Can Direct Latent Model Learning Solve Linear Quadratic Gaussian Control?

Yi Tian YITIAN@MIT.EDU
Massachusetts Institute of Technology

Kaiqing Zhang KAIQING@UMD.EDU
University of Maryland, College Park

Russ Tedrake RUSST@MIT.EDU
Massachusetts Institute of Technology

Suvrit Sra SUVRIT@MIT.EDU
Massachusetts Institute of Technology

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

We study the task of learning state representations from potentially high-dimensional observations, with the goal of controlling an unknown partially observable system. We pursue a *direct latent model learning* approach, where a dynamic model in some latent state space is learned by predicting quantities directly related to planning (e.g., costs) without reconstructing the observations. In particular, we focus on an intuitive cost-driven state representation learning method for solving Linear Quadratic Gaussian (LQG) control, one of the most fundamental partially observable control problems. As our main results, we establish finite-sample guarantees of finding a near-optimal state representation function and a near-optimal controller using the directly learned latent model. To the best of our knowledge, despite various empirical successes, prior to this work it was unclear if such a cost-driven latent model learner enjoys finite-sample guarantees. Our work underscores the value of predicting multi-step costs, an idea that is key to our theory, and notably also an idea that is known to be empirically valuable for learning state representations.

Keywords: Latent model learning, state representation learning for control, linear quadratic Gaussian (LQG)

1. Introduction

We consider state representation learning for control in partially observable systems, inspired by the recent successes of *control from pixels* (Hafner et al., 2019b,a). Control from pixels is an everyday task for human beings, but it remains challenging for learning agents. Methods to achieve it generally fall into two main categories: *model-free* and *model-based* ones. Model-free methods directly learn a visuomotor policy, also known as direct reinforcement learning (RL) (Sutton and Barto, 2018). On the other hand, model-based methods, also known as indirect RL (Sutton and Barto, 2018), attempt to learn a *latent model* that is a compact representation of the system, and to synthesize a policy in the latent model. Compared with model-free methods, model-based ones facilitate generalization across tasks and enable efficient planning (Hafner et al., 2020), and are sometimes more sample efficient (Tu and Recht, 2019; Sun et al., 2019; Zhang et al., 2019).

This manuscript is a shorter version of the technical report. Please refer to its appendix for the missing details.

In latent model-based control, the state of the latent model is also referred to as a *state representation* in the deep RL literature, and the mapping from an observed history to a latent state is referred to as the (state) representation function. *Reconstructing the observation* often serves as a supervision for representation learning for control in the empirical RL literature (Hafner et al., 2019b,a, 2020; Fu et al., 2021; Wang et al., 2022). This is in sharp contrast to model-free methods, where the policy improvement step is completely cost-driven. Reconstructing observations provides a powerful supervision signal for learning a task-agnostic world model, but they are high-dimensional and noisy, so the reconstruction requires an expressive reconstruction function; latent states learned by reconstruction contain irrelevant information for control, which can distract RL algorithms (Zhang et al., 2020; Fu et al., 2021; Wang et al., 2022). This is especially the case for practical visuomotor control tasks, e.g., robotic manipulation and self-driving cars, where the visual images contain predominately task-irrelevant objects and backgrounds.

Various empirical attempts (Schrittwieser et al., 2020; Zhang et al., 2020; Okada and Taniguchi, 2021; Deng et al., 2021; Yang et al., 2022) have been made to bypass observation reconstruction. Apart from observation, the interaction involves two other variables: actions (control inputs) and costs. Inverse model methods (Lamb et al., 2022) reconstruct actions; while other methods rely on costs. We argue that since neither the reconstruction function nor the inverse model is used for policy learning, cost-driven state representation learning is the most direct one. In this paper, we aim to examine the soundness of this methodology in linear quadratic Gaussian (LQG) control, one of the most fundamental partially observable control models.

Parallel to the empirical advances of learning for control from pixels, partially observable linear systems has been extensively studied in the context of learning for dynamic control (Oymak and Ozay, 2019; Simchowitz et al., 2020; Lale et al., 2020, 2021; Zheng et al., 2021; Minasyan et al., 2021; Umenberger et al., 2022). In this context, the representation function is more formally referred to as a *filter*, the optimal one being the Kalman filter. Most existing *model-based* learning approaches for LQG control focus on the linear time-invariant (LTI) case, and are based on the idea of *learning Markov parameters* (Ljung, 1998), the mapping from control inputs to observations. Hence, they need to reconstruct observations by definition. Motivated by the empirical successes in control from pixels, we take a different, cost-driven route, in hope of avoiding reconstructing observations or control inputs, which we refer to as *direct latent model learning*.

We focus on finite-horizon time-varying LQG control and address the following question:

Can direct latent model learning provably solve LQG control?

This work answers the question in the affirmative. Below is an overview of the main results. Additional discussion of related work is deferred to Appendix A in the technical report.

1.1. Overview of main results

Motivated by empirical works on state representation learning for control (Schrittwieser et al., 2020; Zhang et al., 2020) and approximate information states (Subramanian et al., 2020; Yang et al., 2022), we propose a direct model learning method (Algorithm 1), without reconstructing observations or using an inverse model (Mhammedi et al., 2020; Frandsen et al., 2022; Lamb et al., 2022), that has the guarantee informally stated in Theorem 1. In Theorem 1 below, the dependence on dimensions and other system parameters are polynomial.

Theorem 1 (Informal) *Given an unknown time-varying LQG control problem with horizon T, under standard assumptions including stability, controllability (within in ℓ steps) and cost observabil-*

ity, there exists a direct latent model learning algorithm that returns, from n collected trajectories, a state representation function and a controller such that for the LQG control problem

- *at the first ℓ steps, the state representation function is $\mathcal{O}(\ell^{1/2}n^{-1/4})$-optimal and the controller is $\mathcal{O}(\mathcal{O}(1)^{\ell}\ell n^{-1/4})$-optimal;*
- *at the next $T-\ell$ steps, the state representation function is $\mathcal{O}(T^{3/2}n^{-1/2})$-optimal and the controller is $\mathcal{O}(T^4 n^{-1})$-optimal.*

Our method parameterizes the state representation function and the latent system (transition and cost functions) separately. Usually, in empirical works, the state representation and transition functions are jointly learned, and they are, in fact, composed in transition prediction. An interesting finding is that in LQG, the scalar cost is sufficiently informative such that using cumulative cost supervision alone can recover the state representation function. Hence, the representation function and the latent system can be learned sequentially: our method first learns the representation function by predicting *cumulative scalar cost* (Algorithm 2), and then fits the transition and cost functions by minimizing the *transition and cost prediction* errors in the latent space. The learned latent model then enables planning that leads to a near-optimal controller.

Challenges & our techniques. Overall, to establish finite-sample guarantees, a major technical challenge is to deal with the *quadratic regression* problem in cost prediction, arising from the inherent quadratic form of the cost function in LQG. Directly solving the problem for the representation function involves *quartic* optimization; instead, we propose to solve a quadratic regression problem, followed by low-rank approximate factorization. The quadratic regression problem also appears in identifying the cost matrices, which involves concentration for random variables that are fourth powers of Gaussians. We believe these techniques might be of independent interest.

Moreover, the first ℓ-step *latent* states may not be adequately *excited* (having full-rank covariance), which invalidates the use of most system identification techniques. We instead identify only *relevant directions* of the system parameters, and prove that this is sufficient for learning a near-optimal controller by analyzing state covariance mismatch. This fact is reflected in the separation in the statement of Theorem 1; developing finite-sample analysis in this case is technically challenging.

Implications. For practitioners, one takeaway from our work is the benefit of predicting *multi-step cumulative* costs in direct latent model learning. Whereas cost at a single time step may not be revealing enough of the latent state, cumulative cost across multiple steps can be. This idea has been previously used by MuZero (Schrittwieser et al., 2020) in state representation learning for control, and our work can be viewed as a formal understanding of it in the LQG setting.

Notation. Random vectors are denoted by lowercase letters; sometimes they also denote their realized values. Uppercase letters denote matrices, some of which can be random. 0 can denote the scalar zero, zero vector or zero matrix; 1 denotes either the scalar one or a vector consisting of all ones; I denotes an identity matrix. The dimension, when emphasized, is specified in subscripts, e.g., $0_{d_x \times d_x}, 1_{d_x}, I_{d_x}$. Let $a \wedge b$ denote the minimum between scalars a and b. Given vector $v \in \mathbb{R}^d$, $\|v\|$ denotes its ℓ_2-norm. For $P \succcurlyeq 0$, $\|v\|_P := (v^\top P v)^{1/2}$. Semicolon ";" denotes stacking vectors or matrices vertically. For a collection of d-dimensional vectors $(v_t)_{t=i}^{j}$, let $v_{i:j} := [v_i; v_{i+1}; \ldots; v_j] \in R^{d(j-i+1)}$ denote the concatenation along the column. For random variable η, let $\|\eta\|_{\psi_\beta}$ denote its β-sub-Weibull norm, a special case of Orlicz norms (Zhang and Wei, 2022), with $\beta = 1, 2$ corresponding to subexponential and sub-Gaussian norms. $\sigma_i(A), \sigma_{\min}(A), \sigma^{+}_{\min}(A), \sigma_{\max}(A)$ denote its ith largest, minimum, minimum positive, maximum singular values, respectively. $\|A\|_2, \|A\|_F, \|A\|_*$ denote the operator, Frobenius, nuclear

norms of matrix A, respectively. $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product between matrices. $\mathrm{vec}(\cdot)$ and $\mathrm{svec}(\cdot)$ denote flattening a matrix and a symmetric matrix by stacking their columns; $\mathrm{svec}(\cdot)$ does not repeat the off-diagonal elements, but scales them by $\sqrt{2}$ (Schacke, 2004).

2. Problem setup

We study partially observable linear time-varying (LTV) dynamical system

$$x_{t+1} = A_t^* x_t + B_t^* u_t + w_t, \qquad y_t = C_t^* x_t + v_t, \qquad t = 0, 1, \ldots, T-1, \tag{2.1}$$

and $y_T = C_T^* x_T + v_T$. For all $t \geq 0$, we have the notation of state $x_t \in \mathbb{R}^{d_x}$, observation $y_t \in \mathbb{R}^{d_y}$, and control $u_t \in \mathbb{R}^{d_u}$. $(w_t)_{t=0}^{T-1}$ are i.i.d. process noises, sampled from $\mathcal{N}(0, \Sigma_{w_t})$, $(v_t)_{t=0}^{T}$ are i.i.d. observation noises, sampled from $\mathcal{N}(0, \Sigma_{v_t})$. Let initial state x_0 be sampled from $\mathcal{N}(0, \Sigma_0)$.

Let $\Phi_{t,t_0} = A_{t-1}^* A_{t-2}^* \cdots A_{t_0}^*$ for $t > t_0$ and $\Phi_{t,t} = I$. Then $x_t = \Phi_{t,t_0} x_{t_0} + \sum_{\tau=t_0}^{t-1} \Phi_{t,\tau+1} w_\tau$ under zero control input. To ensure the state and the cumulative noise do not grow with time, we make the following uniform exponential stability assumption.

Assumption 1 (Uniform exponential stability) *The system is uniformly exponentially stable. That is, there exists $\alpha > 0, \rho \in (0, 1)$ such that for any $0 \leq t_0 < t \leq T$, $\|\Phi_{t,t_0}\|_2 \leq \alpha \rho^{t-t_0}$.*

Assumption 1 is standard in controlling LTV systems (Zhou and Zhao, 2017; Minasyan et al., 2021), satisfied by a stable LTI system. It essentially says that zero control is a stabilizing policy, and can be relaxed to a given stabilizing policy. Potentially, it can even be relaxed to uniform exponential stabilizability, by using our method for one more step at a time and finding a stabilizing policy incrementally.

Define the ℓ-step controllability matrix

$$\Phi_{t,\ell}^c := [B_t^*, A_t^* B_{t-1}^*, \ldots, A_t^* A_{t-1}^* \cdots A_{t-\ell+2}^* B_{t-\ell+1}^*] \in \mathbb{R}^{d_x \times \ell d_u}$$

for $\ell - 1 \leq t \leq T - 1$, which reduces to the standard controllability matrix $[B, \ldots, A^{\ell-1}B]$ in the LTI setting. We make the following controllability assumption.

Assumption 2 (Controllability) *For all $\ell - 1 \leq t \leq T - 1$, $\mathrm{rank}(\Phi_{t,\ell}^c) = d_x$, $\sigma_{\min}(\Phi_{t,\ell}^c) \geq \nu > 0$.*

Under zero noise, $x_{t+\ell} = \Phi_{t+\ell,t} x_t + \Phi_{t+\ell-1,\ell}^c [u_{t+\ell-1}; \ldots; u_t]$, so Assumption 2 ensures that from any state x, there exist control inputs that drive the state to 0 in ℓ steps, and ν ensures that the equation leading to them is well conditioned. We do not assume controllability for $0 \leq t < \ell - 1$, since we do not want to impose the constraint that $d_u > d_x$. This turns out to present a significant challenge for latent model learning, as seen from the separation of the results before and after the ℓ-steps in Theorem 1.

The quadratic cost functions are given by

$$c_t(x, u) = \|x\|_{Q_t^*}^2 + \|u\|_{R_t^*}^2, \quad 0 \leq t \leq T - 1, \quad c_T(x) = \|x\|_{Q_T^*}^2,$$

for positive semidefinite matrices $(Q_t^*)_{t=0}^T$ and positive definite matrices $(R_t^*)_{t=0}^{T-1}$. Sometimes the cost is defined as a function on observation y. Since the quadratic form $y^\top Q_t^* y = x^\top (C_t^*)^\top Q_t^* C_t^* x$, our analysis still applies if the assumptions on $(Q_t^*)_{t=0}^T$ hold for $((C_t^*)^\top Q_t^* C_t^*)_{t=0}^T$ instead.

(A, C) and $(A, Q^{1/2})$ observabilities are standard assumptions in controlling LTI systems. To differentiate from the former, we call the latter cost observability, since it implies the states are observable through costs. Whereas Markov parameter based approaches need to assume (A, C) observability to identify the system, our cost driven approach does not. Robust control sometimes assumes $(A, Q^{1/2})$ observability with vector cost $Q^{1/2}x$. Here we deal with the more difficult problem of having only the scalar cost. Nevertheless, the notion of cost observability is still important for our approach, formally defined as follows.

Assumption 3 (Cost observability) *For all $0 \le t \le \ell - 1$, $Q_t^* \succcurlyeq \mu^2 I$. For all $\ell \le t \le T$, there exists $m > 0$ such that the cost observability Gramian (Kailath, 1980)*

$$\sum_{\tau=t}^{t+k-1} \Phi_{\tau,t}^\top Q_\tau^* \Phi_{\tau,t} = Q_t^* + (A_t^*)^\top Q_{t+1}^* A_t^* + \ldots + (A_{t+k-2}^* \cdots A_t^*)^\top Q_{t+k-1}^* A_{t+k-2}^* \cdots A_t^* \succcurlyeq \mu^2 I,$$

where $k = m \wedge (T - t + 1)$.

This assumption ensures that without noises, if we start with a nonzero state, the cumulative cost becomes positive in m steps. The special requirement for $0 \le t \le \ell - 1$ results from the difficulty in lacking controllability. The following is a regularity assumption.

Assumption 4 *$(\sigma_{\min}(\Sigma_{v_t}))_{t=0}^T$ are uniformly lower bounded by $\sigma_v > 0$. The operator norms of all matrices in the problem definition are uniformly upper bounded, including $(A_t^*, B_t^*, R_t^*, \Sigma_{w_t})_{t=0}^{T-1}$, $(C_t^*, Q_t^*, \Sigma_{v_t})_{t=0}^T$. In other words, they are all $\mathcal{O}(1)$.*

Let $h_t := [y_{0:t}; u_{0:(t-1)}] \in \mathbb{R}^{(t+1)d_y + td_u}$ denote the available history before deciding control u_t. A policy $\pi = (\pi_t : h_t \mapsto u_t)_{t=0}^{T-1}$ determines at time t a control input u_t based on history h_t. With a slight abuse of notation, let $c_t := c_t(x_t, u_t)$ for $0 \le t \le T - 1$ and $c_T := c_T(x_T)$ denote the cost at each time step. Then, $J^\pi := \mathbb{E}^\pi[\sum_{t=0}^T c_t]$ is the expected cumulative cost under policy π, where the expectation is taken over the randomness in the process noises, observation noises and controls. The objective of LQG control is to find a policy π such that J^π is minimized.

If the system parameters $((A_t^*, B_t^*, R_t^*)_{t=0}^{T-1}, (C_t^*, Q_t^*)_{t=0}^T)$ are known, the optimal control is obtained by combining the Kalman filter

$$z_0^* = L_0^* y_0, \quad z_{t+1}^* = A_t^* z_t^* + B_t^* u_t + L_{t+1}^*(y_{t+1} - C_{t+1}^*(A_t^* z_t^* + B_t^* u_t)), \quad 0 \le t \le T - 1,$$

with the optimal feedback control gains of the linear quadratic regulator (LQR) $(K_t^*)_{t=0}^{T-1}$, where $(L_t^*)_{t=0}^T$ are the Kalman gains; this is known as the *separation principle*. The Kalman gains and optimal feedback control gains are given by

$$L_t^* = S_t^*(C_t^*)^\top (C_t^* S_t^* (C_t^*)^\top + \Sigma_{v_t})^{-1}, \quad K_t^* = -((B_t^*)^\top P_{t+1}^* B_t^* + R_t)^{-1}(B_t^*)^\top P_{t+1}^* A_t^*,$$

where S_t^* and P_t^* are determined by their corresponding Riccati difference equations (RDEs):

$$S_{t+1}^* = A_t^*(S_t^* - S_t^*(C_t^*)^\top (C_t^* S_t^* (C_t^*)^\top + \Sigma_{v_t})^{-1} C_t^* S_t^*)(A_t^*)^\top + \Sigma_{w_t}, \quad S_0^* = \Sigma_0, \tag{2.2}$$

$$P_t^* = (A_t^*)^\top (P_{t+1}^* - P_{t+1}^* B_t^* ((B_t^*)^\top P_{t+1}^* B_t^* + R_t^*)^{-1}(B_t^*)^\top P_{t+1}^*) A_t^* + Q_t^*, \quad P_T^* = Q_T^*. \tag{2.3}$$

We consider data-driven control in an unknown LQG control problem with unknown cost matrices $(Q_t^*)_{t=0}^T$. For simplicity, we assume $(R_t^*)_{t=0}^T$ are known, though our approaches can be readily generalized to the case without knowing them; it suffices to identify them in (3.3).

2.1. Latent model of LQG

Under the Kalman filter, the observation prediction error $i_{t+1} := y_{t+1} - C^*_{t+1}(A^*_t z^*_t + B^*_t u_t)$ is called an *innovation*. It is known that i_t is independent of history h_t and $(i_t)_{t=0}^T$ are independent (Bertsekas, 2012). Now we are ready to present the following proposition that represents the system in terms of the state estimates by the Kalman filter, which we shall refer to as the *latent model*.

Proposition 2 *Let $(z^*_t)_{t=0}^T$ be state estimates given by the Kalman filter. Then,*

$$z^*_{t+1} = A^*_t z^*_t + B^*_t u_t + L^*_{t+1} i_{t+1},$$

where $L^*_{t+1} i_{t+1}$ is independent of z^*_t and u_t, i.e., the state estimates follow the same linear dynamics with noises $L^*_{t+1} i_{t+1}$. The cost at step t can be reformulated as functions of the state estimates by

$$c_t = \|z^*_t\|^2_{Q^*_t} + \|u_t\|^2_{R^*_t} + b_t + \gamma_t + \eta_t,$$

where $b_t > 0$ is a constant, and $\gamma_t = \|z^*_t - x_t\|^2_{Q^*_t} - b_t$, $\eta_t = \langle z^*_t, x_t - z^*_t \rangle_{Q^*_t}$ are both zero-mean subexponential random variables.

Proposition 2 states that 1) the dynamics of the state estimates produced by the Kalman filter remains the same as the original system up to noises, determined by $(A^*_t, B^*_t)_{t=0}^{T-1}$; 2) the costs are still determined by $(Q^*_t)_{t=0}^T$ and $(R^*_t)_{t=0}^{T-1}$, up to constants and noises. Hence, a latent model can be parameterized by $((A_t, B_t)_{t=0}^{T-1}, (Q_t)_{t=0}^T)$ (recall that we assume $(R^*_t)_{t=0}^T$ is known for convenience). Note that observation matrices $(C^*_t)_{t=0}^T$ are *not* involved.

Now let us take a closer look at the state representation function. The Kalman filter can be written as $z^*_{t+1} = \overline{A}^*_t z^*_t + \overline{B}^*_t u_t + L^*_{t+1} y_{t+1}$, where $\overline{A}^*_t = (I - L^*_{t+1} C^*_{t+1}) A^*_t$ and $\overline{B}^*_t = (I - L^*_{t+1} C^*_{t+1}) B^*_t$. For $0 \le t \le T$, unrolling the recursion gives

$$\begin{aligned} z^*_t &= \overline{A}^*_{t-1} z^*_{t-1} + \overline{B}^*_{t-1} u_{t-1} + L^*_t y_t \\ &= [\overline{A}^*_{t-1}\overline{A}^*_{t-2}\cdots\overline{A}^*_0 L^*_0, \ldots, L^*_t][y_0; \ldots; y_t] + [\overline{A}^*_{t-1}\overline{A}^*_{t-2}\cdots\overline{A}^*_1\overline{B}^*_0, \ldots, \overline{B}^*_{t-1}][u_0; \ldots; u_{t-1}] \\ &=: M^*_t[y_{0:t}; u_{0:(t-1)}], \end{aligned}$$

where $M^*_t \in \mathbb{R}^{d_x \times ((t+1)d_y + t d_u)}$. This means the optimal state representation function is linear in the history of observations and controls. A state representation function can then be parameterized by matrices $(M_t)_{t=0}^T$, and the latent state at step t is given by $z_t = M_t h_t$.

Overall, a policy π is a combination of state representation function $(M_t)_{t=0}^{T-1}$ (M_T is not needed) and feedback gain $(K_t)_{t=0}^{T-1}$ in the latent model; in this case, we write $\pi = (M_t, K_t)_{t=0}^{T-1}$. This contrasts with the disturbance-based parameterization (Youla et al., 1976; Wang et al., 2019; Sadraddini and Tedrake, 2020; Simchowitz et al., 2020; Lale et al., 2020).

3. Methodology: direct latent model learning

State representation learning involves history data that contain samples of three variables: observation, control input, and cost. Each of these can potentially be used as a *supervision* signal, and defines a type of state representation learning algorithms. We summarize our categorization as follows.

- Predicting observations defines the class of *observation-reconstruction based* methods, including methods based on Markov parameters (mapping from controls to observations) in linear systems (Lale et al., 2021; Zheng et al., 2021) and methods that learn a mapping from states to observations in more complex systems (Ha and Schmidhuber, 2018; Hafner et al., 2019b,a). This type of method tends to recover all state components.

Algorithm 1 Direct latent model learning for LQG control

1: **Input:** sample size n, input noise magnitude σ_u, singular value threshold $\theta = \Theta(n^{-1/4})$
2: Collect n trajectories using $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$, for $0 \le t \le T-1$, to obtain data in the form of

$$\mathcal{D}_{\text{raw}} = (y_0^{(i)}, u_0^{(i)}, c_0^{(i)}, \ldots, y_{T-1}^{(i)}, u_{T-1}^{(i)}, c_{T-1}^{(i)}, y_T^{(i)}, c_T^{(i)})_{i=1}^n$$

3: Run CoREL($\mathcal{D}_{\text{raw}}$, θ) (Algorithm 2) to obtain state representation function estimate $(\widehat{M}_t)_{t=0}^T$ and latent state estimates $(z_t^{(i)})_{t=0,i=1}^{T,n}$, so that the data are converted to

$$\mathcal{D}_{\text{state}} = (z_0^{(i)}, u_0^{(i)}, c_0^{(i)}, \ldots, z_{T-1}^{(i)}, u_{T-1}^{(i)}, c_{T-1}^{(i)}, z_T^{(i)}, c_T^{(i)})_{i=1}^n$$

4: Run SysID($\mathcal{D}_{\text{state}}$) to obtain system parameter estimates $((\widehat{A}_t, \widehat{B}_t)_{t=0}^{T-1}, (\widehat{Q}_t)_{t=0}^T)$
5: Find feedback gains $(\widehat{K}_t)_{t=0}^{T-1}$ from $((\widehat{A}_t, \widehat{B}_t, R_t^*)_{t=0}^{T-1}, (\widehat{Q}_t)_{t=0}^T)$ by RDE (2.3)
6: **Return:** policy $\hat{\pi} = (\widehat{M}_t, \widehat{K}_t)_{t=0}^{T-1}$

- Predicting controls defines the class of *inverse model* methods, where the control is predicted from states across different time steps (Mhammedi et al., 2020; Frandsen et al., 2022; Lamb et al., 2022). This type of method tends to recover the controllable state components.
- Predicting (cumulative) costs defines the class of *cost-driven latent model learning* methods (Zhang et al., 2020; Schrittwieser et al., 2020; Yang et al., 2022). This type of method tends to recover the state components relevant to the cost.

Our method falls into the cost-driven category, which is more direct than the other two types, in the sense that the cost is directly relevant to planning with a dynamic model. Another reason why we call our method *direct latent model learning* is that compared with Markov parameter-based approaches for linear systems, our approach directly parameterizes the state representation function, without exploiting the structure of the Kalman filter, making our approach closer to empirical practice that was designed for general RL settings.

(Subramanian et al., 2020) proposes to optimize a simple combination of cost and transition prediction errors to learn a latent model. That is, we parameterize a state representation function by matrices $(M_t)_{t=0}^T$ and a latent model by matrices $((A_t, B_t)_{t=0}^{T-1}, (Q_t)_{t=0}^T)$ and then solve

$$\min_{(M_t,Q_t,b_t)_{t=0}^T,(A_t,B_t)_{t=0}^{T-1}} \sum_{t=0}^T \sum_{i=1}^n l_t^{(i)}, \tag{3.1}$$

where $(b_t)_{t=0}^T$ are additional scalar parameters to account for noises, and the loss at step t for trajectory i is defined by

$$l_t^{(i)} = \left(\|M_t h_t^{(i)}\|_{Q_t}^2 + \|u_t^{(i)}\|_{R_t^*}^2 + b_t - c_t^{(i)}\right)^2 + \left(M_{t+1} h_{t+1}^{(i)} - A_t M_t h_t^{(i)} - B_t u_t^{(i)}\right)^2, \tag{3.2}$$

for $0 \le t \le T-1$ and $l_T^{(i)} = \left(\|M_T h_T^{(i)}\|_{Q_T}^2 + b_T - c_T^{(i)}\right)^2$. The optimization problem (3.1) is nonconvex; even if we find the global minimum solution, it is unclear how to establish finite-sample guarantees for it. A main finding of this work is that for LQG, we can solve the cost and transition loss optimization problems sequentially, with the caveat of using cumulative costs.

Our method is summarized in Algorithm 1. It has three steps: cost-driven state representation function learning (CoREL, Algorithm 2), latent system identification (SysID), and planning by

Algorithm 2 CoREL: cost driven state representation learning

1: **Input:** raw data $\mathcal{D}_{\text{raw}}$, singular value threshold $\theta = \Theta((\ell(d_y + d_u))^{1/2} d_x^{3/4} n^{-1/4})$
2: Estimate the state representation function and cost constants by solving

$$(\widehat{N}_t, \widehat{b}_t)_{t=0}^{T} \in \underset{(N_t = N_t^\top, b_t)_{t=0}^{T}}{\operatorname{argmin}} \sum_{t=0}^{T} \sum_{i=1}^{n} (\left\| [y_{0:t}^{(i)}; u_{0:(t-1)}^{(i)}] \right\|_{N_t}^2 + \sum_{\tau=t}^{t+k-1} \|u_\tau^{(i)}\|_{R_\tau^*}^2 + b_t - \bar{c}_t^{(i)})^2, \tag{3.3}$$

where $k = 1$ for $0 \le t \le l-1$ and $k = m \wedge (T - t + 1)$ for $\ell \le t \le T$
3: Find $\widetilde{M}_t \in \mathbb{R}^{d_x \times ((t+1)d_y + t d_u)}$ such that $\widetilde{M}_t^\top \widetilde{M}_t$ is an approximation of $\widehat{N}_t$
4: For all $0 \le t \le \ell - 1$, set $\widehat{M}_t = \textsc{TruncSV}(\widetilde{M}_t, \theta)$; for all $\ell \le t \le T$, set $\widehat{M}_t = \widetilde{M}_t$
5: Compute $\widehat{z}_t^{(i)} = \widehat{M}_t [y_{0:t}^{(i)}; u_{0:t}^{(i)}]$ for all $t = 0, \ldots, T$ and $i = 1, \ldots, n$
6: **Return:** state representation estimate $(\widehat{M}_t)_{t=0}^{T}$ and latent state estimates $(\widehat{z}_t^{(i)})_{t=0,i=1}^{T,n}$

RDE (2.3). This three-step approach is very similar to World Models (Ha and Schmidhuber, 2018) used in empirical RL, except that in the first step, instead of using an autoencoder to learn the state representation function, we use cost values to supervise the representation learning. Most empirical state representation learning methods (Hafner et al., 2019b,a; Schrittwieser et al., 2020) use cost supervision as one loss term; the special structure of LQG allows us to use it alone and have theoretical guarantees.

CoREL (Algorithm 2) is the core of our algorithm. Once the state representation function $(\widehat{M}_t)_{t=0}^{T}$ is obtained, SysID identifies the latent system using ordinary linear and quadratic regression, followed by planning using RDE (2.3) to obtain controller $(\widehat{K}_t)_{t=0}^{T-1}$ from $((\widehat{A}_t, \widehat{B}_t, R_t^*)_{t=0}^{T-1}, (\widehat{Q}_t)_{t=0}^{T})$. SysID consists of the standard regression procedures; the full algorithmic detail is deferred to Appendix E in the technical report. Below we explain the cost-driven state representation learning algorithm (CoREL, Algorithm 2) in detail.

3.1. Learning the state representation function

The state representation function is learned via CoREL (Algorithm 2). Given the raw data consisting of n trajectories, CoREL first solves the regression problem (3.3) to recover the symmetric matrix $\widehat{N}_t$. The target $\bar{c}_t$ of regression (3.3) is defined by

$$\bar{c}_t := c_t + c_{t+1} + \ldots + c_{t+k-1},$$

where $k = 1$ for $0 \le t \le \ell - 1$ and $k = m \wedge (T - t + 1)$ for $\ell \le t \le T$. The superscript in $\bar{c}_t^{(i)}$ denotes the observed $\bar{c}_t$ in the ith trajectory. The quadratic regression has a closed-form solution, by converting it to linear regression using $\|v\|_P^2 = \langle vv^\top, P \rangle_F = \langle \text{svec}(vv^\top), \text{svec}(P) \rangle$.

Why cumulative cost? The state representation function is parameterized by $(M_t)_{t=0}^{T}$ and the latent state at step t is given by $z_t = M_t h_t$. The single-step cost prediction (neglecting control cost $\|u_t\|_{R_t^*}^2$ and constant b_t) is given by $\|z_t\|_{Q_t}^2 = h_t^\top M_t^\top Q_t M_t h_t$. The regression recovers $(M_t^*)^\top Q_t^* M_t^*$ as a whole, from which we can recover $(Q_t^*)^{1/2} M_t^*$ up to an orthonormal transform. If Q_t^* is positive definite and known, then we can further recover M_t^* from it. However, if Q_t^* does not have full rank, information about M_t^* is partially lost, and there is no way

to fully recover M_t^* even if Q_t^* is known. To see why multi-step cumulative cost helps, define $\overline{Q}_t^* := \sum_{\tau=t}^{t+k-1} \Phi_{\tau,t}^\top Q_\tau^* \Phi_{\tau,t}$ for the same k above. Under zero control and zero noise, starting from x_t at step t, the k-step cumulative cost is precisely $\|x_t\|_{\overline{Q}_t^*}^2$. Under the cost observability assumption (Assumption 3), $(\overline{Q}_t^*)_{t=0}^T$ are positive definite.

The normalized parameterization. Still, since $\overline{Q}_t^*$ is unknown, even if we recover $(M_t^*)^\top \overline{Q}_t^* M_t^*$ as a whole, it is not viable to extract M_t^* and $\overline{Q}_t^*$. Such ambiguity is unavoidable; in fact, for every $\overline{Q}_t^*$ we choose, there is an equivalent parameterization of the system such that the system response is exactly the same. In partially observable LTI systems, it is well known that the system parameters can only be recovered up to a similarity transform (Oymak and Ozay, 2019). Since every parameterization is correct, we simply choose $\overline{Q}_t^* = I$, which we refer to as the *normalized parameterization*. Concretely, let us define $x'_t = (\overline{Q}_t^*)^{1/2} x_t$. Then, the new parameterization is given by

$$x'_{t+1} = A_t^{*\prime} x'_t + B_t^{*\prime} u_t + w'_t, \quad y_t = C_t^{*\prime} x'_t + v_t, \quad c'_t(x', u) = \|x'\|_{Q_t^{*\prime}}^2 + \|u\|_{R_t^*}^2,$$

and $c'_T(x') = \|x'\|_{(Q_T^*)'}^2$, where for all $t \geq 0$,

$$A_t^{*\prime} = (\overline{Q}_{t+1}^*)^{1/2} A_t^* (\overline{Q}_t^*)^{-1/2}, \quad B_t^{*\prime} = (\overline{Q}_{t+1}^*)^{1/2} B_t^*, \quad C_t^{*\prime} = C_t^* (\overline{Q}_t^*)^{-1/2},$$
$$w'_t = (\overline{Q}_{t+1}^*)^{1/2} w_t, \quad (Q_t^*)' = (\overline{Q}_t^*)^{-1/2} Q_t^* (\overline{Q}_t^*)^{-1/2}.$$

It is easy to verify that under the normalized parameterization the system satisfies Assumptions 1, 2, 3, and 4, up to a change of some constants in the bounds. Without loss of generality, we assume system (2.1) is in the normalized parameterization; otherwise the recovered state representation function and latent system are with respect to the normalized parameterization.

Low-rank approximate factorization. Regression (3.3) has a closed-form solution; solving it gives $(\widehat{N}_t, \widehat{b}_t)_{t=0}^T$. Constants $(\widehat{b}_t)_{t=0}^T$ account for the state estimation error, and are not part of the state representation function; $d_h \times d_h$ symmetric matrices $(\widehat{N}_t)_{t=0}^T$ are estimates of $(M_t^*)^\top M_t^*$ under the normalized parameterization, where $d_h = (t+1)d_y + td_u$. M_t^* can only be recovered up to an orthonormal transform, since for any orthogonal $S \in \mathbb{R}^{d_x \times d_x}$, $(SM_t^*)^\top SM_t^* = (M_t^*)^\top M_t^*$.

We want to recover $\widetilde{M}_t$ from $\widehat{N}_t$ such that $\widehat{N}_t = \widetilde{M}_t^\top \widetilde{M}_t$. Let $U\Lambda U^\top = \widehat{N}_t$ be its eigenvalue decomposition. Let $\Sigma := \max(\Lambda, 0)$ be the positive semidefinite diagonal matrix containing nonnegative eigenvalues, where "max" applies elementwise. If $d_h \leq d_x$, we can construct $\widetilde{M}_t = [\Sigma^{1/2} U^\top; 0_{(d_x - d_h) \times d_h}]$ by padding zeros. If $d_h > d_x$, however, $\operatorname{rank}(\widehat{N}_t)$ may exceed d_x. Assume that the diagonal elements of Σ are in descending order. Let Σ_{d_x} be the left-top $d_x \times d_x$ block of Σ and U_{d_x} be the left d_x columns of U. By the Eckart-Young-Mirsky theorem, $\widetilde{M}_t = \Sigma_{d_x}^{1/2} U_{d_x}^\top$ is the best approximation among $d_x \times d_h$ matrices in term of the Frobenius norm.

Why singular value truncation in the first ℓ steps? The latent states are used to identify the latent system dynamics, so whether they are sufficiently excited, namely having full-rank covariance, makes a big difference: if not, the system matrices can only be identified partially. Proposition 3 below confirms that the optimal latent state $z_t^* = M_t^* h_t$ indeed have full-rank covariance for $t \geq \ell$.

Proposition 3 *If system (2.1) satisfies Assumptions 2 (controllability) and 4 (regularity), then under control $(u_t)_{t=0}^{T-1}$, where $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$, $\sigma_{\min}(\operatorname{Cov}(z_t^*)) = \Omega(\nu^2)$, M_t^* has rank d_x and $\sigma_{\min}(M_t^*) = \Omega(\nu t^{-1/2})$ for all $\ell \leq t \leq T$.*

Proposition 3 implies that for all $\ell \leq t \leq T$, N_t^* has rank d_x, so if d_x is not provided, this gives a way to discover it. For $\ell \leq t \leq T$, Proposition 3 guarantees that as long as $\widetilde{M}_t$ is close enough

to M_t^*, it also has full rank, and so does $\mathrm{Cov}(\widetilde{M}_t h_t)$. Hence, we simply take the final estimate $\widehat{M}_t = \widetilde{M}_t$. Without further assumptions, however, there is no such guarantee for $(\mathrm{Cov}(z_t^*))_{t=0}^{\ell-1}$ and $(M_t^*)_{t=0}^{\ell-1}$. We make the following minimal assumption to ensure that the minimum positive singular value $(\sigma_{\min}^+(\mathrm{Cov}(z_t^*)))_{t=0}^{\ell-1}$ are uniformly lower bounded.

Assumption 5 *For $0 \le t \le \ell - 1$, $\sigma_{\min}^+(M_t^*) \ge \beta > 0$.*

Still, for $0 \le t \le \ell - 1$, Assumption 5 does not guarantee the rank of $\mathrm{Cov}(\widetilde{M}_t h_t)$, not even its minimum positive singular value; that is why we introduce TRUNCSV that truncates the singular values of $\widetilde{M}_t$ by a threshold $\theta > 0$. Concretely, we take $\widehat{M}_t = (\mathbb{I}_{[\theta,+\infty)}(\Sigma_{d_x}^{1/2}) \odot \Sigma_{d_x}^{1/2}) U_{d_x}^\top$, where the indicator function $\mathbb{I}$ applies elementwise and $\odot$ denotes the Hadamard product. Then, $\widehat{M}_t$ has the same singular values as $\widetilde{M}_t$ except that those below θ are zeroed. We take $\theta = \Theta((\ell(d_y + d_u))^{1/2} d_x^{3/4} n^{-1/4})$ to ensure a sufficient lower bound on the minimum positive singular value of $\widehat{M}_t$ while not increasing the statistical errors.

4. Theoretical guarantees

Theorem 4 below offers finite-sample guarantees for our approach. Overall, it confirms direct latent model learning (Algorithm 1) as a viable path to solving LQG control.

Theorem 4 *Given an unknown LQG control problem, under Assumptions 1, 2, 3, 4 and 5, if we run Algorithm 1 with $n \ge \mathrm{poly}(T, d_x, d_y, d_u, \log(1/p))$, then with probability at least $1 - p$, state representation function $(\widehat{M}_t)_{t=0}^T$ is $\mathrm{poly}(\ell, d_x, d_y, d_u) n^{-1/4}$ optimal in the first ℓ steps, and $\mathrm{poly}(\nu^{-1}, T, d_x, d_y, d_u) n^{-1/2}$ optimal in the next $(T - \ell)$ steps. Also, the learned controller $(\widehat{K}_t)_{t=0}^{T-1}$ is $\mathrm{poly}(\ell, \beta^{-1}, m, d_x, d_y, d_u) c^\ell n^{-1/4}$ optimal for some dimension-free constant $c > 0$ depending on system parameters in the first ℓ steps, and $\mathrm{poly}(T, \nu^{-1}, m, d_x, d_y, d_u, \log(1/p)) n^{-1}$ optimal in the last $(T - \ell)$ steps.*

From Theorem 4, we observe a separation of the convergence rates before and after time step ℓ, resulting from the loss of the full-rankness of $(\mathrm{Cov}(z_t^*))_{t=0}^{\ell-1}$ and $(M_t^*)_{t=0}^{\ell-1}$. In more detail, the proof sketch goes as follows. Quadratic regression guarantees that $\widehat{N}_t$ converges to N_t^* at a rate of $n^{-1/2}$ for all $0 \le t \le T$. Before step ℓ, $\widehat{M}_t$ suffers a square root decay of the rate to $n^{-1/4}$ because M_t^* may not have rank d_x. Since $(\widehat{z}_t)_{t=0}^{\ell-1}$ may not have full-rank covariances, $(A_t^*)_{t=0}^{\ell-1}$ are only recovered partially. As a result, $(\widehat{K}_t)_{t=0}^{\ell-1}$ may not stabilize $(A_t^*, B_t^*)_{t=0}^{\ell-1}$, causing the exponential dependence on ℓ. This means if n is not big enough, this controller may be inferior to zero control, since the system $(A_t^*, B_t^*)_{t=0}^{\ell-1}$ is uniformly exponential stable (Assumption 1) and zero control has suboptimality gap linear in ℓ. After step ℓ, $\widehat{M}_t$ retains the $n^{-1/2}$ convergence rate, and so do $(\widehat{A}_t, \widehat{B}_t)$; the certainty equivalent controller then has an order of n^{-1} suboptimality gap for LQ control (Mania et al., 2019). A full proof is deferred to Appendix E in the technical report.

Theorem 4 states the guarantees for the state representation function $(\widehat{M}_t)_{t=0}^T$ and the controller $(\widehat{K}_t)_{t=0}^{T-1}$ separately. One may wonder the suboptimality gap of $\hat{\pi} = (\widehat{M}_t, \widehat{K}_t)_{t=0}^{T-1}$ in combination; after all, this is the output policy. The new challenge is that a suboptimal controller is applied to a suboptimal state estimation. An exact analysis requires more effort, but a reasonable conjecture is that $(\widehat{M}_t, \widehat{K}_t)_{t=0}^{T-1}$ has the same order of suboptimality gap as $(\widehat{K}_t)_{t=0}^{T-1}$: before step ℓ, the extra suboptimality gap resulted from $(\widehat{M}_t)_{t=0}^{\ell-1}$ can be analyzed by considering perturbation $\widehat{K}_t(\widehat{M}_t - M_t^*)h_t$ on controls; after step ℓ, similar to the analysis of the LQG suboptimality gap in (Mania et al., 2019), the overall suboptimality gap can be analyzed by a Taylor expansion of the value function at $(M_t^*, K_t^*)_{t=\ell}^{T-1}$, with $(\widehat{K}_t \widehat{M}_t - K_t^* M_t^*)_{t=\ell}^{T-1}$ being perturbations.

Acknowledgments

YT, SS acknowledge partial support from the NSF BIGDATA grant (number 1741341). KZ's work was mainly done while at MIT, and acknowledges partial support from Simons-Berkeley Research Fellowship. The authors also thank Xiang Fu, Horia Mania, and Alexandre Megretski for helpful discussions, and the reviewers for constructive feedback.

References

Dimitri Bertsekas. *Dynamic Programming and Optimal Control: Volume I*, volume 1. Athena Scientific, 2012.

Fei Deng, Ingook Jang, and Sungjin Ahn. Dreamerpro: Reconstruction-free model-based reinforcement learning with prototypical representations. *arXiv preprint arXiv:2110.14565*, 2021.

Abraham Frandsen, Rong Ge, and Holden Lee. Extracting latent state representations with linear dynamics from rich observations. In *International Conference on Machine Learning*, pages 6705–6725. PMLR, 2022.

Xiang Fu, Ge Yang, Pulkit Agrawal, and Tommi Jaakkola. Learning task informed abstractions. In *International Conference on Machine Learning*, pages 3480–3491. PMLR, 2021.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019b.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Thomas Kailath. *Linear Systems*, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Logarithmic regret bound in partially observable linear dynamical systems. *Advances in Neural Information Processing Systems*, 33:20876–20888, 2020.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Adaptive control and regret minimization in linear quadratic Gaussian (LQG) setting. In *2021 American Control Conference (ACC)*, pages 2517–2522. IEEE, 2021.

Alex Lamb, Riashat Islam, Yonathan Efroni, Aniket Didolkar, Dipendra Misra, Dylan Foster, Lekan Molu, Rajan Chari, Akshay Krishnamurthy, and John Langford. Guaranteed discovery of controllable latent states with multi-step inverse models. *arXiv preprint arXiv:2207.08229*, 2022.

Lennart Ljung. System identification. In *Signal Analysis and Prediction*, pages 163–173. Springer, 1998.

Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalence is efficient for linear quadratic control. *Advances in Neural Information Processing Systems*, 32, 2019.

Zakaria Mhammedi, Dylan J Foster, Max Simchowitz, Dipendra Misra, Wen Sun, Akshay Krishnamurthy, Alexander Rakhlin, and John Langford. Learning the linear quadratic regulator from nonlinear observations. *Advances in Neural Information Processing Systems*, 33:14532–14543, 2020.

Edgar Minasyan, Paula Gradu, Max Simchowitz, and Elad Hazan. Online control of unknown time-varying dynamical systems. *Advances in Neural Information Processing Systems*, 34, 2021.

Masashi Okada and Tadahiro Taniguchi. Dreaming: Model-based reinforcement learning by latent imagination without reconstruction. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4209–4215. IEEE, 2021.

Samet Oymak and Necmiye Ozay. Non-asymptotic identification of LTI systems from a single trajectory. In *2019 American control conference (ACC)*, pages 5655–5661. IEEE, 2019.

Sadra Sadraddini and Russ Tedrake. Robust output feedback control with guaranteed constraint satisfaction. In *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, pages 1–10, 2020.

Kathrin Schacke. On the Kronecker product. *Master's Thesis, University of Waterloo*, 2004.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering Atari, Go, Chess and Shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Max Simchowitz, Karan Singh, and Elad Hazan. Improper learning for non-stochastic control. In *Conference on Learning Theory*, pages 3320–3436. PMLR, 2020.

Jayakumar Subramanian, Amit Sinha, Raihan Seraj, and Aditya Mahajan. Approximate information state for approximate planning and reinforcement learning in partially observed systems. *arXiv preprint arXiv:2010.08843*, 2020.

Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based RL in contextual decision processes: PAC bounds and exponential improvements over model-free approaches. In *Conference on Learning Theory*, pages 2898–2933. PMLR, 2019.

Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

Stephen Tu and Benjamin Recht. The gap between model-based and model-free methods on the linear quadratic regulator: An asymptotic viewpoint. In *Conference on Learning Theory*, pages 3036–3083. PMLR, 2019.

Jack Umenberger, Max Simchowitz, Juan Carlos Perdomo, Kaiqing Zhang, and Russ Tedrake. Globally convergent policy search for output estimation. In *Advances in Neural Information Processing Systems*, 2022.

Tongzhou Wang, Simon S Du, Antonio Torralba, Phillip Isola, Amy Zhang, and Yuandong Tian. Denoised MDPs: Learning world models better than the world itself. *arXiv preprint arXiv:2206.15477*, 2022.

Yuh-Shyang Wang, Nikolai Matni, and John C Doyle. A system-level approach to controller synthesis. *IEEE Transactions on Automatic Control*, 64(10):4079–4093, 2019.

Lujie Yang, Kaiqing Zhang, Alexandre Amice, Yunzhu Li, and Russ Tedrake. Discrete approximate information states in partially observable environments. In *2022 American Control Conference (ACC)*, pages 1406–1413. IEEE, 2022.

Dante Youla, Hamid Jabr, and Jr Bongiorno. Modern wiener-hopf design of optimal controllers–part ii: The multivariable case. *IEEE Transactions on Automatic Control*, 21(3):319–338, 1976.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

Huiming Zhang and Haoyu Wei. Sharper sub-weibull concentrations. *Mathematics*, 10(13):2252, 2022.

Marvin Zhang, Sharad Vikram, Laura Smith, Pieter Abbeel, Matthew Johnson, and Sergey Levine. Solar: Deep structured representations for model-based reinforcement learning. In *International Conference on Machine Learning*, pages 7444–7453. PMLR, 2019.

Yang Zheng, Luca Furieri, Maryam Kamgarpour, and Na Li. Sample complexity of linear quadratic Gaussian (LQG) control for output feedback systems. In *Learning for Dynamics and Control*, pages 559–570. PMLR, 2021.

Bin Zhou and Tianrui Zhao. On asymptotic stability of discrete-time linear time-varying systems. *IEEE Transactions on Automatic Control*, 62(8):4274–4281, 2017.