Toward Understanding Latent Model Learning in MuZero: A Case Study in 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

Abstract

We study the problem of representation learning for control from partial and potentially high-dimensional observations. We approach this problem via *direct latent model learning*, where one directly learns a dynamical model in some latent state space by *predicting costs*. In particular, we establish *finite-sample guarantees* of finding a near-optimal representation function and a near-optimal controller using the directly learned latent model for infinite-horizon time-invariant Linear Quadratic Gaussian (LQG) control. A part of our approach to latent model learning closely resembles *MuZero*, a recent breakthrough in empirical reinforcement learning, in that it learns latent dynamics *implicitly* by predicting *cumulative costs*. A key technical contribution of this work is to prove persistency of excitation for a new stochastic process that arises from our analysis of quadratic regression in our approach.

1 Introduction

Control with a learned latent model is state of the art in several reinforcement learning (RL) benchmarks, including Go, Atari games, and visuomotor control (Schrittwieser et al., 2020; Ye et al., 2021; Hafner et al., 2023). To better understand this modern machinery, we introduce it to a classical optimal control problem, namely Linear Quadratic Gaussian (LQG) control, and study its theoretical, finite-sample performance. Essential to this approach is the learning of two components: a *state representation* function that maps an observed history to some latent state, and a *latent model* that predicts the transition and cost in the latent state space. The latent model is usually a Markov decision process, using which we obtain a policy in the latent space or execute online planning.

What is the correct objective to optimize for learning a latent model? One popular choice is to learn a function that *reconstructs the observation* from the latent state (Hafner et al., 2019a,b, 2020,

2023). A latent model learned this way is *agnostic to control tasks* and retains all the information about the environment. This class of approaches can achieve satisfactory performance, but are prone to background distraction (Fu et al., 2021). The second class of methods learn an *inverse model* that infers actions from latent states at different time steps (Pathak et al., 2017; Lamb et al., 2022). A latent model learned with this methodology is also task agnostic but extracts control-relevant information. In contrast, *task-relevant* representations can be learned by *predicting costs* in the control task (Oh et al., 2017; Zhang et al., 2020; Schrittwieser et al., 2020). The concept that a good latent state should be able to predict costs is intuitive, and the costs are directly relevant to optimal control. Hence, (Tian et al., 2022) refers to this class of methods as "direct latent model learning", which is the focus of this work.

The direct latent model learning method of particular interest to us is that of MuZero (Schrittwieser et al., 2020). Announced by DeepMind in 2019, MuZero extends the line of works including AlphaGo (Silver et al., 2016), AlphaGo Zero (Silver et al., 2017) and AlphaZero (Silver et al., 2018) by not requiring knowledge of the game rules. MuZero matches the superhuman performance of AlphaZero in Go, shogi and chess, while outperforming model-free RL algorithms in Atari games. MuZero builds upon the powerful planning procedure of Monte Carlo Tree Search, with the major innovation being *learning a latent model*. The latent model replaces the rule-based simulator during planning, and avoids the burdensome planning in pixel space for Atari games.

MuZero is a milestone algorithm in representation learning for control. Intuitively, the algorithm design makes sense, but its complexity has so far inhibited a formal theoretical study. On the other hand, statistical learning theory for linear dynamical systems and control has evolved rapidly in recent years (Tsiamis et al., 2022); for partially observable linear dynamical systems, much of the work relies on learning *Markov parameters*, lacking a direct connection to empirical methods. In this work, we aim to bridge the two areas by studying provable MuZero-style latent model learning in LQG control.

The latent model learning of MuZero features three ingredients: 1) stacking frames, i.e., observations, as input to the representation function; 2) predicting costs, "optimal" values and "optimal" actions from latent states; and 3) implicit learning of latent dynamics by predicting these quantities from latent states at future time steps. These are the defining characteristics of the MuZero-style algorithm that we shall consider. In MuZero, the "optimal" values and actions are found by the powerful online planning procedure. In this work, we simplify the setup by considering data collected using random actions, which are known to suffice for identifying a partially observable linear dynamical system (Oymak and Ozay, 2019). In this setup, the values become those associated with this trivial policy and we do not predict actions since they are random noises anyway. Note that although our study of the above ingredients is directly motivated by MuZero, previous empirical works have also explored them. For example, frame stacking has been a widely used technique to handle partial observability (Mnih et al., 2013, 2015); predicting values for learning a latent model has been studied in (Oh et al., 2017), which also learns the latent state transition implicitly.

Closely related to our work is (Tian et al., 2022), which also considers provable direct latent model learning in LQG, but for the finite-horizon time-varying setting. Our work builds upon it

and complements it in two ways: 1) we extend their algorithm to the time-invariant setting with a *stationary* representation function and latent model, which is closer to practice; 2) we present and analyze a new, MuZero-style latent model learning algorithm. Both 1) and 2) introduce new technical challenges to be addressed. We summarize our contributions as follows.

- We show that two direct latent model learning methods provably solve infinite-horizon time-invariant LQG control by establishing finite-sample guarantees. Both methods only need a single trajectory; one resembles the method in (Tian et al., 2022), and the other resembles MuZero.
- By analyzing the MuZero-style algorithm, we notice the potential issue of coordinate misalignment; that is, costs can be invariant to certain transforms of the latent states, and implicit dynamics learning with one-step transition may not recover consistent coordinates. This insight suggests the need of multi-step transition or other coordinate alignment procedures.
- Technically, we overcome the difficulty of *dependent* data in a single trajectory for latent model learning, by proving a new result about the persistency of excitation for a stochastic process that arises from our analysis of quadratic regression in both methods.

Notation. Given vector $v \in \mathbb{R}^d$, let $\|v\|$ denote its ℓ_2 norm and $\|v\|_P := (v^\top P v)^{1/2}$ for positive semidefinite $P \in \mathbb{R}^{d\times d}$. Semicolon ";" denotes stacking vectors or matrices vertically. For a collection of d-dimensional vectors $(v_t)_{t=i}^j$, let $v_{i:j} := [v_i; v_{i+1}; \dots; v_j] \in \mathbb{R}^{d(j-i+1)}$ 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. For matrix A, let $\sigma_{\min}(A)$, $\|A\|_2$ and $\|A\|_F$ denote its minimum eigenvalue, minimum singular value, operator norm (induced by vector ℓ_2 norms) and Frobenius norm, respectively. $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product between matrices. For square matrix A, let $\lambda_{\min}(A)$ be its minimum eigenvalue and $\rho(A)$ be its spectral radius. Define $\alpha(A) := \sup_{k\geq 0} \|A^k\|_2 \rho(A)^{-k}$. Let $\mathrm{svec}(\cdot)$ denote flattening a symmetric matrix by stacking its columns; it does not repeat the off-diagonal elements, but scales them by $\sqrt{2}$ (Schacke, 2004). The hidden constants in $\mathcal{O}(\cdot)$, $\Omega(\cdot)$ are dimension-free but can depend on system parameters.

2 Problem setup

A partially observable linear time-invariant (LTI) dynamical system is described by

$$x_{t+1} = A^* x_t + B^* u_t + w_t, \quad y_t = C^* x_t + v_t, \tag{2.1}$$

with state $x_t \in \mathbb{R}^{d_x}$, observation $y_t \in \mathbb{R}^{d_y}$, and control $u_t \in \mathbb{R}^{d_u}$ for all $t \geq 0$. Process noises $(w_t)_{t\geq 0}$ and observation noises $(v_t)_{t\geq 0}$ are i.i.d. zero-mean Gaussian random vectors with covariance matrices Σ_w and Σ_v, respectively, and the two sequences are mutually independent. The quadratic cost function is given by

$$c(x, u) = \|x\|^2_{Q^*} + \|u\|^2_{R^*}, \tag{2.2}$$

where $Q^* \succcurlyeq 0$ and $R^* \succ 0$.

A policy/controller π determines an action/control input u_t at time step t based on the history $[y_{0:t}; u_{0:(t-1)}]$ up to this time step. For $t \geq 0$, $c_t := c(x_t, u_t)$ denotes the cost at time step t. Given a policy π, let

$$J^\pi := \limsup_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} c_t\right]$$

denote the average expected cost. The objective of LQG control is to find a policy π such that J^π is minimized.

We make the following standard assumptions.

Assumption 1. *System dynamics (2.1) and cost (2.2) satisfy:*

1. *The system is stable, that is, $\rho(A^*) < 1$.*
2. *(A^*, B^*) is ν-controllable for some $\nu > 0$, that is, the controllability matrix*

   $$\Phi_c(A^*, B^*) := [B^*, A^*B^*, \ldots, (A^*)^{d_x-1}B^*]$$

   has rank d_x and $\sigma_{\min}(\Phi_c(A^*, B^*)) \geq \nu$.
3. *(A^*, C^*) is ω-observable for some $\omega > 0$, that is, the observability matrix*

   $$\Phi_o(A^*, C^*) := [C^*; C^*A^*; \ldots; C^*(A^*)^{d_x-1}]$$

   has rank d_x and $\sigma_{\min}(\Phi_o(A^*, C^*)) \geq \omega$.
4. *(A^*, Σ_w) is v-controllable for some $v > 0$.*
5. *$(A^*, (Q^*)^{1/2})$ is μ-observable for some $\mu > 0$.*
6. *$\Sigma_v \succcurlyeq \sigma_v^2 I$ for some $\sigma_v > 0$; this can always be achieved by inserting Gaussian noises with full-rank covariance matrices to the observations.*
7. *$R^* \succcurlyeq r^2 I$ for some $r > 0$.*
8. *The operator norms of A^*, B^*, C^*, Q^*, R^*, Σ_w, Σ_v, Σ_0 are $\mathcal{O}(1)$ and the singular value lower bounds ν, ω, ν, v, σ_v, r are $\Omega(1)$.*

If the system parameters $(A^*, B^*, C^*, Q^*, R^*, \Sigma_w, \Sigma_v)$ are known, the optimal policy is obtained by combining the Kalman filter

$$z^*_{t+1} = A^*z^*_t + B^*u_t + L^*(y_{t+1} - C^*(A^*z^*_t + B^*u_t)) \tag{2.3}$$

with the optimal feedback gain K^* of the linear quadratic regulator (LQR) such that $u_t = K^*z^*_t$, where L^* is the Kalman gain, and at the initial time step, we can set, e.g., $z^*_0 = L^*y_0$. This fact is known as the *separation principle*, and the Kalman gain and optimal feedback gain are given by

$$L^* = S^*(C^*)^\top(C^*S^*(C^*)^\top + \Sigma_v)^{-1}, \tag{2.4}$$

$$K^* = -((B^*)^\top P^* B^* + R)^{-1}(B^*)^\top P^* A^*, \tag{2.5}$$

where S^* and P^* are determined by their respective discrete-time algebraic Riccati equations (DAREs):

$$S^* = A^*(S^* - S^*(C^*)^\top(C^*S^*(C^*)^\top + \Sigma_v)^{-1}C^*S^*)(A^*)^\top + \Sigma_w, \tag{2.6}$$

$$P^* = (A^*)^\top(P^* - P^*B^*((B^*)^\top P^*B^* + R^*)^{-1}(B^*)^\top P^*)A^* + Q^*. \tag{2.7}$$

Assumptions 1.2 to 1.7 guarantee the existence and uniqueness of positive definite solutions S^* and P^*; Assumption 1.8 further guarantees that their operator norms are $\mathcal{O}(1)$ and minimum singular values are $\Omega(1)$.

We consider data-driven control where the LQG model is unknown, i.e., $(A^*, B^*, C^*, Q^*, \Sigma_w, \Sigma_v)$ are unknown. For simplicity, we assume R^* is known, though our approaches can be readily extended to the case where it is unknown.

2.1 Latent model of LQG

The stationary Kalman filter (2.3) asymptotically produces the optimal *state estimation* in the sense of minimum mean squared errors. With a finite horizon, however, the optimal state estimator is time-varying, given by

$$z^*_{t+1} = A^*z^*_t + B^*u_t + L^*_{t+1}(y_{t+1} - C^*(A^*z^*_t + B^*u_t)), \tag{2.8}$$

where L^*_t is the time-varying Kalman gain, converging to L^* as $t \to \infty$. This convergence is equivalent to that of error covariance matrix $\mathbb{E}[(x_t - z^*_t)(x_t - z^*_t)^\top]$, which happens exponentially fast (Komaroff, 1994). Hence, for simplicity, we assume this error covariance matrix is stationary at the initial time step by the choice of z^*_0 so that $L^*_t = L^*$ for $t \geq 1$; this assumption is common in the literature (Lale et al., 2020, 2021; Jadbabaie et al., 2021). The *innovation* term $i_{t+1} := y_{t+1} - C^*(A^*z^*_t + B^*u_t)$ is independent of the history $(y_0, u_0, y_1, u_1, \ldots, y_{t+1})$ and $(i_t)_{t\geq 1}$ are mutually independent. The following proposition taken from (Tian et al., 2022, Proposition 1) represents the system in terms of the state estimates obtained by the Kalman filter, which we refer to as the *latent model*.

Proposition 1. *Let $(z^*_t)_{t\geq 1}$ be state estimates given by the time-varying Kalman filter. Then, for $t \geq 0$,*

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

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

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

where $b^* > 0$, and $\gamma_t = \|z^*_t - x_t\|^2_{Q^*} - b^*$, $\eta_t = \langle z^*_t, x_t - z^*_t\rangle_{Q^*}$ are both zero-mean subexponential random variables. Moreover, $b^* = \mathcal{O}(1)$ and $\|\gamma_t\|_{\psi_1} = \mathcal{O}(d_x^{1/2})$; if control $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$ for $t \geq 0$, then we have $\|\eta_t\|_{\psi_1} = \mathcal{O}(d_x^{1/2})$.

Proposition 1 shows that the dynamics of the state estimates computed by the time-varying Kalman filter is the same as the original system up to noises; the costs are also the same, up to

constants and noises. Hence, a latent model can be parameterized by (A, B, Q, R^*). A stationary latent policy is a linear controller $u_t = Kz_t$ on latent state z_t, parameterized by feedback gain $K \in \mathbb{R}^{d_u \times d_x}$.

The latent model enables us to find a good latent policy. To learn such a latent model and to deploy a latent policy in the original partially observable system, we need a representation function. Let $\overline{A}^* = (I - L^*C^*)A^*$ and $\overline{B}^* = (I - L^*C^*)B^*$. Then, the Kalman filter can be written as $z^*_{t+1} = \overline{A}^* z^*_t + \overline{B}^* u_t + L^* y_{t+1}$. For $t \geq 0$, unrolling the recursion gives

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

where $M^*_t \in \mathbb{R}^{d_x \times (td_y + td_u + d_x)}$. This means the representation function can be parameterized as linear mappings for full histories (with y_0 replaced by z^*_0). Despite the simplicity, the input dimension of the function grows linearly in time, making estimating the state using the full history intractable for large t; nor is it necessary, since the impact of old data decreases exponentially. Under Assumption 1, $\rho(\overline{A}^*) < 1$ (Bertsekas, 2012, Appendix E.4). With an H-step truncated history, the state estimate can be written as

$$\begin{aligned} z^*_t &= [(\overline{A}^*)^{H-1}L^*, \ldots, L^*] y_{(t-H+1):t} + [(\overline{A}^*)^{H-1}\overline{B}^*, \ldots, \overline{B}^*] u_{(t-H):(t-1)} + \delta_t \\ &=: M^* [y_{(t-H+1):t}; u_{(t-H):(t-1)}] + \delta_t, \end{aligned}$$

where $\delta_t = \overline{A}^H z^*_{t-H}$, whose impact decades exponentially in H and can be neglected for sufficiently large H, since z^*_{t-H} converges to a stationary distribution and its norm is bounded with high probability. Hence, the representation function that we aim to recover is $M^* \in \mathbb{R}^{d_x \times H(d_y + d_u)}$, which takes as input the H-step history $h_t = [y_{(t-H+1):t}; u_{(t-H):(t-1)}]$. Henceforth, we let $d_h := H(d_y + d_u)$. Then, a representation function is parameterized by matrix $M \in \mathbb{R}^{d_x \times d_h}$.

Overall, a policy is a combination of a representation function M and a feedback gain K in the latent model, denoted by $\pi = (M, K)$. Learning to solve LQG control can thus be achieved by: 1) learning representation function M; 2) extracting latent model (A, B, Q, R^*); and 3) finding the optimal K by planning in the latent model. Next, we introduce our approach following this pipeline.

3 Method

In practice, latent model learning methods collect trajectories by interacting with the system with an online policy; the trajectories are used to improve the learned latent model, which in turn improves the policy. In LQG, it is known that the simple setup allows us to learn a good latent model from a single trajectory, collected using zero-mean Gaussian inputs; see e.g., (Oymak and Ozay, 2019). This is also how we assume the data are collected. We note that our results also apply to data from multiple independent trajectories using the same zero-mean Gaussian inputs.

In direct latent model learning, state representations are learned by predicting costs. To learn the transition function in the latent model, two approaches are explored in the literature. The first approach explicitly minimizes transition prediction errors (Subramanian et al., 2020; Hafner et al., 2019a). Algorithmically, the overall loss is a combination of cost and transition prediction errors. The second approach, which MuZero takes, learns transition *implicitly*, by minimizing cost prediction errors at future states generated from the transition function (Oh et al., 2017; Schrittwieser et al., 2020). Algorithmically, the overall loss aggregates the cost prediction errors across multiple time steps. In both approaches, the coupling of different terms in the loss makes finite-sample analysis difficult. As observed in (Tian et al., 2022), the special structure of LQG allows us to learn the representation function independently of learning the transition function. This allows us to formulate both approaches under the same direct latent model learning framework (Algorithm 1).

Algorithm 1 consists of three main steps. Lines 3 to 5 correspond to cost-driven representation learning. Lines 6 to 8 correspond to latent model learning, where the system dynamics can be identified either explicitly, by ordinary least squares (SysId), or implicitly, by future cost prediction (CoSysId, Algorithm 2). Line 8 corresponds to latent policy optimization; in LQG this amounts to solving a DARE. Below we elaborate on cost-driven representation learning, SysId, and CoSysId in order.

3.1 Cost-driven representation learning

The procedure of cost-driven representation learning is almost identical to that in (Tian et al., 2022). The main idea to is perform quadratic regression (3.2) to d_x-step cumulative costs; these correspond to the value prediction in MuZero. By the μ-observability of $(A^*, (Q^*)^{1/2})$ (Assumption 1.5, the cost observability Gramian

$$\overline{Q}^* := \sum_{t=0}^{d_x-1} ((A^*)^t)^\top Q^* (A^*)^t \succcurlyeq \mu^2 I.$$

Under zero control and zero noise, starting from x, the d_x-step cumulative cost is precisely $\|x\|^2_{\overline{Q}^*}$. Hence, $\hat{N}$ estimates $N^* = (M^*)^\top \overline{Q}^* M^*$; up to an orthonormal transform, $\hat{M}$ recovers $M^{*\prime} := (\overline{Q}^*)^{1/2} M^*$, the representation function under an equivalent parameterization, termed the *normalized parameterization* in (Tian et al., 2022), where

$$A^{*\prime} = (\overline{Q}^*)^{1/2} A^* (\overline{Q}^*)^{-1/2},\ B^{*\prime} = (\overline{Q}^*)^{1/2} B,\ C^{*\prime} = C^* (\overline{Q}^*)^{-1/2},$$
$$w'_t = (\overline{Q}^*)^{1/2} w_t,\ Q^{*\prime} = (\overline{Q}^*)^{-1/2} Q^* (\overline{Q}^*)^{-1/2}.$$

Due to the following proposition, the algorithm does not need to know the dimension d_x of the latent model; it can discover d_x from the eigenvalues of $\hat{N}$.

Proposition 2. *Under i.i.d. control $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$ for $t \geq 0$, $\sigma_{\min}(\mathrm{Cov}(z_t^*)) = \Omega(\nu^2)$ for $t \geq d_x$, where ν is defined in Assumption 1.3; as long as $H \geq \frac{\log(a\alpha(\overline{A})\|\mathrm{Cov}(z_t^*)\|\sigma_u^{-2}\nu^{-2})}{\log(\rho(\overline{A})^{-1})}$ for some absolute constant $a > 0$, M^* has rank d_x and $\sigma_{\min}(M^*) \geq \Omega(\nu H^{-1/2})$.*

Algorithm 1 Direct latent model learning for LQG control

1: **Input:** length T, history length H, noise magnitude σ_u
2: Collect a trajectories of length $T+H$ using $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$, for $t \geq 0$, to obtain data in the form of

$$\mathcal{D}_{\text{raw}} = (y_0, u_0, c_0, y_1, u_1, c_1, \ldots, y_{T+H}) \tag{3.1}$$

3: Estimate the state representation function and cost constants by solving

$$\hat{N}, \hat{b}_0 \in \operatorname*{argmin}_{N=N^\top, b_0} \sum_{t=H}^{T+H-1} \left(\|h_t\|_N^2 + b_0 - \bar{c}_t \right)^2, \tag{3.2}$$

where $\bar{c}_t := \sum_{\tau=t}^{t+d_x-1} (c_\tau - \|u_\tau\|_{R^*}^2)$
4: Find $\hat{M} \in \mathbb{R}^{d_x \times H(d_y+d_u)}$ such that $\hat{M}^\top \hat{M}$ is the best approximation of $\hat{N}$ in the Frobenius norm
5: Compute $\hat{z}_t = \hat{M}[y_{(t-H):t}; u_{(t-H):(t-1)}]$ for all $t \geq H$, so that the data are converted to

$$\mathcal{D}_{\text{state}} = (\hat{z}_H, u_H, c_H, \ldots, \hat{z}_{T+H-1}, u_{T+H-1}, c_{T+H-1}, \hat{z}_{T+H})$$

6: Run SysId or CoSysId to obtain dynamics $(\hat{A}, \hat{B})$
7: Estimate the cost function by solving

$$\widetilde{Q}, \hat{b} \in \operatorname*{argmin}_{Q=Q^\top, b} \sum_{t=H}^{T+H-1} (\|\hat{z}_t\|_Q^2 + b - c_t)^2,$$

8: Project $\widetilde{Q}$ to positive semidefinite matrices to obtain $\hat{Q}$
9: Find feedback gain $\hat{K}$ from $(\hat{A}, \hat{B}, \hat{Q}, R^*)$ by DARE (2.7) and (2.5)
10: **Return:** policy $\hat{\pi} = (\hat{M}, \hat{K})$

Proposition 2 is an adaption of (Tian et al., 2022, Proposition 2) to the infinite-horizon LTI setting. Necessarily, this implies that by our choice of H, $d_h = H(d_y + d_u) \geq d_x$. Moreover, since $\overline{Q}^* \succcurlyeq \mu^2 I$, $N^* = (M^*)^\top \overline{Q}^* M^*$ is a $d_h \times d_h$ matrix with rank d_x, and $\lambda_{\min}^+(N^*) \geq \lambda_{\min}(\overline{Q}^*) \lambda_{\min}^2(M^*) = \Omega(\mu^2 \nu^2 H^{-1})$. Hence, if $\hat{N}$ is sufficiently close to N^*, by setting an appropriate threshold on the eigenvalues of $\hat{N}$, the dimension of the latent model equals the number of eigenvalues above it.

To find an approximate factorization of $\hat{N}$, let $\hat{N} = U \Lambda U^\top$ be its eigenvalue decomposition, where the diagonal elements of Λ are listed in a descending order, and U is an orthonormal matrix. Let Λ_{d_x} be the left-top block of Λ and U_{d_x} be the left d_x columns of U. By the Eckart-Young-Mirsky theorem, $\hat{M} = \max(\Lambda_{d_x}, 0)^{1/2} U_{d_x}^\top$ is the best approximate factorization of $\hat{N}$ among $d_x \times d_h$ matrices in terms of the Frobenius norm approximation error.

In the next two subsections, we move on to discuss learning latent dynamics, including the explicit approach SysId and implicit approach CoSysId.

3.2 Explicit learning of system dynamics

Explicit learning of the system dynamics simply minimizes the transition prediction error in the latent space (Subramanian et al., 2020), or more generally, statistical distances between the predicted and estimated distributions of the next latent state, like the KL divergence (Hafner et al., 2019a). In linear systems, it suffices to use the ordinary least squares as the SysId procedure, that is, to solve

$$(\hat{A}, \hat{B}) \in \operatorname*{argmin}_{A,B} \sum_{t=H}^{T+H-1} \|A\hat{z}_t + Bu_t - \hat{z}_{t+1}\|^2.$$

In this linear regression, if $(\hat{z}_t)_{t\geq H}$ are the optimal state estimates $(z_t^*)_{t\geq H}$ (2.8), then (Simchowitz et al., 2018) has shown finite-sample guarantees for $(\hat{A}, \hat{B})$. Here, $\hat{z}_t$ contains errors resulting from the representation function $\hat{M}$ and the residual error δ_t, but as long as T and H are large enough, SysId still has a finite-sample guarantee. We call the algorithm that instantiates Algorithm 1 with SysId CoRel (Cost-driven state Representation Learning). As the time-varying counterpart in (Tian et al., 2022), it provably solves learning for LQG control, as will be shown in Theorem 1.

3.3 Implicit learning of system dynamics (MuZero-style)

An important ingredient of latent model learning in MuZero (Schrittwieser et al., 2020) is to *implicitly* learn the transition function by minimizing the cost prediction error at future latent states generated from the transition function. Let $z_t = Mh_t$ denote the latent state given by representation function M at step t. Let $z_{t,0} = z_t$ and $z_{t,i+1} = Az_{t,i} + Bu_t$ for $i \geq 0$ be future latent states predicted by dynamics (A, B). For a trajectory of length $T + H$ like (3.1), the loss that considers ℓ steps into the future is given by

$$\sum_{t=H}^{T+H-K-1} \sum_{i=0}^{\ell} (\|z_{t,i}\|_Q^2 + \|u_t\|_{R^*}^2 + b - c_t)^2.$$

This loss involves powers of A from A^2 to A^ℓ and is not amenable to analysis. In LQG, our finding is that it suffices to take $\ell = 1$. As mentioned in §1, MuZero also predicts optimal values and optimal actions; in LQG, to handle $Q^* \not\succ 0$, like cost-driven representation learning (see §3.1), we adopt the *cumulative costs* and use the normalized parameterization. Thus, the optimization problem we aim to solve is given by

$$\min_{M,A,B,b} \sum_{t=H}^{T+H-1} \left((\|Mh_t\|^2 + b - \bar{c}_t)^2 + (\|AMh_t + Bu_t\|^2 + b - \bar{c}_{t+1})^2 \right). \tag{3.3}$$

To convexify the optimization problem (3.3), we define $N := M^\top M$ and $N' := [AM, B]^\top [AM, B]$. Then, (3.3) becomes

$$\min_{N,N_1,b} \sum_{t=H}^{T+H-1} \left((\|h_t\|_N^2 + b - \bar{c}_t)^2 + (\|[h_t; u_t]\|_{N_1}^2 + b - \bar{c}_{t+1})^2 \right). \tag{3.4}$$

This minimization problem is convex in N, N_1 and b, and has a closed-form solution; essentially, it consists of two linear regression problems coupled by b. Since constant b is merely a

Algorithm 2 CoSysId: Cost-driven system identification

1: **Input:** data $\mathcal{D}_{\text{raw}}$, representation function $\hat{M}$
2: Estimate the system dynamics by

$$\hat{N}_1, \hat{b}_1 \in \operatorname*{argmin}_{N_1 = N_1^\top, b_1} \sum_{t=H}^{T+H-1} \left(\|[h_t; u_t]\|_{N_1}^2 + b_1 - \bar{c}_{t+1} \right)^2 \tag{3.5}$$

3: Find $\hat{M}_1 \in \mathbb{R}^{d_x \times (Hd_y + (H+1)d_u)}$ such that $\hat{M}_1^\top \hat{M}_1$ is the best approximation of $\hat{N}_1$ in the Frobenius norm
4: Solve $[\widetilde{A}\hat{M}, \widetilde{B}] = \hat{M}_1$ for $\widetilde{A}, \widetilde{B}$
5: Find alignment matrix $\hat{S}_0$ by solving linear regression from $(\hat{M}_1[h_t; u_t])_{t=H}^{T+H-1}$ to $(\hat{z}_{t+1})_{t=H}^{T+H-1}$
6: **Return:** system dynamics estimate $(\hat{A}, \hat{B}) = (S_0\widetilde{A}, S_0\widetilde{B})$

term accounting for the estimation error and not part of the representation function, we can decouple the two regression problems by allowing b to take different values in them. This further simplifies the algorithm: the first regression problem is exactly cost-driven representation learning (§3.1), and the second is cost-driven system identification (CoSysId, Algorithm 2). The algorithm that instantiates Algorithm 1 with CoSysId is called CoReDyL (Cost-driven state Representation and Dynamic Learning). Like CoReL, this MuZero-style latent model learning method provably solves learning for LQG control, as we will show in Theorem 1.

CoSysId has similar steps to cost-driven representation learning (§3.1), except that in Line 5, it requires fitting a matrix $\hat{S}_0$. This is because approximate factorization steps recover M^* and M_1^* up to orthonormal transforms, but there is no guarantee for the two orthonormal matrices to be the same; we need to fit $\hat{S}_0$ to align their coordinates. We note that although CoSysId needs the output $\hat{M}$ from cost-driven representation learning, the two quadratic regressions (3.2) and (3.5) are not coupled and can be solved in parallel.

4 Theoretical guarantees

The following Theorem 1 shows that both CoReL and CoReDyL are guaranteed to solve unknown LQG control.

Theorem 1. *Given an unknown LQG problem satisfying Assumption 1, let $M^{*\prime}$ and $(A^{*\prime}, B^{*\prime}, Q^{*\prime}, R^*)$ be the optimal state representation function and the true system parameters under the normalized parameterization. For a given $p \in (0,1)$, if we run* CoReL *or* CoReDyL *for $T \geq \text{poly}(d_x, d_y, d_u, \log(d_x T/p)))$ for $H \geq \frac{\log((d_y+d_u)^{-1} d_x^{-1/2} T)}{2\log(\rho(\bar{A})^{-1})}$ and $\sigma_u = \Omega(1)$, then there exists an orthonormal matrix $S \in \mathbb{R}^{d_x \times d_x}$, such that with probability at least $1-p$, the representation function $\hat{M}$ satisfies*

$$\|\hat{M} - SM^{*\prime}\|_2 = \mathcal{O}(H(d_y + d_u) d_x^{3/2} T^{-1/2} \log^3(T/p)),$$

and the suboptimality gap of feedback gain $\hat{K}$ in system $(SA^{*\prime}S^\top, SB^{*\prime}, SQ^{*\prime}S^\top, R^*)$ is

$$\mathcal{O}(H^2 (d_x \wedge d_u)(d_y + d_u)^3 d_x^5 T^{-1} \log^6(d_x T/p)).$$

Proof sketch. Central to the analysis is the finite-sample characterization of the *quadratic regression* problem. To solve (3.2), notice that

$$\|h_t\|_N^2 = \left\langle N, h_t h_t^\top \right\rangle_F = \left\langle \mathrm{svec}(N), \mathrm{svec}(h_t h_t^\top) \right\rangle,$$

so this quadratic regression is essentially a linear regression with correlated covariates that are products of Gaussians. By establishing persistency of excitation for $(\mathrm{svec}(h_t h_t^\top))_{t \geq H}$ after a burn-in period and using sub-Weibull martingale concentration, we show that

$$\|\hat{N} - N^*\|_F = \mathcal{O}(H(d_y + d_u) d_x^{3/2} T^{-1/2} \log^3(T/p)),$$
$$\|\hat{N}_1 - N_1^*\|_F = \mathcal{O}((H(d_y + d_u) + d_x) d_x^{3/2} T^{-1/2} \log^3(T/p));$$

our choice of H ensures that the truncation errors are absorbed into these bounds. By the Procrustes-type lemma in (Tu et al., 2016, Lemma 5.4), these two bounds imply bounds of the same order on $\|\hat{M} - SM^{*\prime}\|_F$ and $\|\hat{M}_1 - S_1 M_1^{*\prime}\|_F$, respectively, for some orthonormal matrices S and S_1. Thus, $\|\widetilde{B} - S_1 B^{*\prime}\|_2$ is of the same order as the bound on $\|\hat{N}_1 - N_1^*\|_2$, and by the perturbation bounds of the Moore-Penrose inverse (Wedin, 1973), so is $\|\widetilde{A} - S_1 A^{*\prime} S^\top\|_2$.

As will be remarked below, we need to fit matrix $S_0 = S S_1^\top$ to ensure the next states generated by $(\hat{A}, \hat{B})$ align with those by $\hat{M}$. With an analysis of *perturbed* linear regression, we find that $\|\hat{S}_0 - S_0\|_2$ has the same bound as $\|\hat{N}_1 - N_1^*\|_F$. Therefore, the bounds on $\|\hat{A} - SA^*S^\top\|_2$ and $\|\hat{B} - SB^*\|_2$ are of the same order. Line 7 in Algorithm 1 requires an analysis of perturbed quadratic regression, which guarantees that

$$\|\hat{Q} - SQ^*S^\top\|_F = \mathcal{O}(H(d_y + d_u) d_x^{5/2} T^{-1/2} \log^3(T/p)).$$

Hence, $\|\hat{A} - SA^*S^\top\|_2$, $\|\hat{B} - SB^*S^\top\|_2$ and $\|\hat{Q} - SQ^*S^\top\|_2$ are all $\mathcal{O}(H(d_y + d_u)^{3/2} d_x^{5/2} T^{-1/2} \log^3(d_x T/p))$.

Lastly, we invoke the result on certainty equivalent LQR in (Mania et al., 2019) to certify the suboptimality gap of $\hat{K}$ obtained from $(\hat{A}, \hat{B}, \hat{Q}, R^*)$. This certainty equivalent controller can also be placed by a robust one. Let $K^{*\prime}$ denote the optimal controller under the normalized parameterization. Then, for a large enough burn-in period, the certainty-equivalent controller satisfies that $\|\hat{K} - K^{*\prime}\|_2$ is of the same order as system parameter errors. Note that policy K^*M^* in the original system is independent of the latent model parameterization, and we have that $\|\hat{K}\hat{M} - K^*M^*\|_2$ is of the same order. ■

Remark. Although the next state $z_{t,1}$ generated by $(\widetilde{A}, \widetilde{B})$ yields the correct cumulative cost at the next time step, it mismatches the next state $\hat{z}_{t+1}$ generated by $\hat{M}$ using h_{t+1} by an orthonormal transform $S_0 = S S_1^\top$; that is, $z_{t,1}$ approximates $S_0^\top \hat{z}_{t+1}$ instead of $\hat{z}_{t+1}$. This happens because the cost is invariant to orthonormal transforms of the latent states, and by only predicting one step into the future, the orthonormal transforms from the two quadratic regressions are not guaranteed to be the same. MuZero bypasses this problem by predicting multiple steps; here we simply calculate this alignment matrix S_0 by Line 5 in Algorithm 2.

Remark. Compared with the common system identification methods based on learning Markov parameters (Oymak and Ozay, 2019; Simchowitz et al., 2019), the error bounds of the system parameters produced by CoReDyL (or CoReL) have the same dependence on T, but worse dependence on system dimensions. Moreover, to establish persistency of excitation, CoReDyL

(or CoReL) requires a larger burn-in period. These relative sample inefficiencies are the price we pay for direct latent model learning, which is only supervised by *scalar-valued* costs that are quadratic in the history, instead of *vector-valued* observations that are linear in the history. Hence, we have to address the more challenging quadratic regression problem, which lifts the dimension of the optimization problem. On the other hand, direct latent model learning avoids learning the reconstruction function C^* and can learn task-relevant representations in more complex settings, as demonstrated by empirical studies.

5 Conclusion and future work

We studied direct latent model learning for solving unknown infinite-horizon time-invariant LQG control. We established finite-sample guarantees for two methods, which differ in whether the latent dynamics is learned explicitly or implicitly, with the latter being closer to that used in MuZero Schrittwieser et al. (2020). For MuZero-style latent model learning, our analysis identifies a coordinate misalignment problem, suggesting the value of *multi-step* future prediction. A limitation of this work is that we only consider state representation based on truncated histories, i.e., frame stacking, as used in MuZero; the *recursive form* of the representation function, as the Kalman filter, is also used empirically (Ha and Schmidhuber, 2018; Hafner et al., 2019a), and might be worth further investigation.

Many questions remain to be answered in representation learning for control. Provable generalization of direct latent model learning to nonlinear observations or dynamics is a natural consideration. Moreover, with the ubiquity of visual perception in real-world control systems, what if we have a varying observation function or multiple observation functions, modeling images taken from different angles? In reality, most of the time we do not have a well-defined cost function; learning task-relevant state representations from demonstrations is another intriguing direction.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

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

Xiequan Fan, Ion Grama, and Quansheng Liu. Large deviation exponential inequalities for supermartingales. 2012.

Xiequan Fan, Ion Grama, and Quansheng Liu. Deviation inequalities for martingales with applications. *Journal of Mathematical Analysis and Applications*, 448(1):538–566, 2017.

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.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Ali Jadbabaie, Horia Mania, Devavrat Shah, and Suvrit Sra. Time varying regression with hidden linear dynamics. *arXiv preprint arXiv:2112.14862*, 2021.

N Komaroff. Iterative matrix bounds and computational solutions to the discrete algebraic Riccati equation. *IEEE Transactions on Automatic Control*, 39(8):1676–1678, 1994.

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.

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

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Junhyuk Oh, Satinder Singh, and Honglak Lee. Value prediction network. *Advances in neural information processing systems*, 30, 2017.

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.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

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.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of Go with deep neural networks and tree search. *nature*, 529(7587): 484–489, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *nature*, 550(7676):354–359, 2017.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.

Max Simchowitz, Horia Mania, Stephen Tu, Michael I Jordan, and Benjamin Recht. Learning without mixing: Towards a sharp analysis of linear system identification. In *Conference On Learning Theory*, pages 439–473. PMLR, 2018.

Max Simchowitz, Ross Boczar, and Benjamin Recht. Learning linear dynamical systems with semi-parametric least squares. In *Conference on Learning Theory*, pages 2714–2802. PMLR, 2019.

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.

Yi Tian, Kaiqing Zhang, Russ Tedrake, and Suvrit Sra. Can direct latent model learning solve linear quadratic Gaussian control? *arXiv preprint arXiv:2212.14511*, 2022.

Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12:389–434, 2012.

Anastasios Tsiamis, Ingvar Ziemann, Nikolai Matni, and George J Pappas. Statistical learning theory for control: A finite sample perspective. *arXiv preprint arXiv:2209.05423*, 2022.

Stephen Tu, Ross Boczar, Max Simchowitz, Mahdi Soltanolkotabi, and Ben Recht. Low-rank solutions of linear matrix equations via Procrustes flow. In *International Conference on Machine Learning*, pages 964–973. PMLR, 2016.

Per-Åke Wedin. Perturbation theory for pseudo-inverses. *BIT Numerical Mathematics*, 13:217–232, 1973.

Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering Atari games with limited data. *Advances in Neural Information Processing Systems*, 34:25476–25488, 2021.

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.

Yang Zheng and Na Li. Non-asymptotic identification of linear dynamical systems using multiple trajectories. *IEEE Control Systems Letters*, 5(5):1693–1698, 2020.

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.

A Additional related work

(Oymak and Ozay, 2019) studies the identification of partially observable linear dynamical systems from a single trajectory, which presents a finite-sample analysis of identifying the Markov parameter and a perturbation analysis of the Ho-Kalman algorithm. (Simchowitz et al., 2019) relaxes the stability requirement to marginal stability by using prefiltered least squares to identify the Markov parameter. The method in (Zheng and Li, 2020) applies to unstable systems but requires multiple trajectories. Since the Markov parameter maps control input histories to observations, these methods do not work with costs and use the Markov parameter as an intermediate step to identify the system. By contrast, our methods, entirely driven by the costs and closely connected with empirical methods, directly learn the representation function and the latent model. Directly learning the latent model connects our work to the identification of fully observable linear dynamical systems. (Simchowitz et al., 2018) introduces small-ball conditions to handle dependent data and characterizes the statistical rates for stable and unstable systems, both proving to be useful for our analysis.

Online control of partially observable linear dynamical systems is considered in (Lale et al., 2020, 2021) for stochastic noises and in (Simchowitz et al., 2020) for nonstochastic noises. (Zheng et al., 2021) considers end-to-end sample complexity and is closest to our setup. All these methods rely on the estimation of Markov parameters. For a discussion of the literature in more details and breadth, we refer the reader to the recent survey (Tsiamis et al., 2022).

B Proof of Proposition 2

Lemma 1. *Let x and y be random vectors defined on the same probability space. Then, $\|\mathrm{Cov}(x,y)\|_2 \le \|\mathrm{Cov}(x)^{1/2}\|_2 \|\mathrm{Cov}(y)^{1/2}\|_2$.*

Lemma 2. *Let x, y be random vectors of dimensions d_x, d_y, respectively, defined on the same probability space. Then, $\|\mathrm{Cov}([x;y])\|_2 \le \|\mathrm{Cov}(x)\|_2 + \|\mathrm{Cov}(y)\|_2$.*

For $t \ge d_x$, unrolling the Kalman filter gives

$$\begin{aligned} z_t^* &= A^* z_{t-1}^* + B^* u_{t-1} + L^* i_t \\ &= A^*(A^* z_{t-2}^* + B^* u_{t-2} + L^* i_{t-1}) + L^* i_t \\ &= [B^*, \dots, (A^*)^{d_x-1} B^*][u_{t-1}; \dots; u_{t-d_x}] + (A^*)^{d_x} z_{t-d_x}^* \\ &\quad + [L^*, \dots, (A^*)^{d_x-1} L^*][i_t; \dots; i_{t-d_x+1}], \end{aligned}$$

where $(u_\tau)_{\tau=t-d_x}^{t-1}$, $z_{t-d_x}^*$ and $(i_\tau)_{\tau=t-d_x+1}^{t}$ are independent. For $H \ge d_x$, the matrix multiplied by $[u_{t-1}; \dots; u_{t-d_x}]$ is precisely the controllability matrix $\Phi_c(A^*, B^*)$. Then

$$\begin{aligned} &\mathrm{Cov}(z_t^*) \\ &= \mathbb{E}[z_t^* (z_t^*)^\top] \\ &\succcurlyeq \Phi_c(A^*, B^*) \mathbb{E}[[u_{t-1}; \dots; u_{t-\ell}][u_{t-1}; \dots; u_{t-\ell}]^\top] \Phi_c^\top(A^*, B^*) \\ &= \sigma_u^2 \Phi_c(A^*, B^*) \Phi_c^\top(A^*, B^*). \end{aligned}$$

By the ν-controllability of (A^*, B^*), $\mathrm{Cov}(z_t^*)$ is full-rank and

$$\sigma_{\min}(\mathrm{Cov}(z_t^*)) \ge \sigma_u^2 \nu^2.$$

On the other hand, since $z_t^* = M^* h_t + \delta_t$,

$$\begin{aligned} \mathrm{Cov}(M^* h_t) &= \mathrm{Cov}(z_t^* - \delta_t) \\ &= \mathrm{Cov}(z_t^*) + \mathrm{Cov}(\delta_t) - \mathrm{Cov}(z_t^*, \delta_t) + \mathrm{Cov}(\delta_t, z_t^*). \end{aligned}$$

By Lemma 1,

$$\begin{aligned} \|\mathrm{Cov}(z_t^*, \delta_t)\|_2 &= \|\mathrm{Cov}(\delta_t, z_t^*)\|_2 \\ &\le \|\mathrm{Cov}(z_t^*)^{1/2}\|_2 \|\mathrm{Cov}(\delta_t)^{1/2}\|_2. \end{aligned}$$

Hence, by Weyl's inequality,

$$\begin{aligned} &\lambda_{\min}(\mathrm{Cov}(M^* h_t)) \\ &\ge \lambda_{\min}(\mathrm{Cov}(z_t^*)) - 2\|\mathrm{Cov}(z_t^*)^{1/2}\|_2 \|\mathrm{Cov}(\delta_t)^{1/2}\|_2. \end{aligned}$$

Since $\delta_t = \overline{A}^H z_{t-H}^*$, as long as $H \ge \frac{\log(a\alpha(\overline{A})\|\mathrm{Cov}(z_t^*)\|\sigma_u^{-2}\nu^{-2})}{\log(\rho(\overline{A})^{-1})}$ for some absolute constant $a > 0$,

$$\lambda_{\min}(\mathrm{Cov}(M^* h_t)) \ge \sigma_u^2 \nu^2 / 2.$$

On the other hand,

$$\mathbb{E}[M^* h_t h_t^\top (M^*)^\top] \preccurlyeq \|\mathbb{E}[h_t h_t^\top]\|_2 M^* (M^*)^\top.$$

Since $h_t = [y_{(t-H+1):t}; u_{(t-H):(t-1)}]$ and $(\mathrm{Cov}(y_t))_{t\geq 0}$, $(\mathrm{Cov}(u_t))_{t\geq 0}$ have $\mathcal{O}(1)$ operator norms, by Lemma 2, $\mathrm{Cov}(h_t) = \mathbb{E}[h_t h_t^\top] = \mathcal{O}(H)$. Hence,

$$0 < \sigma_u^2 \nu^2/2 \leq \sigma_{\min}(\mathrm{Cov}(z_t^*)) = \mathcal{O}(H)\sigma_{d_x}^2(M_t^*).$$

This implies that $\mathrm{rank}(M_t^*) = d_x$ and $\sigma_{\min}(M_t^*) = \Omega(\nu H^{-1/2})$.

C Additional discussions on CoSysId

In CoSysId (Algorithm 2), the covariates of quadratic regression in (3.5) are $([h_t; u_t])_{t\geq H}$. One may wonder if we can pursue an alternative approach by fixing M to be $\hat{M}$, and using $([\hat{z}_t; u_t])_{t\geq H}$ as covariates, which have a much lower dimension, though the two quadratic regressions cannot be solved in parallel anymore. Specifically, the new quadratic regression we need to solve is given by

$$\hat{N}_2, \hat{b}_2 \in \underset{N_2 = N_2^\top, b_2}{\mathrm{argmin}} \sum_{t=H}^{T+H-1} \left(\|[\hat{z}_t; u_t]\|_{N_2}^2 + b_2 - \bar{c}_{t+1} \right)^2,$$

where $\hat{z}_t = \hat{M} h_t$ is an approximation of $S z_t^*$. The ground truth for $\hat{N}_2$ is $N_2^* = [SA^*S^\top, SB^*]^\top [SA^*S^\top, SB^*]$, so its approximate factorization recovers $[S_3 A^* S^\top, S_3 B^*]$ for some orthonormal matrix S_3. In a similar way to CoSysId, we still need to fit an alignment matrix $S_2 = S S_3^\top$ to align the coordinates. Let $\widetilde{A}$, $\widetilde{B}$ denote the system parameters recovered from $\hat{N}_2$. The linear regression we now need to solve is from $([\widetilde{A}, \widetilde{B}][\hat{z}_t; u_t])_{t=H}^{T+H-1}$ to $(\hat{z}_{t+1})_{t=H}^{T+H-1}$. However, without further assumptions, $[A^*, B^*]$ does not necessarily have full row rank, and hence, neither does $[\widetilde{A}, \widetilde{B}]$, in which case recovering the entire S_2 is impossible.

D Persistency of excitation

A major difficulty in the analysis of quadratic regression is to establish persistency of excitation, meaning that the minimum eigenvalue of the design matrix of the induced linear regression grows linearly in the size of the data. The following lemma establishes the persistency of excitation for quadratic regressions (3.2) and (3.5).

Lemma 3. *Let $h_t = [y_{(t-H+1):t}; u_{(t-H):(t-1)}]$ be the H-step history at time step t in system (2.1) with $u_t \sim \mathcal{N}(0, \sigma_u^2 I)$ for $t \geq 0$. Define $f_t := \mathrm{svec}(h_t h_t^\top)$. For a given $p \in (0,1)$, as long as $T \geq a_0 H d_h^2 \log(d_h/p)$ for some problem dependent constant $a_0 > 0$, with probability at least $1-p$,*

$$\lambda_{\min}\left(\sum_{t=H}^{T+H-1} f_t f_t^\top \right) = \Omega(T).$$

A linear lower bound on $\lambda_{\min}(\sum_{t=H}^{T+H-1} h_t h_t^\top)$ is a known result for the identification of partially observable linear dynamical systems (Tsiamis et al., 2022). Here, elements of f_t are *products* of Gaussians, making the analysis difficult. If $(h_t)_{t\geq H}$ are independent, which is the case if they are from multiple trajectories, the result has been established in (Jadbabaie et al., 2021; Tian et al., 2022). It can also be proved with the matrix Azuma inequality (Tropp, 2012). Here, we need to aggregate dependent data to estimate a set of *stationary* parameters. The difficulty we face results from both products of Gaussians and the data dependence.

In principle, given enough burn-in time, state x_t, and hence observation y_t and truncated history h_t, will converge to steady-state distributions, and samples with an interval of the order of mixing time are approximately independent, so intuitively, a linear lower bound is correct. But a proof of this type incurs dependence on mixing time; to eschew such dependence, (Simchowitz et al., 2018) introduces the small-ball method. The proof sketch below takes this route, while developing different arguments to handle the products of Gaussians.

Before presenting the proof sketch, we formally define the block martingale small-ball condition (Simchowitz et al., 2018, Definition 2.1).

Definition 1 (Block martingale small ball (BMSB))**.** *Let $(f_t)_{t\geq 1}$ be a stochastic process in $\mathbb{R}^d$ adapted to filtration $(\mathcal{F}_t)_{t\geq 1}$. We say $(f_t)_{t\geq 1}$ satisfies (k, Γ, q)-BMSB condition for $k \in \mathbb{N}^+$, $\Gamma \succcurlyeq 0$ and $q > 0$, if for any $t \geq 1$, for any fixed unit vector $v \in \mathbb{R}^d$, $\frac{1}{k}\sum_{i=1}^k \mathbb{P}(|\langle f_{t+i}, v\rangle| \geq \|v\|_\Gamma \mid \mathcal{F}_t) \geq q$ almost surely.*

Proof sketch. The core of the proof is to show the BMSB condition for $(f_t)_{t\geq H}$, adapted to filtration $(\mathcal{F}_t)_{t\geq H}$, where $\mathcal{F}_t = \sigma(x_0, v_0, u_0, w_0, v_1, \ldots, u_{t-1}, w_{t-1}, v_t)$.

Since every unit vector $v \in \mathbb{R}^{d_h(d_h+1)/2}$ corresponds to a symmetric matrix $M \in \mathbb{R}^{d_h \times d_h}$ with unit Frobenius norm, then

$$\langle f_{t+i}, v\rangle = \langle \mathrm{svec}(h_{t+i}h_{t+i}^\top), \mathrm{svec}(M)\rangle = h_{t+i}^\top M h_{t+i}.$$

Take $\Gamma = \gamma^2 I$ for some $\gamma > 0$. Then, $\|v\|_\Gamma = \gamma$. It suffices to show that for $i > G$ for some $G > 0$,

$$\mathbb{P}(|h_{t+i}^\top M h_{t+i}| \geq \gamma \mid \mathcal{F}_t) \geq q,$$

since if so, we have

$$\begin{aligned}&\frac{1}{2G}\sum_{i=1}^{2G} \mathbb{P}(|h_{t+i}^\top M h_{t+i}| \geq \gamma \mid \mathcal{F}_t)\\ \geq\ &\frac{1}{2G}\sum_{i=G+1}^{2G} \mathbb{P}(|h_{t+i}^\top M h_{t+i}| \geq \gamma \mid \mathcal{F}_t) \geq q/2,\end{aligned}$$

which means $(f_t)_{t\geq H}$ is $(2G, \gamma^2 I, q/2)$-BMSB.

Now let us take a close look at $h_{t+i} = [y_{(t+i-H+1):(t+i)}; u_{(t+i-H):(t+i-1)}]$. Since

$$y_{t+i} = CA^i x_t + \sum_{j=1}^i CA^j(Bu_{t+i-j} + w_{t+i-j}) + v_{t+i},$$

$y_{t+i} \mid F_t$ is Gaussian with mean $CA^i x_t$ and covariance determined by $\sum_{j=1}^i CA^j(Bu_{t+i-j} + w_{t+i-j}) + v_{t+i}$, where we note that v_{t+i} is independent of all other random variables and has

full-rank covariance. Hence, for $i > H$, $h_{t+i} \mid F_t$ is Gaussian and has full-rank covariance. Then intuitively, since $\|M\|_F = 1$, $|h_{t+i}^\top M h_{t+i}| \mid \mathcal{F}_t$ is a well-behaved random variable that can exceed some $\gamma > 0$ with a positive probability q. By Lemma 4, for $i > H$, there exists some absolute constant $a > 0$, such that

$$\mathbb{E}[|h_{t+i}^\top M h_{t+i}| \mid \mathcal{F}_t] \geq a \min\{\sigma_u, \sigma_v\}/d_h.$$

On the other hand, since

$$\begin{aligned} |h_{t+i}^\top M h_{t+i}| &= \left|\left\langle M, h_{t+i} h_{t+i}^\top \right\rangle_F\right| \\ &\leq \|M\|_F \|h_{t+i} h_{t+i}^\top\|_F = h_{t+i}^\top h_{t+i}, \end{aligned}$$

we have $\mathbb{E}[|h_{t+i}^\top M h_{t+i}|^2 \mid \mathcal{F}_t] \leq \mathbb{E}[\|h_{t+i}\|^4 \mid \mathcal{F}_t]$. Since $\|h_{t+i}\| \mid \mathcal{F}_t$ is sub-Gaussian with

$$\|\|h_{t+i}\| \mid \mathcal{F}_t\|_{\psi_2} = \mathcal{O}(\|\mathbb{E}[h_{t+i} h_{t+i}^\top \mid \mathcal{F}_t]\|_2^{1/2}) = \mathcal{O}(1),$$

$\mathbb{E}[|h_{t+i}^\top M h_{t+i}|^2 \mid \mathcal{F}_t] = \mathcal{O}(1)$. By the Paley-Zygmund inequality, for $\theta \in [0,1]$ we have

$$\mathbb{P}(|h_{t+i}^\top M h_{t+i}| \geq \theta a/d_h \mid \mathcal{F}_t) = \Omega((1-\theta)^2 d_h),$$

where the dependence on σ_u, σ_v is hidden in $\Omega(\cdot)$. By taking $\theta = 1/2$, we can see that $(f_t)_{t \geq H}$ satisfies $(k, \gamma^2 I, q)$-BMSB condition for $k = 2H$, $\gamma = a/(2d_h)$ and $q = \Omega(d_h)$. Then, following the analysis in (Simchowitz et al., 2018, Appendix D), we can show that for a given $p \in (0,1)$, as long as $T \geq a_0 H d_h^2 \log(d_h/p)$ for some problem-dependent constant $a_0 > 0$, then with probability at least $1 - p$, we have

$$\lambda_{\min}\left(\sum_{t=H}^{T+H-1} f_t f_t^\top\right) = \Omega(\gamma^2 q^2 T) = \Omega(T),$$

which completes the proof. ■

Lemma 4. *Let x be d-dimensional zero-mean Gaussian random vector with covariance Σ. Let A be $d \times d$ symmetric matrix with unit Frobenius norm. Then $\mathbb{E}[|x^\top A x|] \geq a\lambda_{\min}(\Sigma)/d$ for some absolute constant $a > 0$.*

Proof. Let $y := \Sigma^{-1/2} x$. Then y is a standard Gaussian random vector, and $x^\top A x = y^\top \Sigma^{1/2} A \Sigma^{1/2} y$. Let $U^\top \Lambda U$ be the eigenvalue decomposition of $\Sigma^{1/2} A \Sigma^{1/2}$. Then,

$$\mathbb{E}[|x^\top A x|] = \mathbb{E}[|y^\top U^\top \Lambda U y|] = \mathbb{E}[|z^\top \Lambda z|],$$

where $z := Uy$ is still a standard Gaussian random vector. Since

$$\|\Lambda\|_F \overset{(i)}{=} \|U^\top \Lambda U\|_F = \|\Sigma^{1/2} A \Sigma^{1/2}\|_F \geq \lambda_{\min}(\Sigma)\|A\|_F = \lambda_{\min}(\Sigma),$$

where (i) is due to the unitary invariance of the Frobenius norm, we have

$$\inf_{\|A\|_F = 1} \mathbb{E}[|x^\top A x|] \geq \inf_{\|\Lambda\|_F \geq \lambda_{\min}(\Sigma)} \mathbb{E}[|z^\top \Lambda z|] \overset{(i)}{\geq} a\lambda_{\min}(\Sigma)/d,$$

where (i) is due to Lemma 5. □

Lemma 5. *Let $z_1, z_2, \ldots, z_n$ be d-dimensional independent standard Gaussian random vectors. Let $v = [v_1, v_2, \ldots, v_n]^\top \in \mathbb{R}^n$ be a unit vector. There exists an absolute constant $a > 0$, such that for any such v, $\mathbb{E}[|\sum_{i=1}^n v_i z_i^\top z_i|] \geq a\sqrt{d}/n$.*

Proof. For given a, since $(z_i^\top z_i)_{i=1}^n$ have identical distributions, $\mathbb{E}[|\sum_{i=1}^n v_i z_i^\top z_i|]$ has the same value under permutations of $(v_i)_{i=1}^n$. By the convexity of the absolute function and Jensen's inequality,

$$\begin{aligned}
\mathbb{E}\left[\left|\sum_{i=1}^n v_i z_i^\top z_i\right|\right] &\geq \mathbb{E}\left[\left|\frac{1}{n}\left(\sum_{i=1}^n v_i\right)\left(\sum_{i=1}^n z_i^\top z_i\right)\right|\right] \\
&= \mathbb{E}\left[\left|\frac{1}{n}\left(\sum_{i=1}^n |v_i|\right)\left(\sum_{i=1}^n \operatorname{sign}(v_i) z_i^\top z_i\right)\right|\right] \\
&= \frac{1}{n}\left(\sum_{i=1}^n |v_i|\right)\mathbb{E}\left[\left|\sum_{i=1}^n \operatorname{sign}(v_i) z_i^\top z_i\right|\right]
\end{aligned}$$

Since $\sum_{i=1}^n |v_i| \geq (\sum_{i=1}^n v_i^2)^{1/2} = 1$, we have

$$\mathbb{E}\left[\left|\sum_{i=1}^n v_i z_i^\top z_i\right|\right] \geq \frac{1}{n} \inf_{w \in \{\pm 1\}^n} \mathbb{E}\left[\left|\sum_{i=1}^n w_i z_i^\top z_i\right|\right].$$

It remains to lower bound $\inf_{w \in \{\pm 1\}^n} \mathbb{E}\left[\left|\sum_{i=1}^n w_i z_i^\top z_i\right|\right]$. A constant lower bound is easy to prove. Let p denote the number of $+1$'s and q denote the number of -1's in w, such that $p + q = n$. If $p \neq q$, by Jensen's inequality,

$$\mathbb{E}[|\sum_{i=1}^n w_i z_i^\top z_i|] \geq \mathbb{E}[\sum_{i=1}^{|p-q|} z_i^\top z_i] \geq \mathbb{E}[z_1^\top z_1] = \Omega(d).$$

This can also be seen from the fact that $\mathbb{E}[|X|] \geq |\mathbb{E}[X]|$. If $p = q$, again, an application of Jensen's inequality yields

$$\mathbb{E}[|\sum_{i=1}^n w_i z_i^\top z_i|] \geq \mathbb{E}[|z_1^\top z_1 - z_2^\top z_2|],$$

By definition, $(z_i^\top z_i)_{i=1}^n$ are χ_d^2 random variables; hence, they follow Gamma distributions with $k = d/2$ and $\theta = 2$. By the mean absolute difference formula in https://en.wikipedia.org/wiki/Mean_absolute_difference, $\mathbb{E}[|z_1^\top z_1 - z_2^\top z_2|] = d(4I_{0.5}(d/2, d/2+1) - 2)$, where $I_z(x, y)$ denotes the regularized incomplete Beta function. Numerical calculation shows that $\mathbb{E}[|z_1^\top z_1 - z_2^\top z_2|] = \Theta(\sqrt{d})$. Overall,

$$\inf_{w \in \{\pm 1\}^n} \mathbb{E}[|\sum_{i=1}^n w_i z_i^\top z_i|] = \Omega(\sqrt{d}).$$

Hence, $\mathbb{E}\left[\left|\sum_{i=1}^n v_i z_i^\top z_i\right|\right] \geq \Omega(\sqrt{d}/n)$.

□

E Sum of martingale difference sequences

To upper bound $\sum_{t=H}^{T+H-1} f_t e_t$, one possible approach (Tsiamis et al., 2022) is to use bounds for self-normalized martingales (Abbasi-Yadkori et al., 2011), but the standard self-normalized martingale lemma assumes the noises $(e_t)_{t\geq H}$ are sub-Gaussian.

(Fan et al., 2012, 2017) studies the sum of martingale difference sequence with sub-Weibull distributions, based on which we prove Lemma 6.

Lemma 6. *Let $(\eta_t)_{t\geq 1}$ be a martingale difference sequence adapted to filtration $(\mathcal{F}_t)_{t\geq 1}$. Assume $\eta_t \mid \mathcal{F}_{t-1}$ is θ-sub-Weibull with $\|\eta_t \mid \mathcal{F}_{t-1}\|_{\psi_\theta} \leq K$. Then with probablity at least $1-p$, there exists absolute constants $c, c' > 0$, such that as long as $n \geq c$,*

$$\sum_{t=1}^{T} \eta_t \leq c' K \sqrt{T} (\log(T/p))^{1+\theta^{-1}}.$$

Proof. By the definition of sub-Weibull distributions, $\mathbb{E}[\exp(|\eta_t/K|^\theta) \mid \mathcal{F}_{t-1}] \leq 2$. Define $\epsilon_t = \eta_k/K$. Then by the properties of sub-Weibull distributions, $\mathbb{E}[\epsilon_t^2 \mid \mathcal{F}_{t-1}] \leq a$, for some absolute constant $a > 0$. Hence, $(\epsilon_t)_{t\geq 1}$ satisfies the assumptions required in (Fan et al., 2017, Theorem 3.2) for $\alpha = \theta/(\theta+1)$. Taking $(\phi_t)_{t\geq 1}$ in (Fan et al., 2017, Theorem 3.2) to be ones, we have

$$\mathbb{P}\Big(\sum_{t=1}^{T} \epsilon_t \geq x\sqrt{T}\Big) \leq \exp\Big(-\frac{x^2}{2(c + x^{1+\frac{1}{\theta+1}}/3)}\Big) + 2T\exp\Big(-x^{\frac{\theta}{\theta+1}}\Big).$$

Note that

$$\frac{x^2}{2(c + x^{1+\frac{1}{\theta+1}}/3)} = \frac{x^{\frac{\theta}{\theta+1}}}{2/3 + 2cx^{-1-\frac{1}{\theta+1}}} \geq x^{\frac{\theta}{\theta+1}},$$

if $2cx^{-1-\frac{1}{\theta+1}} \leq 1/3$, that is, $x \geq (6c)^{\frac{\theta+1}{\theta+2}}$. Then, as long as $x \geq 6c$,

$$\mathbb{P}\Big(\sum_{t=1}^{T} \epsilon_t \geq x\sqrt{T}\Big) \leq 3T\exp\Big(-x^{\frac{\theta}{\theta+1}}\Big).$$

Hence,

$$\mathbb{P}\Big(\sum_{t=1}^{T} \epsilon_t \geq 6c + \sqrt{T}(\log(3T/p))^{1+\theta^{-1}}\Big) \leq p.$$

Therefore, there exists absolute constant $c' > 0$, such that if $T \geq \max(36c^2, 3)$, then with probability at least $1-p$,

$$\sum_{t=1}^{T} \theta_t \leq c' K \sqrt{T} (\log(T/p))^{1+\theta^{-1}}.$$

□

Since $\|[f_t]_i \mid \mathcal{F}_{t-1}\|_{\psi_1} = \mathcal{O}(1)$ and $\|e_t \mid \mathcal{F}_{t-1}\|_{\psi_1} \leq E$, the product $[f_t]_i e_t \mid \mathcal{F}_{t-1}$ is $\frac{1}{2}$-sub-Weibull, with the sub-Weibull norm being $\mathcal{O}(E)$. By Lemma 6, with probability at least $1-p$, $\sum_{t=H}^{T+H-1}[f_t]_i e_t = \mathcal{O}(ET^{1/2}\log^3(T/p))$. Then, since $\sum_{t=H}^{T+H-1} f_t e_t$ has $d_h(d_h+1)/2$ components,

$$\Big\|\sum_{t=H}^{T+H-1} f_t e_t\Big\| = \mathcal{O}(d_h E T^{1/2}\log^3(T/p)).$$