Parameterized input inference for approximate stochastic optimal control

Shahbaz P Qadri Syed[1] and He Bai[1]

***Abstract*— Probabilistic inference approaches to stochastic optimal control have attracted significant interest from researchers in the past decade. Existing inference-based optimal control approaches are limited to linear controllers in a finite-horizon model-based setting. Since nonlinear systems typically admit nonlinear optimal controllers, linear controllers may yield sub-optimal trajectories when applied to nonlinear systems. In this paper, we propose a new Expectation-Maximization (EM) based inference algorithm for stochastic optimal control. The algorithm employs nonlinear basis functions to infer nonlinear controllers. We formulate the estimation problem of optimal control as a parameter inference problem. We demonstrate the effectiveness of the algorithm on a simulated nonlinear oscillator system for nonlinear control and a linear thermal system for structured control.**

I. Introduction

The stochastic optimal control (SOC) problem aims at finding a control sequence in the presence of uncertainty for a dynamical system over a finite or infinite horizon such that the control minimizes an expected cost. The uncertainty makes the SOC problem much more challenging than its deterministic counterpart. This uncertainty is either in the form of noisy observations or process noise that approximates model uncertainties in the system.

A solution to the SOC problem can be found by solving the stochastic Hamilton-Jacobi-Bellman (HJB) [1] equation which is a nonlinear PDE. However its numerical solution requires discretization of the space and time that makes it computationally intractable due to the curse of dimensionality [2]. A fast and locally approximate solution to the SOC problem is the Linear Quadratic Gaussian (LQG) case where the SOC problem is solved for the noise free optimal trajectory and a local LQG model is constructed as perturbation around this trajectory. As long as the model is not perturbed too far away from the optimal noise-free trajectory, the local linear quadratic regulator computes a reasonably approximate solution to the original SOC problem. The local LQG can be analytically solved in closed form using Ricatti equations. An algorithm motivated by this approach is the iterative linear quadratic gaussian (*ilqg*) [3] that performs iterative linearization of non-linear system dynamics around the current trajectory and uses the LQG paradigm to obtain update equations to compute locally optimal trajectory.

An alternative promising direction to solve the SOC problem in discrete time was developed in the last decade that reformulates the SOC problem as a graphical model inference problem. References [4], [5], [6], [7], [8] are some examples of inference-based control approaches that have been successfully used in real world applications. The idea of approximate inference for control is also connected to the field of reinforcement learning (RL). ψ-Learning [9] and Soft Q-learning [10] are some of the algorithms developed in this direction. The RL as inference framework has also been studied in the context of risk-sensitive control in [11], [9]. Recently, [12] proposes an approach analogous to the inference based control that relates the optimization-based model predictive control (MPC) to Bayesian estimation for deterministic nonlinear systems.

The aforementioned model based inference approaches to SOC have been derived only in the LQG setting and the non-LQG setting with linear controllers. However, nonlinear systems typically admit nonlinear optimal controllers. Thus, the use of linear optimal controllers can lead to a suboptimal solution to the SOC problem in a non-LQG setting. Recent research in neural network based control, Koopman operators, and Carleman linearziation has motivated the use of controllers parameterized with nonlinear basis functions.

In this paper, we propose a generic inference-based control algorithm to address the SOC problem in discrete time. Particularly, we infer *nonlinear* controllers in a model based inference setting by parameterizing the controller with a nonlinear basis function. This allows us to reformulate the original nonlinear input estimation problem [7], [8] as a parameter inference problem which can be solved using Expectation-Maximization (EM) algorithm.

We perform numerical simulations on a nonlinear oscillator system to demonstrate the effectiveness of the nonlinear controllers inferred by the proposed algorithm over linear controllers. In addition, our algorithm can be easily adapted to produce structured controls. An example of structured control is distributed optimal control for multi-agent systems, where the control of each agent can contain information only from a subset of the agents. We employ a linear thermal dynamics model to demonstrate the ability of the proposed algorithm to infer structured controllers.

The remainder of the paper is organized as follows. Section II presents the formulation of inference-based control approaches and reviews relevant previous work. Section III discusses the *parameterized input inference for control (PIIC)* algorithm for nonlinear control, which is the main contribution of this paper. Section IV presents two simulation examples to demonstrate the effectiveness of the PIIC algorithm for nonlinear and structured control.

[1]Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK, 74078 {shahbaz_qadri.syed, he.bai}@okstate.edu. The work was supported by the National Science Foundation (NSF) under Grant No. 1925147 and 2212582.

Conclusions and future work are provided in Section V.

Notation: Let $y \sim \mathcal{N}(a, A)$ represent a random variable y satisfying a Gaussian distribution in the normal form with mean $a \in \mathbb{R}^d$ and covariance $A \in \mathbb{R}^{d\times d}$ given by

$$\mathcal{N}(a, A) = \frac{1}{(2\pi)^{\frac{d}{2}}|A|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(y-a)^T A^{-1}(y-a)\right),$$

where $|A|$ represents the determinant of A.

II. Inference-based control

A. Formulation of inference-based control

Consider a dynamical system given by

$$x_{t+1} = F(x_t, u_t) + \eta_t, \tag{1}$$

where $x_t \in \mathbb{R}^{n_x}$ and $u_t \in \mathbb{R}^{n_u}$ denote the state and control at time t, respectively, $F\colon \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is a nonlinear mapping of x_t, u_t, $\eta_t \sim \mathcal{N}(0, \Sigma_{\eta_t})$ represents additive Gaussian noise that models the uncertainty in the dynamics. We denote the state-control vector at time t by $\tau_t = [x_t^T \quad u_t^T]^T \in \mathbb{R}^{n_x+n_u}$. Thus, (1) can be rewritten as

$$x_{t+1} = F(\tau_t) + \eta_t, \tag{2}$$

For a given finite-horizon T and a state-control sequence $[x_T, \tau_{0:T-1}]$, we denote the trajectory cost $\mathcal{C}(x_T, \tau_{0:T-1})$, as the summation of the costs per stage and the terminal cost. The considered SOC problem is summarized as

$$\begin{aligned} &\min_{u_{0:T-1}} \mathbb{E}[\mathcal{C}(x_T, \tau_{0:T-1})] \\ &\text{such that } x_{t+1} \sim \mathcal{N}(F(\tau_t), \Sigma_{\eta_t}). \end{aligned} \tag{3}$$

Probabilistic inference approaches formulate the stochastic optimal control problem in (3) as an inference problem on a probabilistic graphical model (PGM). A PGM is a probabilistic model that encodes complex relationships between random variables in the form of a graph. The PGM for the SOC problem is constructed with the state-control sequence as latent variables and the sequence of binary random variables $\mathcal{O}_t \in \{0, 1\}$, $t = 0, \cdots, T$, as observed variables. The binary random variable $\mathcal{O}_t$ represents the notion of optimality or task fulfilment at each time step. In other words, $\mathcal{O}_t = 1$ when optimal state and action are observed at time t. The probabilistic inference approaches relate the probabilities to cost by assuming that the optimality/task fulfilment is observed throughout the trajectory, i.e., $\mathcal{O}_t = 1$, $t = 0, \cdots, T$. This allows modeling of the negative log-likelihood of observation at time t proportional to the stage cost c_t, i.e.

$$\begin{aligned} -\log\left(p(\mathcal{O}_t = 1|\tau_t)\right) \propto c_t(\tau_t) \iff \\ p(\mathcal{O}_t = 1|\tau_t) \propto \exp\{-c_t(\tau_t)\}. \end{aligned} \tag{4}$$

Hence, the likelihood of observing optimality at each time step is high if and only if the cost incurred is low. Then the optimal trajectory is computed as the mean of the conditional or joint posterior distribution of the state-control trajectory given the observations.

B. Previous work

Most of the previous work on probabilistic inference for SOC problems was derived in a Linear Qaudratic Gaussian (LQG) setting to infer a linear controller of the form $p(u_t|x_t) = \mathcal{N}(u_t|\mathbf{K}_t x_t + k_t, \Sigma_t)$. For the LQG case, the dynamics in (1) are linear and the stage cost c_t and terminal cost c_T are quadratic given by

$$F(\tau_t) = \begin{bmatrix} A_t & 0 \\ 0 & B_t \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} + a_t, \tag{5}$$

$$c_t(\tau_t) = \begin{bmatrix} x_t & u_t \end{bmatrix} \begin{bmatrix} Q_t & 0 \\ 0 & R_t \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix}. \tag{6}$$

where $A_t \in \mathbb{R}^{n_x\times n_x}, B_t \in \mathbb{R}^{n_u\times n_u}, a_t \in \mathbb{R}^{n_x}$ represent the system matrices and $Q_t \in \mathbb{R}^{n_x\times n_x}, R_t \in \mathbb{R}^{n_u\times n_u}$ represent the cost matrices.

1) Approximate Inference Control (AICO): The AICO [4] algorithm infers the optimal state trajectories in the LQG case by marginalizing out the controls using the control cost as a prior, i.e., $u_t \sim \mathcal{N}(u_t|0, R_t^{-1})$. This is a consequence of decoupling the state and control cost in (4) and (6) by exploiting the structure of the graph. Then the posterior distribution is approximated using the Gaussian message passing technique on the Maximum a posteriori (MAP) trajectory. The marginalization of controls during inference may yield trajectories that are agnostic to the control constraints and hence infeasible, i.e., no controller can generate the inferred state trajectory. Subsequent work in [5], [13], [14] was proposed to address the drawbacks of AICO and improve its performance.

2) Input inference for control (I2C): The I2C [8], [7] algorithm formulates the SOC problem as an input estimation problem. It infers an optimal linear controller using an EM approach. The expectation step computes the optimal state-control distribution given the inverse temperature parameter α that acts as a scale invariance for the precision of the observation distribution using Kalman smoother like updates. The maximization step finds the inverse temperature that maximizes the expected log likelihood given the state-control distribution. At the end of each pass of the E-step and the M-step the priors of the state and control are updated with the smoothed distributions which are then used to compute a linear controller in the next iteration. The cost is encoded into the optimality variable $\mathcal{O}_t$ through the observations as

$$z_t \sim \mathcal{N}(z_t|h(\tau_t), \Sigma_{\xi_t}), \tag{7}$$

where $z_t \in \mathbb{R}^{n_z}$ denotes the measurement at time t, $h\colon \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_z}$ is a nonlinear mapping of x_t, u_t, $\xi_t \sim \mathcal{N}(0, \Sigma_{\xi_t})$ represents the additive Gaussian noise that models the uncertainty in the measurement. When the cost is quadratic, the likelihood in (4) can be rewritten as

$$p(\mathcal{O}_t = 1|\tau_t) \propto \exp\{-\alpha(\tau_t^T \Gamma_t \tau_t)\} \tag{8}$$

$$= \mathcal{N}(z_t = z_t^*|\tau_t, (\alpha\Gamma_t)^{-1}), \tag{9}$$

where $\Gamma_t = \begin{bmatrix} Q_t & 0 \\ 0 & R_t \end{bmatrix}$ and $z_t^* = 0$. The likelihood that serves as an objective for the inverse temperature optimiza-

tion is given by

$$p(\tau_{0:T}, \mathcal{O}_{0:T} = 1, \Gamma_{0:T}, \alpha) = p(x_0)p(z_T|x_T, \alpha) \prod_{t=0}^{T-1} p(x_{t+1}|\tau_t)p(z_t|\tau_t, \alpha)p(u_t|x_t). \quad (10)$$

As mentioned in the introduction, existing approaches on model-based inference control prescribe a linear control structure with a Gaussian noise for $p(u_t|x_t)$ in (10). In the following section we parameterize the controller with a nonlinear basis function and present an algorithm that yields nonlinear optimal controllers. Particularly, the parameters of the controller are considered latent variables to be inferred using an EM algorithm.

III. Parameterized-input inference for control (PIIC)

A. Nonlinear Control

We assume that the feedback controller u_t at each time step is parameterized by a basis function (possibly nonlinear) of the state, $\mathcal{B}_t(x_t) \in \mathbb{R}^{n_b \times 1}$, and unknown parameters $\Theta_t \in \mathbb{R}^{n_b \times n_u}$ such that

$$u_t = \Theta_t^T \mathcal{B}_t(x_t) + \delta_t \quad (11)$$

$$\Rightarrow p(u_t|x_t) = \mathcal{N}(u_t|\Theta_t^T \mathcal{B}_t(x_t), \Sigma_\delta) \quad (12)$$

where $u_t \in \mathbb{R}^{n_u \times 1}$, $x_t \in \mathbb{R}^{n_x \times 1}$ represent the control and state at time t respectively, and δ_t represents a zero-mean random Gaussian noise with covariance Σ_δ that models the uncertainty in control.

The objective of the PIIC algorithm is to infer the parameters $\Theta_{0:T-1}$ and α that maximize the log-likelihood, i.e.,

$$\Theta^*_{0:T-1}, \alpha^* = \underset{\Theta_{0:T-1}, \alpha}{\text{argmax}} \log[p(\mathcal{O}_{0:T} = 1|\Theta_{0:T-1}, \alpha)]. \quad (13)$$

Here, α refers to the inverse temperature parameter defined in Section II-B.2. The optimization problem in (13) is analytically intractable. Thus, we resort to computing the parameters using an EM algorithm.

To simplify the notation, we denote $\tau_{0:T-1}$ by τ, $\mathcal{O}_{0:T} = 1$ by $\mathcal{O}$ and $\Theta_{0:T-1}$ by $\mathbf{\Theta}$. Then the objective in (13) is rewritten as

$$\log[p(\mathcal{O}|\mathbf{\Theta}, \alpha)] = \log \left[\int p(x_T, \tau, \mathcal{O}|\mathbf{\Theta}, \alpha) d\tau dx_T \right]. \quad (14)$$

Introducing $q(x_T, \tau)$, a known tractable distribution of x_T and τ, we obtain

$$\log[p(\mathcal{O}|\mathbf{\Theta}, \alpha)] = \log \left[\mathop{\mathbb{E}}_{q(x_T,\tau)} \left[\frac{p(x_T, \tau, \mathcal{O}|\mathbf{\Theta}, \alpha)}{q(x_T, \tau)} \right] \right]. \quad (15)$$

Using Jensen's inequality, we further get

$$\log[p(\mathcal{O}|\mathbf{\Theta}, \alpha)] \geq \mathop{\mathbb{E}}_{q(x_T,\tau)} \log \left[\frac{p(x_T, \tau, \mathcal{O}|\mathbf{\Theta}, \alpha)}{q(x_T, \tau)} \right]. \quad (16)$$

It can be shown that the inequality in (16) becomes equality for $q(x_T, \tau) = p(x_T, \tau|\mathcal{O})$. The PIIC algorithm aims at optimizing the right hand side of (16) based on the EM procedure.

B. Update equations

The E-step computes the smoothed state-control distribution given the parameters and the M-step computes the parameters that maximize the expected log posterior over the smoothed distribution. These steps are recursively computed until convergence. In this section we derive the update equations for the parameters $\mathbf{\Theta}$ and α in the M-step. The integrand in (14) is proportional to the joint posterior distribution given by

$$p(x_T, \tau_{0:T-1}, \mathcal{O}_{0:T} = 1, \Theta_{0:T-1}, \alpha) = p(x_0)p(\mathcal{O}_T = 1|x_T) \prod_{t=0}^{T-1} p(x_{t+1}|\tau_t)p(\mathcal{O}_t = 1|\tau_t, \alpha)p(u_t|x_t, \Theta_t). \quad (17)$$

Substituting (17) in the M-step yields

$$\underset{\mathbf{\Theta}, \alpha}{\text{argmax}} \mathop{\mathbb{E}}_{\tau \sim q(x_T,\tau)} \log[p(x_T, \tau, \mathcal{O}|\mathbf{\Theta}, \alpha)] \propto \underset{\mathbf{\Theta}, \alpha}{\text{argmax}} \mathop{\mathbb{E}}_{\tau \sim q(x_T,\tau)} \log \left[p(x_0)p(\mathcal{O}_T = 1|x_T) \prod_{t=0}^{T-1} p(x_{t+1}|\tau_t)p(\mathcal{O}_t = 1|\tau_t, \alpha)p(u_t|x_t, \Theta_t) \right]. \quad (18)$$

To find Θ_t^{k+1}, we take gradient of (18) with respect to Θ_t and set it to zero, which yields

$$\nabla_{\mathbf{\Theta_t}} \mathop{\mathbb{E}}_{\tau \sim q(\tau_t)} \log[\mathcal{N}(u_t|\Theta_t^T \mathcal{B}_t(x_t), \Sigma_{\delta_t})] = 0 \Rightarrow \quad (19)$$

$$\nabla_{\mathbf{\Theta_t}} \mathop{\mathbb{E}}_{\tau \sim q(\tau_t)} (u_t - \Theta_t^T \mathcal{B}_t(x_t))^T \Sigma_{\delta_t}^{-1} (u_t - \Theta_t^T \mathcal{B}_t(x_t)) = 0. \quad (20)$$

Solving (20) gives

$$\Theta_t^{k+1} = \left[\mathop{\mathbb{E}}_{\tau \sim q(\tau_t)} [\mathcal{B}_t(x_t)\mathcal{B}_t(x_t)^T] \right]^{-1} \left[\mathop{\mathbb{E}}_{\tau \sim q(\tau_t)} [\mathcal{B}_t(x_t)u_t^T] \right]. \quad (21)$$

The covariance of the controller Σ_{δ_t} is updated using

$$\Sigma_{\delta_t} = \mathop{\mathbb{E}}_{\tau \sim q(\tau)} (u_t - \Theta^{k+1^T} \mathcal{B}_t(x_t))(u_t - \Theta^{k+1^T} \mathcal{B}_t(x_t))^T. \quad (22)$$

Similarly, to find the α^{k+1} we take gradient of (18) with respect to α and set it to zero, which yields

$$\alpha^{k+1} = \frac{(T-1)n_z + n_{z_T}}{\sum_{t=0}^{T} \text{Tr}(\Gamma_t \mathbb{E}[(z_t^* - z_t)(z_t^* - z_t)^T])}, \quad (23)$$

where $Tr(\cdot)$ denotes the trace operator and the expectation is taken over the smoothed state-control distribution. Note that (23) is the same as in [7], [8].

C. The proposed PIIC algorithm

The formulation presented in the previous section applies to general nonlinear systems in the form of (1) with a controller in (12). In this section, we propose the *unscented-PIIC* algorithm summarized in Algorithm 1. Line 2 in Algorithm 1 corresponds to the E-step that computes the smoothed state-control distribution given the parameters Θ, α. Here we use the unscented I2C algorithm for this purpose. The unscented

Algorithm 1: Unscented-PIIC algorithm

Input : start distribution μ_{x_0}, Σ_{x_0} , goal z_T, system matrices $F(.)$, $h(.)$, cost matrices $Q_{0:T}$, $R_{0:T-1}$, noise covariances $\Sigma_{\xi_{0:T}}$, $\Sigma_{\eta_{0:T}}$, $\Sigma_{\delta_{0:T-1}}$

Output: $\tau^*_{0:T-1}$,Θ^*, α^*.

1 **repeat**

2 Perform unscented I2C to obtain smoothed state control distribution $p(x_T, \tau|\mathcal{O}, \mathbf{\Theta}, \alpha)$

3 Update Θ using (21)

4 Update Σ_δ using (22)

5 Update α using (23)

6 **until** *convergence*;

I2C algorithm is a variant of the Gaussian I2C algorithm [7] that uses unscented transforms [15], [16] to propagate the Gaussian distribution through nonlinear functions. Lines 3-5 in Algorithm 1 correspond to the M-step that computes the parameters and the covariance of the controller Σ_δ, $\forall\ t \in [0, T-1]$, using (21)–(23). These steps are computed iteratively until convergence. The expectations in the M-step are computed using unscented transforms.

The choice of using Gaussian-I2C algorithm in the E-step is arbitrary as any smoothing algorithm that returns the optimal state-control distribution given the controller parameters can be used in the E-step. The inference algorithm in the M-step finds the parameters given the control and the basis function and hence it is independent of the choice of the algorithm in the expectation step. We can show that the PIIC algorithm recovers the I2C algorithm for linear systems with a linear basis function $\mathcal{B}_t(x_t) = [x_t\ \ 1]$. Since the PIIC algorithm is an EM algorithm, convergence to a local maximum is guaranteed [17].

IV. Simulation examples

A. Nonlinear oscillator

Consider the nonlinear oscillator dynamics in [18]:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1 - \frac{1}{2}x_2\cos^2(x_1) + \sin(x_1)u. \end{aligned} \tag{24}$$

For the optimal control problem of minimizing the objective function $\mathcal{C} = \int_0^\infty (x_2^2 + u^2)dt$ subject to the dynamics of the form in (24), [19] shows that an analytical solution to the HJB equation in continuous time is given by

$$u = -x_2\sin(x_1). \tag{25}$$

We solve the optimal control problem of driving the nonlinear oscillator to the origin using Algorithm 1 in a discrete time setting with the following parameters:

$$\begin{aligned} \mu_{x_0} = \begin{bmatrix}3 & 3\end{bmatrix}^T,\ \Sigma_{x_0} = 10^{-4}\mathbb{I}_2,\ \Sigma_{\delta_{0:T-1}} = 1e5, \\ F_t = \begin{bmatrix}x_{1_{t+1}} & x_{2_{t+1}}\end{bmatrix}^T,\ h_t = \begin{bmatrix}\tau_t\end{bmatrix},\ \Sigma_{\eta_{0:T}} = 0, \\ Q_{0:T} = \begin{bmatrix}10^{-8} & 0 \\ 0 & 1\end{bmatrix},\ R_{0:T-1} = 1, \end{aligned}$$

where $\mathbb{I}_n$ denotes an identity matrix of shape $n \times n$. We use the forward Euler method to discretize (24) with a step-size 0.01 seconds and simulate for $T = 1500$ steps. We employ the PIIC algorithm to develop three controllers using a linear basis $\mathcal{B}_t^L(x) = [x_1\ \ x_2]^T$, a nonlinear basis $\mathcal{B}_t^{NL}(x) = [x_1\ \ x_2\ \ x_2\sin x_1\ \ x_2\cos x_1]^T$, and another nonlinear basis $\mathcal{B}_t^{KL}(x) = [x_1\ \ x_2\ \ x_1x_2\ \ x_1^2\ \ x_2^2]^T$. The $\mathcal{B}_t^{NL}$ is due to the optimal solution in (25). The $\mathcal{B}_t^{KL}$ is motivated by Carleman linearziation.

We compare these three controllers with respect to the continuous time optimal solution (25). Figure 1 shows the closed-loop trajectory of the oscillator for the four controllers. We observe that although the states are driven to the goal over time in all the cases, the optimality of the trajectory varies with the choice of the basis function. The nonlinear basis $\mathcal{B}_t^{NL}$ generates the trajectory closest to the optimal HJB solution. However, designing such an accurate basis function for a general nonlinear system is not possible. It is a common practice to use polynomial basis functions such as $\mathcal{B}_t^{KL}$. The trajectory generated by $\mathcal{B}_t^{KL}$ is clearly closer to the optimal HJB solution compared to the linear basis function $\mathcal{B}_t^L$ as shown in Fig. 1. This demonstrates the effectiveness of using nonlinear controllers for nonlinear systems as compared to their linear counterparts.

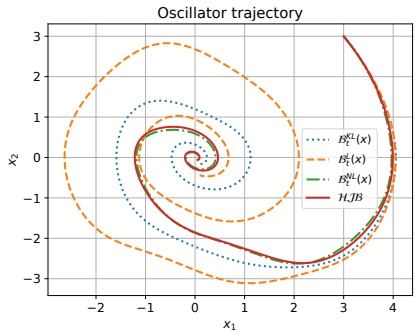
The trajectory costs incurred for the choice of different basis functions are summarized in Table I. For the choice of the basis $\mathcal{B}_t^{NL}$, the corresponding inferred parameter was $\Theta = \begin{pmatrix}0.04268 & -0.0524 & -1.1012 & 0.10083\end{pmatrix}^T$ which approximates (25). We observed that the computational time for the $\mathcal{B}_t^L$ and $\mathcal{B}_t^{KL}$ basis is comparable. Even though $\mathcal{B}_t^{NL}$ is the closest approximation of the analytical HJB solution, it was approximately 7 times slower compared to $\mathcal{B}_t^{KL}$.

TABLE I: Comparison of the controllers using various basis functions for the oscillator model (24) with $x_0 = [3, 3]$.

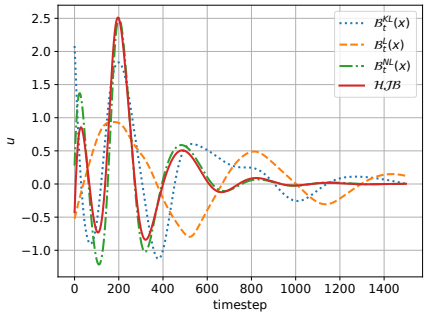
Basis function	Cost
HJB	1838.98
$\mathcal{B}_t^L(x)$	3619.86
$\mathcal{B}_t^{NL}(x)$	1869.47
$\mathcal{B}_t^{KL}(x)$	2410.297

B. Structured temperature control

The update in (21) can be extended to address structured control problems by imposing a structure on the parameter Θ. It can be shown that (21) holds for the non-zero subset of each column of the structured Θ, thereby preserving the structure during inference. In the interest of space, we omit the derivation for structured controllers.



(a) Trajectory of the nonlinear oscillator.



(b) Control inputs.

Fig. 1: Comparison of the state and control trajectories of the nonlinear oscillator for different choices of basis functions.

To demonstrate the structured control we consider the temperature control example in [20]. The problem deals with finding an optimal controller which controls a quantity related to the airflow rate into four zones to achieve a desired temperature. The zones form a rectangle such that zones 1,4 and zones 2,3 do not share a common wall. From a controls perspective, the state $x \in \mathbb{R}^4$ represents the temperature in the zones and $u \in \mathbb{R}^4$ denotes the control input. The evolution of temperature in the zones is given by a linear thermal dynamics model given by

$$x_{t+1} = A_t x_t + a_t + B_t u_t + W_t \tag{26}$$

such that $\forall\ i, j \in \{1, \cdots, 4\}$

$$A_{ij} = \begin{cases} 1 - \frac{\Delta}{\nu_i \zeta_i} - \sum_{j=1}^{4} \frac{\Delta}{\nu_i \zeta_{ij}} & \text{, if } i = j \\ \frac{\Delta}{\nu_i \zeta_{ij}} & \text{, otherwise} \end{cases} \tag{27}$$

$$B_{ij} = \begin{cases} \frac{\Delta}{\nu_i} & \text{, if } i = j \\ 0 & \text{, otherwise} \end{cases} \tag{28}$$

$$a_i = \frac{\Delta}{\nu_i \zeta_i} \epsilon_0 + \frac{\Delta}{\nu_i} \pi_i, \quad W_i = \frac{\sqrt{\Delta}}{\nu_i} w_i, \tag{29}$$

where ϵ_0 is the outdoor temperature, Δ is the time resolution, ν_i is the thermal capacitance of zone i, ζ_i denotes the thermal resistance of windows and walls between zone i and the environment, ζ_{ij} denotes the thermal resistance between zones i, j, π_i represents the constant heat addition from external sources into zone i, and $w_i \sim \mathcal{N}(0, 1)$ represents the process noise in zone i. The optimal controller aims at minimizing a quadratic cost function given by

$$\mathcal{C}_i(x_i, u_i) = (x_i - \epsilon_i^*)^2 + \beta_i u_i^2 \tag{30}$$

where ϵ_i^* is the desired temperature of zone i and β_i is a trade off parameter. A linear controller of the following form is considered

$$u_i = K_i x + k_i. \tag{31}$$

We solve the optimal control problem of achieving a desired temperature in each zone governed by the dynamics in (26)-(29) using Algorithm 1 with the following parameters

$$\mu_{x_0} = \begin{bmatrix} 30 & 27 & 24 & 18 \end{bmatrix}^T \text{°C},\ \Sigma_{x_0} = 10^{-4}\mathbb{I}_4,$$
$$F_t = \begin{bmatrix} x_{1_{t+1}} & x_{2_{t+1}} & x_{3_{t+1}} & x_{4_{t+1}} \end{bmatrix}^T,\ h_t = \begin{bmatrix} \tau_t \end{bmatrix},$$
$$\beta_i = 0.01,\ \Delta = 60 \text{ sec},\ T = 100 \text{ steps},\ \Sigma_{\delta_{0:T-1}} = 1e2,$$
$$Q_{0:T} = \mathbb{I}_4,\ R_{0:T-1} = \beta_i \mathbb{I}_4,\ \epsilon^* = 30\text{°C},$$

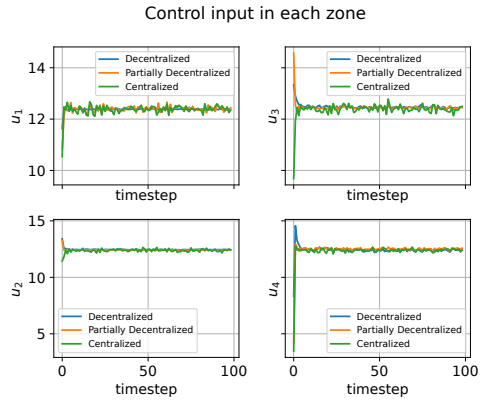
$$\forall i,\ \nu_i = 200 \text{ kJ/°C},\ \pi_i = 1 \text{ kW},$$
$$\forall i,\ \Sigma_\eta = \frac{\Delta \times 6.25}{\nu_i^2},\ \zeta_i = 1\text{°C/kW},$$
$$\forall i, j,\ \zeta_{ij} = \begin{cases} 1 & \text{, if zone } i, j \text{ share a common wall} \\ 0 & \text{, otherwise.} \end{cases}$$

We apply the PIIC algorithm with three control structures. We use a *centralized* structure where the controller of each zone employs the temperature information of all the zones, a *partially decentralized* structure where the controller of each zone employs its own temperature information and the temperature information of one adjacent zone (e.g., the controller in zone 1 employs temperature information of zone 1 and zone 2, the controller in zone 2 employs temperature information of zone 2 and zone 3, and so on), and a *decentralized* structure where the controller of each zone employs only its own temperature information.

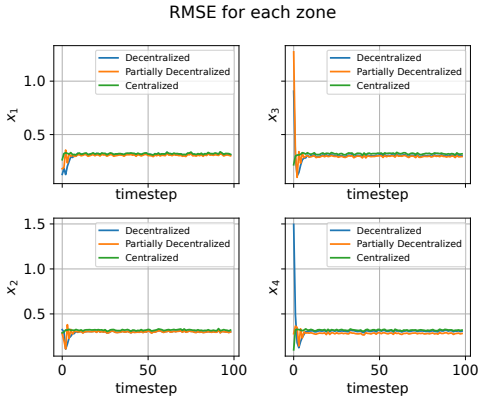
Figure 2 shows the control inputs, and the root mean squared error (RMSE) values of temperature for each zone using the *centralized, partially decentralized, decentralized* structure controllers. We observe that all the three structures are able to generate control inputs appropriately to achieve

TABLE II: The comparison of various controller structures for the HVAC system.

Controller structure	Avg. Cost	Θ_t
Decentralized	651.648	$\begin{pmatrix} -0.091 & 0 & 0 & 0 \\ 0 & 0.180 & 0 & \\ 0 & 0 & 0.385 & 0 \\ 0 & 0 & 0 & 1.102 \\ 14.356 & 8.557 & 4.095 & -11.524 \end{pmatrix}$
Partially decentralized	648.654	$\begin{pmatrix} 0.454 & 0 & 0 & -0.675 \\ -1.005 & 0.179 & 0 & 0 \\ 0 & -0.056 & 0.265 & 0 \\ 0 & 0 & -0.406 & 0.72 \\ 24.368 & 9.756 & 15.504 & 11.498 \end{pmatrix}$
Centralized	645.443	$\begin{pmatrix} 0.582 & -0.876 & -0.878 & -0.006 \\ -0.876 & 0.583 & 0.002 & -0.881 \\ -0.875 & 0.003 & 0.583 & -0.877 \\ 0.004 & -0.875 & -0.874 & 0.588 \\ 37.644 & 37.626 & 37.69 & 37.904 \end{pmatrix}$



(a) Control input



(b) RMSE of temperature

Fig. 2: Evolution of control inputs and RMSE of temperature in the four-zone HVAC system with different controller structures inferred by the PIIC algorithm.

the desired temperature. The average cost for 250 Monte Carlo simulations and the inferred control parameter Θ are given in Table II. The controller with the *centralized* structure incurs the lowest cost whereas the controller with the *decentralized* structure incurs the highest cost. The *decentralized* structure takes the larges number of time steps to reach the desired temperature whereas the *centralized* structure takes the least number of time steps. A similar trend was observed in the RMSE values of the temperatures, and the computational time required by each controller structure. The *partially decentralized* structure that encodes the dependence of zones performs better than the *decentralized* structure overall and is found to be comparable to the *centralized* structure. The lower performance of the *decentralized* structure is attributed to the lack of information resulting from the decoupling of the states.

V. CONCLUSIONS AND FUTURE WORK

We propose the PIIC algorithm which solves the SOC problem as a parameter inference problem. This algorithm allows the use of both linear and nonlinear basis functions to parameterize the controller, which enables us to infer nonlinear optimal feedback control laws for nonlinear systems in a model-based setting. This was a major limitation of the existing inference approaches that prescribe a linear feedback control law for a nonlinear system which yield sub-optimal trajectories. Two simulation examples demonstrate the effectiveness of the proposed algorithm. Future work includes extension of the formulation to encode safety constraints and investigation of the PIIC algorithm for multi-agent applications.

REFERENCES

[1] R. F. Stengel, *Optimal control and estimation*. Courier Corporation, 1994.

[2] E. Todorov, "Optimal control theory," *Bayesian brain: probabilistic approaches to neural coding*, pp. 268–298, 2006.

[3] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005, pp. 300–306.

[4] M. Toussaint, "Robot trajectory optimization using approximate inference," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 1049–1056.

[5] K. Rawlik, M. Toussaint, and S. Vijayakumar, "An approximate inference approach to temporal optimization in optimal control," *Advances in neural information processing systems*, vol. 23, 2010.

[6] E. A. Rückert and G. Neumann, "Stochastic optimal control methods for investigating the power of morphological computation," *Artificial Life*, vol. 19, no. 1, pp. 115–131, 2013.

[7] J. Watson, H. Abdulsamad, R. Findeisen, and J. Peters, "Efficient stochastic optimal control through approximate Bayesian input inference," 2021. [Online]. Available: https://arxiv.org/abs/2105.07693

[8] J. Watson, H. Abdulsamad, and J. Peters, "Stochastic optimal control as approximate input inference," in *Proceedings of the Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, L. P. Kaelbling, D. Kragic, and K. Sugiura, Eds., vol. 100. PMLR, 30 Oct–01 Nov 2020, pp. 697–716. [Online]. Available: https://proceedings.mlr.press/v100/watson20a.html

[9] K. Rawlik, M. Toussaint, and S. Vijayakumar, "On stochastic optimal control and reinforcement learning by approximate inference," *Proceedings of Robotics: Science and Systems VIII*, 2012.

[10] S. Levine, "Reinforcement learning and control as probabilistic inference: Tutorial and review," *arXiv preprint arXiv:1805.00909*, 2018.

[11] E. Noorani and J. S. Baras, "A probabilistic perspective on risk-sensitive reinforcement learning," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 2697–2702.

[12] I. Askari, B. Badnava, T. Woodruff, S. Zeng, and H. Fang, "Sampling-based nonlinear MPC of neural network dynamics with application to autonomous vehicle motion planning," *arXiv preprint arXiv:2205.04506*, 2022.

[13] H. Itoh, Y. Sakai, T. Kadoya, H. Fukumoto, H. Wakuya, and T. Furukawa, "Using model uncertainty for robust optimization in approximate inference control," *Artificial Life and Robotics*, vol. 22, no. 3, pp. 327–335, 2017.

[14] E. Rueckert, M. Mindt, J. Peters, and G. Neumann, "Robust policy updates for stochastic optimal control," in *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2014, pp. 388–393.

[15] S. J. Julier, "The scaled unscented transformation," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)*, vol. 6. IEEE, 2002, pp. 4555–4559.

[16] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal processing, sensor fusion, and target recognition VI*, vol. 3068. SPIE, 1997, pp. 182–193.

[17] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal processing magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[18] A. Amini, Q. Sun, and N. Motee, "Approximate optimal control design for a class of nonlinear systems by lifting Hamilton-Jacobi-Bellman equation," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2717–2722.

[19] V. Nevistić and J. A. Primbs, "Constrained nonlinear optimal control: a converse HJB approach," 1996.

[20] Y. Li, Y. Tang, R. Zhang, and N. Li, "Distributed reinforcement learning for decentralized linear quadratic control: A derivative-free policy optimization approach," 2019. [Online]. Available: https://arxiv.org/abs/1912.09135