

Research in the area of optimal control for hybrid systems stretches back to the seminal work in [30], which highlights the possibility of general nonlinear control by considering piecewise affine systems. In [31], an overview of control approaches for piecewise affine switching dynamics is presented. The authors categorize the literature by distinguishing between externally and internally forced switching mechanisms. The bulk of optimal control approaches in this area focuses on (nonlinear) model predictive control (MPC) [32]. Here we highlight the influential work in [33], which formulates the optimal control problem as a mixed-integer quadratic program (MIQP). This approach was later extended in [34] and [35] to solve a multi-parametric MIQP and arrive at time-variant piecewise affine state-feedback controllers and piecewise quadratic value functions with polyhedral partitions. Recently, more efficient formulations of hybrid control have been proposed [36], which leverage modern techniques from mixed-integer and disjunctive programming to tackle large-scale problems.

Hybrid representations also play a central role in data-driven, general-purpose process modeling and state estimation [37], [38], where different classes of stochastic hybrid systems serve as powerful generative models for complex dynamical behaviors [39], [40], [41]. The dominant paradigm in this domain has been that of probabilistic graphical models (PGM), more specifically, hybrid dynamic Bayesian networks (HDBN) for temporal modeling [42], [43]. One crucial contribution of recent Bayesian interpretations of switching systems is rooted in the Bayesian nonparametric (BNP) view [44], [45], [46], [47]. This perspective theoretically allows for an infinite number of components, thus dramatically increasing the expressiveness of such models. Given the limited scope of this review section, we highlight only recent contributions with high impacts, such as [48] and [17], which successfully develop Markov chain Monte Carlo (MCMC) and stochastic variational inference (SVI) techniques for system identification. More recently, the rise of variational auto-encoders [49] has enabled a new and powerful view of inference techniques [50] for hybrid systems. A distinct drawback of such approaches is their reliance on end-to-end differentiability and the need to relax discrete variables in order to perform inference.

In the domain of learning-for-control, the notion of switching systems is directly related to the paradigm of model-free hierarchical reinforcement learning (HRL) [51], [52], which combines simple representations to build complex policies. Here it is useful to differentiate between two concepts of hierarchical learning, namely *temporal* [53], and *state* abstractions [54]. In their seminal work [55], [56], the authors build on the framework of semi-Markov decision processes (SMDP) [57] to learn activation/termination conditions of temporally extended actions (options) for solving discrete environments. Additionally, pioneering work in optimizing hierarchical control structures with temporally extended actions is developed in [58] and [59]. Recent work has focused on formulations of the SMDP framework that facilitate simultaneous discovery and learning of options [60], [61], [62], [63], [64].

However, the concept of state abstraction - partitioning state-action spaces into sub-regions, each governed by local dynamics and control - carries the most apparent parallels to the classical view of hybrid systems. In [65], a proof of convergence for RL in tabular environments with state abstraction is presented, while [66] does a comprehensive study of different abstraction schemes and gives a formal definition of the problem. Furthermore, recent work has shown promising results in solving complex tasks by combining local policies, albeit while leveraging a complex neural network architecture as an upper-level policy [67].

Switching systems serve as a powerful tool in behavioral cloning. For example, [68] combines hidden Markov models (HMMs) with Gaussian mixture regression to represent trajectory distributions. In contrast, [62] uses a semi-hidden Markov model (HSMM) to learn hierarchical policies, and [69] introduces switching density networks for system identification and behavioral cloning. Finally, a Bayesian framework for the hierarchical policy decomposition is presented in [70], albeit while considering known transition dynamics.

III. CONTRIBUTION

In light of the motivation and reviewed literature from Section I and II, we establish here the overall contribution of our methodology and highlight the main differences that distinguish it from related approaches.

As previously stated, this work strives to cast the problem of nonlinear optimal control into a data-driven hierarchical learning framework. Our aim is to introduce explicit structure and adopt hybrid surrogate models to avoid the opaqueness of recently popularized black-box representations. While this paradigm has been established before, our realization differs from previous attempts in two central aspects:

- *System Modeling:* This work leverages probabilistic hybrid dynamic networks as hierarchical representations of nonlinear dynamics. Contrary to a piecewise autoregressive exogenous systems (PWARX), HDBNs straightforwardly accounts for noise in both discrete and continuous dynamics. They also incorporate nonlinear transition boundaries, thus minimizing partitioning redundancy. Furthermore, HDBNs admit efficient inference methods in data-driven applications. Finally, by pursuing an abstraction over states instead of time, we circumvent the need to infer termination policies of the SMDP framework.
- *Control Synthesis:* We propose a hybrid policy search approach that formulates a non-convex infinite horizon objective and optimizes a piecewise polynomial approximation of the value function with nonlinear partitioning. This approximation is used to derive stationary switching feedback controllers. In contrast, trajectory optimization and model predictive control techniques for hybrid models are often cast as sequential convex programs that assume polyhedral partitions and optimize a fixed horizon objective, yielding time-variant value functions and controls.

IV. PROBLEM STATEMENT

Consider the discrete-time optimal control problem of a stochastic nonlinear dynamical system to be defined as an infinite horizon Markov decision processes (MDP). An MDP is defined over a state space $\mathcal{X} \subseteq \mathbb{R}^d$ and an action space $\mathcal{U} \subseteq \mathbb{R}^m$. The probability of a state transition from state $\mathbf{x}$ to state $\mathbf{x}'$ by applying action $\mathbf{u}$ is governed by the Markovian time-independent density function $p(\mathbf{x}'|\mathbf{x}, \mathbf{u})$. The reward $r(\mathbf{x}, \mathbf{u})$ is a function of the state $\mathbf{x}$ and action $\mathbf{u}$. The state-dependent policy $\pi(\mathbf{u}|\mathbf{x})$, from which the actions are drawn, is a density determining the probability of an action $\mathbf{u}$ given a state $\mathbf{x}$. The general objective in an average-reward infinite horizon optimal control problem is to maximize the average of rewards $V^\pi(\mathbf{x}) = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\big[\sum_{t=1}^{T} r\big]$, where V^π denotes as the state-value function under the policy π, starting from an initial state distribution $\mu_1(\mathbf{x})$.

Given the context of this work and our choice to model the system with hybrid models, we introduce to the MDP formulation a new hidden discrete variable $\mathbf{z}$, an indicator of the currently active local regime. The resulting transition dynamics can then be expressed by a factorized density function $p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z}) = p(\mathbf{z}'|\mathbf{z}, \mathbf{x}, \mathbf{u})p(\mathbf{x}'|\mathbf{x}, \mathbf{u}, \mathbf{z}')$, which we depict as a graphical model in Fig. 2 and discuss in further detail in the upcoming section. In the same spirit of simplification through hierarchical modeling, we employ a mixture of switching polynomial controllers $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$, associated with a piecewise polynomial value function $V^\pi(\mathbf{x}, \mathbf{z})$.

V. HYBRID DYNAMIC BAYESIAN NETWORKS

In this section, we focus on the modeling assumptions for the stochastic switching transition dynamics $p(\mathbf{x}'', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})$, see Section IV. We choose recurrent autoregressive hidden

Markov models (rARHMMs) as a representation, which is a special case of recurrent switching linear dynamical systems (rSLDS) [17], also known as augmented SLDS [71]. In contrast to rSLDS, an rARHMM lacks an observation model and directly describes the internal state up to an additive noise process. We extend rARHMMs to support exogenous and endogenous inputs in order to simulate the open- and closed-loop behaviors of driven dynamics. Fig. 2 depicts the corresponding graphical model, which closely resembles the graph of a PWARX.

An rARHMM with K regions models the trajectory of a dynamical system as follows. The initial continuous state $\mathbf{x}_1 \in \mathbb{R}^d$ and continuous action $\mathbf{u}_1 \in \mathbb{R}^m$ are drawn from a pair of Gaussian and conditional Gaussian distributions,[1] respectively. The initial discrete state $\mathbf{z}_1$ is a random vector modeled by a categorical density parameterized by $\boldsymbol{\varphi}$

$$\mathbf{z}_1 \sim \mathrm{Cat}(\boldsymbol{\varphi}), \ \mathbf{x}_1 \sim \mathrm{N}(\boldsymbol{\mu}_{\mathbf{z}_1}, \boldsymbol{\Omega}_{\mathbf{z}_1}),$$
$$\mathbf{u}_1 \sim \mathrm{N}(\mathbf{K}_{\mathbf{z}_1}\phi(\mathbf{x}_1), \boldsymbol{\Delta}_{\mathbf{z}_1}).$$

The transition of the continuous state $\mathbf{x}_{t+1}$ and actions $\mathbf{u}_t$ are modeled by affine-Gaussian dynamics

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{A}_{\mathbf{z}_{t+1}}\mathbf{x}_t + \mathbf{B}_{\mathbf{z}_{t+1}}\mathbf{u}_t + \mathbf{c}_{\mathbf{z}_{t+1}} + \boldsymbol{\lambda}_t, && \boldsymbol{\lambda}_t \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Lambda}_{\mathbf{z}_{t+1}}), \\ \mathbf{u}_t &= \mathbf{K}_{\mathbf{z}_t}\phi(\mathbf{x}_t) + \boldsymbol{\delta}_t, && \boldsymbol{\delta}_t \sim \mathrm{N}(\mathbf{0}, \boldsymbol{\Delta}_{\mathbf{z}_t}), \end{aligned}$$

where $(\mathbf{A}, \mathbf{B}, \mathbf{c}, \mathbf{K}, \boldsymbol{\Omega}, \boldsymbol{\Lambda}, \boldsymbol{\Delta})$ are matrices and vectors of appropriate dimensions with respect to $\mathbf{x}$ and $\mathbf{u}$. $\phi(\mathbf{x})$ are polynomial state features of arbitrary degree.

The discrete transition probability $p(\mathbf{z}_{t+1}|\mathbf{z}_t, \mathbf{x}_t, \mathbf{u}_t)$ is governed by K categorical distributions parameterized by a state-action dependent multi-class logit link function f [72]

$$\chi_{ij} = p(\mathbf{z}_{t+1} = j | \mathbf{z}_t = i, \mathbf{x}_t, \mathbf{u}_t) \propto \exp\left(f(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\omega}_{ij})\right), \quad (1)$$

where f may have any type of features in $(\mathbf{x}, \mathbf{u})$[2]. The vectors $\boldsymbol{\omega}_{ij}$ parameterize the discrete transition probabilities for all transition combinations $i \to j \, \forall i, j \in [1, K]$. Fig. 1 depicts realizations of different logit link functions leading to various state space partitioning.

The remainder of this article focuses on using these hybrid models in three scenarios:

- An *open-loop* setting that treats the control $\mathbf{u}$ as an exogenous input is used for automatically identifying nonlinear systems via decomposition into continuous and discrete switching dynamics.
- A *closed-loop* setting that assumes the control $\mathbf{u}$ to originate from a nonlinear controller. We show that this setting can simultaneously decompose dynamics and control in a behavioral cloning scenario.
- A *reinforcement learning* setting where we develop a model-based hybrid policy search algorithm to learn switching controllers for general nonlinear systems.

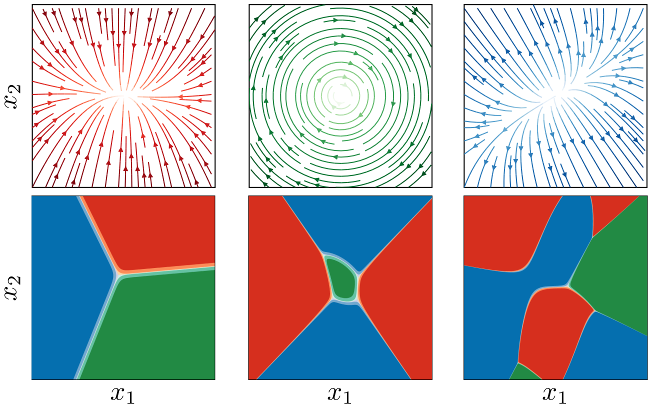

FIGURE 1. A hybrid system with $K = 3$ piecewise affine regimes. The top row depicts the mean unforced continuous transition dynamics in the phase space. The bottom row shows the distinct activation regions of the three dynamics regime across the phase space. We illustrate examples of affine (left), quadratic (middle), and third-order polynomial (right) switching boundaries. Figure reproduced from [18].

VI. BAYESIAN INFERENCE OF HYBRID MODELS

In this section, we sketch the outline of an expectation-maximization/Baum-Welch algorithm [73], [74], [75] for inferring the parameters of an rARHMM given time-series observations. The resulting algorithm can be used two-fold. First, it can be applied to automatically identify hybrid models and approximate the open-loop dynamics of nonlinear systems given state-action observations. Second, it can clone the closed-loop behavior of a nonlinear controller and decompose it into a set of local experts.

Our developed approach is related in some aspects to the Baum-Welch algorithms proposed in [76] and [62]. However, we introduce suitable priors over all parameters and derive a maximum a posteriori (MAP) technique with a stochastic maximization step and hyperparameter optimization. In our experience, the priors significantly regularize the sensitivity of EM with respect to the initial point, making it less prone to getting stuck in bad local minima.

Moreover, a good prior specification is crucial in small data regimes since a vague prior may dominate the predictive posterior and effectively cause under-fitting. We implement a hyperparameter optimization scheme that elevates this concern by optimizing the prior parameters via empirical Bayes [77], thus attenuating the prior influence and improving the predictive performance significantly.

A. MAXIMUM A POSTERIORI OPTIMIZATION

Consider again the rARHMM in Fig. 2 where the continuous state $\mathbf{x}$ and action $\mathbf{u}$ are observed variables, while the K-region indicators $\mathbf{z}$ are hidden. To infer the model parameters, we assume a dataset of N state-action trajectories $\boldsymbol{\mathcal{D}} = \{\boldsymbol{\mathcal{D}}^n\}_{n=1}^N = \{\mathbf{X}^n, \mathbf{U}^n\}_{n=1}^N$, each of length T, where $(\mathbf{X}^n, \mathbf{U}^n, \mathbf{Z}^n)$ represent the time concatenation of an entire trajectory $(\mathbf{x}_{1:T}^n, \mathbf{u}_{1:T}^n, \mathbf{z}_{1:T}^n)$.

The objective corresponding to system identification and behavioral cloning can be cast as a maximization

[1] We parameterize all Gaussian distributions by their precision matrices instead of the more common definition with covariances.

[2] We abuse notation slightly by sometimes using $\mathbf{z}$ to refer to the discrete state index instead of treating it as a one-hot vector.

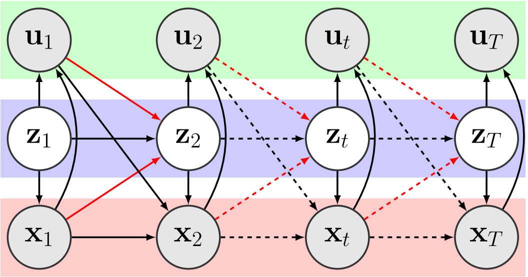

FIGURE 2. Graphical model of rARHMMs extended to support hybrid controls. In rARHMMs, the discrete state explicitly depends on the continuous state and action as highlighted in red. Figure reproduced from [18].

problem of the log-posterior probability of the observations $\{\mathbf{X}^n, \mathbf{U}^n\}_{n=1}^N$, with respect to the free parameter set $\boldsymbol{\theta} = \{\boldsymbol{\varphi}, \boldsymbol{\mu}_k, \boldsymbol{\Omega}_k, \mathbf{A}_k, \mathbf{B}_k, \mathbf{c}_k, \boldsymbol{\Lambda}_k, \mathbf{K}_k, \boldsymbol{\Delta}_k, \boldsymbol{\omega}_{ik}\}_{i,k=1}^K$

$$\boldsymbol{\theta}_{\text{MAP}} := \arg\max_{\boldsymbol{\theta}} \log \prod_{n=1}^{N} \sum_{\mathbf{z}^n} p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n | \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{h}), \tag{2}$$

where $p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n|\boldsymbol{\theta})$ is the complete-data likelihood of a single trajectory and factorizes according to

$$\begin{aligned} p(.|\boldsymbol{\theta}) = {} & p(\mathbf{z}_1^n|\boldsymbol{\varphi}) p(\mathbf{x}_1^n|\boldsymbol{\mu}_{\mathbf{z}_1^n}, \boldsymbol{\Omega}_{\mathbf{z}_1^n}) p(\mathbf{u}_1^n|\mathbf{x}_1^n, \mathbf{K}_{\mathbf{z}_1^n}, \boldsymbol{\Delta}_{\mathbf{z}_1^n}) \\ & \times \prod_{t=2}^{T} p(\mathbf{x}_t^n|\mathbf{x}_{t-1}^n, \mathbf{u}_{t-1}^n, \mathbf{A}_{\mathbf{z}_t^n}, \mathbf{B}_{\mathbf{z}_t^n}, \mathbf{c}_{\mathbf{z}_t^n}, \boldsymbol{\Lambda}_{\mathbf{z}_t^n}) \\ & \times \prod_{t=2}^{T} p(\mathbf{z}_t^n|\mathbf{z}_{t-1}^n, \mathbf{x}_{t-1}^n, \mathbf{u}_{t-1}^n, \boldsymbol{\omega}) \\ & \times \prod_{t=2}^{T} p(\mathbf{u}_t^n|\mathbf{x}_t^n, \mathbf{K}_{\mathbf{z}_t^n}, \boldsymbol{\Delta}_{\mathbf{z}_t^n}), \end{aligned} \tag{3}$$

and $p(\boldsymbol{\theta}|\mathbf{h})$ is the factorized parameter prior

$$\begin{aligned} p(\boldsymbol{\theta}|\mathbf{h}) = {} & p(\boldsymbol{\varphi}) \prod_{k=1}^{K} p(\boldsymbol{\mu}_k|\boldsymbol{\Omega}_k) p(\boldsymbol{\Omega}_k) \\ & \times \prod_{k=1}^{K} p(\mathbf{A}_k|\boldsymbol{\Lambda}_k) p(\mathbf{B}_k|\boldsymbol{\Lambda}_k) p(\mathbf{c}_k|\boldsymbol{\Lambda}_k) p(\boldsymbol{\Lambda}_k) \\ & \times \prod_{k=1}^{K} p(\mathbf{K}_k|\boldsymbol{\Delta}_k) p(\boldsymbol{\Delta}_k) \prod_{i=1}^{K} \prod_{k=1}^{K} p(\boldsymbol{\omega}_{ik}). \end{aligned}$$

We choose all priors to be conjugate or semi-conjugate with respect to their likelihoods. Therefore, we place a normal-Wishart (NW) prior on the initial state distribution $(\boldsymbol{\mu}_k, \boldsymbol{\Omega}_k) \sim \text{NW}(\mathbf{0}, \kappa_0, \boldsymbol{\Psi}_0, \nu_0)$, and a matrix-normal-Wishart (MNW) on the affine transition dynamics $(\mathbf{A}_k, \mathbf{B}_k, \mathbf{c}_k, \boldsymbol{\Lambda}_k) \sim \text{MNW}(\mathbf{0}, \mathbf{R}_0, \boldsymbol{\Phi}_0, \rho_0)$. The initial discrete state takes a Dirichlet prior $\boldsymbol{\varphi} \sim \text{Dir}(\boldsymbol{\tau}_0)$, while the logit link function parameters are governed by a non-conjugate zero-mean Gaussian prior with diagonal precision $\boldsymbol{\omega}_{ik} \sim \text{N}(\mathbf{0}, \alpha\mathbf{I})$. Finally, we place a separate matrix-normal-Wishart prior on the conditional action likelihood $(\mathbf{K}_k, \boldsymbol{\Delta}_k) \sim \text{MNW}(\mathbf{0}, \mathbf{S}_0, \boldsymbol{\Gamma}_0, \varepsilon_0)$. The quantities $(\kappa_0, \boldsymbol{\Psi}_0, \nu_0, \mathbf{R}_0, \boldsymbol{\Phi}_0, \rho_0, \boldsymbol{\tau}_0, \alpha, \mathbf{S}_0, \boldsymbol{\Gamma}_0, \varepsilon_0)$ are hyperparameters aggregated into the hyperparameter set $\mathbf{h}$.

The choice of priors is not restricted to these distributions. Depending on modeling assumptions, one can assume dynamics with diagonal noise matrices and pair them with gamma distribution priors. Moreover, if the system is known to have a state-independent noise process, the K Wishart and gamma priors can be *tied* across components, leading to a more structured representation.

B. BAUM-WELCH EXPECTATION-MAXIMIZATION

On closer examination of (2) and (3), we observe that the optimization problem is non-convex with multiple local optima since the complete-data likelihood $\prod_{n=1}^N p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n|\boldsymbol{\theta})$ can follow complex multi-modal densities. Another technical difficulty is the summation over all possible trajectories of the hidden variables $\mathbf{Z}^n$, which is of computational complexity $\mathcal{O}(NK^T)$ and is intractable in most cases. Expectation-maximization algorithms overcome the latter problem by introducing a variational posterior distribution over the hidden variables $q(\mathbf{Z}^n)$ and deriving a lower bound on the complete log-probability function

$$\begin{aligned} & \log \prod_{n=1}^{N} \sum_{\mathbf{z}^n} p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n, \boldsymbol{\theta}|\mathbf{h}) \\ & \geq \sum_{n=1}^{N} \sum_{\mathbf{z}^n} q(\mathbf{Z}^n) \log \frac{p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n, \boldsymbol{\theta}|\mathbf{h})}{q(\mathbf{Z}^n)}. \end{aligned} \tag{4}$$

We find a point estimate $\boldsymbol{\theta}_{\text{MAP}}$ by following a modified scheme of EM, alternating between an expectation step (E-step), in which the lower bound in (4) is maximized with respect to the variational distributions $q(\mathbf{Z}^n)$ given a parameter estimate $\hat{\boldsymbol{\theta}}$, a maximization step (M-step), that updates $\boldsymbol{\theta}$ given $(\hat{q}(\mathbf{Z}^n), \hat{\mathbf{h}})$, and finally, an empirical Bayes step (EB-step) that updates $\mathbf{h}$ given $(\hat{q}(\mathbf{Z}^n), \hat{\boldsymbol{\theta}})$. A sketch of the overall iterative procedure is presented in Algorithm 1.

1) EXACT EXPECTATION STEP

Maximizing the lower bound with respect to $q(\mathbf{Z}^n)$ is determined by reformulating (4)

$$\begin{aligned} L = {} & \sum_{n=1}^{N} \sum_{\mathbf{z}^n} q(\mathbf{Z}^n) \log \frac{p(\boldsymbol{\mathcal{D}}^n, \mathbf{Z}^n, \boldsymbol{\theta}|\mathbf{h})}{q(\mathbf{Z}^n)} \\ = {} & \sum_{n=1}^{N} \log p(\boldsymbol{\mathcal{D}}^n, \boldsymbol{\theta}|\mathbf{h}) + \sum_{n=1}^{N} \sum_{\mathbf{z}^n} q(\mathbf{Z}^n) \log \frac{p(\mathbf{Z}_n|\boldsymbol{\mathcal{D}}^n, \boldsymbol{\theta})}{q(\mathbf{Z}^n)} \\ = {} & \sum_{n=1}^{N} \log p(\boldsymbol{\mathcal{D}}^n, \boldsymbol{\theta}|\mathbf{h}) - \sum_{n=1}^{N} \text{KL}(q(\mathbf{Z}^n) \,||\, p(\mathbf{Z}^n|\boldsymbol{\mathcal{D}}^n, \boldsymbol{\theta})). \end{aligned}$$

This form of the lower bound implies that the optimal variational distribution $\hat{q}(\mathbf{Z}^n)$ minimizes the Kullback-Leibler divergence (KL) [78], meaning

$$\hat{q}(\mathbf{Z}^n) = p(\mathbf{Z}^n|\mathcal{D}^n, \boldsymbol{\theta}) = p(\mathbf{z}^n_{1:T}|\mathbf{x}^n_{1:T}, \mathbf{u}^n_{1:T}, \boldsymbol{\theta}). \qquad (5)$$

This update tightens the bound if the posterior model $\hat{q}(\mathbf{Z}^n)$ belongs to the same family of the true posterior [15]. Notice that the E-step is independent of the prior $p(\boldsymbol{\theta})$. Moreover, (5) indicates that the E-step reduces to the computation of the smoothed marginals $p(\mathbf{z}^n_t|\mathbf{x}^n_{1:T}, \mathbf{u}^n_{1:T}, \hat{\boldsymbol{\theta}})$ under the current parameter estimate $\hat{\boldsymbol{\theta}}$. Following [73] and [72], we derive a two-filter algorithm, which enables closed-form and exact inference by splitting the smoothed marginals into a forward and backward message[3]

$$\begin{aligned}\boldsymbol{\gamma}^n_t(k) &= p(\mathbf{z}^n_t = k|\mathbf{x}^n_{1:T}, \mathbf{u}^n_{1:T}) \\ &\propto p(\mathbf{z}^n_t = k|\mathbf{x}^n_{1:t}, \mathbf{u}^n_{1:t}) p(\mathbf{x}^n_{t+1:T}, \mathbf{u}^n_{t+1:T}|\mathbf{z}^n_t = k, \mathbf{x}^n_t, \mathbf{u}^n_t) \\ &= \alpha^n_t(k)\beta^n_t(k),\end{aligned}$$

where $\alpha^n_t(k) = p(\mathbf{z}^n_t = k|\mathbf{x}^n_{1:t}, \mathbf{u}^n_{1:t})$ is the message which computes the filtered marginals via a forward recursion

$$\begin{aligned}\alpha^n_t(k) &\propto p(\mathbf{x}^n_t|\mathbf{x}^n_{t-1}, \mathbf{u}^n_{t-1}, \mathbf{z}^n_t = k) p(\mathbf{u}^n_t|\mathbf{x}^n_t, \mathbf{z}^n_t = k) \\ &\quad \times \sum_{j=1}^{K} p(\mathbf{z}^n_t = k|\mathbf{z}^n_{t-1} = j, \mathbf{x}^n_{t-1}, \mathbf{u}^n_{t-1}) \alpha^n_{t-1}(j),\end{aligned}$$

and $\beta^n_t(k) = p(\mathbf{x}^n_{t+1:T}|\mathbf{z}^n_t = k, \mathbf{x}^n_t, \mathbf{u}^n_t)$ is the backward message that performs smoothing by computing the conditional likelihood of future evidence

$$\begin{aligned}\beta^n_t(k) &= \sum_{j=1}^{K} \beta^n_{t+1}(j) p(\mathbf{z}^n_{t+1} = j|\mathbf{z}^n_t = k, \mathbf{x}^n_t, \mathbf{u}^n_t) \\ &\quad \times p(\mathbf{x}^n_{t+1}|\mathbf{x}^n_t, \mathbf{u}^n_t, \mathbf{z}^n_{t+1} = j) p(\mathbf{u}^n_{t+1}|\mathbf{x}^n_{t+1}, \mathbf{z}^n_{t+1} = j).\end{aligned}$$

Additionally, by combining both forward and backward messages, we can compute the two-slice smoothed marginals $p(\mathbf{z}^n_t, \mathbf{z}^n_{t+1}|\mathbf{x}^n_{1:T}, \mathbf{u}^n_{1:T})$ which will be useful during the maximization and empirical Bayes steps

$$\begin{aligned}\xi^n_{t,t+1}(i, j) &= p(\mathbf{z}^n_t = i, \mathbf{z}^n_{t+1} = j|\mathbf{x}^n_{1:T}, \mathbf{u}^n_{1:T}) \\ &\propto p(\mathbf{x}^n_{t+1}|\mathbf{x}^n_t, \mathbf{u}^n_t, \mathbf{z}^n_{t+1} = j) p(\mathbf{u}^n_{t+1}|\mathbf{x}^n_{t+1}, \mathbf{z}^n_{t+1} = j) \\ &\quad \times \alpha^n_t(i) p(\mathbf{z}^n_{t+1} = j|\mathbf{z}^n_t = i, \mathbf{x}^n_t, \mathbf{u}^n_t) \beta^n_{t+1}(j).\end{aligned}$$

2) STOCHASTIC MAXIMIZATION STEP

After performing the E-step and computing the smoothed posteriors, we are able to evaluate the lower bound and maximize it with respect to $\boldsymbol{\theta}$ given $(\hat{q}(\mathbf{Z}^n), \hat{\mathbf{h}})$.

By plugging (3) and (5) into (4), leveraging conditional independence, and disregarding terms independent of $\boldsymbol{\theta}$, we arrive at the expected complete log-probability function $Q(\boldsymbol{\theta}, \boldsymbol{\gamma}, \boldsymbol{\xi}, \hat{\mathbf{h}})$

$$\begin{aligned}Q &= \sum_{n=1}^{N} \sum_{\mathbf{z}^n} \hat{q}(\mathbf{Z}^n) \log p(\mathcal{D}^n, \mathbf{Z}^n, \boldsymbol{\theta}|\hat{\mathbf{h}}) \\ &= \log p(\boldsymbol{\theta}|\hat{\mathbf{h}}) + \sum_{k=1}^{K} \sum_{n=1}^{N} \gamma^n_1 \left[\log \varphi_k + \log \mathrm{N}(\mathbf{x}^n_1|\boldsymbol{\mu}_k, \boldsymbol{\Omega}_k)\right] \\ &\quad + \sum_{k=1}^{K} \sum_{n=1}^{N} \sum_{t=2}^{T} \gamma^n_t \log \mathrm{N}(\mathbf{x}^n_t|\mathbf{A}_k \mathbf{x}^n_{t-1} + \mathbf{B}_k \mathbf{u}^n_{t-1} + \mathbf{c}_k, \boldsymbol{\Lambda}_k) \\ &\quad + \sum_{k=1}^{K} \sum_{n=1}^{N} \sum_{t=1}^{T} \gamma^n_t \log \mathrm{N}(\mathbf{u}^n_t|\mathbf{K}_k \phi(\mathbf{x}^n_{t-1}), \boldsymbol{\Delta}_k) \\ &\quad + \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{n=1}^{N} \sum_{t=2}^{T} \xi^n_{t-1,t} \log \chi_{ij}(\mathbf{x}^n_{t-1}, \mathbf{u}^n_{t-1}, \boldsymbol{\omega}_{ij}).\end{aligned}$$

The function Q is non-convex in $\boldsymbol{\omega}$ when a nonlinear logit link function $f(., \boldsymbol{\omega})$ is chosen as an embedding for the transition probability χ, see (1). In that case, stochastic optimization is recommended [79] as batched noisy gradient estimates allow the algorithm to escape shallow local minima and reduce the computational cost that comes with evaluating the gradients for all data instances.

Consequently, when implementing the M-step, we apply stochastic optimization on the transition parameters $\boldsymbol{\omega}$. We use a stochastic gradient ascent direction with an adaptive learning rate ε and batch size M [79]

$$\boldsymbol{\omega}^{(l+1)} = \boldsymbol{\omega}^l + \frac{\varepsilon}{M} \sum_{m=1}^{M} \nabla_{\boldsymbol{\omega}} Q^{(m)} \big|_{\boldsymbol{\omega}=\boldsymbol{\omega}^l},$$

$$\nabla_{\boldsymbol{\omega}} Q^{(m)} = \nabla_{\boldsymbol{\omega}} \Bigg[\log p(\boldsymbol{\omega}|\alpha) + \sum_{i=1}^{K} \sum_{j=1}^{K} \xi^{(m)} \log \chi_{ij}(\mathbf{x}^{(m)}, \mathbf{u}^{(m)}, \boldsymbol{\omega}_{ij}) \Bigg].$$

Algorithm 1: Expectation-Maximization for System Identification and Behavioral Cloning.

```
input: $\mathcal{D} = \{\mathbf{X}^n, \mathbf{U}^n\}_{n=1}^N, \mathbf{h}, K$
initialize: $\hat{\boldsymbol{\theta}} \sim p(\boldsymbol{\theta}|\mathbf{h}), \hat{\mathbf{h}} \leftarrow \mathbf{h}$
while $\log p(\mathcal{D}, \boldsymbol{\theta}|\mathbf{h})$ not converged do
    // Expectation step
    for $n \leftarrow 1$ to $N$ do
        $\alpha^n, \beta^n \leftarrow$ ForwardBackward$(\mathbf{X}^n, \mathbf{U}^n, \hat{\boldsymbol{\theta}})$
        $\gamma^n, \xi^n \leftarrow$ SmoothedPosteriors$(\alpha^n, \beta^n, \hat{\boldsymbol{\theta}})$
    end
    // Maximization step
    $\hat{\boldsymbol{\theta}} \leftarrow$ Maximize $Q(\boldsymbol{\theta}, \boldsymbol{\gamma}, \boldsymbol{\xi}, \hat{\mathbf{h}})$
    // Empirical Bayes
    $\hat{\mathbf{h}} \leftarrow \hat{\mathbf{h}} + \varrho \nabla_{\mathbf{h}} Q \mid_{\mathbf{h}=\hat{\mathbf{h}}}$
end
output: $\hat{\boldsymbol{\theta}}$
```

[3] We briefly drop the dependency on $\hat{\boldsymbol{\theta}}$ for an uncluttered notation while deriving the forward-backward recursions.

For the parameters with conjugate priors, we derive closed-form optimality conditions. Effectively, we derive the posterior distribution via Bayes' rule and take the mode of each posterior density for a MAP estimate update.

By considering only relevant terms, we write the MAP of the initial gating parameter $\boldsymbol{\varphi}$ as

$$\max_{\boldsymbol{\varphi}} \quad \log \mathrm{Dir}(\boldsymbol{\varphi}|\hat{\boldsymbol{\tau}}_0) + \sum_{k=1}^{K}\sum_{n=1}^{N} \gamma_1^n \log \varphi_k,$$

while the estimate of the initial state parameters $(\boldsymbol{\mu}_k, \boldsymbol{\Omega}_k)$ can be decoupled for each k as follows

$$\max_{(\boldsymbol{\mu},\boldsymbol{\Omega})_k} \quad \log \mathrm{NW}(\boldsymbol{\mu}_k, \boldsymbol{\Omega}_k|(\mathbf{0}, \hat{\kappa}_0, \hat{\boldsymbol{\Psi}}_0, \hat{\nu}_0)_k) + \sum_{n=1}^{N} \gamma_1^n \log \mathrm{N}(\mathbf{x}_1^n|\boldsymbol{\mu}_k, \boldsymbol{\Omega}_k).$$

Analogously, the MAP of the dynamics parameter $(\mathbf{A}_k, \mathbf{B}_k, \mathbf{c}_k, \boldsymbol{\Lambda}_k)$ is also decoupled to k optimizations

$$\max_{(\mathbf{A},\mathbf{B},\mathbf{c},\boldsymbol{\Lambda})_k} \quad \log \mathrm{MNW}(\mathbf{A}_k, \mathbf{B}_k, \mathbf{c}_k, \boldsymbol{\Lambda}_k|(\mathbf{0}, \hat{\mathbf{R}}_0, \hat{\boldsymbol{\Phi}}_0, \hat{\nu}_0)_k) + \sum_{n=1}^{N}\sum_{t=2}^{T} \gamma_t^n \log \mathrm{N}(\mathbf{x}_t^n|\mathbf{A}_k\mathbf{x}_{t-1}^n + \mathbf{B}_k\mathbf{u}_{t-1}^n + \mathbf{c}_k, \boldsymbol{\Lambda}_k),$$

and, finally, to learn closed-loop behavior, we can infer the controller parameters $(\mathbf{K}_k, \boldsymbol{\Delta}_k)$

$$\max_{(\mathbf{K},\boldsymbol{\Delta})_k} \quad \log \mathrm{MNW}(\mathbf{K}_k, \boldsymbol{\Delta}_k|(\mathbf{0}, \hat{\mathbf{S}}_0, \hat{\boldsymbol{\Gamma}}_0, \hat{\varepsilon}_0)_k) + \sum_{n=1}^{N}\sum_{t=1}^{T} \gamma_t^n \log \mathrm{N}(\mathbf{u}_t^n|\mathbf{K}_k\phi(\mathbf{x}_t^n), \boldsymbol{\Delta}_k).$$

Due to space constraints, we will refrain from stating the explicit solution for these optimization problems. Instead, we provide the general outline of how to compute these posteriors and their modes based on the unified notation for exponential family distributions in Appendix A and B.

3) APPROXIMATE EMPIRICAL BAYES

Inference techniques that leverage data-independent assumptions run the risk of prior miss-specification. In our MAP approach, the priors are weakly informative and carry little information. Their main purpose is to regularize greedy updates that might lead to premature convergence. However, when there is little data, the priors, especially those on the precision matrices, may dominate the posterior probability, leading to over-regularization and under-fitting of the objective. Empirical Bayes approaches remedy this issue by integrating out the parameters $\boldsymbol{\theta}$ and optimizing the marginal likelihood with respect to the hyperparameters $\mathbf{h}$ [77]. In our setting, marginalizing all hidden quantities does not admit a closed-form formula. An approximate approach to empirical Bayes is to interleave the E- and M-steps with hyperparameter updates that optimize the lower bound given an estimate of parameters $\hat{\boldsymbol{\theta}}$ and a step size ϱ

$$\mathbf{h}^{(l+1)} = \mathbf{h}^{(l)} + \varrho \, \nabla_{\mathbf{h}} Q \,|_{\mathbf{h}=\mathbf{h}^l} \,,$$

where the gradient of Q with respect to $\mathbf{h}$ reduces to

$$\nabla_{\mathbf{h}} Q = \nabla_{\mathbf{h}} \log p(\hat{\boldsymbol{\theta}}|\mathbf{h}).$$

VII. REINFORCEMENT LEARNING VIA HYBRID MODELS

The last sections focused on the system modeling aspect and how to use hybrid surrogate models to approximate nonlinear dynamics. Now we turn our attention to the problem of using these models to synthesize structured controllers for general nonlinear dynamical systems. One possible approach is to use the learned hybrid models and apply the classical hybrid control methods, which we have reviewed Section II. However, as discussed earlier, these methods suffer from several drawbacks. On the one hand, they rely a polyhedral partitioning of the space. This limitation is severe because it often leads to an explosion in the number of partitions. On the other hand, these methods are often focused on computationally expensive trajectory-centric model predictive control. This class of controllers is disadvantageous in applications that require fast reactive feedback signals with broad coverage over the state-action space.

In this section, we address these points and present an infinite horizon stochastic optimization technique that incorporates the structure of hybrid models. This approach can deal with rARHMMs with arbitrary non-polyhedral partitioning and synthesizes stationary piecewise polynomial controllers. Our algorithm extends the step-based formulation of relative entropy policy search (REPS) [80], [81], [82] by explicitly accounting for the discrete-continuous mixture state variables $(\mathbf{x}, \mathbf{z})$. Our approach, hybrid REPS (Hb-REPS), leverages the state-action-dependent nonlinear switches $p(\mathbf{z}'|\mathbf{z}, \mathbf{x}, \mathbf{u})$ as a task-independent upper-level coordinator to a mixture of K lower-level policies $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$. While the proposed approach shares many features with [62], our formulation relies on a state-abstraction representation of hybrid models and embeds the hierarchical model structure into the optimization problem in order to learn a hierarchy over the global value function. In contrast, [62] operates in the framework of semi-Markov decision processes and optimizes a mixture over termination and feedback policies without considering the existence of a hierarchical structure in the space of dynamics and value functions. For more details on differences between state- and time-abstractions, refer to Section II.

A. INFINITE-HORIZON STOCHASTIC OPTIMAL CONTROL

In the REPS framework, an optimal control problem is presented as an iterative trust-region optimization for a discounted average-reward objective under a stationary state-action distribution $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z})$, (6a). The trust-region is formulated as a KL [78], (6c). Its purpose is to regularize the search direction and limit information loss between iterations. The REPS formulation explicitly incorporates a dynamics consistency constraint, (6b), that describes the evolution of the stochastic state of the system. The following optimization problem is solved during a single iteration of hybrid REPS

$$\max_{\pi,\mu} \quad J = \sum_{\mathbf{z}} \iint r(\mathbf{x}, \mathbf{u})\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}, \tag{6a}$$

$$\text{s.t.} \quad \mu(\mathbf{x}', \mathbf{z}') = (1 - \vartheta)\mu_1(\mathbf{x}', \mathbf{z}') \tag{6b}$$

$$+ \vartheta \sum_{\mathbf{z}} \iint \pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z})p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{u}\mathrm{d}\mathbf{x},$$

$$\mathrm{KL}(\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z}) \,||\, q(\mathbf{x}, \mathbf{u}, \mathbf{z})) \leq \epsilon, \tag{6c}$$

$$\sum_{\mathbf{z}} \iint \pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} = 1, \tag{6d}$$

where $\mu(\mathbf{x}, \mathbf{z})$ is the stationary mixture distribution, $q(\mathbf{x}, \mathbf{u}, \mathbf{z})$ is the trust-region reference distribution, and the constraint in (6d) guarantees the normalization of the state-action distribution. The factor $1 - \vartheta$, $\vartheta \in [0, 1)$, represents the probability of an infinite process to reset to an initial distribution $\mu_1(\mathbf{x}, \mathbf{z})$. The notion of resetting is necessary to ensure ergodicity of the closed-loop Markov process and allows the interpretation of ϑ as a discount factor and regularization of the MDP [82], [83].

B. OPTIMALITY CONDITIONS AND DUAL OPTIMIZATION

To solve the trust-region optimization in (6a)–(6d), we start by constructing the Lagrangian of the primal [84]

$$\begin{aligned} \mathcal{L} = &\sum_{\mathbf{z}} \iint r(\mathbf{x}, \mathbf{u})p(\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} \\ &+ \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}')\Bigg[- \int p(\mathbf{x}', \mathbf{z}', \mathbf{u}')\mathrm{d}\mathbf{u}' \\ &+ (1 - \vartheta) \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z})\mu_1(\mathbf{x}', \mathbf{z}')\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} \\ &+ \vartheta \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z})p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}\Bigg]\mathrm{d}\mathbf{x}' \\ &+ \lambda\left[1 - \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{u}\mathrm{d}\mathbf{x}\right] \\ &+ \eta\left[\epsilon - \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{u}, \mathbf{z})}{q(\mathbf{x}, \mathbf{u}, \mathbf{z})}\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}\right], \end{aligned}$$

where we use $p(\mathbf{x}, \mathbf{u}, \mathbf{z}) = \mu(\mathbf{x}, \mathbf{z})\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$ for convenience and leverage the following identities

$$\mu(\mathbf{x}, \mathbf{z}) = \int p(\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{u},$$

$$\begin{aligned} \mu_1(\mathbf{x}', \mathbf{z}') &= \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z})p_1(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} \\ &= \sum_{\mathbf{z}} \iint p(\mathbf{x}, \mathbf{u}, \mathbf{z})\mu_1(\mathbf{x}', \mathbf{z}')\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}. \end{aligned}$$

The second identity implies that the resetting is only dependent on the parameter ϑ and independent of the state and actions $(\mathbf{x}, \mathbf{u}, \mathbf{z})$ to satisfy the ergodicity property.

The parameters η and λ are the Lagrangian variables associated with (6c) and (6d), while $V(\mathbf{x}, \mathbf{z})$ is the state-value function, which appears naturally in REPS as the Lagrangian function associated with (6b). Next, we take the partial derivative of $\mathcal{L}$ with respect to $p(\mathbf{x}, \mathbf{u}, \mathbf{z})$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial p} = &\; r(\mathbf{x}, \mathbf{u}) - \lambda + (1 - \vartheta) \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}')\mu_1(\mathbf{x}', \mathbf{z}')\mathrm{d}\mathbf{x}' \\ &+ \vartheta \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}')p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}' \\ &- V(\mathbf{x}, \mathbf{z}) - \eta \log \frac{p^*(\mathbf{x}, \mathbf{u}, \mathbf{z})}{q(\mathbf{x}, \mathbf{u}, \mathbf{z})} - \eta, \end{aligned}$$

and set it to zero to get the optimal point

$$p^*(\mathbf{x}, \mathbf{u}, \mathbf{z}) = q(\mathbf{x}, \mathbf{u}, \mathbf{z}) \exp\left[\frac{A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V) - \lambda - \eta}{\eta}\right], \tag{7}$$

where $A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)$ is the advantage function given as

$$\begin{aligned} A(.) = &\; r(\mathbf{x}, \mathbf{u}) + (1 - \vartheta) \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}')\mu_1(\mathbf{x}', \mathbf{z}')\mathrm{d}\mathbf{x}' \\ &+ \vartheta \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}')p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}' - V(\mathbf{x}, \mathbf{z}). \end{aligned} \tag{8}$$

The optimal point $p^*(\mathbf{x}, \mathbf{u}, \mathbf{z}) = \mu^*(\mathbf{x}, \mathbf{z})\pi^*(\mathbf{u}|\mathbf{x}, \mathbf{z})$ has to satisfy the constraint in (6d), which in turn enables us to find the Lagrangian variable λ^*

$$\begin{aligned} 1 &= \sum_{\mathbf{z}} \iint p^*(\mathbf{x}, \mathbf{u}, \mathbf{z})\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} \\ 1 &= \sum_{\mathbf{z}} \iint q(\mathbf{x}, \mathbf{u}, \mathbf{z}) \exp\left[\frac{A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V) - \lambda^* - \eta}{\eta}\right]\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u} \\ \lambda^* &= -\eta + \eta \log \sum_{\mathbf{z}} \iint q(\mathbf{x}, \mathbf{u}, \mathbf{z}) \exp\left[\frac{A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)}{\eta}\right]\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}. \end{aligned}$$

By substituting λ^* back into $p^*(\mathbf{x}, \mathbf{u}, \mathbf{z})$ in (7), we retrieve the normalized density softmax form

$$p^*(\mathbf{x}, \mathbf{u}, \mathbf{z}) = \frac{q(\mathbf{x}, \mathbf{u}, \mathbf{z}) \exp\left[A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)/\eta\right]}{\sum_{\mathbf{z}} \iint q(\mathbf{x}, \mathbf{u}, \mathbf{z}) \exp\left[A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)/\eta\right] \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u}}.$$

Now by plugging the solutions p^* and λ^* back into the Lagrangian, we arrive at the dual function $\mathcal{G}$ as a function of the remaining Lagrangian variables η and V

$$\mathcal{G} = \eta\epsilon + \eta \log \sum_{\mathbf{z}} \iint q(\mathbf{x}, \mathbf{u}, \mathbf{z})\exp\left[\frac{A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)}{\eta}\right]\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{u},$$

where $q(\mathbf{x}, \mathbf{u}, \mathbf{z}) = q(\mathbf{x}, \mathbf{u})q(\mathbf{z}|\mathbf{x}, \mathbf{u})$ and $q(\mathbf{z}|\mathbf{x}, \mathbf{u})$ is the posterior over $\mathbf{z}$ given $\mathbf{x}$ and $\mathbf{u}$. In Section VI, we derived a forward-backward algorithm for inferring this density, allowing us to compute the expectation over $\mathbf{z}$. The expectations over $\mathbf{x}$ and $\mathbf{u}$ are analytically intractable. Therefore, we approximate them given samples from the reference distribution

$q(\mathbf{x}, \mathbf{u})$. The multipliers η and V are then obtained by numerically minimizing the dual $\mathcal{G}(\eta, V)$

$$\arg\min_{\eta, V} \quad \mathcal{G}(\eta, V), \qquad \text{s.t.} \quad \eta \geq 0,$$

that acts as the upper bound on the primal objective.

C. MODELING DYNAMICS AND STATE-VALUE FUNCTION

Up to this point, the derivation of Hb-REPS has been generic. We have made no assumptions on initial distributions $\mu_1(\mathbf{x}, \mathbf{z})$, the dynamics $p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})$, or the value function $V(\mathbf{x}, \mathbf{z})$. Now, we introduce the piecewise affine-Gaussian dynamics and logistic switching described in Section V and assume these representations to be available in parametric form as a result of a separate learning process. Furthermore, we model the state-value function with piecewise n-th degree polynomial functions $V(\mathbf{x}, \mathbf{z}) = \boldsymbol{\tau}_{\mathbf{z}}^{\top} \psi_{\mathbf{z}}(\mathbf{x})$, where $\psi_{\mathbf{z}}(\mathbf{x})$ is the state-feature vector which contains polynomial features of the state $\mathbf{x}$, and $\boldsymbol{\tau}_{\mathbf{z}}$ is the parameter vector assigned to the different regions.

Under these assumptions, we can use the available joint density $\mu_1(\mathbf{x}, \mathbf{z})$ and $p(\mathbf{x}'|\mathbf{x}, \mathbf{u}, \mathbf{z})$ to compute the necessary expectations in (8)

$$\mathbb{E}_{\mathbf{x}_1, \mathbf{z}_1}\left[V(\mathbf{x}', \mathbf{z}')\right] = \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}') \mu_1(\mathbf{x}', \mathbf{z}') \mathrm{d}\mathbf{x}',$$

$$\mathbb{E}_{\mathbf{x}', \mathbf{z}'}\left[V(\mathbf{x}', \mathbf{z}')\right] = \sum_{\mathbf{z}'} \int V(\mathbf{x}', \mathbf{z}') p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z}) \mathrm{d}\mathbf{x}'.$$

This computation allows our approach to capture the stochasticity of the dynamics and delivers an estimate of the advantage function $A(\mathbf{x}, \mathbf{u}, \mathbf{z}, V)$ instead of the temporal difference (TD) error in the general REPS framework [80]. Ultimately, this leads to better estimates of the expected discounted future returns captured by V.

Practically, these integrals can be either naively approximated by applying Monte Carlo integration [85] or, more efficiently, by recognizing the structure of the integrand $V(\mathbf{x}', \mathbf{z}')$ and using Gauss-Hermite cubature rules for exact integration over polynomial functions [86].

D. MAXIMUM-A-POSTERIORI POLICY IMPROVEMENT

A significant advantage of our model-based reinforcement learning approach becomes evident when considering the policy improvement step in the REPS framework. The policy update is incorporated into the optimality condition of the stationary state-action distribution $p(\mathbf{x}, \mathbf{u}, \mathbf{z}) = \pi(\mathbf{u}|\mathbf{x}, \mathbf{z})\mu(\mathbf{x}, \mathbf{z})$ in (7). As a consequence, updating the mixture policies $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$ requires the computation of state probabilities $\mu(\mathbf{x}, \mathbf{z})$, which in turn require knowledge of the dynamics model. This issue is circumvented in other model-free realizations of REPS by introducing a crude approximation to enable a model-free policy update nonetheless. For example, in [87], the authors postulate that the new state distribution $\mu(\mathbf{x}, \mathbf{z})$ is usually *close enough* to the old distribution $q(\mathbf{x}, \mathbf{z})$, thus allowing the ratio $q(\mathbf{x}, \mathbf{z})/\mu(\mathbf{x}, \mathbf{z})$ to be ignored when a weighted maximum-likelihood fit of the actions $\mathbf{u}$ is performed to update π.

While the assumption of *closeness* may be practical and empowers many successful variants of REPS, it is crucial to be aware of its technical ramifications, as it undermines the primary motivation of a relative entropy bound on the state-action distribution in (6c). This aspect is unique in the REPS framework when compared to other state-of-the-art approximate policy iteration algorithms [7], [9], [88], that optimize a similar objective, albeit with a relaxed bound that only limits the change of the action distribution π.

In contrast, our algorithm uses the surrogate hybrid dynamics and updates the policy $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$ with the correct weighting. The optimality condition in (7) is satisfied by computing a weighted maximum a posteriori estimate of the parameters $\boldsymbol{\theta}$ of the state-action distribution $p(\mathbf{x}, \mathbf{u}, \mathbf{z}|\boldsymbol{\theta})$, thus implicitly updating $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$. This procedure is equivalent to a modified Baum-Welch expectation-maximization algorithm that learns the parameters of a closed-loop rARHMM, as derived in Section VI. The difference is that the EM objective in (2) has to be augmented with the importance weights from (7)

$$\arg\max_{\boldsymbol{\theta}} \log \prod_{n=1}^{N} \sum_{\mathbf{z}^n} \mathbf{w}^n p(\mathbf{X}^n, \mathbf{U}^n, \mathbf{Z}^n|\boldsymbol{\theta}) p(\boldsymbol{\theta}),$$

where $(\mathbf{X}^n, \mathbf{U}^n)$ are state-action trajectories collected via interaction with the environment and $\mathbf{w}^n$ are the associated weights resulting from (7)

$$\mathbf{w}^n = \exp\left[A(\mathbf{X}^n, \mathbf{U}^n, \mathbf{Z}^n, V)/\eta\right].$$

This augmentation leads to weighted M- and EB-steps while the E-step is not altered.

Note that during the policy improvement step, we can either assume an a priori estimate of the open-loop dynamics $p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})$ and only update the control parameters corresponding to the conditional $\pi(\mathbf{u}|\mathbf{x}, \mathbf{z})$, or we can iteratively update $p(\mathbf{x}', \mathbf{z}'|\mathbf{x}, \mathbf{u}, \mathbf{z})$ as more data becomes available. A compact sketch of the overall optimization process is available in Algorithm 2.

VIII. EMPIRICAL EVALUATION

In this section, we benchmark different aspects of our approach to system modeling and control synthesis via hybrid models. In the following,

- we assess the predictive performance of rARHMMs at open-loop system identification of nonlinear systems and validate our choice of hybrid surrogate models as a suitable representation.
- we test the ability of rARHMMs to approximate and decompose expert nonlinear controllers in a closed-loop behavioral cloning scenario.
- we deploy rARHMMs in the proposed hierarchical RL algorithm Hb-REPS to solve the infinite horizon stochastic control objective and optimize piecewise polynomial controllers and value functions.

Algorithm 2: Model-Based Relative Entropy Policy Search via Hybrid Models.

```
input: $p(\mathbf{x}', \mathbf{z}' | \mathbf{x}, \mathbf{u}, \mathbf{z})$
initialize: $q(\mathbf{u}|\mathbf{x}, \mathbf{z}), \boldsymbol{\tau}_{\mathbf{z}}, \eta$
while $J$ not converged do
    // Sample interactions
    $(\mathbf{X}, \mathbf{U}) \leftarrow$ Environment$(q)$
    // Policy evaluation
    $\eta^*, \boldsymbol{\tau}^*_{\mathbf{z}}, \mathbf{w}^* \leftarrow$ Minimize $\mathcal{G}(\mathbf{X}, \mathbf{U}, p, \eta, \boldsymbol{\tau}_{\mathbf{z}}, \epsilon)$
    // Policy improvement
    $\pi^* \leftarrow$ BaumWelch$(\mathbf{X}, \mathbf{U}, p, \mathbf{w}^*)$
    // Update parameters
    $q, \boldsymbol{\tau}_{\mathbf{z}}, \eta \leftarrow \pi^*, \boldsymbol{\tau}^*_{\mathbf{z}}, \eta^*$
end
output: $\pi^*(\mathbf{u}|\mathbf{x}, \mathbf{z})$
```

A. PIECEWISE OPEN-LOOP SYSTEM IDENTIFICATION

We start by empirically benchmarking the open-loop learned rARHMMs and their ability to approximate nonlinear dynamics. We compare to popular black-box models in a *long-horizon* and *limited-data* setting.

This evaluation focuses on rARHMMs with exogenous inputs. We learn the dynamics of three simulated deterministic systems; a bouncing ball, an actuation-constrained pendulum, and a cart-pole system. We compare the predictive time forecasting accuracy of rARHMMs to classical non-recurrent autoregressive hidden Markov models (ARHMMs) [4] [16], feed-forward neural nets (FNNs), Gaussian processs (GPs) ,[5] long-short-term memory networks (LSTMs) [89], and recurrent neural networks (RNNs). During the evaluation, we collected segregated training and test datasets. The training dataset is randomly split into 24 groups, each used to train different instances of all models. These instances are then tested on the test dataset. During evaluation, we sweep the test trajectories stepwise and predict the given horizon.

All neural models have two hidden layers, which we test for different layer sizes, $S \in \{16, 32, 64, 128, 256, 512\}$ for FNNs, $S \in \{16, 32, 64, 128, 256\}$ for RNNs, and $S \in \{16, 32, 64, 128\}$ for LSTMs. In the case of (r)ARHMMs, we vary the number of components K, dependent on the task. As a metric, we evaluate the forecast NMSE for a range of horizons averaged over the 24 data splits. We report the result corresponding to the best choice of S and K. Finally, in Table 1, we qualitatively compare the complexity of all representations in terms of their total number of parameters.

1) BOUNCING BALL

This example is a canonical instance of a dual-regime hybrid system due to the hard velocity switch at the moment of impact. We simulate the dynamics with a frequency of 20 Hz and collect 25 training trajectories with different initial heights and velocities, each 30 s long. This dataset is split 24 folds with ten trajectories, 10×150 data points, in each subset. The test dataset consists of 5 trajectories, each 30 s long. We evaluate the NMSE for horizons $h = \{1, 20, 40, 60, 80\}$ time steps. We did not evaluate a GP model in this setting due to the long prediction horizons that led to a very high computational burden. The (r)ARHMMs are tested for $K = 2$. The logit link function of an rARHMM is parameterized by a neural net with one hidden layer containing 16 neurons. The results in Fig. 3 show that the rARHMM approximates the dynamics well and outperforms both ARHMMs and the neural models.

2) PENDULUM AND CART-POLE

These systems are classical benchmarks from the nonlinear control literature. Here we consider two different observation types, one in the wrapped polar space, where the angle space $\theta \in [-\pi, \pi]$ includes a sharp discontinuity, and a second model with smooth observations parameterized with the Cartesian trigonometric features $\{\cos(\theta), \sin(\theta)\}$. Both dynamics are simulated with a frequency of 100 Hz. We collect 25 training trajectories starting from different initial conditions and apply random uniform explorative actions. Each trajectory is 2.5 s long. The 24 splits consist of 10 trajectories each, 10×250 data points. The test dataset consists of 5 trajectories, each 2.5 s long. Forecasting accuracy is evaluated for horizons $h = \{1, 5, 10, 15, 20, 25\}$. The (r)ARHMMs are tested for $K = \{3, 5, 7, 9\}$ on both tasks. The logit link function of the rARHMM is parameterized by a neural net with one hidden layer containing 24 neurons. As shown in Fig. 3, the forecast evaluation provides empirical evidence for the representation power of rARHMMs in both smooth and discontinuous state spaces. FNNs and GPs perform well in the smooth Cartesian observation space and struggle in the discontinuous space, similar to RNNs and LSTMs. Moreover, in Table 1, it is clear that rARHMMs reach comparable predictive performance to state-of-the-art models with a fraction of the parametric complexity.

B. PIECEWISE CLOSED-LOOP BEHAVIORAL CLONING

We want to analyze the closed-loop rARHMM with endogenous inputs as a behavioral cloning framework. The task is to reproduce the closed-loop behavior of expert policies on challenging nonlinear systems. For this purpose, we train two feedback experts on the pendulum and cart-pole. The two environments are simulated at 50 Hz and are influenced by static Gaussian noise with a standard deviation $\sigma = 1 \times 10^{-2}$. The experts are two-layer neural network policies with 4545 parameters (pendulum) and 17537 parameters (cart-pole), optimized with the SAC algorithm [9].

[4] ARHMMs closely resemble rARHMMs. However, the transitions probability in 1 does not depend on the continuous state or action.

[5] With an RBF kernel and hyperparameter optimization.

TABLE 1. System identification: Qualitative comparison of model complexity for the best-performing representations in Fig. 3. The values reflect the total number of parameters of each model. The values in parentheses represent the hidden layer sizes S of the neural models and the number of discrete components K for the (r)ARHMM, respectively.

	Bouncing Ball	Pendulum (Polar)	Pendulum (Cartesian)	Cart-Pole (Polar)	Cart-Pole (Cartesian)
ARHMM	22 (2)	180 (9)	130 (5)	287 (7)	275 (5)
rARHMM	86 (2)	468 (9)	582 (9)	575 (7)	711 (7)
FNN	1250 (32)	546 (64)	1315 (32)	1380 (32)	1445 (32)
RNN	12866 (64)	50306 (128)	3427 (32)	50820 (128)	51077 (128)
LSTM	200450 (128)	51074 (64)	51395 (64)	201732 (128)	202373 (128)

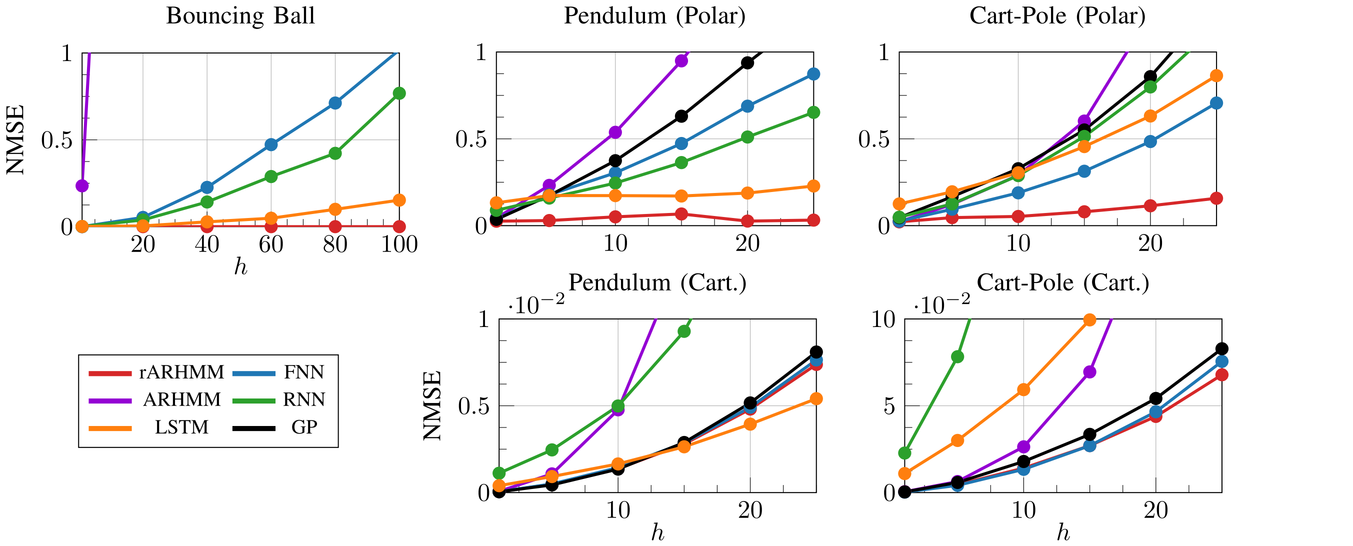

FIGURE 3. System identification: The h-step normalized mean square error (NMSE) of rARHMMs compared to other models. Evaluation is averaged over 24 data splits. Benchmarking on three dynamical systems, a bouncing ball, a pendulum, and a cart-pole. rARHMMs exhibit the most consistent approximation capabilities. Figure reproduced from [18].

For cloning, we construct two 5-regime rARHMMs with piecewise polynomial policies of the third order. The hybrid controllers have a total number of parameters of 100 (pendulum) and 280 (cart-pole). Learning is realized on a dataset of 25 expert trajectories, each 5 s long, for each environment and using the EM technique from Section VI. The decomposed controllers complete the task of swinging up and stabilizing both systems with over 95% success rate. Fig. 4 shows the phase portraits of the unforced dynamics and closed-loop control identified during cloning. Fig. 5 depicts sampled trajectories of the hybrid policies highlighting the switching behavior.

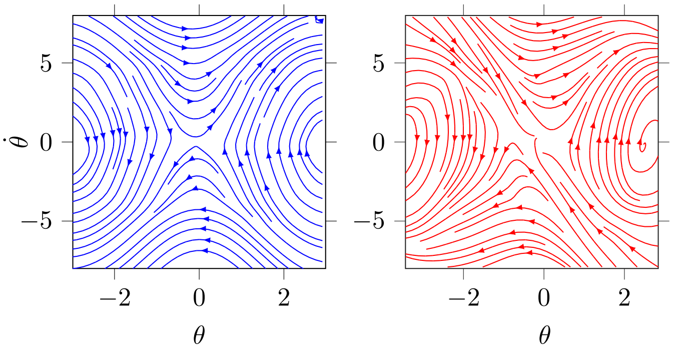

FIGURE 4. Behavioral cloning: The identified unforced dynamics are on the left (blue). The learned model qualitatively captures the phase portrait. On the right (red) are the closed-loop dynamics. The learned stationary hybrid policy with five regions successfully imitates a global nonlinear soft actor-critic (SAC) controller to stabilize the system around the origin. Figure reproduced from [18].

C. NONLINEAR CONTROL SYNTHESIS VIA HYBRID MODELS

Finally, we evaluate the performance of the hybrid policy search algorithm Hb-REPS on two nonlinear stochastic dynamical systems: an actuation-constrained pendulum swing-up and a cart-pole stabilization task. We make no claim to the absolute *sample* efficiency of our RL approach when compared to state-of-the-art RL algorithms. Instead, we aim to provide empirical support for the premise that structured representations that rely on compact piecewise parametric forms can provide an alternative to black-box function approximators with comparable overall performance.

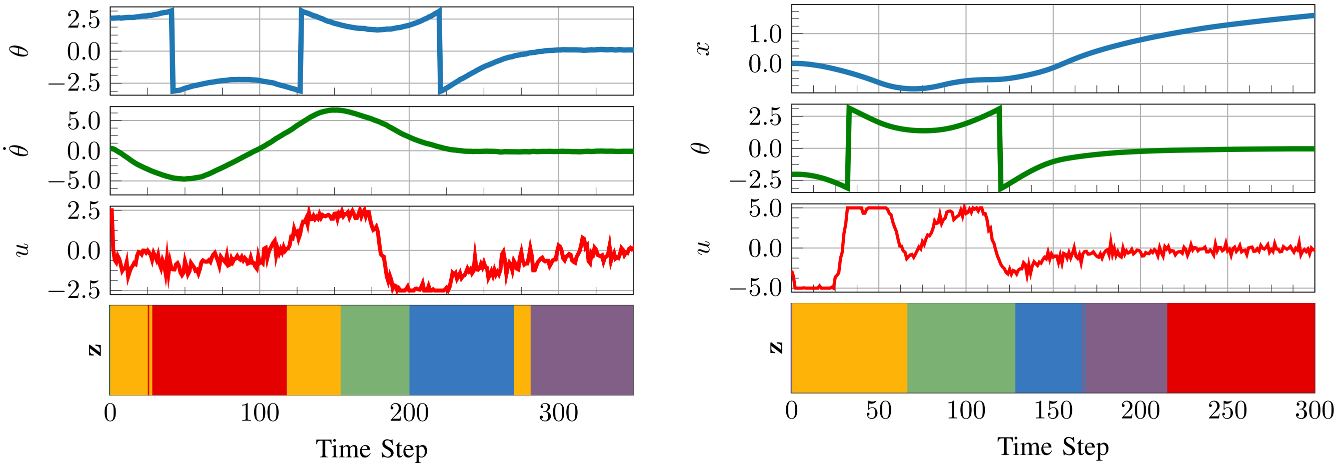

FIGURE 5. **Behavioral cloning: Sample trajectories from the learned hybrid policies on the pendulum (left) and cart-pole (right). Both hybrid controllers are able to consistently solve the tasks while relying on simple local representations of the feedback controllers. The colors indicate the active dynamics and control regimes over time. Figure reproduced from [18].**

Therefore, we compare the performance of Hb-REPS to two baselines. The first is a *vanilla* version of REPS that does not maintain any hierarchical structure and uses nonlinear function approximators with random Fourier features (RFFs) [90] to represent both policy and value function. The second baseline assumes a hierarchical policy structure and a nonlinear value function with Fourier features. This baseline is somewhat akin to what is implemented in [62], albeit with a hierarchy based on state abstraction rather than time. We will refer to this algorithm as hierarchical REPS (Hi-REPS). We assume an offline learning phase in which the hybrid models are learned from pre-collected data.

1) PENDULUM SWING-UP

In this experiment, the actuation-constrained pendulum is simulated at 50 Hz and perturbed by Gaussian noise with a standard deviation $\sigma = 1 \times 10^{-2}$. The REPS agent relies on a policy and value function with 50 and 75 Fourier basis functions, respectively. Hi-REPS assumes a similar form of the value function but with a piecewise third-order polynomial policy over five partitions. Hb-REPS represents both policy and value function with piecewise third-order polynomials over five partitions. Empirical results in Fig. 7 (left half) feature comparable learning performance of all algorithms over ten random seeds. Every iteration involves 5000 interactions with the environment. We provide a phase portrait of the closed-loop behavior for a qualitative assessment of the final stationary hybrid policy.

2) CART-POLE STABILIZATION

This evaluation features a cart-pole constrained by an elastic wall modeled by a spring. The dynamics are linearized around the upright position. The environment is simulated at 100 Hz and perturbed by Gaussian noise with a standard deviation $\sigma = 1 \times 10^{-4}$. The REPS policy and value function both use 25 random Fourier basis functions. Hi-REPS adopts the same value function structure with a two-partition piecewise affine policy. Hb-REPS also assumes a two-partition piecewise affine policy and second-order value function. Fig. 7 (right half) depicts comparable learning performance over ten random seeds. Every iteration involves 2500 interactions with the environment.

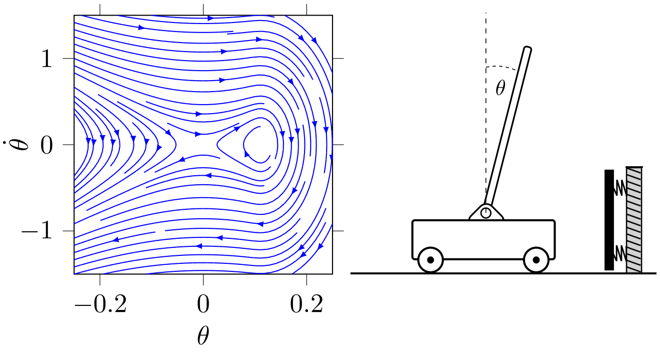

FIGURE 6. **A cart-pole system with an elastic wall constraint: The cart-pole dynamics are linearized around the upright position, and a spring system models the wall. The phase portrait of the unforced angular dynamics is depicted on the left (blue). The aim is to stabilize the pole.**

IX. DISCUSSION

We presented a general framework for data-driven nonlinear system identification and stochastic control based on the structured representation of hybrid surrogate models. To introduce the hybrid structure, we proposed replacing commonly used piecewise affine auto-regressive models with probabilistic hybrid dynamic Bayesian networks, as they offer a range of advantages in data-driven scenarios. Furthermore, we presented a novel reinforcement learning algorithm that leverages the learned hybrid models to synthesize piecewise polynomial feedback controllers for nonlinear systems.

Our hybrid-model-infused reinforcement learning approach is able to reach comparable performance on control tasks with

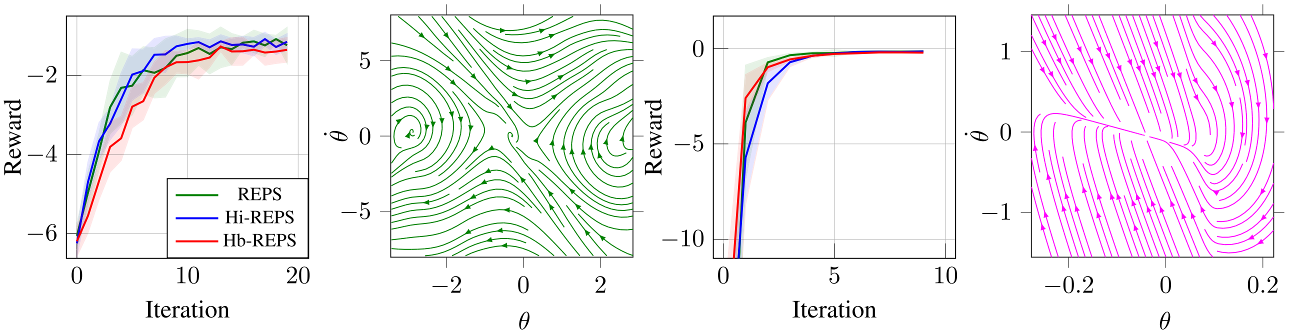

FIGURE 7. Reinforcement learning: REPS, Hi-REPS, and Hb-REPS evaluated on the pendulum swing-up (left) and cart-pole stabilization (right) tasks. By inspecting the learning curves, mean reward with two standard deviations, we conclude that all algorithms perform equally well in terms of the transient and final performance. However, Hb-REPS relies on simpler piecewise polynomial models of the policy and value function, while Hi-REPS and REPS use nonlinear Fourier basis functions. The phase portraits depict the closed-loop behavior achieved by Hb-REPS.

a significant reduction in the complexity of functional representation. Furthermore, in contrast to deterministic hybrid model predictive control, our approach solves the infinite-horizon stochastic optimal control problem by approximating the global value function and lifts the requirement for polyhedral partitioning.

While initial empirical results are encouraging, the application of this work is limited to low-dimensional dynamical systems. Although a viable alternative to expensive mixed-integer optimization, the inference techniques used in this article still present a bottleneck in the face of scalability to higher dimensions. While our MAP approach significantly improves the quality of expectation-maximization solutions, it nevertheless struggles in more challenging environments.

A possible course of action is to investigate Bayesian non-parametric extensions of hybrid dynamic Bayesian networks based on non-conjugate variational inference. Fully Bayesian methods tend to improve learning in large structured models significantly. Another potential avenue of research is to improve the hybrid reinforcement learning framework by considering the control-as-inference paradigm. Such approaches may offer ways of integrating the Bayesian structure of the models into the control optimization and constructing an uncertainty-aware approach that is better equipped to deal with the exploration-exploitation dilemma.

APPENDIX A EXPONENTIAL FAMILY

Our work focuses on random variables with probability density functions belonging to the exponential family. The unified minimal parameterization of this class of distributions lends itself for convenient and efficient posterior computation when paired with conjugate priors.

We assume the natural form for a probability density of a random variable $\mathbf{x}$

$$f(\mathbf{x}|\boldsymbol{\eta}) = h(\mathbf{x}) \exp\left[\boldsymbol{\eta} \cdot \mathbf{t}(\mathbf{x}) - a(\boldsymbol{\eta})\right],$$

where $h(\mathbf{x})$ is the base measure, $\boldsymbol{\eta}$ are the natural parameters, $\mathbf{t}(\mathbf{x})$ are the sufficient statistics and $a(\boldsymbol{\eta})$ is the log-partition function, or log-normalizer. Following the same notation, a conjugate prior $g(\boldsymbol{\eta}|\boldsymbol{\lambda})$ to the likelihood $f(\mathbf{x}|\boldsymbol{\eta})$ has the form

$$g(\boldsymbol{\eta}|\boldsymbol{\lambda}) = h(\boldsymbol{\eta}) \exp\left[\boldsymbol{\lambda} \cdot \mathbf{t}(\boldsymbol{\eta}) - a(\boldsymbol{\lambda})\right],$$

with prior sufficient statistics $\mathbf{t}(\boldsymbol{\eta}) = [\boldsymbol{\eta},\ -a(\boldsymbol{\eta})]^\top$ and hyperparameters $\boldsymbol{\lambda} = [\boldsymbol{\alpha},\ \boldsymbol{\beta}]^\top$. By applying Bayes' rule, we can directly infer the posterior $q(\boldsymbol{\eta}|\mathbf{x})$

$$\begin{aligned} q(\boldsymbol{\eta}|\mathbf{x}) &\propto f(\mathbf{x}|\boldsymbol{\eta}) g(\boldsymbol{\eta}|\boldsymbol{\lambda}) \\ &\propto \exp\left[\boldsymbol{\rho}(\mathbf{x}, \boldsymbol{\lambda}) \cdot \mathbf{t}(\boldsymbol{\eta}) - a(\boldsymbol{\rho})\right], \end{aligned}$$

where the posterior natural parameters $\boldsymbol{\rho}(\mathbf{x}, \boldsymbol{\lambda})$ are a function of the likelihood sufficient statistics $\mathbf{t}(\mathbf{x})$ and prior hyperparameters $[\boldsymbol{\alpha},\ \boldsymbol{\beta}]$

$$\boldsymbol{\rho}(\mathbf{x}, \boldsymbol{\lambda}) = [\boldsymbol{\alpha} + \mathbf{t}(\mathbf{x}),\ \boldsymbol{\beta} + \mathbf{1}]^\top .$$

The structure of the resulting posterior reveals a simple recipe for data-driven inference. By moving into the natural space, the posterior parameters are computed by combining the prior hyperparameters with the likelihood sufficient statistics and log-partition function. By definition, every exponential family distribution has a minimal natural parameterization that leads to a unique decomposition of these quantities [91].

APPENDIX B CONJUGATE POSTERIORS

We present an outline of all M-step updates. We use an adapted form of the exponential natural parameterization, as it offers a clear methodology for deriving and implementing such updates for all relevant distributions.

A. CATEGORICAL WITH DIRICHLET PRIOR

A weighted categorical likelihood over a one-hot random variable $\mathbf{z}$ with size K has the form

$$p(\mathbf{Z}|\boldsymbol{\varphi}) = \prod_{n=1}^{N} \mathrm{Cat}(\mathbf{z}_n|\boldsymbol{\varphi})^{w_n}$$

$$\propto \exp\left\{\begin{bmatrix}\log\varphi 1\\ \vdots\\ \log\varphi K\end{bmatrix}\cdot\begin{bmatrix}\sum_{n=1}^{N} w_{n,1}\\ \vdots\\ \sum_{n=1}^{N} w_{n,K}\end{bmatrix}\right\},$$

where w_{nk} are the importance weights for each category K. The conjugate prior is a Dirichlet $p(\boldsymbol{\varphi})$ distribution

$$\begin{aligned}p(\boldsymbol{\varphi}) &= \mathrm{Dir}(\boldsymbol{\varphi}|\boldsymbol{\tau}_0)\\ &\propto \exp\left\{\begin{bmatrix}\tau_{0,1}-1\\ \vdots\\ \tau_{0,K}-1\end{bmatrix}\cdot\begin{bmatrix}\log\varphi_1\\ \vdots\\ \log\varphi_K\end{bmatrix}\right\},\end{aligned}$$

The posterior $q(\boldsymbol{\varphi})$ is likewise a Dirichlet distribution

$$\begin{aligned}q(\boldsymbol{\varphi}) &= \mathrm{Dir}(\boldsymbol{\varphi}|\boldsymbol{\tau})\\ &\propto \exp\left\{\begin{bmatrix}\tau_{0,1}-1+\sum_{n=1}^{N} w_{n,1}\\ \vdots\\ \tau_{0,K}-1+\sum_{n=1}^{N} w_{n,K}\end{bmatrix}\cdot\begin{bmatrix}\log\varphi_1\\ \vdots\\ \log\varphi_K\end{bmatrix}\right\}.\end{aligned}$$

The maximization step requires computing the mode categorical weights. For a Dirichlet distribution the mode weights are $\hat{\boldsymbol{\varphi}} = (\boldsymbol{\tau}-1)/(\sum_{k=1}^{K}\tau_k - K)$ with $\tau_k > 1$. The parameter vector $\boldsymbol{\tau}$ is given by

$$\tau_k = \tau_{0,k} + \sum_{n=1}^{N} w_{n,k} \quad \forall k \in [1,\, K].$$

A. GAUSSIAN WITH NORMAL-WISHART PRIOR

A weighted Gaussian likelihood over a random variable $\mathbf{x} \in \mathbb{R}^d$ has the following precision-based parameterization

$$\begin{aligned}p(\mathbf{X}|\boldsymbol{\mu},\boldsymbol{\Lambda}) &= \prod_{n=1}^{N}\mathrm{N}(\mathbf{x}_n|\boldsymbol{\mu},\boldsymbol{\Lambda})^{w_n}\\ &\propto \exp\left\{\begin{bmatrix}\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\Lambda}\\ \log|\boldsymbol{\Lambda}|\end{bmatrix}\cdot\begin{bmatrix}\sum_{n=1}^{N} w_n\mathbf{x}_n\\ -\frac{1}{2}\sum_{n=1}^{N} w_n\\ -\frac{1}{2}\sum_{n=1}^{N} w_n\mathbf{x}_n\mathbf{x}_n^\top\\ \frac{1}{2}\sum_{n=1}^{N} w_n\end{bmatrix}\right\},\end{aligned}$$

where w_n are the importance weights. The conjugate prior $p(\boldsymbol{\mu},\boldsymbol{\Lambda})$ is a normal-Wishart distribution with zero mean

$$\begin{aligned}p(\boldsymbol{\mu},\boldsymbol{\Lambda}) &= \mathrm{N}(\boldsymbol{\mu}|\mathbf{0},\kappa_0\boldsymbol{\Lambda})\,\mathrm{W}(\boldsymbol{\Lambda}|\boldsymbol{\Psi}_0,\nu_0)\\ &\propto \exp\left\{\begin{bmatrix}\mathbf{0}\\ -\frac{1}{2}\kappa_0\\ -\frac{1}{2}\boldsymbol{\Psi}_0^{-1}\\ \frac{1}{2}(\nu_0-d)\end{bmatrix}\cdot\begin{bmatrix}\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\Lambda}\\ \log|\boldsymbol{\Lambda}|\end{bmatrix}\right\}.\end{aligned}$$

The resulting posterior $q(\boldsymbol{\mu},\boldsymbol{\Lambda})$ is also a normal-Wishart

$$\begin{aligned}q(\boldsymbol{\mu},\boldsymbol{\Lambda}) &= \mathrm{N}(\boldsymbol{\mu}|\mathbf{m},\kappa\boldsymbol{\Lambda})\,\mathrm{W}(\boldsymbol{\Lambda}|\boldsymbol{\Psi},\nu)\\ &\propto \exp\left\{\begin{bmatrix}\sum_{n=1}^{N} w_n\mathbf{x}_n\\ -\frac{1}{2}\left(\kappa_0+\sum_{n=1}^{N} w_n\right)\\ -\frac{1}{2}\left(\boldsymbol{\Psi}_0^{-1}+\sum_{n=1}^{N} w_n\mathbf{x}_n\mathbf{x}_n^\top\right)\\ \frac{1}{2}\left(\nu_0-d+\sum_{n=1}^{N} w_n\right)\end{bmatrix}\cdot\begin{bmatrix}\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\mu}\\ \boldsymbol{\Lambda}\\ \log|\boldsymbol{\Lambda}|\end{bmatrix}\right\}.\end{aligned}$$

The vector and matrix modes of a normal-Wishart distribution are $\hat{\boldsymbol{\mu}} = \mathbf{m}$ and $\hat{\boldsymbol{\Lambda}} = (\nu-d)\boldsymbol{\Psi}$, respectively. The posterior parameters needed to determine the modes are

$$\kappa = \kappa_0 + \sum_{n=1}^{N} w_n,\ \mathbf{m} = 1/\kappa\sum_{n=1}^{N} w_n\mathbf{x}_n,$$

$$\nu = \nu_0 + \sum_{n=1}^{N} w_n,\ \boldsymbol{\Psi} = \left(\boldsymbol{\Psi}_0^{-1} + \sum_{n=1}^{N} w_n\mathbf{x}_n\mathbf{x}_n^\top - \kappa\,\mathbf{m}\,\mathbf{m}^\top\right)^{-1}.$$

A. LINEAR-GAUSSIAN WITH MATRIX-NORMAL-WISHART PRIOR

A weighted linear-Gaussian likelihood takes a random variable $\mathbf{x} \in \mathbb{R}^d$ and returns a random variable $\mathbf{y} \in \mathbb{R}^m$ according to a linear mapping $\mathbf{A} : \mathbb{R}^d \to \mathbb{R}^m$

$$\begin{aligned}p(\mathbf{Y}|\mathbf{X},\mathbf{A},\mathbf{V}) &= \prod_{n=1}^{N}\mathrm{N}(\mathbf{y}_n|\mathbf{x}_n,\mathbf{A},\mathbf{V})^{w_n}\\ &\propto \exp\left\{\begin{bmatrix}\mathbf{VA}\\ \mathbf{A}^\top\mathbf{VA}\\ \mathbf{V}\\ \log|\mathbf{V}|\end{bmatrix}\cdot\begin{bmatrix}\mathbf{YWX}^\top\\ -\frac{1}{2}\mathbf{XWX}^\top\\ -\frac{1}{2}\mathbf{YWY}^\top\\ \frac{1}{2}\sum_{n=1}^{N} w_n\end{bmatrix}\right\},\end{aligned}$$

where w_n are the weights and $\mathbf{W} = \mathrm{diag}(w_n)$ is the diagonal weight matrix. The data matrices $\mathbf{X}$ and $\mathbf{Y}$ are of size $d \times N$ and $m \times N$, respectively. The conjugate prior $p(\mathbf{A},\mathbf{V})$ is a matrix-normal-Wishart with zero mean

$$\begin{aligned}p(\mathbf{A},\mathbf{V}) &= \mathrm{N}(\mathbf{A}|\mathbf{0},\mathbf{V},\mathbf{K}_0)\,\mathrm{W}(\mathbf{V}|\boldsymbol{\Psi}_0,\nu_0)\\ &\propto \exp\left\{\begin{bmatrix}\mathbf{0}\\ -\frac{1}{2}\mathbf{K}_0\\ -\frac{1}{2}\boldsymbol{\Psi}_0^{-1}\\ \frac{1}{2}(\nu_0-m-1+d)\end{bmatrix}\cdot\begin{bmatrix}\mathbf{VA}\\ \mathbf{A}^\top\mathbf{VA}\\ \mathbf{V}\\ \log|\mathbf{V}|\end{bmatrix}\right\}.\end{aligned}$$

The posterior $q(\boldsymbol{\mu},\boldsymbol{\Lambda})$ is matrix-normal-Wishart

$$\begin{aligned}q(\mathbf{A},\mathbf{V}) &= \mathrm{N}(\mathbf{A}|\mathbf{M},\mathbf{V},\mathbf{K})\,\mathrm{W}(\mathbf{V}|\boldsymbol{\Psi},\nu)\\ &\propto \exp\left\{\begin{bmatrix}\mathbf{YWX}^\top\\ -\frac{1}{2}(\mathbf{K}_0+\mathbf{XWX}^\top)\\ -\frac{1}{2}(\boldsymbol{\Psi}_0^{-1}+\mathbf{YWY}^\top)\\ \frac{1}{2}\left(\nu_0-m-1+d+\sum_{n=1}^{N} w_n\right)\end{bmatrix}\cdot\begin{bmatrix}\mathbf{VA}\\ \mathbf{A}^\top\mathbf{VA}\\ \mathbf{V}\\ \log|\mathbf{V}|\end{bmatrix}\right\}.\end{aligned}$$

The mode mapping and precision of a matrix-normal-Wishart are $\hat{\mathbf{A}} = \mathbf{M}$ and $\hat{\mathbf{\Lambda}} = (\nu - m)\mathbf{\Psi}$, respectively. The standard posterior parameters are

$$\mathbf{K} = \mathbf{K}_0 + \mathbf{X}\mathbf{W}\mathbf{X}^\top, \mathbf{M} = \mathbf{Y}\mathbf{W}\mathbf{X}^\top\mathbf{K}^{-1},$$

$$\nu = \nu_0 + \sum_{n=1}^{N} w_n, \mathbf{\Psi} = \left(\mathbf{\Psi}_0^{-1} + \mathbf{Y}\mathbf{W}\mathbf{Y}^\top - \mathbf{M}\,\mathbf{K}\,\mathbf{M}^\top\right)^{-1}.$$

REFERENCES

[1] I. Fantoni and R. Lozano, *Nonlinear Control for Underactuated Mechanical Systems*. Berlin, Germany: Springer Sci. Bus. Media, 2002.

[2] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, pp. 1238–1274, 2013.

[3] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.

[4] M. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 465–472.

[5] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res.*, vol. 17, pp. 1334–1373, 2016.

[6] D. Hafner et al., "Learning latent dynamics for planning from pixels," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2555–2565.

[7] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1889–1897.

[8] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," in *Proc. Int. Conf. Learn. Representations*, 2016.

[9] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1861–1870.

[10] D. Liberzon, *Switching in Systems and Control*. Berlin, Germany: Springer, 2003.

[11] W. M. Haddad, V. Chellaboina, and S. G. Nersesov, *Impulsive and Hybrid Dynamical Systems (Princeton Series in Applied Mathematics)*. Princeton, NJ, USA: Princeton Univ., 2006.

[12] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton, NJ, USA: Princeton Univ. Press, 2012.

[13] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2017.

[14] Z. Ghahramani and G. E. Hinton, "Variational learning for switching state-space models," *Neural Comput.*, vol. 12, no. 4, pp. 831–864, Apr. 2000.

[15] M. J. Beal, "Variational algorithms for approximate Bayesian inference," Ph.D. dissertation, Univ. College London, London, U.K., 2003.

[16] E. Fox, "Bayesian nonparametric learning of complex dynamical phenomena," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, USA, 2009.

[17] S. W. Linderman, M. J. Johnson, A. C. Miller, R. P. Adams, D. M. Blei, and L. Paninski, "Bayesian learning and inference in recurrent switching linear dynamical systems," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2017, pp. 914–922.

[18] H. Abdulsamad and J. Peters, "Hierarchical decomposition of nonlinear dynamics and control for system identification and policy distillation," in *Proc. Learn. Dyn. Control*, 2020, pp. 904–914.

[19] F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat, "An MPC/hybrid system approach to traction control," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 3, pp. 541–552, May 2006.

[20] P. Menchinelli and A. Bemporad, "Hybrid model predictive control of a solar air conditioning plant," *Eur. J. Control*, vol. 14, pp. 501–515, 2008.

[21] S. Paoletti, A. L. Juloski, G. Ferrari-Trecate, and R. Vidal, "Identification of hybrid systems: A tutorial," *Eur. J. Control*, vol. 13, pp. 242–260, 2007.

[22] A. Garulli, S. Paoletti, and A. Vicino, "A survey on switched and piecewise affine system identification," *Int. Federation Autom. Control*, vol. 45, pp. 344–355, 2012.

[23] R. Vidal, S. Soatto, Y. Ma, and S. Sastry, "An algebraic geometric approach to the identification of a class of linear hybrid systems," in *Proc. IEEE 42nd Int. Conf. Decis. Control*, 2003, pp. 167–172.

[24] A. Bemporad, J. Roll, and L. Ljung, "Identification of hybrid systems via mixed-integer programming," in *Proc. IEEE Conf. Decis. Control*, 2001, pp. 786–792.

[25] A. L. Juloski, S. Weiland, and W. P. M. H. Heemels, "A Bayesian approach to identification of hybrid systems," *IEEE Trans. Autom. Control*, vol. 50, no. 10, pp. 1520–1533, Oct. 2005.

[26] L. Bako, K. Boukharouba, and S. Lecoeuche, "An l_0-l_1 norm based optimization procedure for the identification of switched nonlinear systems," in *Proc. IEEE 49th Conf. Decis. Control*, 2010, pp. 4467–4472.

[27] F. Lauer, G. Bloch, and R. Vidal, "Nonlinear hybrid system identification with kernel models," in *Proc. IEEE 49th Conf. Decis. Control*, 2010, pp. 696–701.

[28] A. Bemporad and S. Di Cairano, "Optimal control of discrete hybrid stochastic automata," in *Proc. Int. Workshop IEEE Hybrid Syst.: Computation Control*, 2005, pp. 151–167.

[29] C. G. Cassandras and J. Lygeros, *Stochastic Hybrid Systems*. Boca Raton, FL, USA: CRC Press, 2006.

[30] E. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Trans. Autom. Control*, vol. 26, no. 2, pp. 346–358, Apr. 1981.

[31] F. Zhu and P. J. Antsaklis, "Optimal control of hybrid switched systems: A brief survey," *Discrete Event Dyn. Syst.*, vol. 25, pp. 345–364, 2015.

[32] E. F. Camacho, D. R. Ramírez, D. Limón, D. M. De La Peña, and T. Alamo, "Model predictive control techniques for hybrid systems," *Annu. Rev. Control*, vol. 34, pp. 21–31, 2010.

[33] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–427, 1999.

[34] A. Bemporad, F. Borrelli, and M. Morari, "Piecewise linear optimal controllers for hybrid systems," in *Proc. IEEE Amer. Control Conf.*, 2000, pp. 1190–1194.

[35] F. Borrelli, M. Baotic, A. Bemporad, and M. Morari, "An efficient algorithm for computing the state feedback optimal control law for discrete time hybrid systems," in *Proc. IEEE Amer. Control Conf.*, 2003, pp. 4717–4722.

[36] T. Marcucci and R. Tedrake, "Mixed-integer formulations for optimal control of piecewise-affine systems," in *Proc. 22nd ACM Int. Conf. Hybrid Syst.: Computation Control*, 2019, pp. 230–239.

[37] G. Ackerson and K. Fu, "On state estimation in switching environments," *IEEE Trans. Autom. Control*, vol. 15, no. 1, pp. 10–17, Feb. 1970.

[38] J. D. Hamilton, "Analysis of time series subject to changes in regime," *J. Econometrics*, vol. 45, pp. 39–70, 1990.

[39] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001.

[40] S. M. Oh, J. M. Rehg, T. Balch, and F. Dellaert, "Data-driven MCMC for learning and inference in switching linear dynamic systems," in *Proc. 20th Nat. Conf. Artif. Intell.*, 2005, pp. 944–949.

[41] B. Mesot and D. Barber, "Switching linear dynamical systems for noise robust speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 6, pp. 1850–1858, Aug. 2007.

[42] D. Koller, N. Friedman, and F. Bach, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[43] U. N. Lerner, "Hybrid Bayesian networks for reasoning about complex systems," Ph.D. dissertation, Stanford Univ., Stanford, CA, USA, 2002.

[44] M. D. Escobar and M. West, "Bayesian density estimation and inference using mixtures," *J. Amer. Stat. Assoc.*, vol. 90, pp. 577–588, 1995.

[45] C. E. Rasmussen, "The infinite Gaussian mixture model," in *Proc. Adv. Neural Inf. Process. Syst.*, 1999.

[46] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen, "The infinite hidden Markov model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002.

[47] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Sharing clusters among related groups: Hierarchical Dirichlet processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005.

[48] E. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "Nonparametric Bayesian learning of switching linear dynamical systems," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009.

[49] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Representations*, 2014.

[50] P. Becker-Ehmck, J. Peters, and P. Van Der Smagt, "Switching linear dynamics for variational Bayes filtering," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 553–562.

[51] A. G. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning," *Discrete Event Dyn. Syst.*, vol. 13, pp. 41–77, 2003.

[52] R. E. Parr, "Hierarchical control and learning for Markov decision processes," Ph.D. dissertation, Univ. California Berkeley, Berkeley, CA, USA, 1998.

[53] D. Precup, "Temporal abstraction in reinforcement learning," Ph.D. dissertation, Univ. Massachusetts Amherst, Amherst, MA, USA, 2000.

[54] D. Andre and S. J. Russell, "State abstraction for programmable reinforcement learning agents," in *Proc. Nat. Conf. Artif. Intell.*, 2002, pp. 119–125.

[55] R. S. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artif. Intell.*, vol. 112, pp. 181–211, 1999.

[56] R. S. Sutton, "Intra-option learning about temporally abstract actions," in *Proc. Int. Conf. Mach. Learn.*, 1998, pp. 556–564.

[57] S. J. Bradtke and M. O. Duff, "Reinforcement learning methods for continuous-time Markov decision problems," in *Proc. Adv. Neural Inf. Process. Syst.*, 1995.

[58] M. Huber and R. A. Grupen, "Learning to coordinate controllers-reinforcement learning on a control basis," in *Proc. Int. Joint Conf. Artif. Intell.*, 1997, pp. 1366–1371.

[59] M. Huber, "A hybrid architecture for adaptive robot control," Ph.D. dissertation, Univ. Massachusetts Amherst, Amherst, MA, USA, 2000.

[60] G. Konidaris and A. G. Barto, "Skill discovery in continuous reinforcement learning domains using skill chaining," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009.

[61] D. J. Mankowitz, T. A. Mann, and S. Mannor, "Adaptive skills adaptive partitions (ASAP)," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016.

[62] C. Daniel, H. Van Hoof, J. Peters, and G. Neumann, "Probabilistic inference for determining options in reinforcement learning," *Mach. Learn.*, vol. 104, pp. 337–357, 2016.

[63] P.-L. Bacon, J. Harb, and D. Precup, "The option-critic architecture," in *Proc. AAAI Conf. Artif. Intell.*, 2017.

[64] M. Smith, H. Hoof, and J. Pineau, "An inference-based policy gradient method for learning options," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4703–4712.

[65] T. G. Dietterich, "State abstraction in MAXQ hierarchical reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000.

[66] L. Li, T. J. Walsh, and M. L. Littman, "Towards a unified theory of state abstraction for MDPs," in *Proc. Int. Symp. Artif. Intell. Math.*, 2006.

[67] R. Akrour, F. Veiga, J. Peters, and G. Neumann, "Regularizing reinforcement learning with state abstraction," in *Proc. IEEE RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 534–539.

[68] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robot. Automat. Mag.*, vol. 17, no. 2, pp. 44–54, Jun. 2010.

[69] M. Burke, Y. Hristov, and S. Ramamoorthy, "Hybrid system identification using switching density networks," in *Proc. Conf. Robot Learn.*, 2020, pp. 172–181.

[70] A. Sosic, A. M. Zoubir, and H. Koeppl, "A Bayesian approach to policy recognition and state representation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1295–1308, Jun. 2018.

[71] D. Barber, "Expectation correction for smoothed inference in switching linear dynamical systems," *J. Mach. Learn. Res.*, vol. 7, 2006, pp. 2515–2540.

[72] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[73] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Statist.*, vol. 41, pp. 164–171, 1970.

[74] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc.*, vol. 39, pp. 1–22, 1977.

[75] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[76] Y. Bengio and P. Frasconi, "An input-output HMM architecture," in *Proc. Adv. Neural Inf. Process. Syst.*, 1995.

[77] J. S. Maritz and T. Lwin, "Empirical Bayes methods with applications," New York, USA: Chapman and Hall/CRC, 1989.

[78] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, pp. 79–86, 1951.

[79] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, pp. 400–407, 1951.

[80] J. Peters, K. Mülling, and Y. Altun, "Relative entropy policy search," in *Proc. AAAI Conf. Artif. Intell.*, 2010, pp. 1607–1612.

[81] H. Van Hoof, J. Peters, and G. Neumann, "Learning of non-parametric control policies with high-dimensional state features," in *Int. Conf. Artif. Intell. Statist.*, 2015, pp. 995–1003.

[82] B. Belousov and J. Peters, "f-Divergence constrained policy improvement," 2017, *arXiv:1801.00056*.

[83] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 2014.

[84] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[85] C. P. Robert, G. Casella, and G. Casella, *Monte Carlo Statistical Methods*. Berlin, Germany: Springer, 1999.

[86] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

[87] M. Deisenroth, G. Neumann, and J. Peters, "A survey on policy search for robotics," *Foundations Trends Robot.*, vol. 2, pp. 1–142, 2013.

[88] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[89] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[90] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008.

[91] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, pp. 1–305, 2008.

HANY ABDULSAMAD received the master's degree in electrical engineering and information technology from the Technical University of Darmstadt, Darmstadt, Germany, and the Ph.D. degree in computer science under the supervision of Jan Peters from the Intelligent Autonomous Systems Group, Technical University of Darmstadt. He is currently a postdoctoral researcher at Aalto University, Espoo, Finland. His research interests include optimal control, statistical learning, and robotics.

JAN PETERS (Fellow, IEEE) is currently a Full Professor (W3) with the Computer Science Department, Technical University of Darmstadt, Darmstadt, Germany. He was the recipient of the Dick Volz Best 2007 U.S. Ph.D. Thesis Runner-Up Award, Robotics: Science & Systems - Early Career Spotlight, INNS Young Investigator Award, IEEE Robotics & Automation Society's Early Career Award, and numerous best paper awards. In 2015, he received an ERC Starting Grant.