Technical University of Munich



Sample-Efficient Reinforcement Learning under Sparse Dynamics

Nikolas Kirschstein

Nora Schneider
1st Supervisor

Georg Manten
2nd Supervisor

Prof. Niki Kilbertus
Reviewer

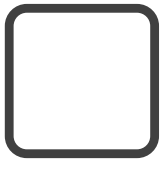
A thesis submitted for the degree of *MSc Informatics*

10 July 2026

Abstract

Sample inefficiency remains a major challenge in reinforcement learning, as learning to act in an unknown dynamical system typically requires costly interactions with that system. Many systems, however, have a sparse structure: a state variable is directly coupled to only a few other variables. We propose SAMBL (*Sparsity-Aware Model-Based Learning*), which exploits this structure to reduce the number of interactions needed to learn a policy to control the system. For the archetypal setting of linear-quadratic control of a d-dimensional system with p inputs, we show that using an ℓ_1-penalised dynamics estimator substantially improves sample efficiency when the true dynamics are row-sparse with at most s nonzeros per row. Our analysis of cumulative regret identifies an initial learning phase, after which cumulative regret grows only logarithmically. Because sparse dynamics can be estimated from fewer samples, SAMBL provably enters this logarithmic low-regret phase sooner than a dense baseline by up to a factor of order $(d + p)/(s \log(d + p))$. Our empirical studies with both synthetic and physical systems confirm this earlier entry into the low-regret regime and the resulting regret improvement. Structure thus makes learning from experience affordable in high-dimensional environments.

Preface



Declaration of AI Usage

The author utilised generative artificial intelligence (AI) as a technical and linguistic tool during the preparation of this thesis. Specifically, *Claude Code* was used to find bugs and improve the performance of existing code, and generate plotting code. *Claude* itself was used to refine the structural flow of the manuscript and assist in the LaTeX formatting. Furthermore, *Grammarly* was used to polish the linguistic style.

The author affirms that all core scientific contributions and theoretical derivations are the result of their original work. All AI-generated suggestions were critically reviewed for technical accuracy and consistency with the underlying research. The author maintains full responsibility for the content, scientific integrity, and final version of this thesis.

Acknowledgements

First and foremost, I thank my supervisors, Nora Schneider and Georg Manten, for their guidance throughout this thesis. Their discussions and sharp questions in weekly meetings, their kind and supportive character, and their thorough feedback on the thesis drafts shaped this work far beyond what I could have achieved on my own.

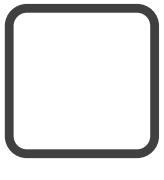
I am equally grateful to Prof. Niki Kilbertus for giving me the opportunity to conduct this research in his group and, what I consider a strong positive exception in academia, being truly and enthusiastically invested in the project by attending supervision meetings, co-brainstorming ideas, and proofreading drafts.

I further thank Prof. Fanny Yang and Prof. Andreas Krause for valuable discussions and feedback on the work-in-progress.

All experiments in this thesis were enabled by the generous computational resources of the Leibniz Supercomputing Centre (LRZ). I also acknowledge having received access to facilities at Helmholtz AI and funding from Studienstiftung.

Finally, I thank my parents and my partner for their unconditional moral support and patience during the past months, without which this thesis would not exist.

Contents



1 Introduction **1**
- 1.1 Learning from Experience . . . 1
- 1.2 The Promise of Sparsity and Sample Efficiency . . . 2
- 1.3 Related Work . . . 3
  - 1.3.1 Continuous-Time Reinforcement Learning . . . 3
  - 1.3.2 Regret in Online Linear-Quadratic Control . . . 4
  - 1.3.3 Sparsity and Structured Estimation . . . 4
  - 1.3.4 Excitation and Identifiability . . . 5
- 1.4 Contributions and Outline . . . 6

2 Background **7**
- 2.1 Reinforcement Learning Paradigms . . . 7
- 2.2 Control Theory meets Reinforcement Learning . . . 8
- 2.3 The Linear-Quadratic Regulator (LQR) . . . 11
  - 2.3.1 Controllability and Stabilisability . . . 11
  - 2.3.2 Optimal Control via Dynamic Programming . . . 13
  - 2.3.3 Closed-Form Solver for the Riccati ODE . . . 15
- 2.4 The Least Absolute Shrinkage and Selection Operator (LASSO) . . . 17
- 2.5 Certainty Equivalence (CE) . . . 19

3 Method **20**
- 3.1 Problem Formulation . . . 20
- 3.2 The SAMBL Algorithm . . . 22
  - 3.2.1 Discretised regression . . . 22
  - 3.2.2 Row-wise LASSO estimation . . . 23
  - 3.2.3 The Riccati ODE and Optimal Feedback . . . 24
  - 3.2.4 Full Algorithm . . . 25
- 3.3 Identifiability Considerations . . . 26
  - 3.3.1 The Drift-Control Asymmetry . . . 26
  - 3.3.2 Bootstrapping Estimates . . . 27

4 Regret Analysis **30**
4.1 Assumptions . . . 30
4.2 Bounding the Estimation Error . . . 33
4.3 Bounding the Per-Episode Regret . . . 36
4.4 Main Results . . . 38
4.5 Satisfying the Assumptions . . . 42

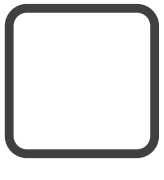
5 Experiments **46**
5.1 Experimental Setup . . . 46
5.1.1 Agents . . . 46
5.1.2 Our LASSO Estimator . . . 47
5.1.3 The Riccati Planner . . . 47
5.1.4 Matched Randomness . . . 48
5.1.5 Evaluation Metrics . . . 48
5.2 Synthetic Systems . . . 49
5.2.1 System Sampling . . . 49
5.2.2 Initialisation . . . 50
5.2.3 Regularisation . . . 51
5.2.4 Main Results . . . 51
5.2.5 Ablations & Robustness . . . 55
5.3 Realistic Systems . . . 58
5.3.1 Coupled Spring-Mass Chain . . . 61
5.3.2 IEEE 39-Bus Power Grid . . . 63
5.4 Discussion . . . 67

6 Conclusion **69**
6.1 Summary . . . 69
6.2 Limitations . . . 70
6.3 Future Work . . . 70

A Appendix **72**
A.1 IEEE 39-Bus Grid Construction . . . 72
A.2 Tabulated Results . . . 74

References **78**

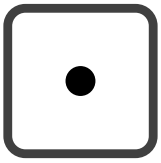
Notation



Symbol	Meaning
d, p	State and control dimensions
$\boldsymbol{x}(t) \in \mathbb{R}^d$, $\boldsymbol{u}(t) \in \mathbb{R}^p$	System state and control input
$\boldsymbol{w}(t) \in \mathbb{R}^d$	Wiener process, a.k.a. Brownian motion
$\boldsymbol{x}_0 \sim P(\boldsymbol{x}_0)$	Initial state, freshly drawn each episode
σ	Diffusion coefficient
$\boldsymbol{\Theta}_\star = [\mathbf{A}_\star\ \mathbf{B}_\star] \in \mathbb{R}^{d\times(d+p)}$	True drift-control matrix
$\boldsymbol{\theta}_{\star,i}$	Row i of $\boldsymbol{\Theta}_\star$ as a column vector
$\mathcal{S}_i = \operatorname{supp}(\boldsymbol{\theta}_{\star,i}), s_i = \lvert\mathcal{S}_i\rvert$	Nonzero entries of row i and their number
$s = \max_i \lvert\mathcal{S}_i\rvert$	Maximum row support size of $\boldsymbol{\Theta}_\star$
$\mathbf{Q} \succeq 0$, $\mathbf{R} \succ 0$	State- and control-cost matrices
$\boldsymbol{\pi}$, $\boldsymbol{\pi}_\star$	Policy and optimal policy
$J(\boldsymbol{\pi})$, $J_\star = J(\boldsymbol{\pi}_\star)$	Expected episode cost and its optimum
$\mathbf{P}_{\boldsymbol{\Theta}}(t)$	Riccati ODE solution for parameters $\boldsymbol{\Theta}$
$\mathbf{K}_{\boldsymbol{\Theta}}(t) = -\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P}_{\boldsymbol{\Theta}}(t)$	Riccati feedback gain, with $\mathbf{K}_\star = \mathbf{K}_{\boldsymbol{\Theta}_\star}$
$F(\boldsymbol{\Theta}) = J(\boldsymbol{\pi}_{\boldsymbol{\Theta}})$	Certainty-mismatch cost
r_m, $R_M = \sum_{m=1}^M r_m$	Per-episode and cumulative regret
T	Episode horizon length
Δt	Euler-Maruyama step size (chosen s.t. Δt divides T)
$H = T/\Delta t$	Euler-Maruyama steps per episode (Δt ensures $H \in \mathbb{N}$)
M	Number of episodes, indexed by m
$N_m = mH$	Total samples after m episodes
m_{explore}	Number of initial pure-exploration episodes
σ_u, $\boldsymbol{\eta}(t) \sim \mathcal{N}(\mathbf{0}, \sigma_u^2\mathbf{I}_p)$	Excitation strength and excitation noise
$\boldsymbol{z}_k = [\boldsymbol{x}_k^\top\ \boldsymbol{u}_k^\top]^\top \in \mathbb{R}^{d+p}$	Concatenated regressor
$\boldsymbol{y}_k = (\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)/\Delta t$	Regression target
$\mathbf{Z} \in \mathbb{R}^{N_m\times(d+p)}$, $\mathbf{Y} \in \mathbb{R}^{N_m\times d}$	Stacked design and response matrices
$\bar{\sigma} = \sigma/\sqrt{\Delta t}$	Regression noise standard deviation
$\boldsymbol{\varepsilon}_k = \bar{\sigma}\,\boldsymbol{w}_k$, $\boldsymbol{w}_k \sim \mathcal{N}(0, \mathbf{I}_d)$	Per-step regression noise and Brownian increment
$\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \bar{\sigma}^2\mathbf{I}_{N_m})$	Stacked noise of the row-i regression ($\neq \boldsymbol{\varepsilon}_k$)
$\hat{\boldsymbol{\Theta}}_m = [\hat{\mathbf{A}}_m\ \hat{\mathbf{B}}_m]$	Parameter estimate after episode m
$\hat{\boldsymbol{\theta}}_{m,i}$	Row i of $\hat{\boldsymbol{\Theta}}_m$ as a column vector
$\boldsymbol{\Delta}_i = \hat{\boldsymbol{\theta}}_{m,i} - \boldsymbol{\theta}_{\star,i}$	Estimation error of row i
λ_m	LASSO regularisation parameter for episode m
c_λ, c_ε	Regularisation schedule and score concentration constants
δ	Failure level of the high-probability guarantees
κ, $\mathcal{C}_i$	Restricted eigenvalue constant and cone of row i
Λ	Per-step estimation complexity
ε_0, m_0	Basin-entry radius and basin-entry episode

Chapter 1

Introduction



1.1 Learning from Experience

Most recent progress in artificial intelligence rests on a single recipe: assemble an enormous corpus of human-generated data and train ever-larger statistical models, on ever-larger computers, to imitate it. The recipe has produced systems that converse fluently, write code, and generate images at a quality that seemed out of reach only a few years ago. However, it may be approaching its limits. The supply of human-generated data is finite and the largest models are already trained on much of the digitised knowledge available (Villalobos et al., 2024). Hence, further upscaling is projected to meet diminishing returns, and fresh training data is increasingly either expensive or synthetic. The resulting systems also remain prone to confidently hallucinating content, hard to interpret, and increasingly unsustainable (Luccioni et al., 2024). According to Silver & Sutton (2025), imitating static human data is asymptotically approaching a performance ceiling, far below what trustworthy and safe intelligent systems require.

Where static data runs out, an agent's own experience does not. A machine can also learn in the way humans and animals do: by acting, observing the consequences, and adjusting its behaviour. This paradigm is known as *reinforcement learning* (RL). Instead of imitating a fixed record of what people have done, an RL agent generates its own stream of observations, based on which to take desirable actions, so its competence is bounded by how much it can interact rather than by how much data exists. This mode of learning is increasingly used at the frontier: from language models that sharpen their reasoning by trial and error (Guo et al., 2025) to robots that acquire motor skills by practising them (Haarnoja et al., 2024).

Experience, however, carries a cost that recorded data does not: the agent's mistakes happen in the real world while it is still learning. In a board game or a simulator, experience is unlimited and errors are free. On a power grid, in a patient, or on a robot, every interaction takes time, causes wear, and carries risk. In such high-stakes applications, a poor decision can become very costly. Learning from experience therefore hinges on *sample efficiency* – how much an agent learns from each interaction. An agent that extracts more information reaches good behaviour sooner and at lower cumulative cost. The excess cost an agent accumulates while still learning, compared with acting optimally from the start, is called *regret*. Achieving a low regret is the quantitative expression of sample-efficient learning from experience.

A key lever for sample efficiency is model-based RL, in which the agent learns an explicit model of the environment's dynamics and derives its actions from it. The model is a compact, reusable summary of all past experience, and it is where prior knowledge about the dynamics can be injected. The agent repeatedly interacts with the unknown dynamical system, and must learn to act well in it while still learning what it is. The systems where this matters are often described by hundreds or thousands of interacting variables, so learning them from scratch is inefficient. But such systems are rarely arbitrary: physical interaction is typically local, so each variable directly influences only a few others, and the system is often highly *structured* despite its size. This thesis studies, in the canonical setting of a linear dynamical system, how exploiting the sparse structure reduces the number of interactions needed to learn effective actions.

1.2 The Promise of Sparsity and Sample Efficiency

Real systems are sparse. For instance, a power grid couples each bus only to its electrical neighbours, and a mechanical structure couples each mass to the ones it is physically joined to. Gene-regulatory, epidemiological, and traffic networks are all governed by sparse interaction graphs (Newman, 2003). Rarely is a real system a dense web in which every variable influences every other. When the dynamics are linear, or linearised at an operating point, the graph sparsity becomes matrix sparsity: most entries of the dynamics-describing matrix are exactly zero. Sparsity is not a modelling convenience but a physical fact, and hence prior knowledge an agent should make use of.

Experience is expensive. In many relevant control settings, interaction is costly and cannot be scaled up at will. While experimentation is possible, it has a price: a robot wears and risks damaging its hardware with every trial, an exploratory dose in a clinical trial is paid for in patient risk, and a control action on a power grid is taken on a safety-critical system in real time. Sample efficiency is therefore not a matter of computational budget but of feasibility and cost. A method that needs fewer, better-chosen interactions to reach good performance makes the difference between a deployable agent and an impractical one whose sample demand explodes in high dimensions.

Structure breaks the curse of dimensionality. There is also a precise theoretical reason to expect sparsity to help. Estimating a dense d-dimensional linear system needs a number of samples that grows with the ambient parameter count, on the order of d^2. If instead each row of the dynamics depends on at most $s \ll d$ variables, the sample cost drops to the order of $s \log d$. This is a well-known gap between dense and sparse estimation in high-dimensional statistics that may pay off in model-based RL.

The question this thesis asks is what the statistical gap buys *for control*: how much sooner does a sparsity-aware agent learn to act well, and how does that advantage scale as the dimension grows while the sparsity level stays fixed? We study this question in the canonical setting that contains all of its ingredients: episodic, continuous-time, linear-quadratic (LQ) control of a d-dimensional system with p control inputs, whose dynamics parameters are unknown and row-sparse.

1.3 Related Work

This thesis sits at the intersection of three lines of research: reinforcement learning in continuous time, regret analysis of online linear-quadratic control, and sparse high-dimensional estimation. We review each in turn, and then the treatment of excitation and identifiability that ties them together. Throughout, the recurring axis of comparison is whether a method exploits *structure* in the dynamics and whether its guarantees are stated as *regret*. Note that for finite-horizon problems, regret is stated in terms of the number of episodes M, while for infinite-horizon problems it is stated in terms of the number of overall time steps T.*

1.3.1 Continuous-Time Reinforcement Learning

Most reinforcement learning is formulated in discrete time, but the systems it is meant to control usually evolve continuously. The continuous-time formulation was pioneered by Doya (2000), who derived value-estimation and actor-critic algorithms from the Hamilton-Jacobi-Bellman equation and a continuous-time temporal-difference error, without discretising time, state, or actions. The area has seen a revival, driven on the theoretical side by the exploratory stochastic-control formulation of Wang et al. (2020), who add an entropy regulariser to a controlled diffusion and show that in the linear-quadratic case the optimal exploratory control is Gaussian – a principled justification for Gaussian exploration. This has grown into a substantial body of work on q-learning for diffusions (Tang & Zhou, 2024), on reconciling the discrete-time and continuous-time formulations (Carmona & Lauriere, 2025; Berner et al., 2026), and on model-based methods that learn the dynamics as neural ordinary differential equations (Yildiz et al., 2021). A particularly relevant strand studies efficient exploration in continuous-time model-based RL through optimism over calibrated dynamics models, establishing sublinear-regret guarantees (Treven et al., 2023, 2024; Iten et al., 2026).

Regret guarantees in continuous time are more recent. Gao & Zhou (2024) obtain logarithmic-in-time regret, with a matching lower bound, for continuous-time average-reward Markov decision processes. For the linear-quadratic case specifically, Huang et al. (2025) give a model-free actor-critic that learns the optimal policy directly, attaining sublinear $\widetilde{O}(M^{3/4})$ regret over M episodes without ever estimating the dynamics. Closest to our setting is the work of Basei et al. (2022), who study *episodic, finite-horizon, continuous-time linear-quadratic* reinforcement learning: the agent learns the dynamics coefficients across M episodes, and a greedy (purely exploiting) least-squares algorithm attains $O\big((\ln M)(\ln\ln M)\big)$ regret, non-asymptotically. Crucially, their analysis assumes *dense* parameter matrices, so its sample complexity scales with the full ambient dimension. We adopt their episodic continuous-time setting and ask the orthogonal question of how exploiting sparsity sharpens the dimension dependence.

*Not to be confused with the fixed episode horizon T of our episodic setting (Chapter 3).

1.3.2 Regret in Online Linear-Quadratic Control

The regret theory that makes this dimension dependence precise is most developed in discrete time, where the linear-quadratic regulator (LQR) with unknown dynamics is the canonical testbed for the sample complexity of control; its theory is now fairly mature. Dean et al. (2020) gave the first end-to-end finite-sample sub-optimality guarantee, identifying the system with a coarse least-squares estimate and synthesising a robust controller. Mania et al. (2019) sharpen this for certainty-equivalent control, i.e., planning as if the current estimate were the true system, showing that its sub-optimality gap scales with the *square* of the parameter-estimation error, so a plug-in estimate is already near-optimal. This fast-rate guarantee underlies the certainty-equivalent design we adopt. The online problem, where the agent is judged on the regret accumulated while learning rather than on a final sub-optimality gap, had in fact been opened earlier by Abbasi-Yadkori & Szepesvári (2011), whose optimism-based algorithm first attained $\sqrt{T}$ regret for adaptive LQR. The picture is by now sharp: Simchowitz & Foster (2020) prove that the optimal regret for single-trajectory average-cost online LQR is $\Theta(\sqrt{T})$, attained by certainty-equivalent control with a vanishing amount of injected Gaussian noise, and that no method can do better. In particular, logarithmic regret is *impossible* in this setting despite the problem's apparent strong convexity.

Logarithmic regret becomes attainable only when the problem is made easier along some axis. It is classical that adaptive control of stochastic regression models incurs only logarithmic cost in the minimum-variance regime (Lai, 1986). More recently, logarithmic regret has been established for online control with *known* dynamics and adversarial strongly-convex costs (Agarwal et al., 2019), when the dynamics are only *partially* unknown (Jedra & Proutiere, 2022), and for reinforcement learning with linear function approximation (He et al., 2021). The episodic finite-horizon structure we work in – inherited from Basei et al. – is one such favourable axis, and underlies the logarithmic regret we establish. The reduction of adaptive control to non-asymptotic estimation that all of this work relies on is surveyed by Tsiamis et al. (2023) and Ziemann et al. (2023). The underlying idea that curvature turns slow $1/\sqrt{n}$ rates into fast $1/n$ rates is a general phenomenon studied by van Erven et al. (2015). Across the continuous- and discrete-time literature alike, these analyses share one feature: the estimator is dense, so its sample complexity, and hence its regret constant, carries the full ambient dimension of the dynamics.

1.3.3 Sparsity and Structured Estimation

The statistical benefit of sparsity is well understood. The LASSO (least squares with a sparsity-inducing ℓ_1 penalty) recovers a sparse parameter from far fewer samples than its ambient dimension would suggest, at a rate governed by the number of nonzeros rather than the dimension, under a restricted-eigenvalue condition on the design (Bühlmann & Van De Geer, 2011; Hastie et al., 2015; Wainwright, 2019). Transferring this benefit to dynamical systems, Fattahi et al. (2019) show that an ℓ_1-regularised estimator identifies

a sparse linear system from a single trajectory with a sample complexity that grows only logarithmically in the ambient dimension. However, crucially, this is open-loop *identification*, with no control objective or regret. Sparse dictionary learning of dynamics has also been combined with model-based RL for nonlinear systems, improving sample efficiency and interpretability, though without regret guarantees (Zolman et al., 2025). The recovery of sparse structure by continuous ℓ_1-regularised optimisation is by now a general tool.

The one prior work that combines sparse dynamics, linear-quadratic control, *and* regret is Ibrahimi et al. (2012). They study a high-dimensional LQ system whose combined parameter matrix is row-sparse, estimate it with a per-row ℓ_1 LASSO, and prove a regret bound growing as $\sqrt{T}$ on a single continuing trajectory, with sparsity entering the sample-complexity constants. Their setting is nonetheless quite different from ours: it is discrete-time, single-trajectory, and average-cost, it relies on a supplied stabilising controller and a mutual-incoherence condition, and its regret is *linear* in the state dimension. This thesis, however, studies row-sparse estimation in the *continuous-time, episodic, finite-horizon* setting and characterises how the sparse-versus-dense advantage *increases with the state dimension*, an effect none of the above analyses isolate.

Sparsity has also been exploited in networked control, but on the *controller* rather than the dynamics. In power systems, sparsity-promoting optimal control designs wide-area feedback with few communication links (Dörfler et al., 2014), and data-driven reinforcement learning has been used to the same end (Dizche et al., 2019; Coulson et al., 2019; Yan & Xu, 2020). These methods presuppose a known model and place sparsity on the feedback policy to reduce communication. In contrast, we place it on the *identified dynamics* and learn them online, which is the actual estimation problem an operator faces when the model itself is uncertain.

1.3.4 Excitation and Identifiability

The three threads above meet at one question: does a closed loop that *exploits* a good policy still generate data informative enough to *identify* the system? Whether it does is a classical concern in adaptive control. In the deterministic case, identifiability requires *persistent excitation* of the input (Green & Moore, 1986), and the standard remedy when a feedback law would otherwise starve identification is to inject exploratory perturbations (Ljung, 1999; Söderström & Stoica, 1989). The tension between exploiting a good policy and exploring the system enough to keep learning is exactly what the regret framework quantifies. As Wang et al. (2025) show for nonlinear control, the very same problem can admit logarithmic regret when the optimal policy is persistently exciting and only $\sqrt{T}$ regret when it is not. Our finite-horizon setting is favourable in this respect, because the time-variation of the optimal policy makes the closed loop self-exploring (Basei et al., 2022). However, we show that this self-exploration, while enough for a dense estimator, does not by itself supply the curvature the LASSO's guarantees require. Identifying this gap and its solution is one of the contributions of this thesis.

1.4 Contributions and Outline

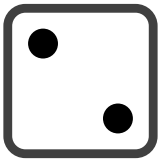
In the setting established above – episodic, continuous-time LQ control of an unknown row-sparse system – the overarching research question is what exploiting that sparsity is worth, measured in regret and the statistical speed of learning. After a thorough review of the necessary theoretical background in Chapter 2, which a knowledgeable reader may skip or only consult for reference, the subsequent chapters carry the central contributions of this thesis:

- **A sparsity-aware RL algorithm with identifiability analysis (Chapter 3).** We propose SAMBL (*Sparsity-Aware Model-Based Learning*), a certainty-equivalent model-based RL algorithm with a row-wise LASSO estimator that exploits the row-sparsity of the dynamics. We then analyse when the self-generated closed-loop data suffice to identify the system, exposing a structural asymmetry between the readily identified drift (state-coupling) block and the weakly excited control block, a degenerate zero-initialisation trap, and the minimal solution that we adopt.
- **A rigorous regret analysis exhibiting the sparse advantage (Chapter 4).** We prove a regret decomposition into a finite transient and a logarithmic tail, driven by a sparse estimation rate of order ds rather than the dense $d(d+p)$. This yields a certified basin-entry speedup of order $(d+p)/(s\log(d+p))$: SAMBL provably reaches the low-regret regime sooner, by a factor that *grows with the dimension.*A doubling schedule turns this into a logarithmic regret guarantee independent of basin entry, adapting the dense result of Basei et al. (2022) to the sparse setting.
- **An empirical study across synthetic and physical systems (Chapter 5).** On a controlled synthetic ensemble, a scalable spring-mass chain, and the IEEE 39-bus power grid, we show that the sparse advantage materialises and, as predicted, increases with dimension, from roughly 1.3× at moderate dimension to 1.9–2.3× at $d = 500$. Furthermore, the advantage grows as the rows get sparser and reverses once they are no longer sparse. We further examine a high-dimensional identifiability collapse of pure greedy play that the exploration episode averts, and we evaluate the robustness of the advantage to the LASSO penalty, excitation, cost weighting, and discretisation. Our source code is publicly available:

  https://github.com/nkirschi/SAMBL

Chapter 2

Background



Reinforcement learning (RL) studies how an agent should act in an unknown environment to minimise a long-run cost, learning from the consequences of its own actions rather than from a fixed dataset of labelled examples. This chapter collects the concepts on which the rest of the thesis builds. We first locate our setting among the standard RL paradigms (Section 2.1). Then, we formalise the control problem as a stochastic differential equation and specialise it to the linear-quadratic case, whose optimal policy is given in closed form (Sections 2.2 and 2.3). Finally, we review the two statistical and algorithmic ingredients the method rests on: the LASSO as a sparsity-exploiting estimator (Section 2.4), and the certainty-equivalence principle that turns an estimate into a policy (Section 2.5).

2.1 Reinforcement Learning Paradigms

The RL literature is organised along several orthogonal axes. We situate the thesis by explaining each axis and stating the choice this work makes and why.

Continuous-time vs. discrete-time. Most RL is formulated in discrete time as a Markov decision process, whereas the physical systems we care about evolve continuously. We thus adopt the *continuous-time* formulation and treat the discretisation as an artefact of estimation and simulation rather than a property of the system. As Chapter 5 shows, this distinction is not merely aesthetic: the identifiability of the dynamics and the estimator's behaviour depend on the continuous-time structure, and the aggregate information in a trajectory is invariant to the discretisation step.

Model-based vs. model-free. A model-free agent learns a value function or a policy directly, without ever forming an explicit model of the environment's dynamics. A *model-based* agent instead estimates the dynamics and derives a policy from that estimate, for example by planning. Model-based methods tend to be more sample-efficient, because a dynamics model is a compact and reusable summary of experience, and because structural prior knowledge about the environment can be injected directly into the model. Our approach is therefore model-based: the agent estimates the dynamics and plans against it, and the sparsity prior enters precisely through the estimator.

Finite-horizon vs. infinite-horizon. An infinite-horizon problem optimises for a single stationary policy that minimises a discounted or average cost forever, whereas a finite-horizon problem optimises over a fixed window $[0, T]$ in episodes and admits a time-varying optimal policy. We work in the *finite-horizon*, episodic setting, because the time variation of the optimal policy allows to identify the dynamics without deliberate excitation, a property that the infinite-horizon problem lacks. This makes our experiments less dependent on concrete choices of exploration strategy.

Online vs. offline. An offline agent learns from a fixed, previously collected dataset, whereas an online agent learns while it acts and is judged on the cost it incurs during learning. Our setting is *online*: the agent controls the system over M episodes and its performance is measured by the excess cost paid for not knowing the dynamics from the outset. Sample efficiency is therefore not a matter of storage but of cost: every poorly informed action is paid for in regret.

Off-policy vs. on-policy. An on-policy agent improves its behaviour using only data generated by the policy it is currently executing. In contrast, an *off-policy* agent may also learn from data produced by other policies, such as earlier versions of itself or a separate exploratory policy. Our scheme is off-policy: the estimator pools all trajectories collected so far, generated by the sequence of different policies that successive estimates induced, and re-fits the dynamics on their union. Reusing this heterogeneous history is what contributes to sample efficiency.

2.2 Control Theory meets Reinforcement Learning

Model-based RL and the classical engineering discipline of control theory share quite some overlap that is worth shedding light on. Control theory is concerned with steering a dynamical system toward desired behaviour, classically by *feedback*: measuring the state and choosing inputs that counteract deviations from a target. Where RL starts from an unknown environment and asks how good behaviour can be *learnt* from interaction, control theory traditionally starts from a mathematical model of the system and asks how good behaviour can be *derived* from it, with guarantees. The two traditions meet in the problem this thesis studies: an agent that estimates a model from interaction, as in RL, and plans against the estimate exactly, as in control. This section develops the formal object the two fields share and assembles a dictionary between their vocabularies.

Reinforcement learning and control theory study similar problems under different names, and the thesis uses both vocabularies freely. The agent is the *controller*, its environment the *plant*, and its actions the *control inputs*. The transition kernel of a Markov decision process becomes the drift-diffusion pair of (2.1), and a policy becomes a *feedback law*. Where RL maximises cumulative reward, control theory *minimises* cumulative cost, so rewards correspond to negative cost rates and returns to negative costs-to-go, and we adopt the minimisation convention throughout.

RL	**control theory**
agent	controller
environment	plant
state	state
action	control input
policy	feedback law
transition kernel	drift and diffusion
reward	(negative) cost rate
return	(negative) episode cost
value function	(negative) cost-to-go
Bellman equation	HJB equation
planning	controller synthesis
model learning	system identification
exploration	excitation, probing
regret	transient performance loss

Table 1: Translation guide between the vocabularies of reinforcement learning and control theory. We use all terms interchangeably in this thesis.

On a deeper level, the Bellman equation of dynamic programming becomes, in continuous time, the Hamilton-Jacobi-Bellman partial differential equation of Section 2.3, learning a model of the environment is *system identification*, and the exploration-exploitation dilemma reappears as the classical tension between *excitation* and regulation. Even the paradigms of Section 2.1 have classical counterparts in *adaptive control*, the branch of control theory that regulates systems whose parameters are unknown: model-based RL corresponds to *indirect* adaptive control, which identifies a model and derives a controller from it, and model-free RL to *direct* adaptive control, which tunes the controller without an intermediate model. Table 1 summarises the terminology into a dictionary.

A controlled dynamical system in continuous time is naturally described by a *stochastic differential equation* (SDE). For a state $\boldsymbol{x}(t) \in \mathbb{R}^d$ and a control input $\boldsymbol{u}(t) \in \mathbb{R}^p$ over the horizon $[0, T]$, it reads

$$\mathrm{d}\boldsymbol{x}(t) = \underbrace{\boldsymbol{\mu}(\boldsymbol{x}(t), \boldsymbol{u}(t))}_{\text{drift} \in \mathbb{R}^d} \mathrm{d}t + \underbrace{\boldsymbol{\Sigma}(\boldsymbol{x}(t), \boldsymbol{u}(t))}_{\text{diffusion} \in \mathbb{R}^{d \times d}} \mathrm{d}\boldsymbol{w}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.1}$$

where the *drift* $\boldsymbol{\mu} : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d$ is the deterministic instantaneous velocity, the *diffusion* $\boldsymbol{\Sigma} : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^{d \times d}$ scales the injected noise, and $\boldsymbol{w}$ is a d-dimensional Brownian motion – an important special case of stochastic processes, for whose general treatment we refer to Yong & Zhou (1999, Chapter 1).

Definition B.1 (Brownian motion)**.** A d-dimensional *Brownian motion* (or *Wiener process*) is a stochastic process $(\boldsymbol{w}(t))_{t\geq 0}$ with values in $\mathbb{R}^d$ and $\boldsymbol{w}(0) = \mathbf{0}$ such that

(i) its increments over disjoint time intervals are independent,

(ii) each increment is Gaussian with $\boldsymbol{w}(t) - \boldsymbol{w}(s) \sim \mathcal{N}\big(\mathbf{0}, (t-s)\,\mathbf{I}\big)$ for $0 \leq s < t$, and

(iii) its sample paths are continuous.

Brownian motion arises as the scaling limit of a random walk, and property (ii) fixes its characteristic scale: an increment over a step Δt has standard deviation $\sqrt{\Delta t}$ rather than Δt. This $\sqrt{\Delta t}$ scaling is why the sample paths, though continuous, are nowhere differentiable, and why ordinary calculus does not apply to them. Because $\boldsymbol{w}$ has no derivative, the differential notation of (2.1) is shorthand for the defining integral equation $\boldsymbol{x}(t) = \boldsymbol{x}_0 + \int_0^t \boldsymbol{\mu}\,\mathrm{d}s + \int_0^t \boldsymbol{\Sigma}\,\mathrm{d}\boldsymbol{w}(s)$, whose last term is an *Itô integral*: the limit of Riemann sums that evaluate the integrand at the left endpoint of each subinterval, so that the integrand never anticipates the noise. A unique solution exists whenever drift and diffusion are Lipschitz in the state and grow at most linearly (Yong & Zhou, 1999, Chapter 1). These conditions are trivially met by the linear drift and constant diffusion used in this thesis.

The drift-diffusion pair plays the role that the transition kernel plays in a discrete-time Markov decision process. An Euler-Maruyama discretisation makes the analogy explicit: on the grid $t_k \coloneqq k\,\Delta t$, the discretised state evolves as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{\mu}(\boldsymbol{x}_k, \boldsymbol{u}_k)\,\Delta t + \boldsymbol{\Sigma}(\boldsymbol{x}_k, \boldsymbol{u}_k)\,\sqrt{\Delta t}\,\boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{2.2}$$

with $\boldsymbol{u}_k \coloneqq \boldsymbol{\pi}(t_k, \boldsymbol{x}_k)$ and $\sqrt{\Delta t}\,\boldsymbol{w}_k$ standing for the Brownian increment $\boldsymbol{w}(t_{k+1}) - \boldsymbol{w}(t_k)$, whose law is given by Definition B.1(ii). Each step is thus a Gaussian transition whose mean follows the drift and whose covariance follows the diffusion.

A policy $\boldsymbol{\pi} : [0, T] \times \mathbb{R}^d \to \mathbb{R}^p$ maps time and state to a control $\boldsymbol{u}(t) = \boldsymbol{\pi}(t, \boldsymbol{x}(t))$, which closes the feedback loop: substituting it into (2.1) turns the controlled system into an autonomous stochastic process. A cost rate $\ell : \mathbb{R}^d \times \mathbb{R}^p \to [0, \infty)$ assigns an instantaneous cost to every state–control pair, and its accumulation along the trajectory defines the expected cumulative cost of a policy,

$$J(\boldsymbol{\pi}) \coloneqq \mathbb{E}\left[\int_0^T \ell\big(\boldsymbol{x}(t), \boldsymbol{u}(t)\big)\,\mathrm{d}t\right], \tag{2.3}$$

with the expectation taken over the Brownian motion $\boldsymbol{w}$ and any randomness in the initial state $\boldsymbol{x}_0$. The optimisation problem is to attain the optimal cost $J_\star \coloneqq \inf_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$, with the infimum running over all policies. Dynamics (2.1) and cost (2.3) together constitute the continuous-time counterpart of the reinforcement-learning problem.

The formulation (2.1) admits a spectrum of modelling choices. At one extreme, we may parametrise the drift $\boldsymbol{\mu}$ by a flexible function approximator, such as a neural network, and estimate the full state-dependent diffusion $\boldsymbol{\Sigma}$, yielding a very general but

data-hungry model. At the other extreme, we impose strong structure: a *linear* drift $\boldsymbol{\mu}(\boldsymbol{x}, \boldsymbol{u}) = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$, with fixed matrices $\mathbf{A} \in \mathbb{R}^{d\times d}$ and $\mathbf{B} \in \mathbb{R}^{d\times p}$, and an isotropic, state-independent diffusion $\boldsymbol{\Sigma} = \sigma\mathbf{I}$ with a scalar noise level $\sigma > 0$. This linear regime is the setting studied in this thesis. It is the narrowest of the useful models, but it is the one for which optimal control is exactly solvable and for which sharp finite-sample statements are within reach. These properties together facilitate a clean study of the value of structure, which is the main concern of this thesis.

2.3 The Linear-Quadratic Regulator (LQR)

Specialising the general problem of Section 2.2 to linear dynamics and a quadratic cost yields the *linear-quadratic regulator* (LQR), the cornerstone of optimal control. The state now evolves according to the linear SDE

$$\mathrm{d}\boldsymbol{x}(t) = \big(\mathbf{A}\,\boldsymbol{x}(t) + \mathbf{B}\,\boldsymbol{u}(t)\big)\,\mathrm{d}t + \sigma\,\mathrm{d}\boldsymbol{w}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.4}$$

where the *drift matrix* $\mathbf{A} \in \mathbb{R}^{d\times d}$ governs the autonomous motion of the state, the *input matrix* $\mathbf{B} \in \mathbb{R}^{d\times p}$ maps the control into the state space, and $\sigma > 0$ scales the isotropic noise. The cost rate is quadratic in both arguments:

$$\ell(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{x}^\top \mathbf{Q}\,\boldsymbol{x} + \boldsymbol{u}^\top \mathbf{R}\,\boldsymbol{u}, \tag{2.5}$$

where the symmetric *state-cost matrix* $\mathbf{Q} \succeq 0$ (positive semi-definite) penalises deviations of the state from the origin, and the symmetric *control-cost matrix* $\mathbf{R} \succ 0$ (positive definite) penalises control effort. Therefore, the objective (2.3) trades regulation accuracy against actuation cost. The solution of this problem is a central fact from control theory on which the rest of the thesis rests, so we derive it in full.

Throughout this section only, the parameters $(\mathbf{A}, \mathbf{B})$ are assumed *known*, whereas in Chapter 3 we no longer require this. We first delimit which linear systems can be controlled at all (Section 2.3.1), then derive the optimal policy from dynamic programming (Section 2.3.2), and finally show that the resulting Riccati equation admits a closed-form solution (Section 2.3.3).

2.3.1 Controllability and Stabilisability

For a control problem to be well-posed, the input must be able to steer the state in the directions that matter. This ability is captured by two classical notions, which we state for the noiseless part $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$ of the dynamics (2.4).

Definition B.2 (Controllability)**.** The pair $(\mathbf{A}, \mathbf{B})$ is *controllable* if for every pair of states $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{R}^d$ and every time $t_1 > 0$ there exists a control $\boldsymbol{u} : [0, t_1] \to \mathbb{R}^p$ steering the system $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$ from $\boldsymbol{x}(0) = \boldsymbol{x}_0$ to $\boldsymbol{x}(t_1) = \boldsymbol{x}_1$.

Lemma B.1 (Hautus test for controllability)**.** *For* $\mathbf{A} \in \mathbb{R}^{d\times d}$ *and* $\mathbf{B} \in \mathbb{R}^{d\times p}$*, the following are equivalent (Hespanha, 2018; Chen, 2014):*

(i) The pair $(\mathbf{A}, \mathbf{B})$ *is controllable.*

(ii) $\operatorname{rank}[\,\mathbf{B}\ \ \mathbf{AB}\ \ \cdots\ \ \mathbf{A}^{d-1}\mathbf{B}\,] = d.$ *(Kalman rank condition)*

(iii) $\operatorname{rank}[\,\lambda\mathbf{I} - \mathbf{A}\ \ \mathbf{B}\,] = d$ *for every* $\lambda \in \mathbb{C}$*.* *(Hautus condition)*

Since $\lambda\mathbf{I} - \mathbf{A}$ alone has full rank whenever λ is not an eigenvalue of $\mathbf{A}$, condition (iii) needs to be checked only at the eigenvalues of $\mathbf{A}$. The Hautus condition makes the failure mode transparent: controllability fails precisely when some left eigenvector $\boldsymbol{v}$ of $\mathbf{A}$ (a vector with $\boldsymbol{v}^\top\mathbf{A} = \lambda\boldsymbol{v}^\top$) is orthogonal to every column of $\mathbf{B}$, in which case the mode $\boldsymbol{v}$ evolves untouched by any control. However, requiring *every* mode to be steerable is often more than an application needs. For regulation it suffices that the modes which would otherwise grow can be steered, which is the weaker notion of stabilisability.

Definition B.3 (Stabilisability)**.** The pair $(\mathbf{A}, \mathbf{B})$ is *stabilisable* if for every initial state $\boldsymbol{x}(0) \in \mathbb{R}^d$ there exists an input signal $\boldsymbol{u} : [0, \infty) \to \mathbb{R}^p$ such that the solution of $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$ satisfies $\lim_{t\to\infty} \boldsymbol{x}(t) = \mathbf{0}$.

Lemma B.2 (Hautus test for stabilisability)**.** *For* $\mathbf{A} \in \mathbb{R}^{d\times d}$ *and* $\mathbf{B} \in \mathbb{R}^{d\times p}$*, the following are equivalent (Hespanha, 2018; Chen, 2014):*

(i) The pair $(\mathbf{A}, \mathbf{B})$ *is stabilisable.*

(ii) There exists $\mathbf{K} \in \mathbb{R}^{p\times d}$ *such that* $\mathbf{A} + \mathbf{BK}$ *is Hurwitz, i.e., all its eigenvalues have negative real part.* *(feedback stabilisation)*

(iii) $\operatorname{rank}[\,\lambda\mathbf{I} - \mathbf{A}\ \ \mathbf{B}\,] = d$ *for every* $\lambda \in \mathbb{C}$ *with* $\operatorname{Re}(\lambda) \geq 0$*.* *(Hautus condition)*

Condition (ii) is the operational form: the constant linear feedback law $\boldsymbol{u} = \mathbf{K}\boldsymbol{x}$ closes the loop into $\dot{\boldsymbol{x}} = (\mathbf{A} + \mathbf{BK})\boldsymbol{x}$, whose Hurwitz property makes every trajectory decay to the origin. That the seemingly much weaker condition (i) implies it means nothing is lost by passing from arbitrary input signals to constant linear feedback. The uncontrollable modes evolve autonomously, unaffected by any input, so stabilisability forces them to decay on their own, while the controllable modes can be relocated into the stable half-plane by a suitable feedback. The Hautus condition (iii) correspondingly relaxes its controllability counterpart from all modes to the unstable ones: uncontrollable modes are permitted as long as they are already stable.

For the finite-horizon problem of this thesis neither notion is needed for well-posedness. Yet, stabilisability is the standing assumption of the infinite-horizon problem, where a constant stabilising policy must exist. Crucially for us, the condition separates benign from degenerate systems. We therefore invoke it when constructing well-posed systems for the experiments (Chapter 5).

2.3.2 Optimal Control via Dynamic Programming

Definition B.4 (Value Function)**.** The *value function* is defined as the optimal cost-to-go under the dynamics (2.4):

$$V(t, \boldsymbol{x}(t)) \coloneqq \inf_{\boldsymbol{\pi}} \mathbb{E}\left[\int_t^T \big(\boldsymbol{x}(s)^\top \mathbf{Q}\, \boldsymbol{x}(s) + \boldsymbol{u}(s)^\top \mathbf{R}\, \boldsymbol{u}(s)\big)\mathrm{d}s \;\middle|\; \boldsymbol{x}(t) = \boldsymbol{x}\right]. \tag{2.6}$$

The value function measures the least expected cost still to come from state $\boldsymbol{x}$ at time t. Recall that the optimisation problem of Section 2.2 asks for $V(0, \boldsymbol{x}_0)$ and a policy attaining it. The key structural property of V is the *dynamic programming principle*: for any slice length $0 \le h \le T - t$, acting optimally on $[t, T]$ means acting optimally on the initial slice $[t, t+h]$ and continuing optimally from wherever the system lands,

$$V(t, \boldsymbol{x}(t)) = \inf_{\boldsymbol{\pi}} \mathbb{E}\left[\int_t^{t+h} \ell\big(\boldsymbol{x}(s), \boldsymbol{u}(s)\big)\mathrm{d}s \;+\; V\big(t+h, \boldsymbol{x}(t+h)\big) \;\middle|\; \boldsymbol{x}(t) = \boldsymbol{x}\right]. \tag{2.7}$$

The principle holds because the dynamics are Markov: past and future interact only through the current state, so everything after $t+h$ is summarised by the value at the state reached then, and a policy suboptimal on the tail could be improved there without affecting the head. In this vein, the value function can be re-expressed as the solution of a partial differential equation.

Proposition B.1 (HJB equation)**.** *Let the value function (2.6) be continuously differentiable in t and twice continuously differentiable in $\boldsymbol{x}$. Then it satisfies the* Hamilton–Jacobi–Bellman *(HJB) equation*

$$-\partial_t V(t, \boldsymbol{x}(t)) = \min_{\boldsymbol{u} \in \mathbb{R}^p} \left\{ \underbrace{\boldsymbol{x}^\top \mathbf{Q} \boldsymbol{x} + \boldsymbol{u}^\top \mathbf{R} \boldsymbol{u}}_{\textit{instantaneous cost}} + \underbrace{\nabla_{\boldsymbol{x}} V(t, \boldsymbol{x}(t))^\top (\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u})}_{\textit{cost transport along drift}} + \underbrace{\frac{\sigma^2}{2} \operatorname{tr}\big(\nabla_{\boldsymbol{x}}^2 V(t, \boldsymbol{x}(t))\big)}_{\textit{noise-induced cost}} \right\} \tag{2.8}$$

with terminal condition $V(T, \cdot) = 0$.

Proof. The terminal condition is immediate: at $t = T$ no running cost remains. Expanding the continuation term in (2.7) using Itô's formula (Yong & Zhou, 1999, Theorem 5.5) from stochastic calculus yields

$$\begin{aligned} V\big(t+h, \boldsymbol{x}(t+h)\big) = V(t, \boldsymbol{x}(t)) \\ + \int_t^{t+h} \Big(\partial_t V(t, \boldsymbol{x}(t)) + \nabla_{\boldsymbol{x}} V(t, \boldsymbol{x}(t))^\top (\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}) + \frac{\sigma^2}{2} \operatorname{tr}\big(\nabla_{\boldsymbol{x}}^2 V(t, \boldsymbol{x}(t))\big)\Big)\mathrm{d}s \\ + \int_t^{t+h} \sigma\, \nabla_{\boldsymbol{x}} V(t, \boldsymbol{x}(t))^\top \mathrm{d}\boldsymbol{w}. \end{aligned}$$

Taking the conditional expectation eliminates the last term: an Itô integral never anticipates the noise, so its increments have mean zero. Substituting the remainder into (2.7) cancels the $V(t, \boldsymbol{x}(t))$ appearing on both sides. Dividing by h and sending $h \to 0$ then replaces the time integrals by their integrands at time t. In this limit, the infimum over policies on the shrinking slice collapses to a minimum over the single control value applied at $(t, \boldsymbol{x}(t))$. This is a minimum rather than an infimum, because the expression is a strictly convex quadratic in $\boldsymbol{u}$ and hence attains it. Rearranging yields (2.8). □

The right-hand side weighs the three forces acting on the cost-to-go: the instantaneous cost, the transport of the value along the drift, and the sensitivity of the cost to the process noise. The terminal condition encodes that no cost accrues beyond the horizon. The smoothness assumption is fulfilled in our setting: the derivation below exhibits an explicit solution that is quadratic in the state and thus smooth.

For linear dynamics and quadratic cost, the value function is itself quadratic in the state. Furthermore, the symmetry of dynamics and cost under $(\boldsymbol{x}, \boldsymbol{u}) \mapsto (-\boldsymbol{x}, -\boldsymbol{u})$ rules out a linear term. We therefore make the quadratic ansatz

$$V(t, \boldsymbol{x}(t)) = \boldsymbol{x}^\top \mathbf{P}(t)\, \boldsymbol{x} + c(t)$$

with a symmetric matrix $\mathbf{P}(t) \in \mathbb{R}^{d \times d}$ and a scalar offset $c(t) \in \mathbb{R}$, so that $\nabla_{\boldsymbol{x}} V = 2\mathbf{P}(t)\boldsymbol{x}$ and $\nabla_{\boldsymbol{x}}^2 V = 2\mathbf{P}(t)$. The only $\boldsymbol{u}$-dependent and therefore minimisation-relevant terms in (2.8) are then $\boldsymbol{u}^\top \mathbf{R}\boldsymbol{u} + 2\boldsymbol{x}^\top \mathbf{P}\mathbf{B}\boldsymbol{u}$. Completing the square as

$$\boldsymbol{u}^\top \mathbf{R}\boldsymbol{u} + 2\boldsymbol{x}^\top \mathbf{P}\mathbf{B}\boldsymbol{u} = \big(\boldsymbol{u} + \mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}\boldsymbol{x}\big)^\top \mathbf{R}\, \big(\boldsymbol{u} + \mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}\boldsymbol{x}\big) - \boldsymbol{x}^\top \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}\boldsymbol{x},$$

shows that, provided $\mathbf{R} \succ 0$, there is a unique minimiser, and it is *linear* in the state:

$$\boldsymbol{\pi}_\star(t, \boldsymbol{x}(t)) = \mathbf{K}(t)\, \boldsymbol{x}, \qquad \mathbf{K}(t) = -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}(t). \tag{2.9}$$

The optimal policy is thus a time-varying linear feedback with gain $\mathbf{K}(t)$, determined entirely by $\mathbf{P}(t)$. Substituting the minimiser back into (2.8) and symmetrising the drift term via $2\boldsymbol{x}^\top \mathbf{P}\mathbf{A}\boldsymbol{x} = \boldsymbol{x}^\top(\mathbf{A}^\top \mathbf{P} + \mathbf{P}\mathbf{A})\boldsymbol{x}$ leaves an identity that must hold for every $\boldsymbol{x}$,

$$-\boldsymbol{x}^\top \dot{\mathbf{P}}(t)\, \boldsymbol{x} - \dot{c}(t) = \boldsymbol{x}^\top \big(\mathbf{Q} + \mathbf{A}^\top \mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}(t)\big)\boldsymbol{x} + \sigma^2 \operatorname{tr}\big(\mathbf{P}(t)\big).$$

Comparing the parts quadratic in $\boldsymbol{x}$ forces $\mathbf{P}$ to satisfy the backward *Riccati differential equation*

$$-\dot{\mathbf{P}}(t) = \mathbf{A}^\top \mathbf{P}(t) + \mathbf{P}(t)\mathbf{A} - \mathbf{P}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}(t) + \mathbf{Q}, \qquad \mathbf{P}(T) = \mathbf{0}, \tag{2.10}$$

and comparing the scalar parts forces $\dot{c}(t) = -\sigma^2 \operatorname{tr}(\mathbf{P}(t))$ with $c(T) = 0$, i.e. $c(t) = \sigma^2 \int_t^T \operatorname{tr}(\mathbf{P}(s))\, \mathrm{d}s$. The quadratic term $-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}$ encodes the cost reduction that the optimal control extracts, and it is what makes (2.10) nonlinear. The computation above is an ansatz followed by term matching. A classical result in optimal control confirms that it indeed solves the problem (Yong & Zhou, 1999, Chapter 6).

Theorem B.1 (LQR optimality)**.** *Let* $\mathbf{Q} \succeq 0$ *and* $\mathbf{R} \succ 0$*. Then:*

(i) The Riccati ODE (2.10) *admits a unique solution* $\mathbf{P}$ *on* $[0, T]$*, and* $\mathbf{P}(t)$ *is symmetric positive semi-definite for every* $t \in [0, T]$*.*

(ii) The feedback policy $\boldsymbol{\pi}_\star$ *of* (2.9) *is optimal, i.e.* $J(\boldsymbol{\pi}_\star) \leq J(\boldsymbol{\pi})$ *for every policy* $\boldsymbol{\pi}$ *whose state and control processes are square-integrable over* $[0, T]$*.*

(iii) The optimal cost is $J(\boldsymbol{\pi}_\star) = \mathbb{E}\big[\boldsymbol{x}_0^\top \mathbf{P}(0)\, \boldsymbol{x}_0\big] + \sigma^2 \int_0^T \operatorname{tr}\big(\mathbf{P}(t)\big)\, \mathrm{d}t$*.*

Part (i) is not automatic: the quadratic term can drive solutions of a Riccati equation to escape in finite time, and it is exactly $\mathbf{Q} \succeq 0$ together with $\mathbf{R} \succ 0$ that supplies the a priori bound ruling this out. Parts (ii) and (iii) follow by a verification argument: applying Itô's formula to $V(t, \boldsymbol{x}(t))$ under an arbitrary admissible policy and integrating, the HJB equation (2.8) makes V a lower bound on the achievable cost-to-go, and the bound is attained by $\boldsymbol{\pi}_\star$.

Two features of Theorem B.1 shape the rest of the thesis. First, the optimal gain (2.9) does not depend on the noise level σ, so the optimal feedback of the stochastic problem coincides with that of its noiseless counterpart. Second, the dynamics enter the entire solution only through $(\mathbf{A}, \mathbf{B})$ in (2.10): once an estimate of the dynamics is available, a policy is one Riccati solve away. While the nonlinearity of (2.10) might suggest that this solve requires numerical integration, the next subsection shows that it does not.

2.3.3 Closed-Form Solver for the Riccati ODE

The optimal finite-horizon gain $\mathbf{K}(t) = -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}(t)$ is fixed by the solution $\mathbf{P}$ of the differential Riccati equation. Transforming time to the variable $\tau = T - t$, it integrates *forward* from $\mathbf{P}(0) = \mathbf{0}$,

$$\frac{\mathrm{d}\mathbf{P}}{\mathrm{d}\tau} = \mathbf{Q} + \mathbf{A}^\top \mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}. \tag{2.11}$$

When the dynamics $(\mathbf{A}, \mathbf{B})$ and cost $(\mathbf{Q}, \mathbf{R})$ are constant, which is true within a single planning problem, (2.11) is autonomous and it turns out that it can be solved in closed form rather than by numerical integration (Vaughan, 1969).

The quadratic term $-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}$ is what makes (2.11) nonlinear. The classical approach for a Riccati equation by Vaughan (1969) is to write its solution as a *ratio* $\mathbf{P} = \mathbf{Y}\mathbf{X}^{-1}$ of two matrices that evolve *linearly*, trading the nonlinear matrix ODE for a larger linear one. Concretely, posit

$$\dot{\mathbf{X}} = \mathbf{M}_{11}\mathbf{X} + \mathbf{M}_{12}\mathbf{Y}, \qquad \dot{\mathbf{Y}} = \mathbf{M}_{21}\mathbf{X} + \mathbf{M}_{22}\mathbf{Y},$$

for a constant block matrix $\mathbf{M}$ to be determined (dots denote $\mathrm{d}/\mathrm{d}\tau$). Differentiating the ratio, with $\frac{\mathrm{d}}{\mathrm{d}\tau}\mathbf{X}^{-1} = -\mathbf{X}^{-1}\dot{\mathbf{X}}\,\mathbf{X}^{-1}$ (obtained by differentiating $\mathbf{X}\mathbf{X}^{-1} = \mathbf{I}$), gives

$\dot{\mathbf{P}} = \dot{\mathbf{Y}}\,\mathbf{X}^{-1} - \mathbf{P}\,\dot{\mathbf{X}}\,\mathbf{X}^{-1}$. Substituting the posited dynamics and $\mathbf{Y}\mathbf{X}^{-1} = \mathbf{P}$ yields

$$\dot{\mathbf{P}} = \big(\mathbf{M}_{21} + \mathbf{M}_{22}\mathbf{P}\big) - \mathbf{P}\big(\mathbf{M}_{11} + \mathbf{M}_{12}\mathbf{P}\big) = \mathbf{M}_{21} + \mathbf{M}_{22}\,\mathbf{P} - \mathbf{P}\,\mathbf{M}_{11} - \mathbf{P}\,\mathbf{M}_{12}\,\mathbf{P}. \quad (2.12)$$

Matching (2.12) term by term with the Riccati equation (2.11) requires $\mathbf{M}_{11} = -\mathbf{A}$, $\mathbf{M}_{12} = \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{\top}$, $\mathbf{M}_{21} = \mathbf{Q}$, and $\mathbf{M}_{22} = \mathbf{A}^{\top}$, so $(\mathbf{X}, \mathbf{Y})$ obeys the larger *linear* system

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix}\mathbf{X}\\ \mathbf{Y}\end{bmatrix} = \mathbf{M}\begin{bmatrix}\mathbf{X}\\ \mathbf{Y}\end{bmatrix}, \qquad \mathbf{M} \coloneqq \begin{bmatrix}-\mathbf{A} & \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{\top}\\ \mathbf{Q} & \mathbf{A}^{\top}\end{bmatrix}, \quad (2.13)$$

with the $2d \times 2d$ Hamiltonian matrix $\mathbf{M}$. The initial condition $\mathbf{P}(0) = \mathbf{0}$ corresponds to $\mathbf{X}(0) = \mathbf{I}$, $\mathbf{Y}(0) = \mathbf{0}$.

The payoff of linearity is a closed-form flow. Exactly as the scalar linear ODE $\dot{x} = mx$ has solution $x(\tau) = e^{m\tau}x(0)$, the linear system (2.13) has solution $[\mathbf{X}(\tau); \mathbf{Y}(\tau)] = \exp(\mathbf{M}\tau)\,[\mathbf{X}(0); \mathbf{Y}(0)]$, where $\exp(\cdot)$ is the matrix exponential. We only ever need $\mathbf{P}$ at the control times $\tau_k = k\,\Delta t$, so it suffices to advance the pair one step at a time with the single matrix $\mathbf{E} \coloneqq \exp(\mathbf{M}\,\Delta t)$, computed once,

$$\begin{bmatrix}\mathbf{X}_{k+1}\\ \mathbf{Y}_{k+1}\end{bmatrix} = \mathbf{E}\begin{bmatrix}\mathbf{X}_{k}\\ \mathbf{Y}_{k}\end{bmatrix}, \qquad \mathbf{E} = \begin{bmatrix}\mathbf{E}_{11} & \mathbf{E}_{12}\\ \mathbf{E}_{21} & \mathbf{E}_{22}\end{bmatrix}, \quad (2.14)$$

partitioned into $d \times d$ blocks. To turn this into a recurrence for $\mathbf{P}$ itself, form the ratio $\mathbf{P}_{k+1} = \mathbf{Y}_{k+1}\mathbf{X}_{k+1}^{-1}$ from the two block rows of (2.14),

$$\mathbf{P}_{k+1} = \big(\mathbf{E}_{21}\mathbf{X}_k + \mathbf{E}_{22}\mathbf{Y}_k\big)\big(\mathbf{E}_{11}\mathbf{X}_k + \mathbf{E}_{12}\mathbf{Y}_k\big)^{-1}, \quad (2.15)$$

and substitute $\mathbf{Y}_k = \mathbf{P}_k\mathbf{X}_k$. This factors an $\mathbf{X}_k$ out of each parenthesis, and the two copies cancel in the product. This results in a recurrence in $\mathbf{P}$ alone,

$$\mathbf{P}_{k+1} = \big(\mathbf{E}_{21} + \mathbf{E}_{22}\,\mathbf{P}_k\big)\big(\mathbf{E}_{11} + \mathbf{E}_{12}\,\mathbf{P}_k\big)^{-1}, \qquad \mathbf{P}_0 = \mathbf{0}, \quad (2.16)$$

the matrix analogue of a Möbius (fractional-linear) map $z \mapsto (e_{21} + e_{22}z)/(e_{11} + e_{12}z)$. Exponentiating the partitioned Hamiltonian once and propagating $\mathbf{P}$ stepwise is the method of Davison & Maki (1973). Iterating (2.16) for $k = 0, \ldots, H$ produces $\mathbf{P}$ at every grid point $\tau_k = k\,\Delta t$ from one $2d \times 2d$ matrix exponential and H bounded $d \times d$ steps. The scheme is *exact* up to round-off, since it carries no discretisation error, being the true flow of (2.11). Furthermore, its $\mathcal{O}(Hd^3)$ cost is *independent of the spectrum* of $\mathbf{A}$, so a near-unstable estimate, which makes the flow stiff, costs no more than a benign one. Standard numerical integrators, such as explicit Runge–Kutta (RK45) or implicit Radau, attain the same accuracy but become prohibitively slow as the dimension grows, so we use the closed-form recurrence throughout, falling back to numerical integrators only for the rare step at which $\mathbf{E}_{11} + \mathbf{E}_{12}\mathbf{P}_k$ is near-singular.

2.4 The Least Absolute Shrinkage and Selection Operator (LASSO)

The key ingredient of our method is a statistical estimator that turns the sparsity prior into a sample-efficiency gain. Consider the linear model $\boldsymbol{y} = \mathbf{Z}\boldsymbol{\theta}_\star + \boldsymbol{\varepsilon}$ with design $\mathbf{Z} \in \mathbb{R}^{N\times q}$, noise $\boldsymbol{\varepsilon} \in \mathbb{R}^N$ that is sub-Gaussian with scale $\sigma > 0$, and an unknown parameter $\boldsymbol{\theta}_\star \in \mathbb{R}^q$ that is *sparse*, with at most $s \ll q$ nonzero entries at unknown locations. Ordinary least squares ignores this structure and pays for all q coordinates; its error scales as q/N. The ideal estimator would penalise the number of nonzeros $\|\boldsymbol{\theta}\|_0$, but that objective is non-convex and combinatorial. Instead, the *least absolute shrinkage and selection operator* (LASSO) replaces the ℓ_0 count by its tightest convex surrogate, the ℓ_1 norm (Tibshirani, 1996):

$$\hat{\boldsymbol{\theta}} \in \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^q} \frac{1}{2N}\big\|\boldsymbol{y} - \mathbf{Z}\boldsymbol{\theta}\big\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1, \tag{2.17}$$

where $\lambda > 0$ trades data fit against sparsity. The geometry of the ℓ_1 ball, whose corners lie on the coordinate axes, makes the solution *exactly* sparse: coordinates below a threshold set by λ are set to zero rather than merely shrunk, so (2.17) performs selection and estimation at once (Hastie et al., 2015).

The statistical value of this substitution is captured by the LASSO's oracle inequality. Provided that the regularisation dominates the noise, $\lambda \gtrsim \sigma\sqrt{\log q/N}$, and the design satisfies a *restricted eigenvalue* (RE) condition, the error obeys

$$\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_\star\|_2^2 \;=\; O\!\left(\frac{s\log q}{N}\right), \tag{2.18}$$

replacing the dense q/N rate by $s\log q/N$ (Bühlmann & Van De Geer, 2011; Wainwright, 2019). The gain is dramatic when $s \ll q$: only $s\log q$ rather than q samples are needed to reach a target accuracy. This drop in dimensional dependence is the statistical engine behind the sample-complexity advantage that Chapter 4 converts into a control-regret advantage. The RE condition is benign for independent designs but becomes the crux in the closed-loop setting, where the design is generated by the agent's own feedback policy.

The selection operator. The estimator's name promises *selection*, and the mechanism is visible in the scalar prototype of (2.17): minimise $f(\theta) = \frac{1}{2}(\theta - z)^2 + \lambda|\theta|$ over $\theta \in \mathbb{R}$. The objective is convex but not differentiable at 0, so the minimiser is characterised by the subdifferential condition $0 \in \partial f(\theta)$, where $\partial|\theta| = \{\operatorname{sign}(\theta)\}$ for $\theta \neq 0$ and $\partial|\theta| = [-1, 1]$ at $\theta = 0$. For $\theta > 0$ stationarity reads $\theta - z + \lambda = 0$, consistent with $\theta > 0$ only if $z > \lambda$; the case $\theta < 0$ mirrors it; and $\theta = 0$ is optimal precisely when $0 \in -z + \lambda\,[-1, 1]$, i.e. when $|z| \le \lambda$. The three cases combine into the *soft-thresholding*

operator

$$\begin{aligned} S_\lambda(z) &\coloneqq \arg\min_{\theta\in\mathbb{R}} \{\tfrac{1}{2}(\theta - z)^2 + \lambda|\theta|\} \\ &= \operatorname{sign}(z)\, \max\big(|z| - \lambda,\, 0\big) \\ &= \begin{cases} z - \lambda, & z > \lambda, \\ 0, & |z| \le \lambda, \\ z + \lambda, & z < -\lambda. \end{cases} \end{aligned} \tag{2.19}$$

The entire interval $[-\lambda, \lambda]$ is mapped to *exactly* zero. This dead zone is the selection: a coordinate whose signal does not clear the threshold λ is eliminated rather than merely shrunk, which is how the corner geometry of the ℓ_1 ball manifests computationally.

Cyclic coordinate descent. The full problem (2.17) reduces to the scalar prototype one coordinate at a time. Fix all coordinates except the j-th and collect the *partial residual* $\boldsymbol{r}^{(j)} \coloneqq \boldsymbol{y} - \sum_{k\neq j} \mathbf{Z}_{:,k}\, \theta_k$. As a function of θ_j alone, the objective is

$$\frac{1}{2N}\big\|\boldsymbol{r}^{(j)} - \mathbf{Z}_{:,j}\, \theta_j\big\|_2^2 + \lambda|\theta_j| + \text{const} = \frac{a_j}{2}\, \theta_j^2 - z_j\, \theta_j + \lambda|\theta_j| + \text{const},$$

with curvature $a_j \coloneqq \|\mathbf{Z}_{:,j}\|_2^2/N$ and correlation $z_j \coloneqq \langle \mathbf{Z}_{:,j},\, \boldsymbol{r}^{(j)}\rangle/N$, whose exact minimiser is the rescaled threshold update $\theta_j \leftarrow S_\lambda(z_j)/a_j$. *Cyclic* coordinate descent cycles through $j = 1, \ldots, q$ in order, applying this update with the most recent values of all other coordinates, and repeats until the updates stagnate (Wu & Lange, 2008; Friedman et al., 2010). One cycle costs $\mathcal{O}(Nq)$ flops, or $\mathcal{O}(q^2)$ once the Gram matrix $\mathbf{Z}^\top\mathbf{Z}$ is precomputed and the correlations are updated incrementally. The latter approach is used in this thesis, so per-cycle cost is independent of the sample count.

Correctness. Each update minimises its coordinate exactly, so the objective decreases monotonically; what is not automatic is that the procedure reaches the *global* minimum. For a general non-differentiable convex function, coordinatewise minimisation can lock into a point that is optimal along every axis yet suboptimal along a diagonal direction, in which case cyclic descent fails. The LASSO escapes this failure because its non-smooth part is *separable*: the objective is a differentiable convex quadratic plus a sum of one-dimensional terms $\lambda|\theta_j|$. For such composite functions the subdifferential splits coordinatewise, so the global optimality condition is exactly the conjunction of the per-coordinate conditions $0 \in \nabla_j\, g(\boldsymbol{\theta}) + \lambda\, \partial|\theta_j|$, and a fixed point of the cycle satisfies each of them by construction. A coordinatewise minimum of the LASSO objective is therefore a global minimum, and the convergence of the cyclic iterates to one is guaranteed by the theorem of Tseng for convex objectives with separable non-smooth parts (Wu & Lange, 2008).

2.5 Certainty Equivalence (CE)

This section finally bridges estimation and control. Once the agent holds an estimate $\hat{\mathbf{\Theta}} = [\,\hat{\mathbf{A}}\ \ \hat{\mathbf{B}}\,]$ of the true dynamics $\mathbf{\Theta}_\star = [\,\mathbf{A}_\star\ \ \mathbf{B}_\star\,]$, it must decide how to act. *Certainty equivalence* (CE) is the simplest and most widely used answer: treat the current estimate as if it were the truth, compute the policy that would be optimal under it, and execute that policy. Here, this means solving the Riccati equation (2.10) at $\hat{\mathbf{\Theta}}$ and applying the resulting linear-quadratic gain (2.9). The name originates in linear-quadratic-Gaussian control, where the state is observed only through noisy measurements and must itself be estimated. There, with *known* parameters, the optimal controller depends only on the mean of the state estimate and not on its uncertainty. Hence, estimation and control separate exactly, the celebrated *separation principle* (Yong & Zhou, 1999, Chapter 6). The term is thus used in three related senses: (1) the optimal LQ gain is independent of the noise level (Theorem B.1), (2) the controller is independent of the state-estimate uncertainty, and (3) a plug-in policy treats the parameter estimate as if it were exact.

When the *parameters* are unknown, no such exact separation is available, and CE becomes a principled heuristic rather than an optimal strategy. It deliberately ignores the value of information: it never takes an action *because* that action would be informative, only because it looks best under the current estimate. The theoretically optimal alternative, dual control (Feldbaum, 1960), would balance the immediate cost of control against the long-run value of the information an action reveals, but it is intractable in all but the smallest problems. CE sidesteps this intractability, and a line of recent work shows that the sacrifice is mild. Mania et al. (2019) prove that certainty equivalence is *efficient*: the suboptimality of the plug-in controller scales with the square of the parameter-estimation error, so a good estimate is converted into a disproportionately good policy. Building on this, Simchowitz & Foster (2020) show that for the online LQR certainty-equivalent control, augmented with a vanishing amount of exploratory noise, attains the optimal regret rate: "naive" exploration is essentially all that is needed. The finite-horizon analogue of this quadratic conversion is exactly what our regret analysis establishes for the episodic setting (Section 4.3).

CE is attractive here for a further reason: it keeps the sparsity structure confined to the estimator. The planner is an off-the-shelf Riccati solve that neither knows nor cares that $\mathbf{\Theta}_\star$ is sparse; all of the structure exploitation happens inside the LASSO. This clean separation is what lets us isolate the effect of sparse estimation on control performance, simply by swapping the estimator while holding the planner fixed. The one place CE needs care is the feedback between policy and data noted in Section 2.1: because the greedy policy shapes its own training data, a purely certainty-equivalent scheme can fail to excite the system enough to identify it, which is why the algorithm we develop in the next chapter augments pure CE with an exploratory episode.

Chapter 3

Method



Exploiting dynamics sparsity to control an unknown system requires three ingredients: a precise statement of the control problem, an algorithm that solves it while making use of that structure, and an assurance that the algorithm's generated data can identify the system at all. This chapter develops them in turn. We first formalise the setting (Section 3.1): an agent steers an unknown row-sparse linear system to minimise a quadratic cost over a finite horizon. We then propose our algorithm, SAMBL (Section 3.2), which follows the principle of certainty equivalence but replaces the usual least-squares estimator with a row-wise LASSO that exploits the sparsity. Finally, we ask when the closed-loop data generated by this scheme suffice to identify the parameters in principle (Section 3.3), a prerequisite for the regret guarantees of Chapter 4.

3.1 Problem Formulation

We consider an agent interacting with an unknown linear stochastic system in continuous time, with the goal of minimising a quadratic cost over a sequence of fixed-length episodes. The agent knows the cost structure and the system dimensions but not the dynamics. It must learn the dynamics from the trajectories it observes while paying for the suboptimality of its actions. The setting formulation follows Basei et al. (2022), while the row-sparsity assumption on the dynamics is the ingredient we add. This section makes the setting precise and introduces the regret objective that Chapter 4 analyses.

Dynamics. The system state $\boldsymbol{x}(t) \in \mathbb{R}^d$ evolves according to the linear stochastic differential equation

$$\mathrm{d}\boldsymbol{x}(t) = \big(\mathbf{A}_\star \boldsymbol{x}(t) + \mathbf{B}_\star \boldsymbol{u}(t)\big)\,\mathrm{d}t + \sigma\,\mathrm{d}\boldsymbol{w}(t), \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \qquad t \in [0, T] \tag{3.1}$$

where $\boldsymbol{u}(t) \in \mathbb{R}^p$ is the control, $\sigma \in \mathbb{R}_{>0}$ is the diffusion coefficient, T is the finite time horizon, and $\boldsymbol{w}(t)$ is a d-dimensional Wiener process (Brownian motion, see Definition B.1). The drift and control parameter matrices $\mathbf{A}_\star \in \mathbb{R}^{d\times d}$ and $\mathbf{B}_\star \in \mathbb{R}^{d\times p}$ are unknown to the agent. We collect them into a single parameter matrix

$$\boldsymbol{\Theta}_\star \coloneqq [\mathbf{A}_\star\ \mathbf{B}_\star] \in \mathbb{R}^{d\times(d+p)}$$

and denote by $\boldsymbol{\theta}_{\star,i}$ the i-th row, written as a column vector. The dimensions d, p, the noise level σ, and the horizon T are known. Only the known σ is a substantive assumption. In practice, it can be estimated from regression residuals.

Sparsity. We assume $\boldsymbol{\Theta}_\star$ to be *row-sparse*: each row $\boldsymbol{\theta}_{\star,i}$ has at most s nonzero entries, with $s \ll d + p$. Formally,

$$|\mathcal{S}_i| \leq s \quad \text{for all } i \in [d], \quad \text{where } \mathcal{S}_i \coloneqq \operatorname{supp}(\boldsymbol{\theta}_{\star,i}) \coloneqq \{j : \theta_{\star,i,j} \neq 0\}$$

The support sets $\{\mathcal{S}_i\}_{i=1}^d$ are unknown. We refer to s as the *row support size* throughout. Note that since s counts nonzero entries, a larger s means a *less* sparse system. The sparsity assumption captures modelling scenarios where the law of each state variable depends on at most s other state and control variables.

Policy. The actions of the agent obey a time-dependent policy $\boldsymbol{\pi} : [0, T] \times \mathbb{R}^d \to \mathbb{R}^p$ that maps the current state $\boldsymbol{x}(t)$ and time t to a control action $\boldsymbol{u}(t) \coloneqq \boldsymbol{\pi}(t, \boldsymbol{x}(t))$.

Cost. Over the course of an episode, a given policy $\boldsymbol{\pi}$ incurs a quadratic episode cost

$$J(\boldsymbol{\pi}) \coloneqq \mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{w}(t)} \left[\int_0^T \left(\boldsymbol{x}(t)^\top \mathbf{Q}\, \boldsymbol{x}(t) + \boldsymbol{u}(t)^\top \mathbf{R}\, \boldsymbol{u}(t) \right) \mathrm{d}t \right], \tag{3.2}$$

where the expectation is taken over the Brownian motion driving the dynamics and any randomness in the initial state. The state cost matrix $\mathbf{Q} \in \mathbb{R}^{d \times d}$ and control cost matrix $\mathbf{R} \in \mathbb{R}^{p \times p}$ are symmetric positive (semi-)definite, specifically $\mathbf{Q} \succeq 0$ and $\mathbf{R} \succ 0$. Both are known to the agent. This is a conventional assumption in the LQR literature; extending the method to unknown cost matrices is a natural direction for future work.

Objective. The goal of the agent is to interact with the environment to learn a policy that minimises the cost:

$$\boldsymbol{\pi}_\star \in \arg\inf_{\boldsymbol{\pi}} J(\boldsymbol{\pi}), \qquad J_\star \coloneqq J(\boldsymbol{\pi}_\star). \tag{3.3}$$

The infimum is taken over all time-dependent policies and attained by a policy that depends on the unknown true parameters $\boldsymbol{\Theta}_\star$ (see Section 3.2.3), which can be expressed in closed form when $\boldsymbol{\Theta}_\star$ is given.

Episodes. The agent interacts with the environment over M episodes, indexed by $m \in \{1, \ldots, M\}$. Each episode is an independent roll-out of the dynamics (3.1) over the time interval $[0, T]$, starting from a freshly sampled initial state $\boldsymbol{x}_0 \sim P(\boldsymbol{x}_0)$ from a potentially unknown distribution P, and using the current policy of the agent.

Regret. The performance of the learnt policies $\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots, \boldsymbol{\pi}_M$ over M episodes is measured by the *cumulative regret* against the optimal policy (3.3):

$$R_M \coloneqq \sum_{m=1}^{M} \underbrace{\left(J(\boldsymbol{\pi}_m) - J_\star \right)}_{=: r_m}. \tag{3.4}$$

The next section develops an algorithm to minimise the cost and achieve favourable regret under the sparsity structure of $\boldsymbol{\Theta}_\star$.

3.2 The SAMBL Algorithm

The key challenge in our setting is that the agent does not have access to the true dynamics $\boldsymbol{\Theta}_\star$ and thus cannot compute the optimal policy directly. To address this, we adopt the principle of *certainty equivalence* (cp. Section 2.5): at each episode, the agent treats its current estimate $\hat{\boldsymbol{\Theta}}$ as if it were the truth, computes the optimal policy under the estimated system, and executes it. The data collected during the episode is added to the dataset and used to update the dynamics estimate. This section specifies the three components needed to make this concrete: first, the discretised regression data that the estimator consumes (Section 3.2.1); second, the row-wise LASSO estimator to estimate $\hat{\boldsymbol{\Theta}}$ (Section 3.2.2); third, the planner that maps the parameter estimate to a policy (Section 3.2.3). The full algorithm assembling these pieces is given in Section 3.2.4.

3.2.1 Discretised regression

While the dynamics (3.1) are continuous in time, a practical system only ever observes discrete samples. For simplicity, as our focus is on regret and not SDE simulation, we use the basic Euler-Maruyama method (Kloeden & Platen, 1992) with a fixed step size $\Delta t > 0$. When choosing Δt so that $H := T/\Delta t$ is an integer, an episode comprises H discrete steps. At step k corresponding to time t_k, the next state is determined as

$$\begin{aligned} \boldsymbol{x}_{k+1} &:= \boldsymbol{x}_k + \Delta t\, \boldsymbol{\Theta}_\star \boldsymbol{z}_k + \sigma\sqrt{\Delta t}\, \boldsymbol{w}_k, \\ \boldsymbol{w}_k &\sim \mathcal{N}(0, \mathbf{I}_d), \\ \boldsymbol{u}_k &:= \boldsymbol{\pi}(t_k, \boldsymbol{x}_k), \\ \boldsymbol{z}_k &:= \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{u}_k \end{bmatrix}, \end{aligned} \tag{3.5}$$

where $\boldsymbol{w}_k$ is the discretised Brownian increment, $\boldsymbol{u}_k$ is the applied control, and the stacked regressor $\boldsymbol{z}_k \in \mathbb{R}^{d+p}$ collects the state and control.

Rearranging (3.5) gives the regression form

$$\begin{aligned} \boldsymbol{y}_k &:= \frac{\boldsymbol{x}_{k+1} - \boldsymbol{x}_k}{\Delta t} = \boldsymbol{\Theta}_\star \boldsymbol{z}_k + \boldsymbol{\varepsilon}_k, \\ \boldsymbol{\varepsilon}_k &:= \bar{\sigma}\, \boldsymbol{w}_k \sim \mathcal{N}(0, \bar{\sigma}^2 \mathbf{I}_d), \\ \bar{\sigma} &:= \frac{\sigma}{\sqrt{\Delta t}}. \end{aligned} \tag{3.6}$$

The regression target $\boldsymbol{y}_k$ is the empirical state increment per unit time, and the noise $\boldsymbol{\varepsilon}_k$ is isotropic Gaussian with standard deviation $\bar{\sigma}$. The discretised problem is therefore a standard multivariate linear regression with H samples per episode, regressor $\boldsymbol{z}_k$, and target $\boldsymbol{y}_k$.

After m episodes, the agent has collected $N_m := mH$ samples, all of which are retained. We stack the data row-wise to produce the *design matrix* $\mathbf{Z} \in \mathbb{R}^{N_m \times (d+p)}$ and the *response matrix* $\mathbf{Y} \in \mathbb{R}^{N_m \times d}$. The i-th column $\mathbf{Y}_{:,i}$ contains the N_m samples of the i-th state coordinate's increment, and admits the row-wise regression model

$$\mathbf{Y}_{:,i} = \mathbf{Z}\,\boldsymbol{\theta}_{\star,i} + \boldsymbol{\varepsilon}_i, \qquad \boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \bar{\sigma}^2 \mathbf{I}_{N_m}). \tag{3.7}$$

The d rows of $\boldsymbol{\Theta}_\star$ can therefore be estimated *in parallel* from d separate regression problems sharing the same design matrix $\mathbf{Z}$.

Note that while we choose an equidistant discretisation grid for simplicity, the regression formulation does not depend on this due to the definition of the targets in (3.6). More generally, Euler-Maruyama can be replaced by any discretisation scheme whose targets remain *linear* in the sparse continuous-time parameter $\boldsymbol{\Theta}_\star$, so that the regression form (3.6) and everything built on it carry over unchanged. We leave the consideration of advanced discretisation schemes to future work.

3.2.2 Row-wise LASSO estimation

By the sparsity assumption, each row $\boldsymbol{\theta}_{\star,i}$ has at most s nonzero entries with $s \ll d + p$. The natural estimator that exploits this structure is the LASSO (Tibshirani, 1996), which encourages sparse solutions (cp. Section 2.4). At episode m, the agent estimates the system parameters by solving the LASSO objective

$$\hat{\boldsymbol{\theta}}_{m,i} \in \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \mathbb{R}^{d+p}} \frac{1}{2N_m} \left\| \mathbf{Y}_{:,i} - \mathbf{Z}\boldsymbol{\theta} \right\|_2^2 + \lambda_m \|\boldsymbol{\theta}\|_1, \qquad i \in [d], \tag{3.8}$$

where the regularisation parameter $\lambda_m > 0$ controls the trade-off between data fit and sparsity, with larger λ_m inducing more zero entries. The specific form of λ_m is determined by a score-concentration argument and specified later in Lemma 1. Stacking the row estimates gives the joint parameter estimate

$$\hat{\boldsymbol{\Theta}}_m := [\hat{\mathbf{A}}_m \; \hat{\mathbf{B}}_m] \in \mathbb{R}^{d \times (d+p)}.$$

The row-wise formulation is the natural choice because the rows of $\boldsymbol{\Theta}_\star$ have independent sparsity patterns and independent noise across the response coordinates. Estimating each row separately decouples the problem into d instances of the standard LASSO, each with the same design matrix but a different response vector.

The optimisation problem (3.8) is convex and can be solved exactly by the cyclic coordinate descent reviewed in Section 2.4 (Wu & Lange, 2008; Friedman et al., 2010). The analysis in Chapter 4 treats $\hat{\boldsymbol{\theta}}_{m,i}$ as the exact minimiser. In Chapter 5 we discuss the optimisation error of the solver.

3.2.3 The Riccati ODE and Optimal Feedback

By the principle of certainty equivalence, the agent uses the current estimate of the system parameters to obtain the optimal policy. To do so, we apply the classical LQR optimality theorem derived in Section 2.3 (Theorem B.1): for linear dynamics (3.1) and quadratic cost (3.2), the optimal policy is a *linear* but time-varying function of the state, parametrised by a time-varying *gain matrix* $\mathbf{K}_\star$:

$$\boldsymbol{\pi}_\star(t, \boldsymbol{x}) = \mathbf{K}_\star(t)\, \boldsymbol{x}, \qquad \mathbf{K}_\star : [0, T] \to \mathbb{R}^{p \times d}. \tag{3.9}$$

The form of the optimum is a standard consequence of the dynamic programming principle for finite-horizon linear-quadratic control (cp. Section 2.3). The remainder of this section defines the general gain $\mathbf{K}_{\boldsymbol{\Theta}}(t)$ associated with any candidate parameter matrix $\boldsymbol{\Theta}$.

Given parameters $\boldsymbol{\Theta} = [\mathbf{A}\ \mathbf{B}]$, the corresponding optimal finite-horizon gain is obtained from the backward *Riccati differential equation* (cp. (2.10)), now instantiated at $\boldsymbol{\Theta}$ in place of the true parameters. The matrix-valued function $\mathbf{P}_{\boldsymbol{\Theta}} : [0, T] \to \mathbb{R}^{d \times d}$ is defined as the unique solution of

$$-\dot{\mathbf{P}}_{\boldsymbol{\Theta}}(t) = \mathbf{A}^\top \mathbf{P}_{\boldsymbol{\Theta}}(t) + \mathbf{P}_{\boldsymbol{\Theta}}(t)\mathbf{A} - \mathbf{P}_{\boldsymbol{\Theta}}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_{\boldsymbol{\Theta}}(t) + \mathbf{Q}, \qquad \mathbf{P}_{\boldsymbol{\Theta}}(T) = \mathbf{0}. \tag{3.10}$$

Existence and uniqueness of the solution on $[0, T]$ are guaranteed under the $\mathbf{Q} \succeq 0$ and $\mathbf{R} \succ 0$ assumptions we made (see Theorem B.1(i) and Yong & Zhou, 1999, Chapter 6). The corresponding gain is

$$\mathbf{K}_{\boldsymbol{\Theta}}(t) \coloneqq -\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P}_{\boldsymbol{\Theta}}(t). \tag{3.11}$$

The gain $\mathbf{K}_{\boldsymbol{\Theta}}$ is optimal for the system whose dynamics are exactly defined by $\boldsymbol{\Theta}$: if the agent's estimate were correct, executing $\mathbf{K}_{\boldsymbol{\Theta}}$ would incur the minimal cost $J_\star$. Under certainty equivalence, the agent computes $\mathbf{K}_{\hat{\boldsymbol{\Theta}}}$ for its current estimate $\hat{\boldsymbol{\Theta}}$ and applies it to the true system, whose dynamics parameters $\boldsymbol{\Theta}_\star$ differ from $\hat{\boldsymbol{\Theta}}$. The resulting suboptimality – the gap between the cost of $\mathbf{K}_{\hat{\boldsymbol{\Theta}}}$ under $\boldsymbol{\Theta}_\star$ and the optimal cost $J_\star$ – is precisely the per-episode regret analysed in Chapter 4. Setting $\boldsymbol{\Theta} = \boldsymbol{\Theta}_\star$ recovers the true optimal gain $\mathbf{K}_\star = \mathbf{K}_{\boldsymbol{\Theta}_\star}$ from (3.9).

We refer to (3.10) as the Riccati *differential* equation to distinguish it from the algebraic Riccati equation that arises in the more prominent infinite-horizon setting. The two are conceptually related but technically distinct: the differential equation is integrated over the finite horizon $[0, T]$ and produces a time-varying gain $\mathbf{K}_{\boldsymbol{\Theta}}(t)$, whereas the algebraic equation produces a constant gain $\mathbf{K}_\infty$ which is optimal for the infinite-horizon cost. The time dependence turns out to be crucial for identifiability, as we will see in Section 3.3.

3.2.4 Full Algorithm

The three components discussed in Sections 3.2.1 to 3.2.3 combine into a single episodic loop summarised in Algorithm 1. At the start of episode m the agent has a parameter estimate $\hat{\mathbf{\Theta}}_{m-1}$, computes the corresponding gain $\mathbf{K}_{\hat{\mathbf{\Theta}}_{m-1}}(t)$ by solving the Riccati ODE (3.10), and executes the induced policy $\boldsymbol{\pi}_m(t, \boldsymbol{x}(t)) \coloneqq \mathbf{K}_{\hat{\mathbf{\Theta}}_{m-1}}(t)\boldsymbol{x}(t)$ throughout the episode. The trajectory collected is appended to the dataset, and the estimate is updated by solving the LASSO objective (3.8). We choose to time re-estimation with episode boundaries for simplicity of presentation. However, many asynchronous re-estimation schemes are possible and we leave their consideration to future work.

Algorithm 1 SAMBL: Sparsity-Aware Model-Based Learning
(here for episodic continuous-time linear-quadratic control)

Input: horizon T, step size Δt, episodes M, exploration episodes m_{explore}, regularisation schedule $\{\lambda_m\}_{m=1}^{M}$, initial estimate $\hat{\mathbf{\Theta}}_0$
for $m = 1, \dots, M$ **do**
  if $m \le m_{\text{explore}}$ **then**
    Roll out episode under a piecewise constant exploration policy, playing an independent $\boldsymbol{u}_k \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_p)$ over each step interval $[t_k, t_{k+1})$.
  else
    Solve the Riccati ODE (3.10) at $\hat{\mathbf{\Theta}}_{m-1}$ to obtain gain $\mathbf{K}_{\hat{\mathbf{\Theta}}_{m-1}}$.
    Roll out episode under the policy $\boldsymbol{\pi}_m(t, \boldsymbol{x}(t)) = \mathbf{K}_{\hat{\mathbf{\Theta}}_{m-1}}(t)\, \boldsymbol{x}(t)$ as per (3.5).
  end if
  Collect samples $(\boldsymbol{z}_k, \boldsymbol{y}_k)_{k=1}^{H}$ from the rollout and append them to the dataset.
  Solve the row-wise LASSO (3.8) on the dataset to obtain $\hat{\mathbf{\Theta}}_m$.
end for

Our Algorithm 1 plays pure certainty equivalence: apart from the estimator, no part of the loop knows about sparsity, and the executed gain is always the cost-optimal one for the current estimate. Chapter 4 establishes its regret bound, and Chapter 5 evaluates it against a dense baseline that differs only in the estimator.

The only concessions to exploration are two forms of state-independent control stimulation, one transient and one persistent. The transient one is built in: the first m_{explore} episodes are rolled out under pure Gaussian control before certainty equivalence takes over – a standard device in learning-based control (Dean et al., 2020) that we keep minimal, with the canonical choice $m_{\text{explore}} = 1$. The persistent one is optional: the executed control can be augmented with isotropic noise,

$$\boldsymbol{u}(t) = \mathbf{K}_{\hat{\mathbf{\Theta}}_{m-1}}(t)\, \boldsymbol{x}(t) + \boldsymbol{\eta}(t), \qquad \boldsymbol{\eta}(t) \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I}_p),$$

of excitation strength $\sigma_u \ge 0$, with the algorithm above corresponding to $\sigma_u = 0$. Both approaches exist solely to stimulate the control directions independently of the state, and both are priced in regret. Why such stimulation is needed at all – and why certainty-equivalent play alone may fail to sustain it – is the subject of the next section.

3.3 Identifiability Considerations

The certainty-equivalent scheme of Section 3.2 generates its own training data: the trajectories used to estimate $\boldsymbol{\Theta}_\star$ are produced by the very policy that the current estimate induces. Therefore, the design matrix $\mathbf{Z}$ is shaped by the closed-loop policy, and whether it carries enough information to identify $\boldsymbol{\Theta}_\star$ is not automatic. This section examines the information content in closed-loop data, identifies a structural asymmetry between the state and control blocks, and motivates the excitation and initialisation choices introduced in Section 3.2.

3.3.1 The Drift-Control Asymmetry

Only partial identifiability in infinite-horizon settings. One might hope that the intrinsic noise driving (3.1) removes any need for deliberate exploration by the agent, in contrast to the deterministic setting where sustained exploration, called *persistent excitation of the control* (Green & Moore, 1986), is vital to ensure identifiability, despite being a sufficient condition. The hope is only partially justified. Under a fixed feedback law $\boldsymbol{u}(t) = \mathbf{K}\boldsymbol{x}(t)$, where the gain $\mathbf{K}$ is time-independent, the state evolves as $\mathrm{d}\boldsymbol{x}(t) = (\mathbf{A}_\star + \mathbf{B}_\star\mathbf{K})\boldsymbol{x}(t)\,\mathrm{d}t + \sigma\,\mathrm{d}\boldsymbol{w}$. Hence, the data informs the learner about the *closed-loop* matrix $\mathbf{A}_\star + \mathbf{B}_\star\mathbf{K}$ but not about $\mathbf{A}_\star$ and $\mathbf{B}_\star$ separately. In the infinite-horizon setting, where the optimal gain $\mathbf{K}$ is constant, this is fatal: for any perturbation $\boldsymbol{\delta} \in \mathbb{R}^{d\times p}$, the pair $(\mathbf{A}_\star + \boldsymbol{\delta}\mathbf{K},\ \mathbf{B}_\star - \boldsymbol{\delta})$ induces the identical closed-loop matrix,

$$(\mathbf{A}_\star + \boldsymbol{\delta}\mathbf{K}) + (\mathbf{B}_\star - \boldsymbol{\delta})\mathbf{K} = \mathbf{A}_\star + \mathbf{B}_\star\mathbf{K}, \tag{3.12}$$

and is therefore indistinguishable from $(\mathbf{A}_\star, \mathbf{B}_\star)$ on the basis of closed-loop data alone. Persistent excitation is the standard remedy in this setting.

Full identifiability for finite horizon thanks to self-exploration. Our finite-horizon problem escapes this degeneracy because the optimal gain $\mathbf{K}_\star(t)$ *varies in time*. The perturbation argument above relies on a single constant $\mathbf{K}$, but when the gain sweeps through a range of values over $[0, T]$, no single $\boldsymbol{\delta}$ reconciles the closed-loop matrices at all times. Basei et al. (2022, Proposition 2.1) make this precise and show: whenever the matrix $\mathbf{B}_\star^\top \mathbf{Q}\,\mathbf{B}_\star$ is nonsingular, the full parameter $\boldsymbol{\Theta}_\star$ is identifiable under the optimal time-varying policy without any added excitation on the control. For $\mathbf{Q} = \mathbf{I}$ the condition reduces to $\mathbf{B}_\star$ having full column rank, since $\ker(\mathbf{B}_\star) = \ker(\mathbf{B}_\star^\top\mathbf{B}_\star)$. Basei et al. refer to this as the *self-exploration* property: the time variation of the optimal gain supplies, for free, the stimulation that would otherwise require deliberate excitation.

The state-control asymmetry of the sparse estimator. The self-exploration property rules out exact non-identifiability in the finite-horizon setting: under the condition above, the time-varying optimal gain makes the closed-loop state-control process informative about both $\mathbf{A}_\star$ and $\mathbf{B}_\star$. This is a population identifiability statement, however, and it does not by itself guarantee that the finite-sample regression design is

well conditioned. The genuine challenge is therefore identification from finite data, and here the two blocks of $\boldsymbol{\Theta}_\star$ are not on equal footing. The LASSO in particular needs more than an informative population design: support recovery rests on a condition that demands not just that every support direction be visible to the design, but that the empirical Gram have curvature *bounded below by a fixed constant* along each row's support (see Assumption 2 in Chapter 4). A direction can be visible by contributing positive curvature and hence to full rank, yet have curvature arbitrarily close to zero, in which case the condition fails however small the constant. The closed-loop data does not supply this quantitative curvature symmetrically across the two blocks of $\boldsymbol{\Theta}_\star$.

To see why, consider the regressor row $\boldsymbol{z}_k^\top = [\boldsymbol{x}_k^\top, \boldsymbol{u}_k^\top]$ under the deterministic policy $\boldsymbol{u}_k = \mathbf{K}(t_k)\boldsymbol{x}_k$. The state block $\boldsymbol{x}_k$ is driven directly by the diffusion $\sigma\,\mathrm{d}\boldsymbol{w}$ and carries genuine stochastic variation at every step. The control block, by contrast, is a deterministic linear function of the state at each instant: conditional on $\boldsymbol{x}_k$, the control $\boldsymbol{u}_k$ has zero variance. The columns of $\mathbf{Z}$ corresponding to the control block therefore lie, at each timestep, in the span of the state columns. Crucially, the only variation that distinguishes them comes from the time variation of $\mathbf{K}(t)$ across the episode. This is enough to make the joint Gram nonsingular, but it does not guarantee the stronger restricted eigenvalue condition along the control-support directions, and when it does hold the associated constant may be very small. The drift block faces no such obstruction, because its columns inherit the diffusion's variation directly.

The consequence is a structural asymmetry: under purely closed-loop data, the drift parameters $\mathbf{A}_\star$ are reliably recoverable by the LASSO, whereas the control parameters $\mathbf{B}_\star$ are not, even though both are identifiable in principle. The asymmetry has no analogue in passive sparse regression, where the design is not constrained to lie in the column space of part of itself. Due to the asymmetry, a sparse agent may require deliberate excitation where the dense agents do not. We examine this empirically in Chapter 5.

3.3.2 Bootstrapping Estimates

The asymmetry has an extreme degenerate case that is worth stating precisely, because it dictates how the algorithm must be initialised. If the control block of the initial estimate is set to zero, the control is never exercised and $\mathbf{B}_\star$ is never identified, regardless of how many episodes are run.

Example (Zero-control trap)**.** Consider Algorithm 1 with no excitation ($\sigma_u = 0$) and no pure-exploration phase. If the initial estimate has zero control block $\hat{\mathbf{B}}_0 = \mathbf{0}$, then the induced gain $\mathbf{K}_{\hat{\boldsymbol{\Theta}}_0}(t) = \mathbf{0}$ vanishes identically, every control is zero, and $\hat{\mathbf{B}}_m = \mathbf{0}$ for all $m \geq 0$. That is, the control block is never identified.

Proof. The gain (3.11) has $\hat{\mathbf{B}}_0^\top$ as a left factor. With $\hat{\mathbf{B}}_0 = \mathbf{0}$ it vanishes identically, $\mathbf{K}_{\hat{\boldsymbol{\Theta}}_0}(t) = \mathbf{0}$, whatever the Riccati solution $\mathbf{P}_{\hat{\boldsymbol{\Theta}}_0}$. Hence $\boldsymbol{u}_k = \mathbf{0}$ throughout the first episode, and the control columns of the design matrix $\mathbf{Z}$ are identically zero. Splitting

the regressor into its state and control parts, $\mathbf{Z} = [\mathbf{Z}_A \ \mathbf{Z}_B]$ with $\mathbf{Z}_B = \mathbf{0}$, the row-$i$ LASSO objective (3.8) after the first episode becomes

$$\frac{1}{2N_1}\left\|\mathbf{Y}_{:,i} - \mathbf{Z}_A(\hat{\boldsymbol{\theta}}_{1,i})_A\right\|_2^2 + \lambda_1\left\|(\hat{\boldsymbol{\theta}}_{1,i})_A\right\|_1 + \lambda_1\left\|(\hat{\boldsymbol{\theta}}_{1,i})_B\right\|_1.$$

The control coefficients $(\hat{\boldsymbol{\theta}}_{1,i})_B$ enter the objective only through the non-negative penalty term, while the data-fit term is independent of them. The objective is therefore strictly decreased by setting $(\hat{\boldsymbol{\theta}}_{1,i})_B = \mathbf{0}$ if it is not yet zero, so every minimiser has zero control block $\hat{\mathbf{B}}_1 = \mathbf{0}$. Induction shows $\hat{\mathbf{B}}_m = \mathbf{0}$ for all m. □

The same conclusion holds when the ℓ_1 penalty is replaced by an ℓ_2 penalty. The trap is therefore not a pathology of the LASSO but a property of the closed-loop data. It confirms that bootstrapping $\mathbf{B}_\star$ requires the control to be exercised in a manner not dictated by a zero or close-to-zero estimate.

Two mechanisms break the deadlock, each supplying control stimulation that the greedy closed loop fails to generate on its own:

- The mechanism we adopt by default is an *initial exploration episode*: for the first episode ($m_{\text{explore}} = 1$, Algorithm 1) the control is a state-independent, zero-order-hold signal that holds a fresh draw $\boldsymbol{u}_k \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_p)$ over each step interval $[t_k, t_{k+1})$, so the control directions are exercised from the outset regardless of the current estimate, seeding $\hat{\mathbf{B}}$ away from the trap before certainty equivalence resumes. Because this stimulation is independent of the state, it also supplies restricted eigenvalue curvature along the control-support directions on the first episode. Chapter 5 shows that this single episode is what keeps the control block identifiable as the dimension grows.
- Without such an episode ($m_{\text{explore}} = 0$), the trap can still be escaped by initialising the control block away from zero, $\hat{\mathbf{B}}_0 \neq \mathbf{0}$, so that the induced gain is non-trivial and the control is exercised from the first step. This removes the structural obstruction of Section 3.3.2, but does not by itself supply the restricted eigenvalue curvature for $\mathbf{B}_\star$. The control remains a deterministic function of the state, and its signal may fall below the LASSO threshold. To keep the penalised estimator from shrinking this freshly initialised control block back to zero on the small, weakly informative first-episode design – which would re-enter the trap – the penalty is relaxed to a near-OLS warmup ($\lambda_1 = 10^{-6}$) on the first episode only, with the theoretical schedule of Lemma 1 taking over from the second episode.

Orthogonally to the bootstrapping, the estimator can profit from the optional excitation of Section 3.2.4, which adds state-independent noise $\boldsymbol{\eta}(t) \sim \mathcal{N}(\mathbf{0}, \sigma_u^2\mathbf{I}_p)$ to the control. This gives the control columns variation independent of the state, which the drift columns cannot explain, and so restores the restricted eigenvalue condition along the control-support directions unconditionally, at the cost of increased regret during the excitation phase. The trade-off between the self-exploration of the time-varying gain and the guarantees of deliberate excitation is a central empirical question of Chapter 5.

This chapter has set up the problem and the method. Section 3.1 cast sparse continuous-time control as an episodic cost minimisation problem in an unknown row-sparse dynamical system $\mathbf{\Theta}_\star$. Section 3.2 developed SAMBL, our proposed algorithm based on sparse system parameter estimation as well as its variants. Section 3.3 examined what closed-loop data can reveal, and motivated the excitation and initialisation choices the algorithm offers. The next chapter analyses the regret guarantees of SAMBL.

Chapter 4

Regret Analysis



Our SAMBL algorithm of Chapter 3 is built to exploit intrinsic system sparsity. This chapter now proves that it pays off, bounding the regret of a SAMBL agent and quantifying its advantage over a dense baseline. The argument has several moving parts: an estimation-error bound for the row-wise LASSO under adaptively collected data, a perturbation inequality that converts parameter error into excess control cost, and a summation of that cost across episodes. We first collect the assumptions the analysis rests on (Section 4.1). We then bound the estimation error of the sparse estimator (Section 4.2) and, through a value-perturbation inequality derived from optimality of the Riccati solution, the per-episode regret it induces (Section 4.3). Summing over episodes yields the main regret and speedup results (Section 4.4). Finally, we return to the assumptions and show how the closed-loop scheme can satisfy them (Section 4.5).

4.1 Assumptions

The regret analysis rests on three additional assumptions for our problem class. The first two are statistical conditions on the data the agent collects, and the third is a regularity condition on the planning map. The statistical conditions are stated for a generic sample size N and a generic design matrix $\mathbf{Z}$. We state the three assumptions here and interpret them while deferring the question of when they hold under the adaptive, closed-loop data of Algorithm 1 to Section 4.5, where we give sufficient conditions in terms of the system primitives.

Score concentration. The first assumption controls correlations between the regressors and the noise, ensuring that the LASSO penalty can separate genuine signal from noise. Recall from (3.7) that the i-th row is estimated from the model $\mathbf{Y}_{:,i} = \mathbf{Z}\,\boldsymbol{\theta}_{\star,i} + \boldsymbol{\varepsilon}_i$ with $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \bar{\sigma}^2 \mathbf{I}_N)$. The quantity $\frac{1}{N}\mathbf{Z}^\top \boldsymbol{\varepsilon}_i$ is the *score*: the gradient of the least-squares loss at the true parameter. Its supremum norm must be small for the signal to be distinguishable from the noise.

Assumption 1 (Adaptive score concentration)**.** There exists a constant $c_\varepsilon > 0$ such that for every row $i \in [d]$, every sample size N, and every failure level $\delta \in (0, 1)$,

$$\left\| \frac{1}{N} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i \right\|_\infty \;\le\; c_\varepsilon\, \bar{\sigma} \sqrt{\frac{\log\big((d+p)/\delta\big)}{N}} \qquad \text{with probability at least } 1-\delta. \tag{4.1}$$

The rate on the right hand side is a standard sub-Gaussian score bound. What is not standard is the setting: because the regressors are generated by the closed-loop policy, the score is a sum of dependent terms rather than independent ones, and the usual i.i.d. concentration argument does not apply directly. That a bound of the standard form nonetheless holds rests on the Gaussian structure of the diffusion noise. We make this precise in Section 4.5.

Restricted eigenvalue. The second assumption guarantees that the regression objective has enough curvature to make the estimate well-determined. In high dimensions the empirical Gram $\frac{1}{N}\mathbf{Z}^\top\mathbf{Z}$ is typically rank-deficient, so it cannot be positive definite in all directions. For sparse recovery, however, curvature is needed only in the directions that matter – those lying in a cone around the support of each row.

Assumption 2 (Restricted eigenvalue on row cones)**.** For each row support $\mathcal{S}_i = \operatorname{supp}(\boldsymbol{\theta}_{\star,i})$ there exists a constant $\kappa > 0$ such that

$$\boldsymbol{v}^\top\left(\frac{1}{N}\mathbf{Z}^\top\mathbf{Z}\right)\boldsymbol{v} \;\geq\; \kappa\,\|\boldsymbol{v}\|_2^2 \qquad \text{for all } \boldsymbol{v} \in \mathcal{C}_i, \tag{4.2}$$

where $\mathcal{C}_i := \left\{\boldsymbol{v} \in \mathbb{R}^{d+p} : \|\boldsymbol{v}_{\mathcal{S}_i^c}\|_1 \leq 3\,\|\boldsymbol{v}_{\mathcal{S}_i}\|_1\right\}$ is the cone of vectors whose mass off the support does not greatly exceed their mass on it.

Restricting the curvature requirement to the cone $\mathcal{C}_i$ is what allows the condition to hold despite the Gram being globally rank-deficient. The LASSO solution provably lies in this cone (a fact established in Section 4.2), so curvature outside it is never needed. The restricted eigenvalue condition is essentially an identifiability requirement – it ensures that no sparse direction in the support cone is invisible to the design. As we shall see in Section 4.5, it is precisely this condition that the closed-loop data fails to supply symmetrically across the drift and control blocks of $\boldsymbol{\Theta}_\star$.

Closed-loop regularity. The third assumption concerns the map from parameters to the planning solution, and is what makes the value-perturbation argument of Section 4.3 possible.

Assumption 3 (Uniform closed-loop regularity)**.** There is a neighbourhood $\mathcal{U}$ of $\boldsymbol{\Theta}_\star$ such that for every $\boldsymbol{\Theta} \in \mathcal{U}$

(i) the Riccati ODE (3.10) admits a unique solution $\mathbf{P}_{\boldsymbol{\Theta}} \in C^1\big([0,T];\mathbb{R}^{d\times d}\big)$,

(ii) the solution map $\boldsymbol{\Theta} \mapsto \mathbf{P}_{\boldsymbol{\Theta}}$ is twice continuously differentiable, and

(iii) the second moments of the controlled state process under the gain $\mathbf{K}_{\boldsymbol{\Theta}}$ are uniformly bounded over episodes.

Parts (i) and (ii) are properties of the Riccati equation and follow from the implicit function theorem applied to (3.10), as shown by Basei et al. (2022, Lemma 3.1). Part (iii) requires the closed-loop dynamics under $\mathbf{K}_{\boldsymbol{\Theta}}$ to be stable enough that the state does not diverge in mean square. Unlike (i) and (ii) it is not guaranteed by the structure

of the problem and we monitor it empirically in Chapter 5. Second moments suffice for the analysis because the cost (3.2), the covariance of the controlled state, and the perturbation bounds of Section 4.3 are all second-order objects.

A practitioner's view. Before putting the assumptions to work, it is worth asking what they demand of the world rather than of the analysis: if SAMBL were run on an unknown real system, how much should its operator worry, and about which condition? The modelling premises of Section 3.1 – linearity, Gaussian noise, row sparsity – delimit the problem class itself and raise the separate question of model adequacy, which Chapter 1 has already addressed. Within the class, the three assumptions differ sharply in how likely they are to fail, how visibly they fail, and what can be done about a failure. Here we discuss the intuitive meaning of the assumptions, while Section 4.5 at the end of the chapter develops precise sufficient conditions.

Assumption 1 (adaptive score concentration) warrants the least concern. It asks only that the noise does not conspire with the regressors, and Section 4.5 reduces it to a bound on the actual regressor energy, a condition that any experiment with bounded signals meets and that state and action clipping enforce by construction. Its one unverifiable ingredient is the light tail of the noise itself, which no data set can certify because the noise is never observed However, heavier tails would inflate constants rather than break the mechanism, and they would afflict the dense baseline just as much, so this assumption is a property of the problem rather than a vulnerability of our method.

Assumption 3 (closed-loop regularity) deserves moderate concern, exclusively in part (iii). On an unstable system, a poor early estimate can induce a destabilising gain, and the state may grow substantially within an episode before fresh data corrects the model. This is a genuine operational hazard, but a conspicuous one: a diverging trajectory becomes visible immediately, and the standard safeguards, such as conservative initial play, clipping, and the initial exploration episode of Algorithm 1, act on it directly. Moreover, Section 4.5 shows that the boundedness required here is the same condition that delivers Assumption 1, so a single control-side safeguard serves both.

The condition to worry about most is Assumption 2 (restricted eigenvalue), and the worry peaks, in fact, under *successful* operation: high dimension, a well-regulated loop, low process noise, and long stretches of purely certainty-equivalent play. It is the one assumption that the algorithm's own success erodes: under greedy play the control is a deterministic function of the state, so the control block of the design is excited only through the time variation of the gain (Section 3.3), and the better the loop regulates, the less variation remains. Its failure is also the only silent one: nothing diverges and no cost spikes, but the estimate simply stops improving along the control directions while regret quietly accumulates. The redeeming feature, developed in Section 4.5, is that this most fragile assumption is also the most repairable: the design matrix is observed, so the loss of curvature is detectable online from the empirical Gram, and Proposition 4 shows that any positive amount of injected excitation restores the missing curvature.

The overall picture is therefore reassuring in an unexpected way: the assumptions' ranking by fragility runs opposite to their ranking by verifiability. The condition most likely to fail is the one a practitioner can monitor and repair, while the condition that cannot be checked even in principle is the one least likely to matter.

With the three assumptions in place, the analysis proceeds in two stages. Section 4.2 bounds the parameter estimation error using Assumptions 1 and 2. Section 4.3 converts this error into a per-episode regret bound using Assumption 3. Section 4.4 combines the two into the regret theorem and its corollaries.

4.2 Bounding the Estimation Error

This section bounds the parameter estimation error of the row-wise LASSO (3.8) using Assumptions 1 and 2. The argument follows the classical LASSO analysis, due to Bickel et al. (2009) and given in textbook form by Wainwright (2019, Chapter 7): we specify the regularisation schedule so that it dominates the score uniformly over all episodes, derive a basic inequality from the optimality of the estimate, use the restricted eigenvalue condition to convert that inequality into an error bound, and aggregate the per-row bounds into a joint rate.

Conditional on the two assumptions, this chain is deterministic and standard, and we claim no novelty for it: Lemma 2 and Lemma 3 restate the textbook argument for a single row in our notation. Our setting enters at the two ends of the chain. At the front, the schedule of Lemma 1 must dominate the score simultaneously over all M episodes of the learning trajectory rather than for a single fixed regression. At the back, Proposition 1 aggregates the d per-row bounds into the joint Frobenius rate that the regret analysis consumes.

The estimation bound holds on the event that the score is dominated by half the regularisation parameter, simultaneously across all episodes and all rows. The following lemma specifies the schedule $\{\lambda_m\}_{m=1}^{M}$ that guarantees this.

Lemma 1 (Uniform score bound)**.** *Choose the regularisation schedule*

$$\lambda_m = c_\lambda \, \bar{\sigma} \sqrt{\frac{\log\big((d+p)Md/\delta\big)}{N_m}}, \qquad c_\lambda \geq 2c_\varepsilon. \tag{4.3}$$

Under Assumption 1, with probability at least $1-\delta$*, the score event*

$$\left\| \frac{1}{N_m} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i \right\|_\infty \leq \frac{\lambda_m}{2} \tag{4.4}$$

holds simultaneously for all episodes $m \leq M$ *and all rows* $i \in [d]$*.*

Proof. Apply Assumption 1 to the pair (m, i) at failure level $\delta/(Md)$. The bound (4.1) gives

$$\left\| \frac{1}{N_m} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i \right\|_\infty \leq c_\varepsilon \, \bar{\sigma} \sqrt{\frac{\log\big((d+p)Md/\delta\big)}{N_m}} = \frac{c_\varepsilon}{c_\lambda} \, \lambda_m \leq \frac{\lambda_m}{2},$$

where the last inequality uses $c_\lambda \geq 2c_\varepsilon$. Each pair's bound holds with probability at least $1 - \delta/(Md)$, so a union bound over the Md pairs (m, i) gives the simultaneous statement with probability at least $1 - \delta$. □

The union bound over all M episodes and d rows is what introduces the factor Md inside the logarithm of (4.3), beyond the factor $d + p$ that the per-coordinate score bound already carries. We allocate the failure budget δ uniformly across the Md pairs. A non-uniform allocation, e.g., spending less of it on the early, data-poor episodes, improves the leading constant of the bound, but not its order. Therefore, we keep the uniform allocation for simplicity. The schedule decays as $N_m^{-1/2}$, so the penalty shrinks as data accumulates.

On the score event, the optimality of the LASSO estimate yields a one-sided inequality relating the prediction error to the error on the support. Throughout, we write $\boldsymbol{\Delta}_i := \hat{\boldsymbol{\theta}}_{m,i} - \boldsymbol{\theta}_{\star,i}$ for the estimation error vector of row i at episode m, and $s_i := |\mathcal{S}_i| \leq s$ for the true support size. The episode index m is suppressed where it is clear from context.

Lemma 2 (Basic inequality)**.** *On the score event* (4.4)*,*

$$\underbrace{\frac{1}{2N_m} \|\mathbf{Z}\boldsymbol{\Delta}_i\|_2^2}_{\textit{prediction error}} \leq \underbrace{\frac{3\lambda_m}{2} \, \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1}_{\textit{on-support budget}} - \underbrace{\frac{\lambda_m}{2} \, \big\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\big\|_1}_{\textit{off-support penalty}}. \tag{4.5}$$

Proof. By optimality of $\hat{\boldsymbol{\theta}}_{m,i}$ in (3.8),

$$\frac{1}{2N_m} \big\|\mathbf{Y}_{:,i} - \mathbf{Z}\hat{\boldsymbol{\theta}}_{m,i}\big\|_2^2 + \lambda_m \|\hat{\boldsymbol{\theta}}_{m,i}\|_1 \leq \frac{1}{2N_m} \big\|\mathbf{Y}_{:,i} - \mathbf{Z}\boldsymbol{\theta}_{\star,i}\big\|_2^2 + \lambda_m \|\boldsymbol{\theta}_{\star,i}\|_1.$$

Substituting $\mathbf{Y}_{:,i} = \mathbf{Z}\boldsymbol{\theta}_{\star,i} + \boldsymbol{\varepsilon}_i$, expanding, and cancelling the common term $\|\boldsymbol{\varepsilon}_i\|_2^2/(2N_m)$ gives

$$\frac{1}{2N_m} \|\mathbf{Z}\boldsymbol{\Delta}_i\|_2^2 \leq \langle \tfrac{1}{N_m} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i, \, \boldsymbol{\Delta}_i \rangle + \lambda_m \big(\|\boldsymbol{\theta}_{\star,i}\|_1 - \|\hat{\boldsymbol{\theta}}_{m,i}\|_1 \big). \tag{4.6}$$

Score term. By Hölder's inequality and (4.4),

$$|\langle \tfrac{1}{N_m} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i, \, \boldsymbol{\Delta}_i \rangle| \overset{\text{Hölder}}{\leq} \|\tfrac{1}{N_m} \mathbf{Z}^\top \boldsymbol{\varepsilon}_i\|_\infty \|\boldsymbol{\Delta}_i\|_1 \overset{(4.4)}{\leq} \tfrac{\lambda_m}{2} \|\boldsymbol{\Delta}_i\|_1 = \tfrac{\lambda_m}{2} (\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 + \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1).$$

Support decomposition.

- On $\mathcal{S}_i$, we have $(\hat{\boldsymbol{\theta}}_{m,i})_{\mathcal{S}_i} = (\boldsymbol{\Delta}_i)_{\mathcal{S}_i} + (\boldsymbol{\theta}_{\star,i})_{\mathcal{S}_i}$ and by the reverse triangle inequality

$$\|(\hat{\boldsymbol{\theta}}_{m,i})_{\mathcal{S}_i}\|_1 = \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i} + (\boldsymbol{\theta}_{\star,i})_{\mathcal{S}_i}\|_1 \geq \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 - \|(\boldsymbol{\theta}_{\star,i})_{\mathcal{S}_i}\|_1$$

- On $\mathcal{S}_i^c$, we have $(\hat{\boldsymbol{\theta}}_{m,i})_{\mathcal{S}_i^c} = (\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c} + \underbrace{(\boldsymbol{\theta}_{\star,i})_{\mathcal{S}_i^c}}_{0} = (\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}$

Taken together: $\|\hat{\boldsymbol{\theta}}_{m,i}\|_1 \geq \|\boldsymbol{\theta}_{\star,i}\|_1 - \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 + \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1$, so

$$\|\boldsymbol{\theta}_{\star,i}\|_1 - \|\hat{\boldsymbol{\theta}}_{m,i}\|_1 \leq \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 - \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1.$$

Substituting both bounds into (4.6) gives

$$\begin{aligned}\frac{1}{2N_m}\|\mathbf{Z}\boldsymbol{\Delta}_i\|_2^2 &\leq \frac{\lambda_m}{2}\big(\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 + \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1\big) + \lambda_m\big(\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 - \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1\big) \\ &= \frac{3\lambda_m}{2}\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 - \frac{\lambda_m}{2}\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1.\end{aligned}$$

□

Dropping the non-negative prediction-error term from (4.5) shows that the error lies in the cone of Assumption 2, on which the restricted eigenvalue condition then supplies the curvature needed to bound the error.

Lemma 3 (Cone-based error control)**.** *Under Assumption 2, on the score event (4.4), the error $\boldsymbol{\Delta}_i$ lies in the cone $\mathcal{C}_i$ and satisfies*

$$\|\boldsymbol{\Delta}_i\|_2 \leq \frac{3\lambda_m\sqrt{s_i}}{\kappa}, \qquad \|\boldsymbol{\Delta}_i\|_1 \leq \frac{12\lambda_m s_i}{\kappa}. \tag{4.7}$$

Proof. Dropping the non-negative quadratic prediction-error term in the basic inequality (4.5) gives $\frac{\lambda_m}{2}\|\boldsymbol{\Delta}_{i,\mathcal{S}_i^c}\|_1 \leq \frac{3\lambda_m}{2}\|\boldsymbol{\Delta}_{i,\mathcal{S}_i}\|_1$, which is the cone condition $\boldsymbol{\Delta}_i \in \mathcal{C}_i$. On the cone, Assumption 2 provides curvature:

$$\kappa\|\boldsymbol{\Delta}_i\|_2^2 \overset{\text{Ass. 2}}{\leq} \frac{1}{N_m}\|\mathbf{Z}\boldsymbol{\Delta}_i\|_2^2 \overset{(4.5)}{\leq} 3\lambda_m \underbrace{\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1}_{\mathbf{1}^\top|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}|} \overset{\text{C.-S.}}{\leq} 3\lambda_m\sqrt{s_i}\,\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_2 \leq 3\lambda_m\sqrt{s_i}\,\|\boldsymbol{\Delta}_i\|_2,$$

where the second inequality is (4.5) after dropping the off-support term, the third is Cauchy–Schwarz on the s_i support coordinates, and the fourth uses $\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_2 \leq \|\boldsymbol{\Delta}_i\|_2$. For $\boldsymbol{\Delta}_i = \mathbf{0}$ the ℓ_2 bound is trivial and otherwise, dividing by $\|\boldsymbol{\Delta}_i\|_2$ gives it. The ℓ_1 bound follows from the cone condition:

$$\|\boldsymbol{\Delta}_i\|_1 = \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 + \|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i^c}\|_1 \overset{\boldsymbol{\Delta}_i\in\mathcal{C}_i}{\leq} 4\|(\boldsymbol{\Delta}_i)_{\mathcal{S}_i}\|_1 \leq 4\sqrt{s_i}\,\|\boldsymbol{\Delta}_i\|_2 \leq 12\lambda_m s_i/\kappa.$$

□

Aggregating the per-row ℓ_2 bounds over the d rows gives the rate for the full parameter matrix in Frobenius norm.

Proposition 1 (Joint sparse estimation rate)**.** *Under Assumptions 1 and 2, on the score event (4.4), the joint estimate at episode m satisfies*

$$\left\|\hat{\boldsymbol{\Theta}}_m - \boldsymbol{\Theta}_\star\right\|_F^2 \leq \frac{9\, d\, s\, \lambda_m^2}{\kappa^2} \overset{(4.3)}{=} \frac{9\, d\, s\, c_\lambda^2\, \bar{\sigma}^2}{\kappa^2} \cdot \frac{\log\big((d+p)Md/\delta\big)}{N_m}. \tag{4.8}$$

Proof. The squared Frobenius error decomposes by rows as $\|\hat{\boldsymbol{\Theta}}_m - \boldsymbol{\Theta}_\star\|_F^2 = \sum_{i=1}^d \|\boldsymbol{\Delta}_i\|_2^2$. By the ℓ_2 bound of Lemma 3, $\|\boldsymbol{\Delta}_i\|_2^2 \leq 9\lambda_m^2 s_i/\kappa^2$. Summing over i and using $s_i \leq s$ for all i gives $\sum_i \|\boldsymbol{\Delta}_i\|_2^2 \leq 9\lambda_m^2(\sum_i s_i)/\kappa^2 \leq 9ds\lambda_m^2/\kappa^2$. Substituting the schedule (4.3) gives the second form. □

Proposition 1 is the estimation rate that the regret analysis consumes. Per row, it matches the classical LASSO rate (2.18) with ambient dimension $q = d + p$, up to the union-bound factor Md inside the logarithm, and the d rows supply the remaining factor d. The dependence on the problem is through the factor ds: the rate scales with the row dimension d and the row support size s, rather than with the ambient parameter count $d(d+p)$ that a dense estimator would incur. The next section converts this estimation rate into a per-episode regret bound, and Section 4.4 shows that the ds scaling is precisely what yields the sparse advantage over the dense $d(d+p)$ rate.

4.3 Bounding the Per-Episode Regret

Section 4.2 bounds how far the parameter estimate $\hat{\boldsymbol{\Theta}}_m$ lies from the truth. To turn this into a regret bound we need to know how the cost responds to a parameter error: a small estimation error should translate into a small excess cost. This section establishes that translation. The central object is the *certainty-mismatch cost*

$$F(\boldsymbol{\Theta}) \coloneqq J(\boldsymbol{\pi}_{\boldsymbol{\Theta}}), \tag{4.9}$$

the cost of the policy $\boldsymbol{\pi}_{\boldsymbol{\Theta}}(t, \boldsymbol{x}(t)) = \mathbf{K}_{\boldsymbol{\Theta}}(t)\boldsymbol{x}(t)$ induced by the dynamics estimate $\boldsymbol{\Theta}$, executed on the true system with dynamics $\boldsymbol{\Theta}_\star$. By construction $F(\boldsymbol{\Theta}_\star) = J_\star$, the optimal cost, since the gain optimal for the true dynamics is the globally optimal policy (cp. Theorem B.1(ii)). The per-episode regret of Section 3.1 is exactly the excess of F over its minimum: with the gain $\mathbf{K}_{\hat{\boldsymbol{\Theta}}_{m-1}}$ executed in episode m,

$$r_m = J(\boldsymbol{\pi}_{\hat{\boldsymbol{\Theta}}_{m-1}}) - J_\star = F(\hat{\boldsymbol{\Theta}}_{m-1}) - F(\boldsymbol{\Theta}_\star). \tag{4.10}$$

Bounding regret therefore reduces to bounding the increment of F away from its minimiser $\boldsymbol{\Theta}_\star$. The following proposition provides two such bounds: a Lipschitz bound valid on a neighbourhood, and a sharper quadratic bound valid close to the minimiser.

The quadratic bound is the finite-horizon episodic analogue of the certainty-equivalence efficiency established by Mania et al. (2019) for the infinite-horizon LQR.

Proposition 2 (Value perturbation)**.** *Under Assumption 3, F is twice continuously differentiable on a compact convex neighbourhood $\mathcal{U}$ of $\boldsymbol{\Theta}_\star$, and the following hold.*

(i) ***Lipschitz bound.*** *For all $\boldsymbol{\Theta} \in \mathcal{U}$,*

$$F(\boldsymbol{\Theta}) - F(\boldsymbol{\Theta}_\star) \le L_1 \|\boldsymbol{\Theta} - \boldsymbol{\Theta}_\star\|_F, \qquad L_1 := \sup_{\boldsymbol{\Theta}' \in \mathcal{U}} \|\nabla F(\boldsymbol{\Theta}')\|_F. \tag{4.11}$$

(ii) ***Local quadratic bound.*** *For all $\boldsymbol{\Theta} \in \mathcal{U}$,*

$$F(\boldsymbol{\Theta}) - F(\boldsymbol{\Theta}_\star) \le L_2 \|\boldsymbol{\Theta} - \boldsymbol{\Theta}_\star\|_F^2, \qquad L_2 := \tfrac{1}{2} \sup_{\boldsymbol{\Theta}' \in \mathcal{U}} \|\nabla^2 F(\boldsymbol{\Theta}')\|_{\mathrm{op}}. \tag{4.12}$$

Proof. Twice continuous differentiability of F on $\mathcal{U}$ follows from Assumption 3(ii): the cost $J(\boldsymbol{\pi}_{\boldsymbol{\Theta}})$ is a smooth functional of the Riccati solution $\mathbf{P}_{\boldsymbol{\Theta}}$, which is twice continuously differentiable in $\boldsymbol{\Theta}$, and Assumption 3(iii) ensures the relevant second moments of the state are finite so that F and its derivatives are well-defined. Now denote $\boldsymbol{\theta} := \mathrm{vec}(\boldsymbol{\Theta})$ and $f(\boldsymbol{\theta}) := F(\boldsymbol{\Theta})$, and note that $\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|_2 = \|\boldsymbol{\Theta} - \boldsymbol{\Theta}_\star\|_F$.

(i) By the mean value theorem, $f(\boldsymbol{\theta}) - f(\boldsymbol{\theta}_\star) = \nabla f(\tilde{\boldsymbol{\theta}})^\top (\boldsymbol{\theta} - \boldsymbol{\theta}_\star)$ for some $\tilde{\boldsymbol{\theta}}$ on the segment $[\boldsymbol{\theta}_\star, \boldsymbol{\theta}]$. Cauchy-Schwarz gives $f(\boldsymbol{\theta}) - f(\boldsymbol{\theta}_\star) \le \|\nabla f(\tilde{\boldsymbol{\theta}})\|_2 \, \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|_2 \le L_1 \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|_2$.

(ii) Since $\boldsymbol{\pi}_{\boldsymbol{\Theta}_\star}$ is the cost-minimising policy for the true dynamics (Theorem B.1(ii)), $\boldsymbol{\theta}_\star$ minimises f and therefore $\nabla f(\boldsymbol{\theta}_\star) = \mathbf{0}$. Taylor's theorem with the Lagrange remainder then gives, for some $\tilde{\boldsymbol{\theta}}$ on the segment $[\boldsymbol{\theta}_\star, \boldsymbol{\theta}]$,

$$f(\boldsymbol{\theta}) - f(\boldsymbol{\theta}_\star) = \tfrac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_\star)^\top \nabla^2 f(\tilde{\boldsymbol{\theta}}) \, (\boldsymbol{\theta} - \boldsymbol{\theta}_\star) \le L_2 \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|_2^2. \qquad \square$$

The two bounds play complementary roles in the regret analysis. The Lipschitz bound (4.11) holds throughout the neighbourhood and is used in the early episodes, when the estimate may be far from the truth. It gives a per-episode regret of order $\|\hat{\boldsymbol{\Theta}}_{m-1} - \boldsymbol{\Theta}_\star\|_F$. The quadratic bound (4.12) is sharper when the estimate is close to $\boldsymbol{\Theta}_\star$, where it gives a per-episode regret of order $\|\hat{\boldsymbol{\Theta}}_{m-1} - \boldsymbol{\Theta}_\star\|_F^2$. Combined with the estimation rate of Proposition 1, the Lipschitz bound yields a per-episode regret decaying as $N_{m-1}^{-1/2}$ and the quadratic bound one decaying as N_{m-1}^{-1}.

Remark (Why the finite horizon is essential)*.* The crux of the quadratic bound is the vanishing gradient $\nabla F(\boldsymbol{\Theta}_\star) = \mathbf{0}$, which relies on $\boldsymbol{\pi}_{\boldsymbol{\Theta}_\star}$ being a *minimiser* of the cost at the true dynamics. This holds because $\mathbf{K}_\star = \mathbf{K}_{\boldsymbol{\Theta}_\star}$ is the solution of the finite-horizon Riccati differential equation (3.10), which Theorem B.1 certifies as exactly the cost-minimising gain for the finite-horizon objective (3.2). Had the algorithm instead planned with the algebraic Riccati equation, the resulting constant gain $\mathbf{K}_\infty$ would be optimal for the

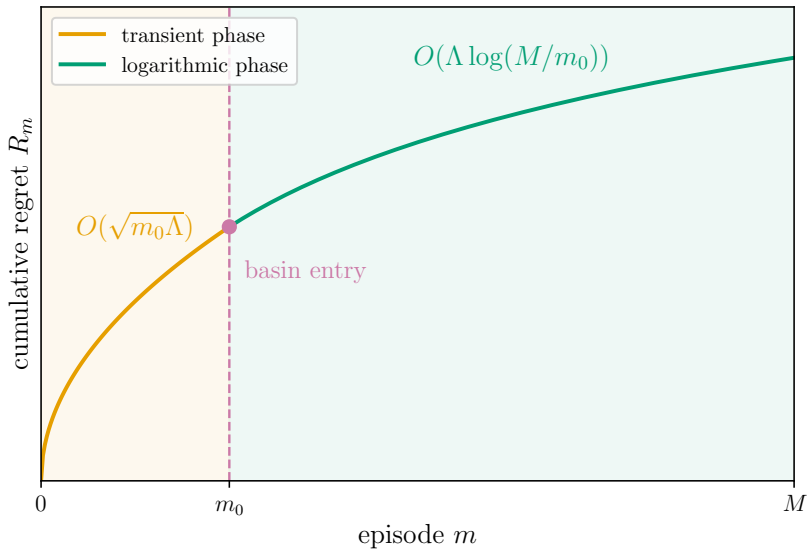


Figure 1: Illustration of the regret decomposition afforded by Theorem 1.

infinite-horizon cost but *not* for the finite-horizon cost J. Then $\boldsymbol{\Theta}_\star$ would not minimise F, the gradient $\nabla f(\boldsymbol{\theta}_\star)$ would not vanish, and only the Lipschitz bound would survive.

4.4 Main Results

We now combine the estimation rate of Section 4.2 with the per-episode bounds of Section 4.3 to obtain regret guarantees. The estimation rate controls how fast the parameter error shrinks and the per-episode bounds convert that error into per-episode regret. The latter give two regimes: the Lipschitz bound governs the early episodes, when the estimate is far from the truth, and the sharper quadratic bound governs the later episodes, once the estimate has entered the region where it applies. The transition between the two, as illustrated in Figure 1, takes place in the basin-entry episode, and locating it is what reveals the sparse advantage.

Throughout, we abbreviate the per-step estimation complexity as

$$\Lambda := \frac{d\, s\, \bar{\sigma}^2 \log\big((d+p)Md/\delta\big)}{H}, \tag{4.13}$$

so that the estimation rate of Proposition 1 reads $\|\hat{\boldsymbol{\Theta}}_m - \boldsymbol{\Theta}_\star\|_F^2 \le C_0\, \Lambda H / N_m = C_0\, \Lambda / m$ for a constant C_0 absorbing $9c_\lambda^2/\kappa^2$, using $N_m = mH$.

Theorem 1 (Regret decomposition)**.** *Under Assumptions 1–3, with the schedule (4.3) and $N_m = mH$, the following hold with probability at least $1-\delta$.*

(i) ***Global bound.*** *For all $m \geq 2$, the Lipschitz bound gives $r_m \leq C_1\sqrt{\Lambda/(m-1)}$, and therefore*

$$R_M \leq r_1 + C_1\sqrt{\Lambda}\sum_{m=2}^{M}(m-1)^{-1/2} = O\big(\sqrt{M\Lambda}\big). \tag{4.14}$$

(ii) ***Transient–logarithmic decomposition.*** *Suppose there exists an episode m_0 such that $\|\hat{\boldsymbol{\Theta}}_m - \boldsymbol{\Theta}_\star\|_F \leq \varepsilon_0$ for all $m \geq m_0$, where ε_0 is the radius of the neighbourhood $\mathcal{U}$ of Proposition 2. For those m, the local quadratic bound gives $r_m \leq C_2\Lambda/(m-1)$, and therefore*

$$R_M = \sum_{m=1}^{m_0} r_m + \sum_{m=m_0+1}^{M} r_m \leq \underbrace{r_1 + C_1\sqrt{m_0\Lambda}}_{\textit{transient}} + \underbrace{C_2\,\Lambda \log\frac{M}{m_0}}_{\textit{logarithmic tail}} = O\big(\Lambda \log M\big). \tag{4.15}$$

Here C_1 absorbs L_1, c_λ, and κ^{-1}, and C_2 absorbs L_2, c_λ^2, and κ^{-2}.

Proof. On the score event of Lemma 1, which holds with probability at least $1-\delta$, the estimation rate of Proposition 1 gives $\|\hat{\boldsymbol{\Theta}}_{m-1} - \boldsymbol{\Theta}_\star\|_F \leq \sqrt{C_0\Lambda/(m-1)}$ for each $m \geq 2$.

Part (i). The per-episode regret is $r_m = F(\hat{\boldsymbol{\Theta}}_{m-1}) - F(\boldsymbol{\Theta}_\star)$ by (4.10). The Lipschitz bound of Proposition 2(i) gives $r_m \leq L_1\|\hat{\boldsymbol{\Theta}}_{m-1} - \boldsymbol{\Theta}_\star\|_F \leq C_1\sqrt{\Lambda/(m-1)}$. Summing over m and using $\sum_{k=1}^{M-1} k^{-1/2} \leq 2\sqrt{M}$ (lower Riemann sum) yields (4.14).

Part (ii). Split the sum at m_0. For $m \leq m_0$, apply the Lipschitz bound as in part (i): $\sum_{m=1}^{m_0} r_m \leq r_1 + C_1\sqrt{m_0\Lambda}$. For $m > m_0$, the estimate lies in $\mathcal{U}$ by hypothesis, so the quadratic bound of Proposition 2(ii) applies: $r_m \leq L_2\|\hat{\boldsymbol{\Theta}}_{m-1} - \boldsymbol{\Theta}_\star\|_F^2 \leq C_2\Lambda/(m-1)$. Summing the tail, $\sum_{m=m_0+1}^{M}(m-1)^{-1} \leq 1 + \log(M/m_0)$, gives the logarithmic term. Combining the two phases yields (4.15). □

The two phases have a clear interpretation. The transient term $C_1\sqrt{m_0\Lambda}$ is the cost of the initial learning period, during which the estimate is too inaccurate for the quadratic bound to apply. It is a finite constant, independent of M. The logarithmic tail is the residual cost once the estimate has entered the basin: each subsequent episode contributes a regret decaying as $1/m$, whose sum over M episodes grows only logarithmically. Crucially, m_0 depends on the estimation complexity but not on M, so the transient cost does not grow with the horizon. The leading $O(\sqrt{M\Lambda})$ term of part (i) is a worst-case bound that ignores basin entry. When basin entry occurs early, part (ii) is the operative bound.

The transient term in (4.15) scales with the basin-entry episode m_0: the earlier the estimate enters the basin, the less regret accumulates before the log-regime takes over. We now compare when an estimator's guarantee *certifies* basin entry, and show the sparse certificate arrives earlier than the dense one, by a sparsity-dependent factor.

Corollary 1 (Certified basin entry and the sparse advantage)**.** *For an estimator with error guarantee* $\|\hat{\mathbf{\Theta}}_m - \mathbf{\Theta}_\star\|_F^2 \le C\psi/N_m$*, define the* certified basin-entry episode $\bar{m}_0 := \min\{m \in \mathbb{N} : C\psi/N_m \le \varepsilon_0^2\}$*, the first episode at which the guarantee places the estimate in the neighbourhood* $\mathcal{N}$ *of Proposition 2. The sparse rate of Proposition 1 certifies basin entry at*

$$\bar{m}_0^{\text{sparse}} = \Theta\left(\frac{\bar{\sigma}^2\, d\, s\, \log\big((d+p)Md/\delta\big)}{H\varepsilon_0^2}\right). \tag{4.16}$$

Ordinary least squares (OLS) on all $d(d+p)$ *entries of* $\mathbf{\Theta}_\star$*, achieves the standard rate* $\|\hat{\mathbf{\Theta}}-\mathbf{\Theta}_\star\|_F^2 = O\big(\bar{\sigma}^2(d(d+p)+\log(Md/\delta))/N\big)$*, which is minimax-optimal up to constants for dense estimation (Wainwright, 2019, Chapter 15), and its guarantee certifies basin entry at*

$$\bar{m}_0^{\text{dense}} = \Theta\left(\frac{\bar{\sigma}^2\big(d(d+p)+\log(Md/\delta)\big)}{H\varepsilon_0^2}\right). \tag{4.17}$$

Treating M *and* δ *as fixed, the ratio of certified basin-entry episodes satisfies*

$$\frac{\bar{m}_0^{\text{dense}}}{\bar{m}_0^{\text{sparse}}} \;=\; \Omega\left(\frac{d+p}{s\log(d+p)}\right) \quad \text{as } d\to\infty. \tag{4.18}$$

Hence the sparse guarantee certifies entry into the quadratic regime earlier whenever $s\log(d+p) \ll d+p$*.*

Proof. Certified entry requires the guarantee to fall below the basin radius, $C\psi/N_m \le \varepsilon_0^2$, which holds once $N_m \ge C\psi/\varepsilon_0^2$. Since $N_m = mH$, the certified basin-entry episode is *exactly* $\bar{m}_0 = \lceil C\psi/(H\varepsilon_0^2)\rceil$. Whenever certifying entry requires more than one episode (the case of interest), we have $C\psi > H\varepsilon_0^2$ and so $x \le \lceil x\rceil < x+1 \le 2x$ places $\bar{m}_0$ within a factor of two of $C\psi/(H\varepsilon_0^2)$. This two-sided control is what gives the $\Theta(\cdot)$ statements and, below, the lower bound on the ratio. Substituting the sparse and dense complexities $\psi_{\text{sparse}} = \bar{\sigma}^2 ds\log((d+p)Md/\delta)$ and $\psi_{\text{dense}} = \bar{\sigma}^2(d(d+p)+\log(Md/\delta))$ results in (4.16) and (4.17), respectively.

For the ratio, the constants $\bar{\sigma}^2$, H, and ε_0^2 cancel, while the two estimators' constants C and the ceilings change the ratio by at most a constant factor, absorbed into $c > 0$. Since $\log(Md/\delta) \ge 0$, dropping it from the numerator gives a lower bound:

$$\frac{\bar{m}_0^{\text{dense}}}{\bar{m}_0^{\text{sparse}}} \ge \frac{c\,(d(d+p)+\cancel{\log(Md/\delta)})}{ds\log((d+p)Md/\delta)} \ge \frac{c\,\cancel{d}(d+p)}{\cancel{d}s\log((d+p)Md/\delta)} = \frac{c\,(d+p)}{s\big(\log(d+p)+\log(Md/\delta)\big)}.$$

With M and δ fixed, $\log(Md/\delta)$ is a constant while $\log(d+p)\to\infty$, so the denominator is $s\log(d+p)(1+o(1))$ and the ratio is $\Omega\big((d+p)/(s\log(d+p))\big)$ as $d\to\infty$. □

The dense rate carries the full parameter count $d(d+p)$ where the sparse rate carries ds. This is precisely the sample-complexity gap between dense and sparse estimation, and (4.18) is its consequence for control. Note that the bracket in the dense rate is $d(d+p)+\log(Md/\delta)$, not $d(d+p+\log(Md/\delta))$. The logarithmic concentration term enters additively, because the Frobenius-norm concentration is carried out jointly over all rows rather than row by row. The substantive comparison is therefore ds against $d(d+p)$, and the speedup $(d+p)/(s\log(d+p))$ is large precisely in the high-dimensional sparse regime that this thesis addresses. We stress that Corollary 1 compares the two estimators' *guarantees*: the minimax optimality of the dense rate means no dense estimator can certify entry substantially earlier, but on a benign instance either estimator may enter the true basin before its certificate says so.

In theory, we can improve even further. The leading $\sqrt{M}$ term in Theorem 1(i) is an artefact of updating the estimate every episode, so that early episodes act on poor estimates built from little data. A standard doubling schedule removes it, as Basei et al. (2022) have already shown for the dense estimation setting. We adapt it for sparse estimation.

Corollary 2 (Logarithmic regret with doubling)**.** *Modify Algorithm 1 so that the estimate is updated only at episodes $m = 2^0, 2^1, 2^2, \dots$, and the gain is held fixed within each batch: for $2^\ell \le m < 2^{\ell+1}$, the agent uses the estimate $\hat{\mathbf{\Theta}}^{(\ell)}$ computed from the first $2^\ell H$ samples. Then under Assumptions 1–3, with probability at least $1-\delta$,*

$$R_M = O\left(\frac{d\,s\,\bar{\sigma}^2 \log\big((d+p)Md/\delta\big)}{H}\,\log M\right) = O\big(\Lambda \log M\big). \tag{4.19}$$

Proof. There are $L = \lfloor \log_2 M \rfloor + 1$ batches. In batch ℓ, the estimate $\hat{\mathbf{\Theta}}^{(\ell)}$ is built from $N^{(\ell)} = 2^\ell H$ samples, so by Proposition 1, $\|\hat{\mathbf{\Theta}}^{(\ell)} - \mathbf{\Theta}_\star\|_F^2 \le C_0\Lambda/2^\ell$. Once ℓ is large enough that the estimate lies in the basin, the quadratic bound gives per-episode regret $r_m \le L_2\|\hat{\mathbf{\Theta}}^{(\ell)} - \mathbf{\Theta}_\star\|_F^2 \le C_2\Lambda/2^\ell$. Batch ℓ contains 2^ℓ episodes, so its total contribution is $2^\ell \cdot C_2\Lambda/2^\ell = C_2\Lambda$, independent of ℓ. Summing over the $O(\log M)$ batches gives (4.19). The batches before basin entry contribute $O(1)$ each by the global bound and do not change the order. □

The mechanism is that doubling makes each batch contribute a constant amount of regret: the per-episode error decays as $2^{-\ell}$ while the batch size grows as 2^ℓ, and the two cancel. This matches the rate obtained by Basei et al. (2022) for dense estimation up to the replacement of the dense complexity $d(d+p)$ by the sparse complexity ds inside Λ. It is this constant in which the sparse advantage appears, as in Corollary 1. In practice the every-episode update of Theorem 1 is simpler and performs well empirically, since the $\sqrt{M}$ term is a worst-case bound that overestimates the transient when basin entry is early. Further parameter re-estimation timing strategies are left to future work.

4.5 Satisfying the Assumptions

The regret guarantees of Section 4.4 hold under Assumptions 1–3. This section explores when those assumptions are themselves justified. Assumptions 1 and 3 admit sufficient conditions in terms of the system primitives. Assumption 2 is the delicate one, and we give a sufficient condition for it under excitation while connecting its failure mode to the closed-loop identifiability discussion of Section 3.3. Throughout, the recurring requirement is that the closed-loop trajectories remain well-behaved, i.e., bounded in an appropriate sense, which links the statistical assumptions back to the control-side regularity of Assumption 3.

Assumption 1 follows from a single condition on the design matrix: that the columns of $\mathbf{Z}$ do not carry too much energy. Recall from Section 4.2 that the score $\frac{1}{N}\mathbf{Z}^\top \boldsymbol{\varepsilon}_i$ is a sum of martingale differences, each with conditional mean zero given the past. The following proposition, adapted for our setting from Wainwright (2019), turns a bound on the regressor energy into the sub-Gaussian tail that Assumption 1 requires.

Proposition 3 (Score concentration)**.** *Suppose the columns of* $\mathbf{Z}$ *are uniformly bounded in energy: there exists* $B > 0$ *such that*

$$\max_{j\in[d+p]} \sqrt{\frac{1}{N}\sum_{k=1}^{N} z_{k,j}^2} \le B \tag{4.20}$$

along every trajectory. Then for each row $i \in [d]$ *and every* $\delta \in (0,1)$*,*

$$\left\| \frac{1}{N}\mathbf{Z}^\top \boldsymbol{\varepsilon}_i \right\|_\infty \le B\,\bar{\sigma}\sqrt{\frac{2\log\big(2(d+p)/\delta\big)}{N}} \qquad \text{with probability at least } 1-\delta, \tag{4.21}$$

so Assumption 1 holds with c_ε *a constant multiple of* B*, i.e.,* $c_\varepsilon \ge B\sqrt{2}(1+\sqrt{\frac{\log(2)}{\log((d+p)/\delta)}})$*.*

Proof. Fix a row i and coordinate j, and let $S_j = \frac{1}{N}\sum_{k=1}^{N}\xi_k$ with $\xi_k = z_{k,j}\varepsilon_{k,i}$. The terms ξ_k are not independent as the regressor row $\boldsymbol{z}_k$ depends on the past trajectory, so a standard Hoeffding bound does not apply. They do, however, form a martingale-difference sequence: $z_{k,j}$ is measurable with respect to the history $\mathcal{F}_{k-1}$, while $\varepsilon_{k,i} \mid \mathcal{F}_{k-1} \sim \mathcal{N}(0,\bar{\sigma}^2)$ is the fresh increment, so $\mathbb{E}[\xi_k \mid \mathcal{F}_{k-1}] = 0$.

We bound the moment generating function of $\sum_{k=1}^{N}\xi_k$. Using the Gaussian identity $\mathbb{E}[e^{tg}] = e^{t^2v^2/2}$ for $g \sim \mathcal{N}(0,v^2)$ conditionally, $\mathbb{E}[e^{t\xi_k} \mid \mathcal{F}_{k-1}] = \exp(\frac{t^2}{2}z_{k,j}^2\bar{\sigma}^2)$. This still carries the random factor $z_{k,j}^2$, so the per-step moment-generating functions cannot simply be multiplied. The remedy is to subtract the accumulated conditional variance:

define

$$M_k \coloneqq \exp\left(t\sum_{l=1}^{k}\xi_l - \frac{t^2\bar{\sigma}^2}{2}\sum_{l=1}^{k}z_{l,j}^2\right) = \begin{cases} M_{k-1}\exp\left(t\xi_k - \frac{t^2\bar{\sigma}^2}{2}z_{k,j}^2\right) & \text{for } k>0 \\ 1 & \text{for } k=0 \end{cases},$$

so the correction cancels the conditional moment-generating function at each step, i.e., $\mathbb{E}[M_k \mid \mathcal{F}_{k-1}] = M_{k-1}$. Thus $(M_k)_{k=0}^N$ is a martingale with $\mathbb{E}[M_N] = \ldots = \mathbb{E}[M_0] = 1$, and

$$\mathbb{E}\Big[\exp\Big(t\textstyle\sum_k \xi_k\Big)\Big] = \mathbb{E}\Big[M_N \exp\Big(\frac{t^2\bar{\sigma}^2}{2}\textstyle\sum_k z_{k,j}^2\Big)\Big] \le \exp\Big(\frac{t^2\bar{\sigma}^2 NB^2}{2}\Big),$$

where the inequality uses the energy bound (4.20) in the form $\sum_k z_{k,j}^2 \le NB^2$, which holds surely and can be pulled out of the expectation. Hence $\sum_k \xi_k$ is sub-Gaussian with variance proxy $\bar{\sigma}^2 NB^2$. A Chernoff bound gives $\mathbb{P}(|S_j| > a) \le 2\exp(-Na^2/(2\bar{\sigma}^2B^2))$. Setting the right-hand side to $\delta/(d+p)$, solving for a, and taking a union bound over the $d+p$ coordinates yields (4.21), the $\log(d+p)$ being the cost of the union bound. □

The energy bound (4.20) is a pathwise condition on the controlled trajectory, slightly stronger than the second-moment control of Assumption 3(iii): it bounds the realised regressor energy along every trajectory rather than only in expectation. It holds whenever the closed-loop state and control remain bounded over the episode. This can be achieved by state and action clipping, which enforces it by construction. The two conditions share the same underlying requirement that the closed-loop dynamics do not diverge.

Assumption 2 is the one that closed-loop data does not supply automatically. Under the deterministic optimal policy, the self-exploration property of Section 3.3 makes the empirical Gram full rank. This is enough for the dense estimator, but it does not guarantee curvature along the control-support directions, because the control carries no variation independent of the state. This is the state–control asymmetry: in the degenerate case of Section 3.3.2 the control-block curvature is exactly zero. Deliberate excitation removes the obstruction, and the following proposition makes the mechanism precise at the population level.

Proposition 4 (Restricted eigenvalue under excitation)**.** *Suppose the agent applies the excited policy $\boldsymbol{u}_k = \mathbf{K}(t_k)\boldsymbol{x}_k + \boldsymbol{\eta}_k$ with $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \sigma_u^2\mathbf{I}_p)$ drawn independently of the state and across steps, and that the initial state is non-degenerate, $\mathbb{E}[\boldsymbol{x}_0\boldsymbol{x}_0^\top] \succeq \sigma^2\Delta t\,\mathbf{I}_d$. Write any $\boldsymbol{v} \in \mathbb{R}^{d+p}$ in its state and control components $\boldsymbol{v} = (\boldsymbol{v}_A, \boldsymbol{v}_B)$. Then the population Gram admits the decomposition*

$$\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\,\boldsymbol{v} = \frac{1}{N}\sum_{k=1}^{N}\big(\boldsymbol{v}_A+\mathbf{K}(t_k)^\top\boldsymbol{v}_B\big)^\top\mathbb{E}[\boldsymbol{x}_k\boldsymbol{x}_k^\top]\,\big(\boldsymbol{v}_A+\mathbf{K}(t_k)^\top\boldsymbol{v}_B\big) + \sigma_u^2\,\|\boldsymbol{v}_B\|_2^2, \quad (4.22)$$

and satisfies the restricted eigenvalue bound

$$\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\,\boldsymbol{v} \;\ge\; \kappa\,\|\boldsymbol{v}\|_2^2 \quad \textit{for all } \boldsymbol{v} \in \mathbb{R}^{d+p}, \qquad \kappa \asymp \min\big(\sigma^2\Delta t,\, \sigma_u^2\big), \quad (4.23)$$

where the implied constant depends only on $K_{\max} \coloneqq \sup_k \|\mathbf{K}(t_k)\|$.

Proof. Fix a step k, condition on $\boldsymbol{x}_k$, and write $\mathbf{K}_k = \mathbf{K}(t_k)$. The excited regressor is $\boldsymbol{z}_k = [\boldsymbol{x}_k^\top,\ (\mathbf{K}_k\boldsymbol{x}_k + \boldsymbol{\eta}_k)^\top]^\top$ with $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \sigma_u^2\mathbf{I}_p)$ independent of $\boldsymbol{x}_k$, so its conditional second moment is

$$\mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top \mid \boldsymbol{x}_k] = \begin{bmatrix} \boldsymbol{x}_k\boldsymbol{x}_k^\top & \boldsymbol{x}_k\boldsymbol{x}_k^\top\mathbf{K}_k^\top \\ \mathbf{K}_k\boldsymbol{x}_k\boldsymbol{x}_k^\top & \mathbf{K}_k\boldsymbol{x}_k\boldsymbol{x}_k^\top\mathbf{K}_k^\top + \sigma_u^2\mathbf{I}_p \end{bmatrix},$$

the $\sigma_u^2\mathbf{I}_p$ arising from $\mathbb{E}[\boldsymbol{\eta}_k\boldsymbol{\eta}_k^\top]$. Contracting with $\boldsymbol{v} = (\boldsymbol{v}_A, \boldsymbol{v}_B)$ and using that the two off-diagonal blocks are mutual transposes gives

$$\boldsymbol{v}^\top\mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top \mid \boldsymbol{x}_k]\boldsymbol{v} = \boldsymbol{v}_A^\top\boldsymbol{x}_k\boldsymbol{x}_k^\top\boldsymbol{v}_A + 2\,\boldsymbol{v}_A^\top\boldsymbol{x}_k\boldsymbol{x}_k^\top\mathbf{K}_k^\top\boldsymbol{v}_B + \boldsymbol{v}_B^\top\mathbf{K}_k\boldsymbol{x}_k\boldsymbol{x}_k^\top\mathbf{K}_k^\top\boldsymbol{v}_B + \sigma_u^2\|\boldsymbol{v}_B\|_2^2.$$

Write $a = \boldsymbol{x}_k^\top\boldsymbol{v}_A$ and $b = \boldsymbol{x}_k^\top\mathbf{K}_k^\top\boldsymbol{v}_B$. The three state-driven terms equal a^2, $2ab$, and b^2 respectively, so they sum to $(a+b)^2$. This results in

$$\boldsymbol{v}^\top\mathbb{E}[\boldsymbol{z}_k\boldsymbol{z}_k^\top \mid \boldsymbol{x}_k]\boldsymbol{v} = \big((\boldsymbol{v}_A + \mathbf{K}_k^\top\boldsymbol{v}_B)^\top\boldsymbol{x}_k\big)^2 + \sigma_u^2\|\boldsymbol{v}_B\|_2^2.$$

Taking expectations over $\boldsymbol{x}_k$ and averaging over k gives (4.22).

The isotropic diffusion makes the state second moments non-degenerate at every step. From the closed-loop recursion $\boldsymbol{x}_{k+1} = (\mathbf{I} + \Delta t\,\mathbf{A}_{\mathrm{cl},k})\boldsymbol{x}_k + \Delta t\,\mathbf{B}_\star\boldsymbol{\eta}_k + \sigma\sqrt{\Delta t}\,\boldsymbol{w}_k$, with $\mathbf{A}_{\mathrm{cl},k} = \mathbf{A}_\star + \mathbf{B}_\star\mathbf{K}_k$ and the increments $\boldsymbol{\eta}_k, \boldsymbol{w}_k$ fresh and independent of $\boldsymbol{x}_k$,

$$\boldsymbol{\Sigma}_{x,k+1} = (\mathbf{I} + \Delta t\,\mathbf{A}_{\mathrm{cl},k})\,\boldsymbol{\Sigma}_{x,k}\,(\mathbf{I} + \Delta t\,\mathbf{A}_{\mathrm{cl},k})^\top + \Delta t^2\sigma_u^2\,\mathbf{B}_\star\mathbf{B}_\star^\top + \sigma^2\Delta t\,\mathbf{I}_d \ \succeq\ \sigma^2\Delta t\,\mathbf{I}_d,$$

where $\boldsymbol{\Sigma}_{x,k} = \mathbb{E}[\boldsymbol{x}_k\boldsymbol{x}_k^\top]$. Together with the non-degenerate initial condition we have $\boldsymbol{\Sigma}_{x,k} \succeq \sigma^2\Delta t\mathbf{I}_d$ for every k.

The two terms of (4.22) control complementary directions. The first term is a sum of squares and thus non-negative, giving the floor

$$\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\boldsymbol{v} \geq \sigma_u^2\|\boldsymbol{v}_B\|_2^2, \qquad \text{so} \qquad \|\boldsymbol{v}_B\|_2^2 \leq \frac{1}{\sigma_u^2}\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\boldsymbol{v}.$$

Writing $\boldsymbol{w}_k = \boldsymbol{v}_A + \mathbf{K}_k^\top\boldsymbol{v}_B$ and $\bar{\boldsymbol{w}} = \frac{1}{N}\sum_k \boldsymbol{w}_k = \boldsymbol{v}_A + \bar{\mathbf{K}}^\top\boldsymbol{v}_B$ with $\bar{\mathbf{K}} = \frac{1}{N}\sum_k \mathbf{K}_k$, the per-step curvature $\boldsymbol{\Sigma}_{x,k} \succeq \sigma^2\Delta t\mathbf{I}_d$ and Jensen's inequality $\frac{1}{N}\sum_k \|\boldsymbol{w}_k\|_2^2 \geq \|\bar{\boldsymbol{w}}\|_2^2$ give

$$\sigma^2\Delta t\|\bar{\boldsymbol{w}}\|_2^2 \leq \frac{1}{N}\sum_k \boldsymbol{w}_k^\top\boldsymbol{\Sigma}_{x,k}\boldsymbol{w}_k \leq \boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\boldsymbol{v}, \qquad \text{so} \qquad \|\bar{\boldsymbol{w}}\|_2^2 \leq \tfrac{1}{\sigma^2\Delta t}\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\boldsymbol{v}.$$

Recovering $\boldsymbol{v}_A = \bar{\boldsymbol{w}} - \bar{\mathbf{K}}^\top\boldsymbol{v}_B$ via $\|\boldsymbol{v}_A\|_2^2 \leq 2\|\bar{\boldsymbol{w}}\|_2^2 + 2K_{\max}^2\|\boldsymbol{v}_B\|_2^2$ and adding $\|\boldsymbol{v}_B\|_2^2$,

$$\|\boldsymbol{v}\|_2^2 \leq \Big(\tfrac{2}{\sigma^2\Delta t} + \tfrac{2K_{\max}^2+1}{\sigma_u^2}\Big)\boldsymbol{v}^\top\mathbb{E}[\tfrac{1}{N}\mathbf{Z}^\top\mathbf{Z}]\boldsymbol{v},$$

which yields (4.23) with $\kappa \asymp \min(\sigma^2\Delta t, \sigma_u^2)$, for every $\boldsymbol{v}$. □

The decomposition (4.22) isolates the two sources of curvature. The first term is the curvature supplied by the state trajectory: it acts on the combined direction $\boldsymbol{v}_A + \mathbf{K}(t_k)^\top \boldsymbol{v}_B$ and, thanks to the dynamics' diffusion, is bounded below along the state directions. The second term is the floor $\sigma_u^2 \|\boldsymbol{v}_B\|_2^2$ contributed by the excitation, which is present for any $\sigma_u > 0$ regardless of the gain. Combining the two, the population Gram is positive definite on all of $\mathbb{R}^{d+p}$ with a restricted eigenvalue bounded below by a constant multiple of $\min(\sigma^2 \Delta t, \sigma_u^2)$. This both explains why any positive excitation breaks the trap of Section 3.3.2 and quantifies the curvature that excitation supplies.

Remark. Proposition 4 bounds the *population* Gram $\mathbb{E}[\frac{1}{N}\mathbf{Z}^\top \mathbf{Z}]$, while Assumption 2 concerns the *empirical* Gram $\frac{1}{N}\mathbf{Z}^\top \mathbf{Z}$. The passage from one to the other requires uniform concentration of the empirical Gram over the support cone, which holds with high probability under uniformly bounded fourth moments of the controlled state. We do not pursue this moment bound, and so treat Assumption 2 as a condition ensured by excitation at the population level and verified empirically. We also note that Proposition 4 supplies the control-block curvature only when the excitation is active ($\sigma_u > 0$); under pure certainty-equivalent play the restricted eigenvalue along $\mathbf{B}_\star$ rests on the state trajectory alone, which Section 3.3.2 shows can fail to supply it. The regret guarantee for SAMBL is therefore conditional on Assumption 2 holding empirically.

Assumption 3 divides into a part that follows from the structure of the Riccati equation and a part that must be monitored. Parts (i) and (ii) on the existence, uniqueness, and twice-differentiability of the solution map $\boldsymbol{\Theta} \mapsto \mathbf{P}_{\boldsymbol{\Theta}}$ follow from the implicit function theorem applied to the Riccati equation (3.10), as established by Basei et al. (2022, Lemma 3.1), and hold on a neighbourhood of $\boldsymbol{\Theta}_\star$ under the assumptions on $\mathbf{Q}$ and $\mathbf{R}$. Part (iii) on the uniform boundedness of the second moments of the controlled state requires the closed-loop dynamics under $\mathbf{K}_{\boldsymbol{\Theta}}$ to be stable enough that the state does not diverge in mean square. This is not guaranteed by the structure of the problem: a poor early estimate can induce a destabilising gain. We monitor it empirically in Chapter 5, and note that it is the same boundedness that guarantees the energy bound (4.20) of Proposition 3, tying the two statistical assumptions to a single control-side condition.

This chapter has established the regret guarantees that SAMBL enjoys based on the assumptions in Section 4.1. First, Section 4.2 bounded the parameter estimation error at the sparse rate ds rather than the dense rate $d(d+p)$. Then, Section 4.3 converted that error into per-episode regret through the certainty-mismatch cost, using the finite-horizon optimality of the Riccati gain to obtain a local quadratic bound. Finally, Section 4.4 combined the two into a regret decomposition whose transient cost is governed by the basin-entry episode, yielding the certified basin-entry speedup $(d+p)/(s\log(d+p))$ over dense estimation.Section 4.5 examined when the underlying assumptions hold, isolating the restricted eigenvalue condition and closed-loop stability as the two conditions specific to the adaptive setting. The next chapter turns from theory to experiment, testing whether the predicted advantage materialises and whether the identifiability obstructions of Section 3.3 manifest as the theory anticipates.

Chapter 5

Experiments



The analysis of Chapter 4 makes three predictions about what sparsity buys an agent that controls an unknown, sparse linear system: (1) the sparse estimator should incur a smaller regret growth constant (Theorem 1), (2) it should enter the quadratic basin around the optimum from fewer samples compared to dense estimators (Corollary 1), and (3) the resulting advantage should increase with the dimension, scaling as the sample-complexity ratio $(d+p)/(s\log(d+p))$. This chapter evaluates these theoretical insights empirically. We introduce the experimental setup in Section 5.1. We then report the results for synthetic sparse systems in Section 5.2 for varying dimensionality and sparsity. Thereafter, we turn to two real-world systems in Section 5.3 where sparsity is given by the underlying physics: a scalable spring-mass chain and the IEEE 39-bus power grid. Section 5.4 then discusses the central insights.

5.1 Experimental Setup

All our experiments were performed on AMD EPYC 7642 compute nodes with 48 physical cores (96 threads) and roughly 400 GB of RAM. They are reproducible by running the open-sourced code at `https://github.com/nkirschi/SAMBL`.

5.1.1 Agents

We benchmark our method against a dense counterpart and the theoretically optimal policy. To this end, every experiment compares the same five agents, summarised in Table 2. An *oracle* is given the true parameters $\mathbf{\Theta}_\star$ and plays the corresponding optimal time-varying gain $\mathbf{K}_\star$, achieving the optimal cost $J_\star$. Therefore, the oracle has zero regret by construction and is the ground truth against which we measure regret for all other agents. The remaining agents form a 2×2 design over two independent axes.

- The first axis is the estimator: A *sparse* agent uses our Algorithm 1 with the regularisation schedule λ_m. A *dense* agent replaces LASSO estimation with the natural baseline, ordinary least squares (OLS). To ensure OLS has a unique solution, we apply a negligible ℓ_2 penalty of 10^{-6}, giving OLS the benefit of the doubt.
- The second axis is exploration: a *greedy* agent plays pure certainty equivalence, executing the gain computed from its current estimate ($\sigma_u = 0$). In contrast, an *excited* agent enables the optional excitation of Section 3.2.4, adding the isotropic exploration signal $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I})$ to the control.

	dense (baseline)	*sparse* (ours)
greedy	OLS	LASSO
excited	OLS + control noise	LASSO + control noise
oracle	knows true parameters	

Table 2: The five agents. The table shows the four learning agents as a 2×2 design over the estimator (columns) and exploration (rows). The fifth agent, the oracle, knows the true $\boldsymbol{\Theta}_\star$ and plays optimally.

The four combinations (*dense-greedy*, *dense-excited*, *sparse-greedy*, *sparse-excited*) isolate the effect of the ℓ_1 penalty and of deliberate excitation separately and in combination. In this nomenclature, *sparse-greedy* is exactly SAMBL (Algorithm 1), and the other three are variants of it. Because the dense and sparse agents share the same data, planner, and noise, and differ only in the estimator, any gap between them is attributable to the sparsity-exploiting penalty alone.

5.1.2 Our LASSO Estimator

Our estimator solves the d independent row-wise LASSO problems of (3.8). Each is a quadratic program in $\boldsymbol{\theta} \in \mathbb{R}^{d+p}$ that depends on the data only through the Gram matrix $\mathbf{Z}^\top\mathbf{Z}$ and the cross term $\mathbf{Z}^\top\mathbf{Y}_{:,i}$. Therefore, we do not keep raw trajectories but accumulate $\mathbf{Z}^\top\mathbf{Z}$ and $\mathbf{Z}^\top\mathbf{Y}$ across episodes, where a new episode contributes only its own block. The per-episode solve cost is therefore independent of how much data has been collected, which matters because all data is retained and $N_m = mH$ grows linearly in the episode count. Each row is minimised by cyclic coordinate descent (Wu & Lange, 2008) on the accumulated Gram matrix, warm started from the previous episode's coefficients. Warm starting is effective because consecutive estimates change little once the support has stabilised. The dense agent shares the same incremental-Gram machinery but solves the standard least-squares normal equations in closed form, with a negligible 10^{-6} diagonal shift that guarantees invertibility before the design reaches full rank.

5.1.3 The Riccati Planner

Given a parameter estimate, the planner of Section 3.2.3 maps it to a feedback policy by solving the differential Riccati equation (3.10) over the time horizon. For computational efficiency, we solve it in closed form with the classical Hamiltonian matrix-exponential scheme based on Vaughan (1969) and Davison & Maki (1973). We describe it in detail in Section 2.3.3. The relevant consideration for experiment execution is that $2d \times 2d$ matrix exponential and H bounded $d \times d$ Möbius steps produce $\mathbf{P}$ at the control grid points $\tau_j = j\,\Delta t$ (where the gain is used) exactly up to round-off and at a cost $\mathcal{O}(Hd^3)$ independent of the spectrum of the estimate. While established solvers like RK45 also pay $\mathcal{O}(d^3)$ per step, their adaptive step count heavily depends on the spectrum.

5.1.4 Matched Randomness

Regret is heavy-tailed: an unlucky early estimate can briefly destabilise a system and dominate the average. To separate the effect of the estimator from this noise, all agents within a seed are driven by identical randomness. They share the same process noise realisation, the same initial states, and the same excitation draws. The only thing that differs between two agents in a given seed is their estimate and the control it induces. This is a matched-pairs design that cancels the dominant source of variance between agents and allows a modest number of seeds to expose the true differences between those agents. For all our metrics, we report the median over 20 seeds with the interquartile range as a spread. We do not report mean and confidence intervals, because the regret distribution is skewed and the median is the more stable summary.

5.1.5 Evaluation Metrics

The primary metric is regret, the excess cost an agent pays for not knowing $\mathbf{\Theta}_\star$, defined per episode as $r_m = J(\boldsymbol{\pi}_{\hat{\mathbf{\Theta}}_{m-1}}) - J_\star$ and cumulatively as $R_M = \sum_{m=1}^{M} r_m$ (cp. (3.4)). To evaluate the relative advantage of using a sparse agent, we report the dense-to-sparse cumulative regret ratio. Regret is what the theory bounds and is the quantity of ultimate interest, but on its own it does not explain *why* one agent beats another, so we track four additional diagnostics.

Parameter error. The relative Frobenius distance of the parameter estimate to the true parameters, reported both jointly as $\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}_\star\|_F / \|\mathbf{\Theta}_\star\|_F$ and separately for the drift block as $\|\hat{\mathbf{A}} - \mathbf{A}_\star\|_F / \|\mathbf{A}_\star\|_F$ and the control block as $\|\hat{\mathbf{B}} - \mathbf{B}_\star\|_F / \|\mathbf{B}_\star\|_F$. The two behave very differently, and the split exposes the asymmetry between identifying $\mathbf{A}_\star$ and $\mathbf{B}_\star$.

Support recovery. The precision, recall, and F_1 score of the estimated nonzero pattern against the truth, where an estimated entry $\hat{\theta}_{ij}$ counts as nonzero when $|\hat{\theta}_{ij}| > 0.05$. Because we ensure that the true nonzero entries all exceed 0.1 (Section 5.2.1), this threshold separates genuine structure from estimation noise. These support metrics measure whether the LASSO uncovers the right structure and not merely achieves a small error.

Restricted eigenvalue. Verifying the exact restricted eigenvalue condition (4.2) would require an infeasible minimisation over a cone set. Instead, we track a proxy: the minimum over all rows i of the smallest eigenvalue of the *support-restricted* Gram matrix $\mathbf{Z}_{\mathcal{S}_i}^\top \mathbf{Z}_{\mathcal{S}_i}$. This is an upper bound on the restricted-eigenvalue constant κ that controls the estimation rate (Proposition 1) and therefore its being bounded away from zero constitutes a necessary condition. The smallest eigenvalue of the *full* Gram matrix would instead be a *lower* bound on κ, but a trivial one: it is essentially zero whenever the accumulated design is rank-deficient, in particular before $N_m = mH$ exceeds $d + p$, and whenever two regressors are near-collinear. Restricting to the support discards these irrelevant near-null directions.

Basin entry. We identify the first episode at which the estimate is accurate enough for the cost suboptimality to be quadratic in the parameter error, the regime in which Proposition 2 applies and the logarithmic tail of Theorem 1 sets in. As the true basin radius ε_0 is unknown, we use a per-seed proxy: the threshold is the *larger* of the two paired agents' final relative parameter errors $\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}_\star\|_F / \|\mathbf{\Theta}_\star\|_F$ and we record, for each agent, the first episode at which its error falls below this threshold. The ratio of the two episodes is the empirical basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$, the realised counterpart of the certified speedup of Corollary 1.

5.2 Synthetic Systems

5.2.1 System Sampling

The synthetic ensemble draws each system at random subject to three conditions. First, *exact sparsity*: every row of $\mathbf{A}_\star$ has precisely s_A nonzeros and every row of $\mathbf{B}_\star$ precisely s_B nonzeros, placed uniformly at random. In this vein, the row support sizes are controlled rather than incidental. Second, *full control coverage*: we reject any $\mathbf{B}_\star$ with an all-zero column, since a zero column is a control input disconnected from the dynamics that would also make the self-exploration matrix $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star$ singular and violate the identifiability condition of Section 3.3. Third, *stabilisability*, checked via the Hautus test as described in Lemma B.2. This is less an algorithmic assumption than a well-posedness requirement, and is exactly the condition under which the associated algebraic Riccati equation admits a solution (Kucera, 1972, Theorem 2). Notably we do *not* require open-loop stability, so a sampled system may be unstable and a poor early estimate may transiently destabilise the closed loop.

Nonzero entries are drawn from $\pm\,\text{Uniform}[0.1, 1.9]$, symmetric about zero so that the sign carries no prior information, with a gap around zero. The gap creates a clear signal-to-noise separation, i.e., no entry is so small as to be unidentifiable. While this makes the recovery problem somewhat easier than a general system with arbitrarily small couplings, it eliminates a source of high seed variance and allows us to focus on the effect of sparsity, which is what we care about in this work. For a similar reason, we scale each control column to unit norm, giving every input *equal authority* ($\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star \approx \mathbf{I}$). This removes the following sampling artefact: a raw random sparse $\mathbf{B}_\star$ has wildly varying column norms, so its self-exploration matrix grows ill-conditioned as p increases (Figure 5b), which would confound the results with a control identifiability effect rather than the sparsity effect we study as dimensionality grows. The sparsity pattern of $\mathbf{B}_\star$ and the drift block $\mathbf{A}_\star$ are untouched, and the only effect is that $\mathbf{B}_\star$ is no longer *dimension-dependently* harder. Unless stated otherwise, experiments use the canonical configuration of Table 3, varying only the dimensions d, p while holding everything else fixed.

Table 3: Canonical configuration for the synthetic systems. The dimensions d and p are varied while all other quantities are held fixed.

Quantity	**Symbol**	**Value**
dimension	d	$\{10, 20, 50, 100, 200, 500\}$
control dimension	p	$d/2$
drift and control support size	$s_A,\ s_B$	$2,\ 1$
process noise	σ	0.3
horizon, step size, steps	$T,\ \Delta t, H$	$1,\ 0.02,\ 50$
initial-state scale	σ_{x_0}	1
excitation scale	σ_u	0.05
episodes, initial exploration episodes	$M,\ m_{\text{explore}}$	$1000,\ 1$
seeds	$-$	20
regularisation, failure level	c_λ, δ	$0.02,\ 0.05$
$\mathbf{A}_\star$ entry magnitude	$[a_{\min},\ a_{\max}]$	$[0.1,\ 1.9]$
$\mathbf{B}_\star$ entry magnitude	$[b_{\min},\ b_{\max}]$	$[0.1,\ 1.9]$

5.2.2 Initialisation

We initialise both parameter blocks as scaled (truncated) identity matrices, $\hat{\mathbf{A}}_0 = -a_0\mathbf{I}_d$ and $\hat{\mathbf{B}}_0 = b_0\mathbf{I}_{d\times p}$. A diagonal prior is full rank and carries no directional or dimensional bias. The negative sign on $\hat{\mathbf{A}}_0$ makes the initial drift estimate mildly contractive, so that the very first planning problem, solved before any data is seen, is well-posed. It remains to set the scales a_0 and b_0. Since nonzero entries are drawn symmetrically around zero, we have $\mathbb{E}[\mathbf{\Theta}_\star] = 0$ and the first moment gives no information about scale. We therefore match the *expected squared row norm* of each initial estimate to that of the corresponding true block.

Each nonzero entry c is drawn from $\pm\,\text{Uniform}[c_{\min}, c_{\max}]$, giving a root mean square of

$$\text{RMS}(c) \coloneqq \sqrt{\mathbb{E}[c^2]} = \sqrt{\frac{1}{c_{\max} - c_{\min}} \int_{c_{\min}}^{c_{\max}} t^2 \mathrm{d}t} = \sqrt{\frac{c_{\min}^2 + c_{\min}c_{\max} + c_{\max}^2}{3}}. \tag{5.1}$$

For the default configuration ($a_{\min} = b_{\min} = 0.1$, $a_{\max} = b_{\max} = 1.9$) both evaluate to $\text{RMS}(a) = \text{RMS}(b) \approx 1.13$. Row i of $\mathbf{A}_\star$ and $\mathbf{B}_\star$ has s_A and s_B independent nonzero entries a and b, respectively, so $\mathbb{E}[\|(\mathbf{A}_\star)_{i,:}\|_2^2] = s_A\,\mathbb{E}[a^2]$ and $\mathbb{E}[\|(\mathbf{B}_\star)_{i,:}\|_2^2] = s_B\,\mathbb{E}[b^2]$.

For $\hat{\mathbf{A}}_0 = -a_0\mathbf{I}_d$, row i has squared norm a_0^2. For $\hat{\mathbf{B}}_0 = b_0\mathbf{I}_{d\times p}$, row $i \le p$ has squared norm b_0^2. Setting each equal to the corresponding expectation:

$$a_0 = \sqrt{s_A} \cdot \text{RMS}(a) \approx 1.13\sqrt{s_A}, \qquad b_0 = \sqrt{s_B} \cdot \text{RMS}(b) \approx 1.13\sqrt{s_B}. \tag{5.2}$$

We stress that under the canonical protocol ($m_{\text{explore}} = 1$) the precise choice of a_0 and b_0 is fully irrelevant: the first episode plays pure random exploration regardless of the current estimate, and the estimator is refit from that data before certainty equivalence ever consults $\hat{\mathbf{\Theta}}_0$, so the prior is overwritten before it influences a single control. The initialisation becomes consequential only when the exploration episode is disabled ($m_{\text{explore}} = 0$, an ablation in Section 5.2.5). In this case, the very first policy is planned directly from $\hat{\mathbf{\Theta}}_0$, so a poorly scaled prior can seed the estimate far from the truth and delay basin entry. The moment-matched scales in (5.2) are thus a safeguard for the greedy-from-the-start ablation, not a factor in the main results.

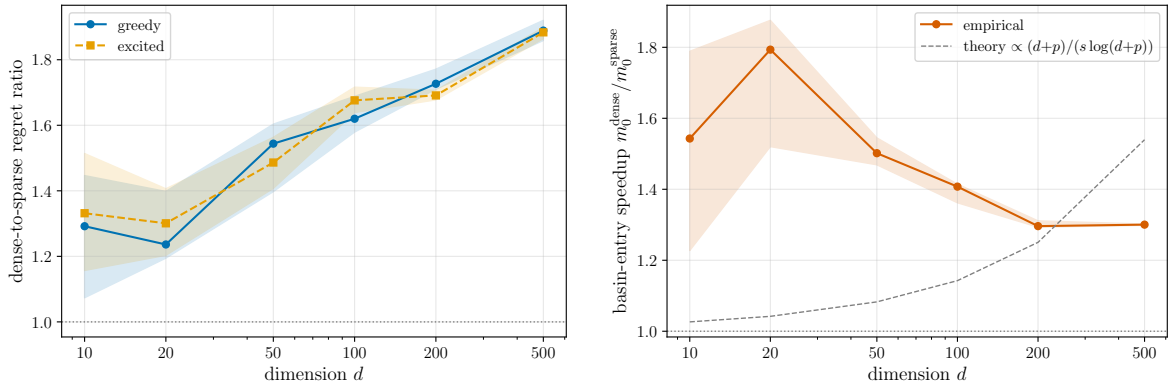
5.2.3 Regularisation

The regularisation strength follows the theoretical schedule λ_m of Lemma 1, which gives the correct $\sqrt{\log / N}$ decay. The worst-case constant c_λ it prescribes is conservative and in practice leads to immediate shrinkage of the control block estimate to $\hat{\mathbf{B}} = \mathbf{0}$, so we fix a calibrated value $c_\lambda = 0.02$ across every experiment. The calibration itself is deferred to the ablations in Section 5.2.5.

5.2.4 Main Results

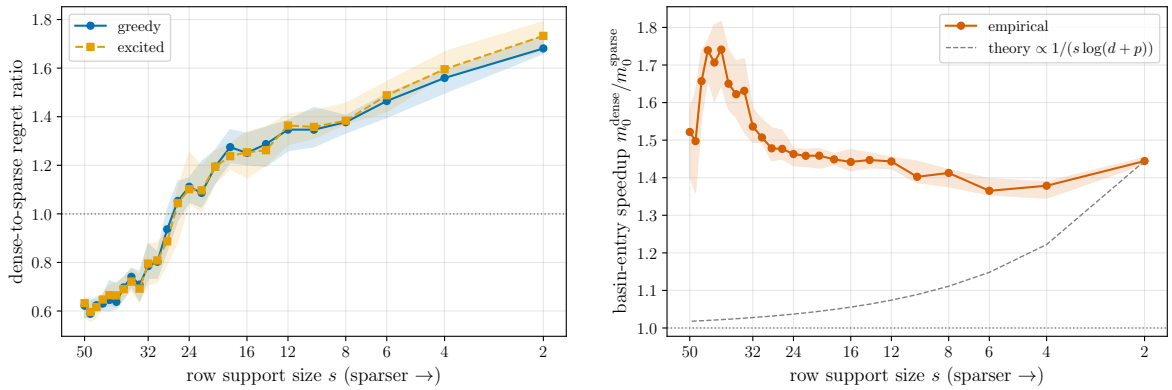
The regret advantage is significant and increases with dimension. In Figure 2 we report the results for the dense and sparse agents for varying the system dimensionality. The dense-to-sparse regret ratio is consistently above one at every dimension (Figure 2a), rising from about 1.3 at $d = 10$ to about 1.9 at $d = 500$ for both greedy and excited agents. This shows sparse estimation affords a significant regret advantage that grows with dimension. Figure 2b shows the empirical basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$, the realised counterpart of the certified speedup (4.18): it stays consistently above 1 for all dimensions. Thus, the sparse estimator reaches the quadratic basin from fewer samples at every d, as predicted by our analysis. However, it does not increase with dimension, contrary to what the theory suggests. We hypothesise that this is likely due to the dense estimator performing far above its worst-case bound in high dimensions on our well-behaved systems. The advantage of using a sparse estimator thus increases with dimension in the regret-ratio sense, but not in the relative basin-entry-time sense.

The regret advantage grows for sparser systems. In Figure 3, we report the results for the dense and sparse agents for varying the system sparsity. For systems with $d = 100$ state variables and $p = 50$ control variables, we vary the joint row support size $s = s_A + s_B$ from 50 down to 2 in steps of two (Figure 3), so that s maps to a clean row density $s/(d+p) = s/150$. The sparse estimator outperforms the dense one in terms of regret with largest benefits in the deeply sparse regime, with a dense-to-sparse regret ratio around 1.6 for the sparsest rows ($s \leq 6$, i.e., at most 4% density). It then decreases steadily as the rows fill in to about 1.2 by $s = 20$ (13% density), and eventually *reverses*: the ratio passes through parity at $s \approx 28$ (19% density) and falls to about 0.6



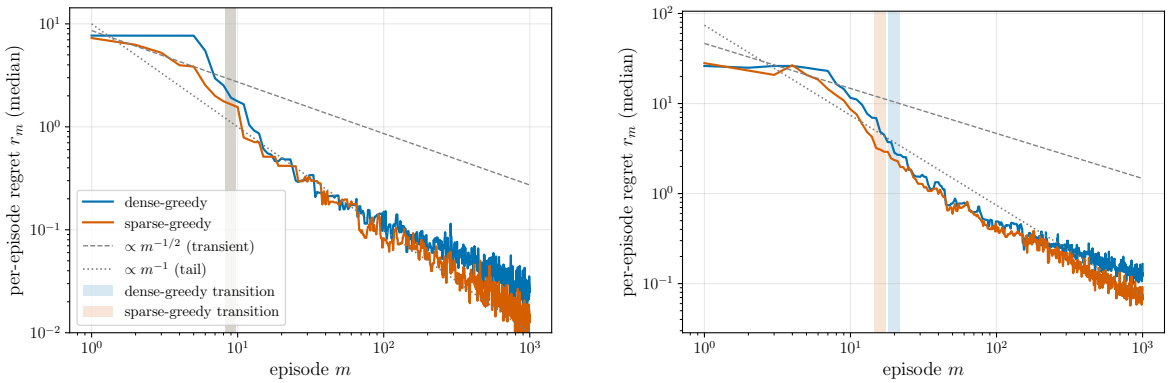
(a) Dense-to-sparse regret ratio vs. d (b) Basin-entry speedup ratio vs. d

Figure 2: Dimensional scaling of performance (medians across 20 seeds, IQR shaded). **(a)** Final dense-to-sparse regret ratio for the greedy and excited agent pairs, versus dimension. It exceeds one (grey dotted) at every dimension and increases with d. **(b)** The basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$ against the theory-predicted shape (dashed grey, fitted constant) and one (grey dotted).

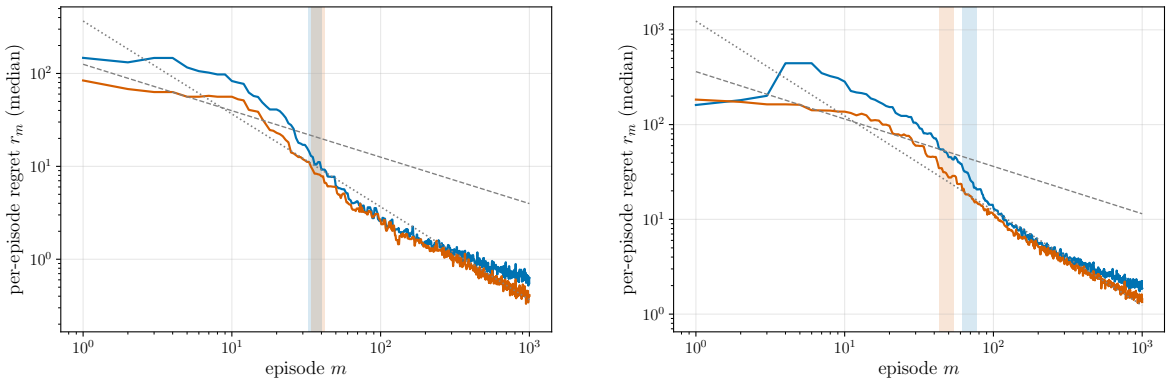


(a) Dense-to-sparse regret ratio vs. s (b) Basin-entry speedup ratio vs. s

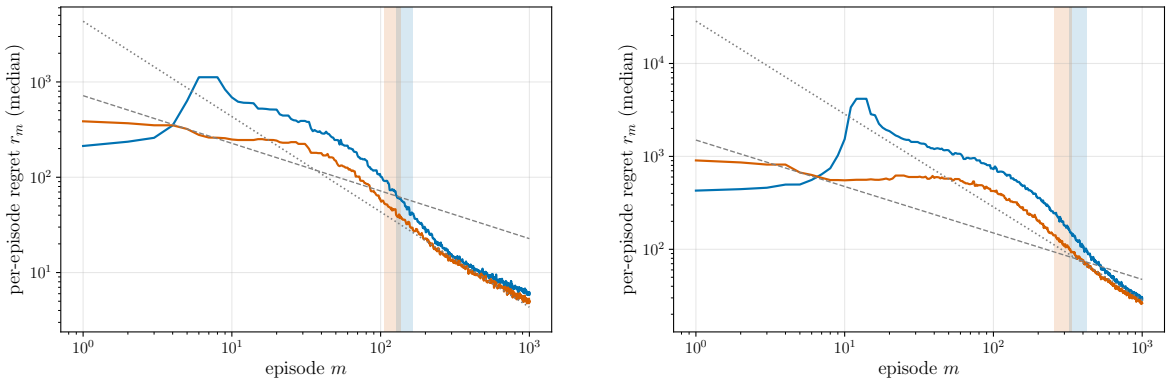
Figure 3: Sparsity scaling of performance at fixed $d = 100$, $p = 50$ with varying s (medians across 20 seeds, IQR shaded). The horizontal axis is reversed so that the system gets sparser to the right. **(a)** Final dense-to-sparse regret ratio for the greedy and excited agent pairs, as a function of the row support size: above one for sparse rows, but crossing parity near $s \approx 28$ (19% density) and dropping to about 0.6 by $s = 50$, where the dense agent clearly leads. **(b)** The basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$ against the theory-predicted shape (dashed grey, fitted constant) and one (grey dotted). It stays around 1.4–1.7 across the whole range, *not* mirroring the regret reversal, even though the $1/s$ shape of the speedup ratio predicts a decay (dashed).



(a) $d = 10$ (b) $d = 20$



(c) $d = 50$ (d) $d = 100$



(e) $d = 200$ (f) $d = 500$

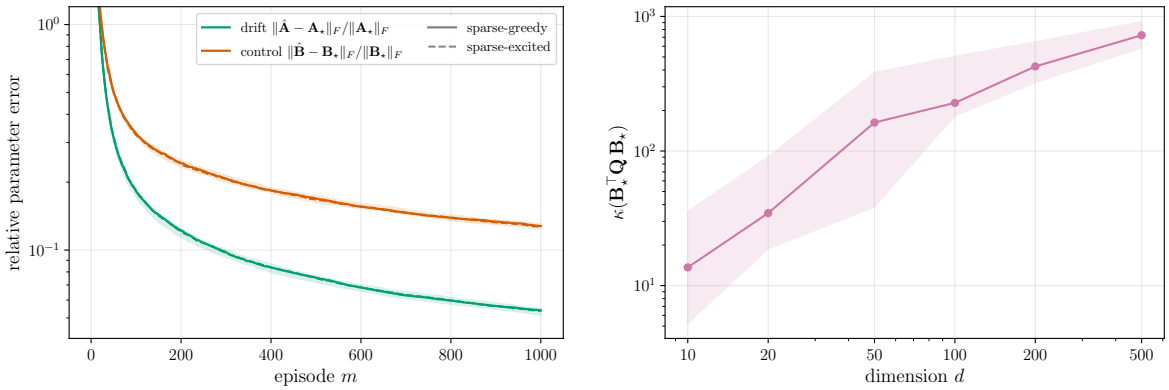
Figure 4: Per-episode regret of the greedy agents (median, smoothed) with reference slopes, for every dimension of the synthetic sweep. The transient decays as $m^{-1/2}$ and the tail as m^{-1} (cp. Theorem 1(ii)). The knee marks entry into the quadratic basin. The shaded bands mark each agent's basin-entry, determined by the steepest descent of its log-log curve. The legend in panel (a) applies to all panels.

by $s = 50$ (33% density), where the dense agent is the better one. This is the decay the worst-case speedup $(d+p)/(s\log(d+p))$ predicts, carried past its useful range: once the rows are no longer sparse, the LASSO over-shrinks genuine couplings, and its structural bias turns from an asset into a liability. The crossover is a clean confirmation that the advantage is *structure*-dependent: sparsity pays precisely when the dynamics are sparse. The basin-entry speedup proxy, by contrast, does not exhibit this reversal (Figure 3b): it stays above one across the whole range, hovering around 1.4–1.7. The sparse agent thus still reaches the basin from fewer samples even where its *final* regret is worse.

The regret changes regimes and basin entry takes place. The observed absence of an increase in the speedup factor with dimension might suggest that within the horizon of $M = 1000$ episodes the agents never enter the quadratic basin, and that the regret is a transient effect. Figure 4 shows this is not the case by decomposing the empirically observed per-episode regret over the episodes. The per-episode regret indeed splits into the two regimes predicted by Theorem 1: an initial transient decaying roughly as $m^{-1/2}$ and a tail decaying as m^{-1}, the quadratic-basin regime. Basin entry takes place at the transition, which the sparse agents reaches slightly earlier than the dense agent. The effect is real but modest in this regime. The decomposition plots visually confirm the previous notion that the earlier basin entry is not accelerated in higher dimensions.

The sparse estimator identifies the drift block better than the control block. Figure 5a reports the errors of the estimated system parameters $\mathbf{A}_\star$ and $\mathbf{B}_\star$ over the episodes. While the drift block $\mathbf{A}_\star$ is recovered to high accuracy within a few dozen episodes, the control block $\mathbf{B}_\star$ plateaus about 2.5 times higher. We hypothesize that this is because the state is persistently excited by the process noise, and thus identifying $\mathbf{A}_\star$ is easier from abundant data. At the same time, $\mathbf{B}_\star$ is informed only through the control's effect on the next state, which the self-exploration condition $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star \succ 0$ guarantees is present but may be a rather weak signal. Due to this asymmetry, the sparse advantage is moderate, as sparsity sharpens the part that is already easy. Without the equal-authority scaling of Section 5.2.1 it would get increasingly harder with dimension: Figure 5b shows the conditioning of $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star$ rising sharply with d. The $\mathbf{B}_\star$ estimation difficulty is what the initial exploration episode addresses, by supplying the restricted-eigenvalue curvature that greedy self-exploration lacks (Section 5.2.5). With that episode in place, the additional σ_u of the excited agents is redundant: the excited and greedy curves in Figure 5a nearly coincide.

A single high-dimensional point under the microscope. The sweep views above compress each dimension into one number per agent, so we also inspect one point in full per-episode detail. Figure 6 does so at the largest dimension $d = 500$, showing all four per-episode diagnostics side by side. Cumulative regret (Figure 6a) separates the estimator groups within the first tens of episodes and the gap grows monotonically to the roughly 1.9 final ratio of Figure 2a, with non-overlapping interquartile bands. The restricted Gram eigenvalue (Figure 6b) suggests that the design stays identifiable: seeded



(a) Drift versus control recovery

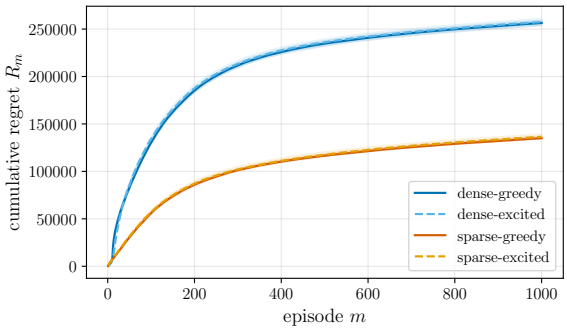
(b) Effect of no column normalisation.

Figure 5: Asymmetry between drift and control blocks (medians across 20 seeds, IQR shaded). **(a)** Relative error of the drift and control blocks ($d = 50$): $\mathbf{A}_\star$ is recovered to high accuracy while $\mathbf{B}_\star$ plateaus about 2.5 times higher. **(b)** A *non*-column-normalised random-sparse $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star$ becomes worse conditioned as d grows. The experiments scale control inputs to equal authority to remove this artefact, so $\mathbf{B}_\star$ is uniformly rather than dimension-dependently hard.

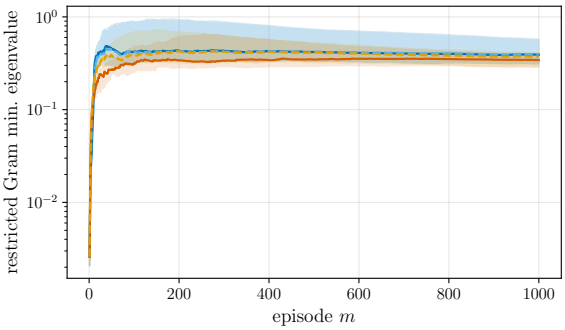
by the exploration episode, the curvature stays comfortably bounded away from zero over the entire horizon instead of collapsing – the failure mode dissected in Section 5.2.5. The relative parameter error (Figures 6c and 6d) shows where this regret gap comes from: the sparse estimator descends through the transient faster and settles at a lower plateau. Finally, the support recovery F_1 score (Figures 6e and 6f) confirms that the sparse advantage is structural. The LASSO recovers the true support pattern more accurately than thresholded least squares throughout. In both parameter error and support recovery, the accuracy advantage concentrates in the drift block while the control block suffers from false positive entries, echoing the asymmetry of Figure 5.

5.2.5 Ablations & Robustness

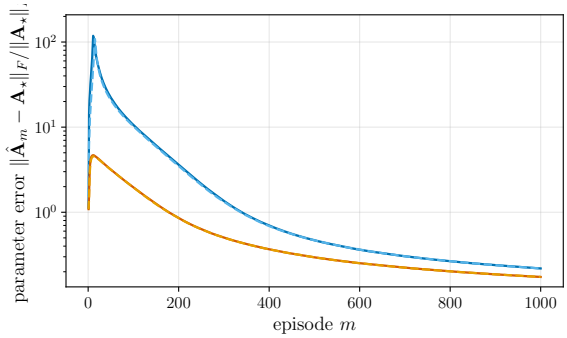
The exploration episode is vital in high dimensions. The results above use the canonical protocol with a single initial exploration episode ($m_{\text{explore}} = 1$). Without it, i.e. pure greedy certainty equivalence ($m_{\text{explore}} = 0$), the sparse advantage *reverses* at high dimension. Figure 7a plots the basin-entry speedup for both protocols: they agree in the median up to $d = 100$, where the first seeds start to collapse below 1, which means the dense agent enters the basin *first* in high dimensions. In contrast, just one initial exploration episode holds the speedup at roughly 1.3 throughout. The cause is a control block identifiability failure (Section 3.3): under greedy play $\boldsymbol{u}(t) = \mathbf{K}(t)\boldsymbol{x}(t)$ is close to collinear with the state, so the control block is stimulated only incidentally, and that self-exploration extinguishes once the closed loop stabilises. Figure 7b shows the restricted eigenvalue proxy metric collapsing for greedy at $d \geq 200$, while the exploration



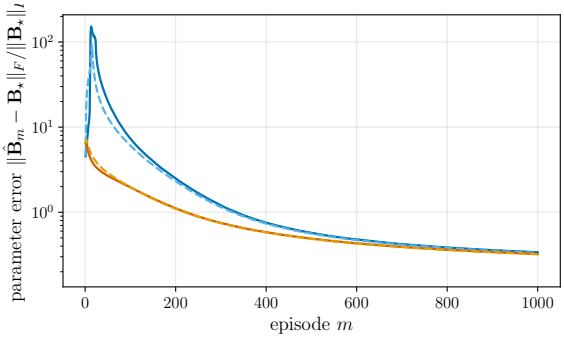
(a) Cumulative regret



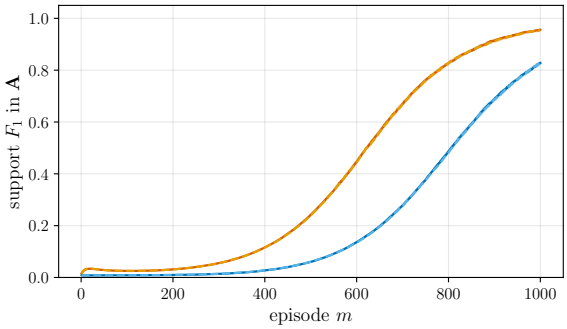
(b) Restricted Gram minimum eigenvalue



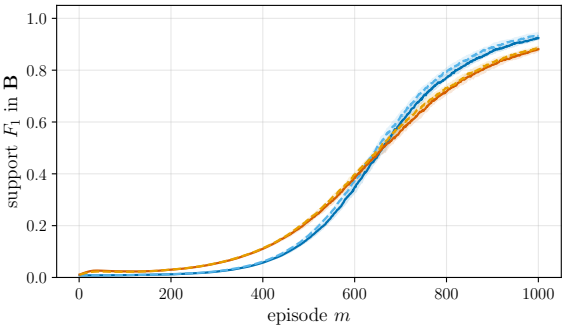
(c) Relative parameter error in $\mathbf{A}_\star$



(d) Relative parameter error in $\mathbf{B}_\star$

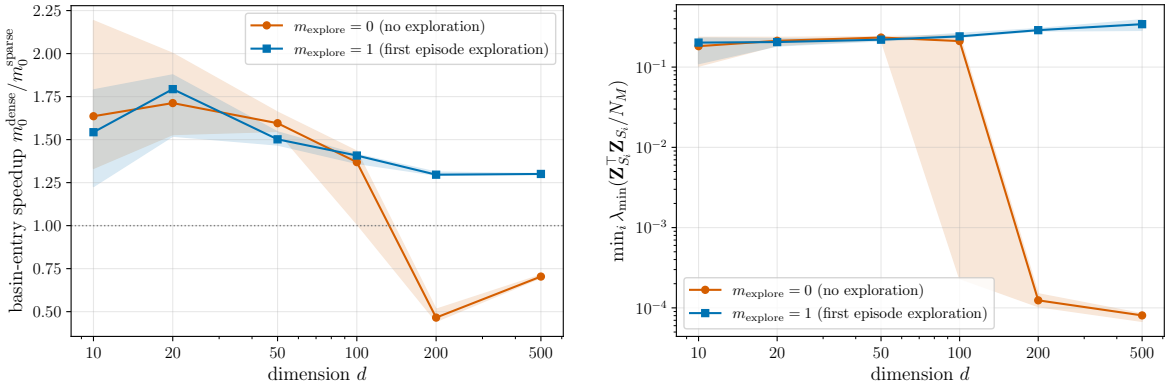


(e) Support F_1 in $\mathbf{A}_\star$



(f) Support F_1 in $\mathbf{B}_\star$

Figure 6: Per-episode diagnostics at the largest synthetic dimension ($d = 500$, $p = 250$). All show the median and the IQR shaded, and the legend in panel (a) applies to all. **(a)** Cumulative regret separates the sparse from the dense agents early, with non-overlapping bands. **(b)** The minimum eigenvalue of the (restricted) empirical Gram matrix is comfortable bounded away from zero after the exploration episode, suggesting identifiability throughout. **(c)(d)** Relative parameter error of the drift and control blocks: the sparse estimator peaks lower and descends earlier in both, but its final advantage concentrates in the drift block. **(e)(f)** LASSO recovers the drift support almost exactly, while false positive entries in the control block cap both methods.



(a) Basin-entry speedup ratio vs. d (b) Restricted eigenvalue proxy vs. d

Figure 7: Greedy certainty equivalence ($m_{\text{explore}} = 0$) versus one initial exploration episode ($m_{\text{explore}} = 1$), across dimension (medians across 20 seeds, IQR shaded). **(a)** The basin-entry speedup dives below 1 without initial exploration at $d \geq 200$, i.e., the sparse advantage reverses. **(b)** The control-block restricted-Gram minimum eigenvalue collapses without and stays away from zero with the initial exploration.

episode keeps it at the same order of magnitude as for lower dimensions. A single episode of state-independent exploration, played before the estimate has stabilised, supplies the stimulation that greedy play never accumulates, at negligible regret cost. There are three further ways that avoid this collapse: using (1) the excited agent, (2) cheaper control cost, or (3) a smaller c_λ. All of them keep the estimate lively enough, confirming the phenomenon is an excitation deficit rather than an estimator flaw. However, we found the initial exploration episode to be the most robust guarantor of identifiability, and it is thus the one we adopt in the canonical protocol.

The regularisation schedule scaler must not be too large. We fix the LASSO schedule scaling factor $c_\lambda = 0.02$ throughout. Figure 8 calibrates it once on the synthetic ensemble. The median regret (Figure 8a) and the parameter error (Figure 8b) both keep improving as c_λ rises to about 0.05, the lowest point on the grid at every tested dimension, before both blow up at $c_\lambda = 0.1$ where the control block is over-shrunk. The lower median at $c_\lambda = 0.05$ is problematic, however, as it comes with a heavy tail. Already there a handful of seeds destabilise, the worst over fifty times the median, whereas $c_\lambda = 0.02$ yields no such failures at any dimension and a tight interquartile spread. We therefore fix the more conservative $c_\lambda = 0.02$, trading a modestly higher median regret for robustness across seeds. An even smaller c_λ still is, as noted, one of the levers that fixes the high-dimensional collapse of the previous ablation.

On top of the exploration episode, excitation does not pay. Figure 9 sweeps the excitation scale σ_u of the sparse-excited agent, with $\sigma_u = 0$ recovering sparse-greedy. Across $d \in \{10, 20, 50\}$ the final regret is already minimal at $\sigma_u = 0$ and only rises with σ_u: once the exploration episode has broken the identifiability trap, the extra

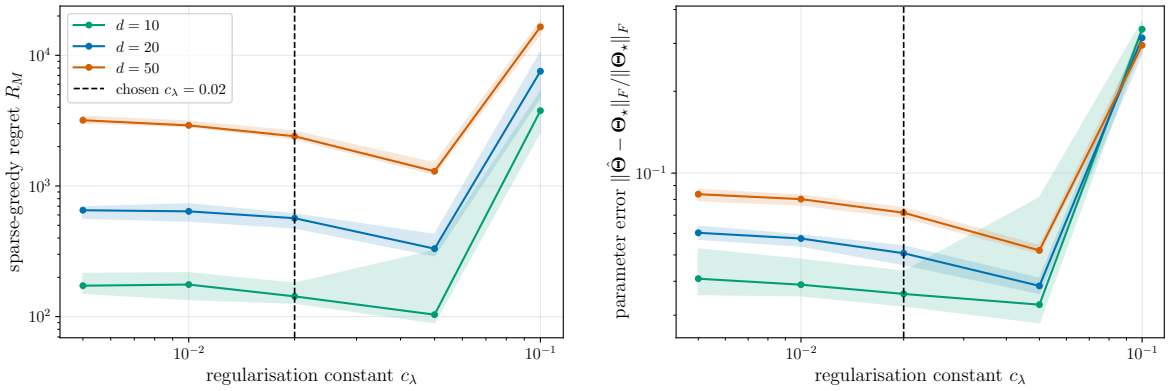
injected energy costs more than the marginal $\mathbf{B}_\star$-recovery it buys, so deliberate excitation never helps and large σ_u is several-fold worse (a factor of six at $d = 10$, $\sigma_u = 0.4$). Excitation is thus redundant on top of the kick, in line with the result that naive exploration is optimal for online LQR (Simchowitz & Foster, 2020). The canonical $\sigma_u = 0.05$ used for the excited agents elsewhere is the smallest value that leaves their regret indistinguishable from the greedy agents'.

The advantage is robust to the cost weighting. Figure 10a sweeps the control-cost scale r at $d = 100$, with $q = 1$ fixed as only the ratio q/r matters by scale-invariance. The dense/sparse regret ratio holds at ≈1.6–1.8 across the entire range, for both the greedy and the excited agents, whose curves nearly coincide. Cheaper control ($r < 1$) drives the state harder, while expensive control ($r > 1$) makes the optimal gains timid and would starve the incidental self-excitation of $\mathbf{B}_\star$. It is this regime in which we observed the sparse-greedy agent collapse *without* an exploration episode. With the exploration episode in place, this problem vanishes: Figure 10b shows that even at the largest r the sparse-greedy and sparse-excited agents descend cleanly to the lowest regret. Together with the equal-authority scaling of Section 5.2.1, this leaves the advantage insensitive to the cost geometry.

Discretisation affects only the control block. Our Euler–Maruyama scheme uses a step $\Delta t = 0.02$. Figure 11 varies it from 0.005 to 0.2 (i.e., $H = 200$ down to 5 steps per episode) at $d = 100$. The drift error is Δt-*invariant* across the entire range (Figure 11b): a finer step provides proportionally more but noisier samples ($\bar{\sigma} = \sigma/\sqrt{\Delta t}$) whose aggregate information is Δt-independent, so finer sampling neither helps nor hurts the drift estimate. The control block behaves differently: while first low and flat up to $\Delta t = 0.05$, it degrades at coarser steps, where the few-step episodes leave the exploration episode too little to excite $\mathbf{B}_\star$ (parameter error rising to 0.66 at $H = 5$). The regret tracks this (Figure 11a): nearly constant for $\Delta t \leq 0.1$, it grows significantly at the coarser steps. The canonical $\Delta t = 0.02$ sits in the flat regime, so our results do not hinge on the discretisation.

5.3 Realistic Systems

The synthetic ensemble is convenient but its sparsity is arbitrary, and holding the row support size s fixed while d grows has to be imposed by hand. Physical systems supply genuine, dimension-independent sparsity for free, because locality of interaction is a structural fact rather than a modelling choice. We study two such systems whose dynamics fall squarely in our setting: a coupled spring-mass chain, a controlled and scalable family on which to repeat the dimensional sweep, and the IEEE 39-bus power grid, a single fixed system that serves as an established real-world test. Both turn out to be marginally stable, which removes the heavy-tailed cost of occasional destabilisation and yields a cleaner regret signal than the synthetic ensemble.



(a) Regret (b) Parameter error

Figure 8: Calibration of the regularisation constant c_λ, one curve per dimension (medians across 20 seeds, IQR shaded). **(a)** Final sparse-greedy regret: the median is lowest near $c_\lambda = 0.05$ but heavy-tailed (some seeds destabilise), and $c_\lambda = 0.1$ blows up. **(b)** Parameter error, averaged over the final 20 episodes: likewise lowest near $c_\lambda = 0.05$ and blowing up at $c_\lambda = 0.1$. We adopt the more conservative, seed-robust $c_\lambda = 0.02$.

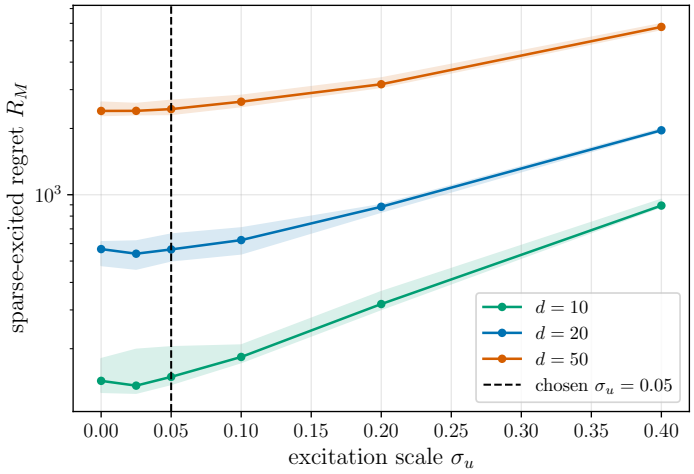
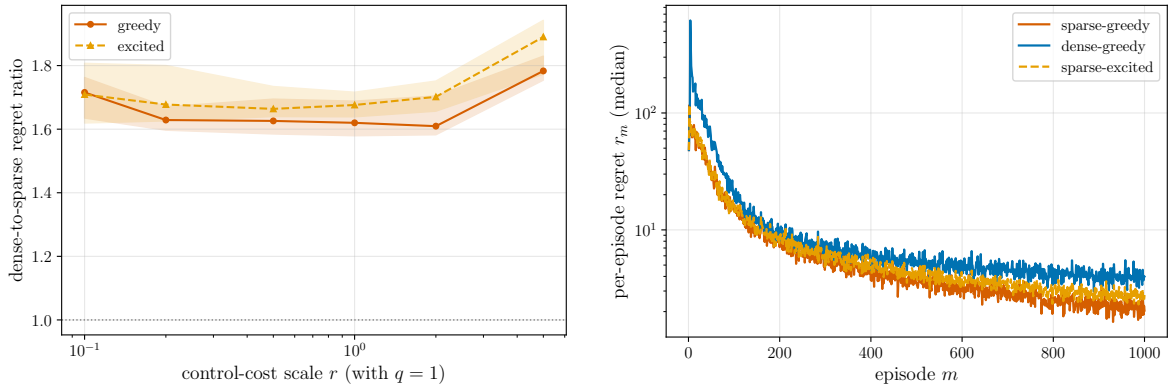


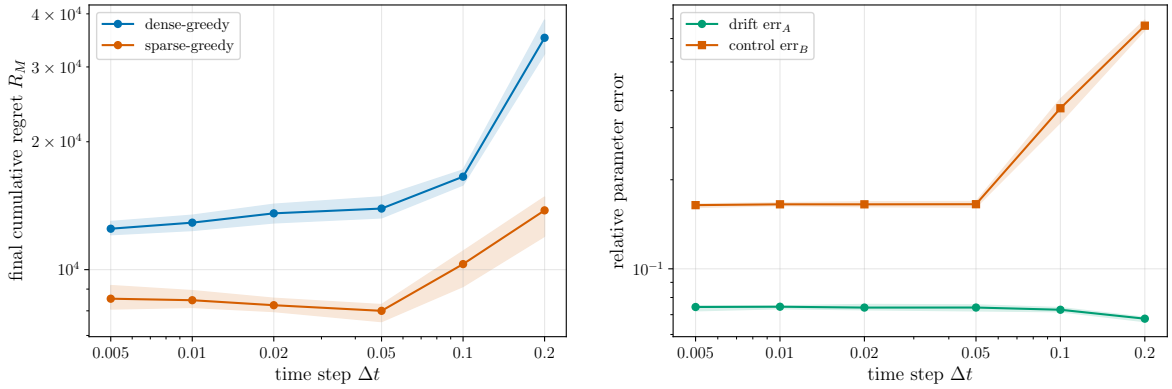
Figure 9: Final regret of the sparse-excited agent versus the excitation scale σ_u (medians across 20 seeds, IQR shaded). The left end $\sigma_u = 0$ corresponds to sparse-greedy. At the canonical cost the curves are flat-to-rising: deliberate excitation does not lower regret and large σ_u is costly. The marked $\sigma_u = 0.05$ is the level used in our studies.



(a) Advantage vs cost

(b) Per-episode regret at $r = 5$

Figure 10: Cost sensitivity at $d = 100$ ($q = 1$, medians across 20 seeds, IQR shaded). **(a)** Dense/sparse final-regret ratio versus the control-cost scale r: both the greedy and excited ratios hold near 1.6–1.8 across the range and nearly coincide. **(b)** Per-episode regret at the largest r: the sparse agents descend cleanly to the lowest regret, while dense-greedy suffers a brief early destabilisation before recovering.



(a) Final regret

(b) Drift vs control error

Figure 11: Sensitivity to the Euler step Δt at $d = 100$ ($H = T/\Delta t$ steps per episode, medians across 20 seeds, IQR shaded). **(a)** Final regret of the dense and sparse greedy agents: stable for $\Delta t \leq 0.1$ and increasing at the coarser steps. **(b)** The sparse agent's drift and control error: the drift error is Δt-invariant (its aggregate information is Δt-independent), while the control block degrades at coarse Δt.

5.3.1 Coupled Spring-Mass Chain

The synthetic systems draw their sparsity pattern at random, which is convenient but leaves the structure physically arbitrary. To evaluate the dimensional scaling of the sparse advantage (Corollary 1) on a system whose sparsity is dictated by physics rather than chosen, we use a chain of n point masses coupled by linear springs to their nearest neighbours and to fixed walls at both ends (Figure 12, left).

Let q_i and $\dot{q}_i$ be the horizontal displacement and velocity* of mass i, with mass m_i and spring constants k_i, where spring k_i joins mass $i-1$ to mass i and the wall springs are k_1 and k_{n+1}. With the wall convention $q_0 = q_{n+1} = 0$, Hooke's law gives the undamped equation of motion

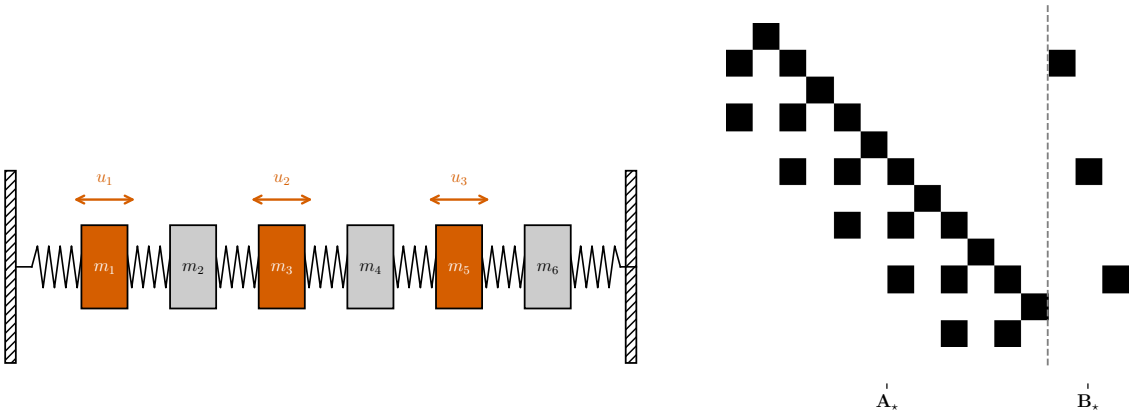
$$m_i \ddot{q}_i = k_i\, q_{i-1} - (k_i + k_{i+1})\, q_i + k_{i+1}\, q_{i+1} + \mathbb{1}_{\{i \text{ actuated}\}}\, u. \tag{5.3}$$

Stacking the state in interleaved order $\boldsymbol{x} = [q_1, \dot{q}_1, \ldots, q_n, \dot{q}_n]^\top \in \mathbb{R}^d$ with $d = 2n$ casts the chain into the LQR form (3.1). The drift $\mathbf{A}_\star$ is banded with row-sparsity fixed independently of n. Each position row reduces to $\frac{\mathrm{d}}{\mathrm{d}t} q_i = \dot{q}_i$, a single unit entry, while each velocity row collects the displacements of the mass and its two neighbours. This amounts to at most three nonzeros per row, two at the chain ends, and (being undamped) no velocity couplings. Each actuator j drives a single mass a_j through one entry $(\mathbf{B}_\star)_{2a_j, j} = 1/m_{a_j}$. The largest joint row support size of $\boldsymbol{\Theta}_\star = [\mathbf{A}_\star \mid \mathbf{B}_\star]$ is thus $s = 4$, attained at an actuated interior velocity row (its own and two neighbour displacements, plus one control) and *constant* as d grows. Figure 12 (right) shows the pattern.

Each seed draws a fresh chain: the masses m_i and the $n+1$ spring constants k_i are sampled independently and uniformly from $[0.5, 1.5]$ and $[0.5, 2]$. Hence, the *structure* is fixed but the *values* vary, yielding an ensemble of heterogeneous chains. We actuate every other mass for $p = \lfloor n/2 \rfloor$ evenly spaced actuators, which means $p/d = 1/4$. Acting on distinct masses also keeps the control block non-degenerate, $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star \succ 0$, as required for self-exploration (Section 3.3). The chosen ranges keep the stiffness-to-mass ratios k_i/m_i bounded away from zero, so the nonzero entries of $\boldsymbol{\Theta}_\star$ obey a uniform signal gap and stay recoverable. The chain is discretised on the same equidistant grid as the synthetic ensemble and enters the row-wise regression (3.7) unchanged.

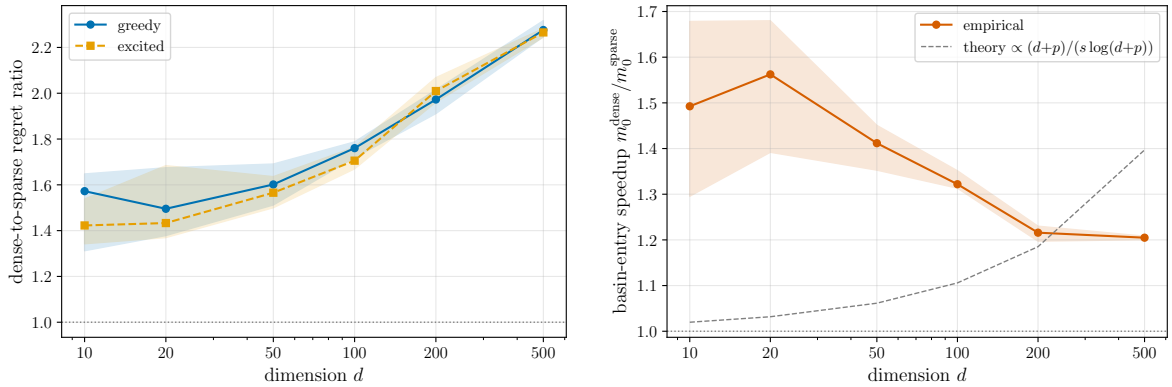
Figure 13 repeats the dimensional sweep on the chain. We evaluate $d = 2n \in \{10, 20, 50, 100, 200, 500\}$ at the canonical horizon $T = 1$ with $H = 50$ steps per episode. The picture mirrors the synthetic ensemble but is cleaner: the dense-to-sparse regret ratio exceeds one at every dimension with tight bands (Figure 13a), running from about 1.6 at $d = 10$ to about 2.3 at $d = 500$, and the basin-entry speedup (Figure 13b) staying above 1 throughout. That the regret advantage is largest at the top of the sweep, on a system whose row sparsity is genuinely fixed by physics, is a clean confirmation of the dimensional scaling prediction.

*We follow the convention of q for the generalised coordinate, reserving x for the full state.



(a) Spring-mass chain ($n = 6$, fixed walls) (b) Sparsity of $[\mathbf{A}_\star \mid \mathbf{B}_\star]$ ($d = 12$)

Figure 12: The coupled spring-mass chain benchmark. **(a)** n masses on linear springs between fixed walls; every other mass is actuated (highlighted, with control force u_j). **(b)** The induced sparsity pattern of $\mathbf{\Theta}_\star = [\mathbf{A}_\star \mid \mathbf{B}_\star]$ for $n = 6$ ($d = 12$). The drift block $\mathbf{A}_\star$ is banded – position rows carry a single unit entry and velocity rows couple only to a mass and its neighbours (three drift nonzeros); with one control entry per actuated mass in $\mathbf{B}_\star$, the largest joint row support is a fixed $s = 4$ regardless of d.



(a) Dense-to-sparse regret ratio vs. d (b) Basin-entry speedup ratio vs. d

Figure 13: Dimensional scaling on the spring chain ($s = 4$, $p = d/4$, medians across 20 seeds, IQR shaded). **(a)** Final dense-to-sparse regret ratio versus dimension: the sparse agent is better at every d, and the advantage grows with d. **(b)** Basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$ versus dimension: the sparse agent enters the basin first at every d.

5.3.2 IEEE 39-Bus Power Grid

While the spring chain is realistic, it is still our construction. As a truly real-world example we add the IEEE 39-bus "New England" system, the canonical benchmark for power-system dynamics (Athay et al., 1979; Pai, 1989). Notably, in this domain sparse networked linear control is not an analogy for our setting but the actual engineering problem.

An electric power grid is a network of synchronous generators and loads tied together by transmission lines. In normal operation every generator's rotor turns in synchrony at the nominal frequency (50 or 60 Hz), and the dynamic state of the system is each machine's rotor phase angle θ_i together with its frequency deviation $\omega_i = \dot{\theta}_i$ from nominal. Electricity is not stored in the network, so generation must match consumption from instant to instant. A surplus of mechanical power accelerates the rotors and frequency rises, a deficit decelerates them and frequency falls. Each generator therefore obeys Newton's law for a rotating mass, the *swing equation*, in which the rotor inertia M_i resists changes in frequency, a damping term D_i dissipates them, and the electrical power exchanged with the rest of the grid couples the machine to its neighbours. Real power flows from the higher to the lower phase angle along each line, in proportion to the line susceptance $B_{ij} = 1/x_{ij}$ (the reciprocal of its reactance) and, to first order about the operating point, proportionally to the angle difference $\theta_i - \theta_j$. Summed over a bus's incident lines this is exactly a graph Laplacian coupling. The grid is held at its setpoint by the generators' governors, which adjust mechanical power input to balance frequency excursions, while fluctuations of demand and, increasingly, of renewable generation act as exogenous disturbances. Keeping frequency within a narrow band is a safety-critical, real-time control task: large excursions trip protective relays and, in the worst case, cascade into blackouts (Kundur et al., 2004).

This problem instantiates our setting almost perfectly. The linearised dynamics are continuous-time and linear, the natural objective penalises angle and frequency deviation against governor effort (a quadratic LQ cost), the demand and renewable fluctuations are precisely the stochastic forcing $\sigma\,\mathrm{d}\boldsymbol{w}$, and operation around a schedule is naturally episodic. Crucially, the parameter matrix is genuinely and physically row-sparse as each bus couples only to its handful of electrical neighbours. Yet the system is high-dimensional, with real power grids running to hundreds or thousands of buses. The dynamics are also uncertain in practice: line parameters drift with temperature and loading, the topology changes as lines are switched in and out or fail, and the displacement of heavy synchronous machines by low-inertia renewables steadily reshapes the inertia distribution M (Milano et al., 2018). Operators must therefore identify and re-identify grid models from measurement data, and doing so sample-efficiently is the practical payoff the sparse estimator promises. The IEEE 39-bus system is thus not a contrived sparse matrix but a faithful, safety-critical instance of the problem this thesis studies. Structurally it is the same second-order network as the spring chain in Section 5.3.1 but now on a real, irregular graph whose sparsity pattern is given by the physical grid.

We model the grid as a structure-preserving network of swing oscillators, a standard reduction in the power-systems literature (Bergen & Hill, 1981; Dörfler et al., 2013). Each bus i carries a voltage phase angle θ_i and frequency deviation $\omega_i = \dot{\theta}_i$, and obeys the linearised swing equation

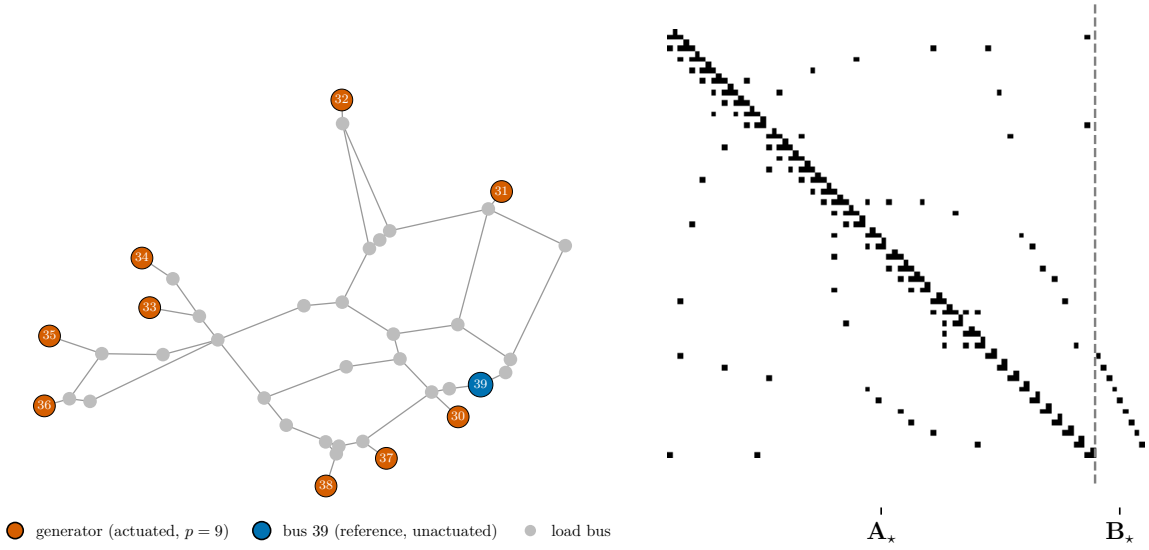
$$M_i\ddot{\theta}_i + D_i\dot{\theta}_i = -\sum_j L_{ij}\,\theta_j + \mathbb{1}_{\{i \text{ is a generator}\}}\,u_i, \tag{5.4}$$

where M_i is the inertia, D_i the damping, and L the susceptance-weighted network Laplacian: $L_{ij} = -1/x_{ij}$ for a line of reactance x_{ij} between buses i and j (otherwise 0), with $L_{ii} = -\sum_{j\neq i} L_{ij}$. Because L inherits the grid's connectivity, each bus couples only to its electrical neighbours. In the interleaved state $\boldsymbol{x} = [\theta_1, \omega_1, \ldots, \theta_n, \omega_n]^\top$ of dimension $d = 2n$, this is exactly the form (3.1) with the same block sparsity as the spring chain: angle rows $\frac{\mathrm{d}}{\mathrm{d}t}\theta_i = \omega_i$ contribute a single entry while frequency rows carry the bus's Laplacian neighbourhood. The only difference is that here the support follows an irregular graph instead of a band.

The standard New England case has $n = 39$ buses, 46 transmission lines, and 10 generators (Figure 14, left), giving $d = 78$ with both angle and frequency states. We actuate the nine New England generators (buses 30–38), so $p = 9$, and leave bus 39 as the unactuated reference, which aggregates the rest of the interconnection. The reactances and generator inertias are the published values, while load buses are assigned a nominal inertia. Appendix A.1 gives the full construction, the data sources, and the explicit state-space assembly. Each control enters one generator's frequency equation, so $\mathbf{B}_\star$ has a single nonzero per column. The resulting $\mathbf{\Theta}_\star = [\mathbf{A}_\star \mid \mathbf{B}_\star]$ is only 3.2% dense, with row support up to $s = 7$ set by the maximum bus degree (cp. Figure 14, right).

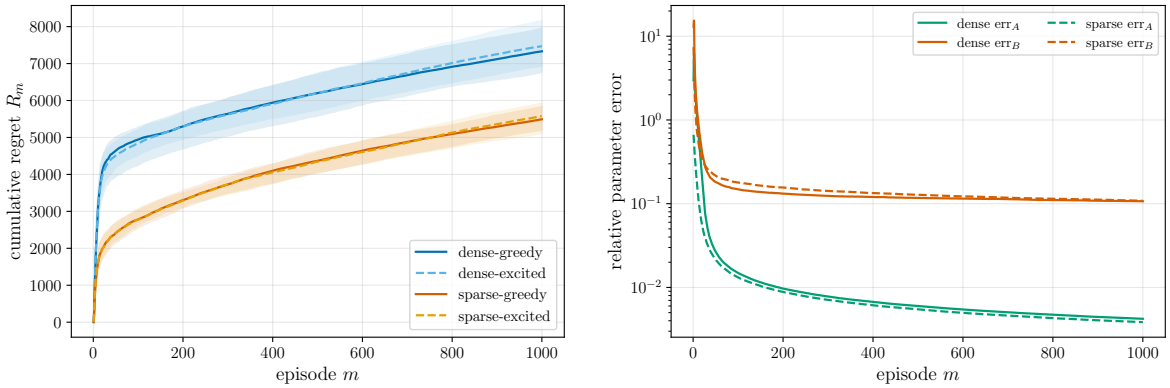
Two modelling choices keep the power grid problem well-posed. First, the physical control authority $1/M_i$ is tiny against the process noise, because generator inertias are large. Therefore, we express the input in units of the median generator inertia as $(\mathbf{B}_\star)_{\cdot,j} = M_{\text{ref}}/M_{a_j}$. This rescales only u and lifts the control above the noise floor so that $\mathbf{B}_\star$ is well-identifiable. The drift $\mathbf{A}_\star$ depends solely on the ratios L/M and D/M and is unchanged. Second, uniform damping $D_i > 0$ renders every mode asymptotically stable except the reference mode, which is controllable from the generators. The system is thus stabilisable. Excluding the heavy reference bus from actuation also keeps $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star$ well-conditioned, satisfying the self-exploration condition (Section 3.3). As a single fixed system this is only a snapshot, since across seeds only the noise varies. We evaluate it at the canonical $T = 1$ with $H = 50$ steps per episode.

On this real system the sparse advantage is at its most pronounced and least ambiguous. Figure 15a shows that both sparse agents incur markedly lower cumulative regret than both dense agents from the first dozen episodes onward, and the interquartile bands do not overlap. The deterministic system and matched seeds make the gap statistically unambiguous in a way the noisier synthetic ratios are not. The median dense-to-sparse ratio is about 1.34 for both the greedy and the excited pair – smaller than the ratios



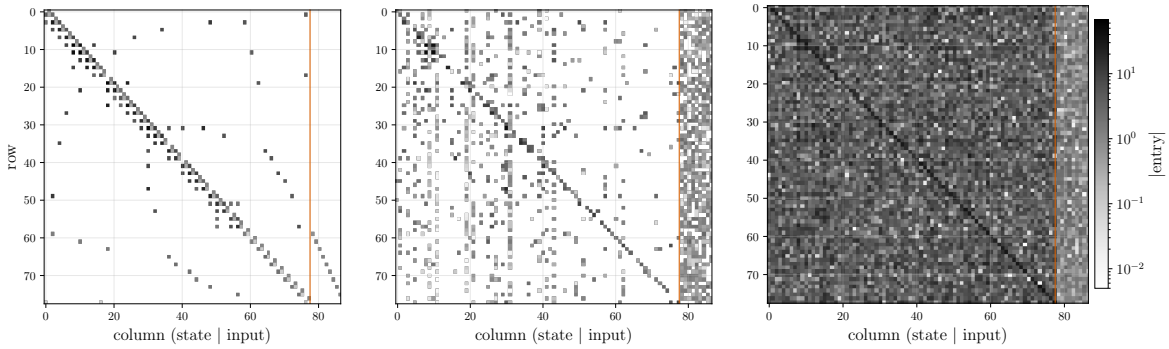
(a) IEEE 39-bus topology (39 buses, 46 lines) (b) Sparsity of $[\mathbf{A}_\star \mid \mathbf{B}_\star]$ ($d = 78$)

Figure 14: The IEEE 39-bus power grid. **(a)** The New England topology with 39 buses, 46 lines, the nine actuated generators (highlighted) and bus 39 as the unactuated reference. **(b)** The induced sparsity of $\mathbf{\Theta}_\star = [\mathbf{A}_\star \mid \mathbf{B}_\star]$ ($d = 78$, 3.2% nonzero). As for the chain the angle rows carry a single entry, but the frequency rows now scatter over the irregular grid graph rather than forming a band. $\mathbf{B}_\star$ has one entry per actuated generator.



(a) Cumulative regret (b) Drift vs control recovery

Figure 15: Results on the IEEE 39-bus power grid ($d = 78$, $p = 9$, medians across 20 seeds, IQR shaded). **(a)** Cumulative regret: the sparse agents separate from the dense agents early, with non-overlapping bands. **(b)** Drift versus control recovery for the dense and sparse greedy agents: sparsity accelerates recovery of $\mathbf{A}_\star$ to roughly 0.004, while $\mathbf{B}_\star$ is at the common floor near 0.11.



(a) True parameters (b) LASSO estimate (c) OLS estimate

Figure 16: Topology recovery on the IEEE 39-bus grid after the initial episode: log-magnitude heatmaps of $\mathbf{\Theta} = [\mathbf{A} \mid \mathbf{B}]$ for a median-regret seed. **(a)** The true parameters, whose support encodes the grid's wiring (3.2% dense). **(b)** Our sparse (LASSO) estimate after the single exploration episode ($H = 50$ samples): 82% of entries are exactly zero and the coupling skeleton is already recognisable. **(c)** The dense (least-squares) estimate from the same data: underdetermined, it returns noise at the magnitude of the largest true couplings. The vertical line separates the state block from the input block. The control block is noisy for both estimators, showing the drift-control asymmetry of Figure 15b. By $m = 1000$ both estimators recover the support almost perfectly.

at the largest synthetic and spring dimensions, but the *cleanest* we observe, with non-overlapping confidence bands. That the grid is also the sparsest system we test, at 3.2% density, is consistent with the dimensional scaling observation. Figure 15b locates the source of the advantage. The drift block is recovered to a relative error of about 0.004 by both estimators, but the sparse estimator reaches that accuracy from fewer samples. The control block, by contrast, plateaus near 0.11 for both. The advantage is therefore almost entirely a drift-identification effect: sparsity recovers the network topology faster, while the control block remains the common hard floor. This is reminiscent of the same asymmetry seen on the synthetic ensemble, now on a real grid.

The wiring can be read off the estimate. Beyond the regret numbers, the estimate itself is interpretable on this system: the support of $\hat{\mathbf{\Theta}}$ *is* the learned wiring of the grid. Figure 16 makes this visible by comparing what the two greedy agents believe the grid looks like after their very first episode – the pure-exploration episode, whose $H = 50$ samples are fewer than the $d + p = 87$ unknowns of each row regression. Our LASSO already sketches a recognisable skeleton of the network, setting 82% of the entries exactly to zero, while the underdetermined least-squares estimate is noise at the full magnitude of the largest true couplings, with no visible structure. The control block is noisy even for the LASSO – the drift–control asymmetry of Figure 15b, visible in a single snapshot. This head start is transient by construction: as data accumulate, both

estimators recover the support essentially perfectly (support $F_1 \approx 1$ by $m = 1000$ at the threshold of Table 3). Notably, the analysis of Chapter 4 promises only parameter-error decay, never support recovery, so that the wiring emerges legibly – and early – is an interpretability bonus of the ℓ_1 geometry: the certainty-equivalent planner acts on a structurally sound map of the grid from the second episode onward.

It is natural to ask how our agents perform against an established IEEE 39-bus controller, but no such comparison is well-posed. The classical power-systems literature synthesises wide-area, LQR, and $\mathcal{H}_\infty$ damping controllers from a *known* state-space model, obtained by linearising the grid at its operating point (Dörfler et al., 2014). These methods presuppose exactly the $[\mathbf{A}_\star \mid \mathbf{B}_\star]$ that our agents must learn, and thus measure controller synthesis, not dynamics learning. The data-driven literature instead controls directly from measurements, e.g., with model-free deep reinforcement learning (Yan & Xu, 2020) or behavioural predictive control (Coulson et al., 2019). These works report closed-loop cost or damping rather than regret against the optimal policy and, where they invoke sparsity, place it on the controller gain rather than the identified dynamics (Dizche et al., 2019). None of these offers a comparable regret baseline for an agent that identifies a row-sparse $[\mathbf{A}_\star \mid \mathbf{B}_\star]$ online. We therefore benchmark internally by holding everything else fixed across the dense and sparse estimators, exactly as in the synthetic and spring-chain studies, so that the regret gap isolates the estimator alone.

5.4 Discussion

Taken together, our experiments support the theory's qualitative predictions while sharpening the quantitative picture. The clearest result is that, as predicted, the sparse regret advantage increases with dimension. It does so on the synthetic ensemble, on the spring chain with its physically fixed sparsity, and most cleanly on the 3.2%-dense power grid.

The other two predictions, earlier basin entry and a smaller regret growth constant, also hold, but more modestly than the worst-case theoretical analysis suggests. The reason might be the same across every experiment: the worst-case ratio $(d+p)/(s\log(d+p))$ assumes the dense least-squares estimator pays for all $d+p$ coordinates of each row, but on the well-excited designs studied here ordinary least squares performs far above its worst case, so the realised advantage is lower than the order of magnitude the bound permits. The sparse advantage is therefore real and dimension-dependent, but the worst-case bound is loose. Leveraging sparsity therefore helps reliably but not dramatically in this regime. The sparsity sweep completes the picture with the complementary axis: holding the dimension fixed and filling in the rows, the advantage grows as the rows get sparser but *reverses* once they are no longer sparse – crossing parity around 19% density and leaving the dense estimator ahead thereafter. Sparsity thus pays precisely to the extent that the dynamics are sparse, exactly the structure-dependence the $(d+p)/(s\log(d+p))$ ratio encodes.

Another theme recurs everywhere: the asymmetry between identifying the drift and the control. The state is persistently excited by process noise, so $\mathbf{A}_\star$ is recovered quickly and accurately, whereas $\mathbf{B}_\star$ is informed only weakly through self-exploration and remains the bottleneck. It worsens with dimension as $\mathbf{B}_\star^\top \mathbf{Q} \mathbf{B}_\star$ becomes ill-conditioned, and is only partly remedied by deliberate excitation. This asymmetry is another reason why the advantage is moderate: sparsity sharpens the block that is already easy. The power grid case study isolates the sparse advantage as a pure drift-identification effect with non-overlapping confidence bands.

Several limitations qualify these conclusions. First, the synthetic ensemble is in some respects benign: the signal gap (every nonzero entry exceeds 0.1 in magnitude) makes recovery easier than a general system with arbitrarily small couplings, and the equal-authority scaling of Section 5.2.1 deliberately removes a control-conditioning confound that a raw physical system might exhibit. Both choices isolate the sparsity effect but give a slight advantage to the estimator. Second, the budget $M = 1000$ does not reach the asymptotic regime in which the speedup is stated. Even so, the per-episode regret has entered its m^{-1} quadratic-basin tail well before the end (Figure 4), so the measured advantage reflects the basin regime rather than a short-budget transient. Third, the dimensional reach is bounded by computation, not statistics: the dense $\mathcal{O}(d^3)$ Riccati solve makes $d = 1000$ already impractical on our hardware, so the very high-dimensional regime in which the worst-case speedup predicts the largest gains remains untested, and confirming it would need more computational resources. Fourth, the power grid is a single fixed topology rather than a family, so its decisive, non-overlapping result speaks to one network. Finally, the entire setting is linear. Extending the study to nonlinear dynamics and to genuinely high dimensions are the natural next steps, both of which the machinery developed here is designed to support.

Chapter 6

Conclusion



6.1 Summary

This thesis shows what exploiting sparsity buys an agent that must learn to control an unknown linear system from its own experience. We studied the episodic, continuous-time, linear-quadratic setting with a row-sparse dynamics matrix, and addressed the question on three fronts.

On the methodological front (Chapter 3), we proposed SAMBL, which instantiates certainty-equivalent control with a row-wise LASSO estimator, so that all of the structure exploitation is confined to the estimator while the planner remains an off-the-shelf Riccati solver. Analysing the self-generated closed-loop data, we found that the time-varying optimal policy makes the loop self-exploring enough to identify the drift, but that a structural asymmetry leaves the control block only weakly excited. In fact, so weakly that the LASSO's restricted-eigenvalue curvature can fail along the control directions, with a degenerate zero-initialisation trap as the extreme case. A single initial episode of state-independent exploration removes the obstruction at negligible cost, and is the mechanism we adopt.

On the theoretical front (Chapter 4), we proved a regret decomposition into a finite transient and a logarithmic tail, driven by a sparse estimation rate of order $ds\log(d+p)$ in place of the dense $d(d+p)$. The consequence for control is a certified basin-entry speedup of order $(d+p)/(s\log(d+p))$: SAMBL provably reaches the low-regret regime sooner, by a factor that grows with the dimension, and a doubling schedule converts this into a logarithmic cumulative-regret guarantee.

On the empirical front (Chapter 5), the predicted advantage materialised on a synthetic ensemble, a spring-mass chain, and the IEEE 39-bus power grid. As predicted, it increases with dimension, reaching 1.9-fold on the synthetic ensemble and 2.3-fold on the spring chain at $d = 500$. The experiments also revealed a high-dimensional collapse of pure greedy play, caused by the control-block identifiability failure and averted by the exploration episode. Furthermore, they showed that the advantage is structure-dependent in the strongest sense: it grows as the rows get sparser and reverses once they are no longer sparse. While the effects are real and increase with dimension, that increase is far shallower than the growth of the theoretical speedup factor, because OLS appears to perform far above its worst case on the well-excited designs studied here.

6.2 Limitations

Several assumptions restrict the scope of our conclusions. The setting is linear-quadratic throughout, with dynamics that are linear, a cost that is quadratic, and an isotropic diffusion with known variance. The cost matrices are likewise assumed known. The synthetic ensemble is in some respects benign: a signal gap keeps every nonzero coupling identifiable, and the equal-authority scaling of the control inputs deliberately removes a control-conditioning confound that a raw physical system might exhibit. The regret guarantees rest on a restricted-eigenvalue condition that we can verify only through a proxy and, on the estimation side, treat the LASSO solution as an exact minimiser. The row-sparsity assumption is uniform: every row carries at most s nonzeros, rather than an average level that would permit occasional dense rows. The horizon of $M = 1000$ episodes does not reach the asymptotic regime in which the speedup is stated, although the per-episode regret has entered its logarithmic tail well before the end, so the measured advantage reflects the basin regime rather than a short-budget transient. Finally, the dimensional reach is bounded by computation rather than statistics: the dense cubic-time Riccati ODE solver makes the very-high-dimensional regime, in which the worst-case speedup predicts the largest gains, impractical on our hardware.

6.3 Future Work

Many of the assumptions mentioned above suggest a direction:

- **Infinite-horizon control.** The finite-horizon structure is what makes the closed loop self-exploring. A natural theoretical continuation would be to extend the analysis to the infinite-horizon, constant-gain setting, where identification is genuinely harder and persistent excitation is unavoidable.
- **Re-estimation timing.** We re-estimate at episode boundaries for simplicity. Sub-episode or event-triggered re-estimation could shorten the transient, at a computational cost that trades off against the regret saved. In turn, towards the later episodes, re-estimating only when the change in parameter estimate exceeds a threshold, or following the doubling schedule, would save computation without hurting regret. Characterising these trade-offs is an open question.
- **Irregular observation and control.** The regression formulation does not require an equidistant grid, so adaptive or irregular sampling of observations and control updates fits naturally and may further improve sample efficiency. Being able to decide *when* to sense and act, as in the time-adaptive continuous-time RL literature, would be a powerful extension.
- **Scaling the planner.** The sparse advantage is expected to be largest exactly where the Riccati ODE solve is most expensive. Therefore, a scalable planner (e.g., low-rank or sparsity-aware) is what would let the predicted very-high-dimensional gains be tested.

- **Other sparsity patterns.** Relaxing the sparsity assumption to an average level with occasional dense rows would leave the algorithm unchanged, and the estimation rate would have ds replaced by the total support $\sum_i s_i$. The restricted-eigenvalue condition, however, would then have to hold over the cone of the densest row, where the weakly excited control directions offer the least curvature. The rate thus follows the average density but the curvature follows the maximum. Whether a few dense rows bottleneck the advantage in practice remains open.
- **Unknown cost structure.** Our setting assumes the cost matrices $\mathbf{Q}, \mathbf{R}$ are known, but in practice they may be unknown or only partially known. In that case, they must be identified from data alongside the dynamics. Once they are estimated, planning can proceed as in this thesis due to the linear-quadratic closed form. If additionally the cost is structured, like the dynamics, the estimation machinery developed here may transfer directly.
- **General, non-quadratic costs.** Extending beyond the quadratic cost to a general convex or Huber objective would broaden applicability, but forfeits the closed-form linear feedback of the linear-quadratic case. Planning would then rely on numerical optimal control. Possibilities include convex model-predictive control for a convex cost with linear dynamics, or iterative LQ methods (iLQR/DDP) that reapproximate the cost quadratically and reuse the Riccati solver.
- **Nonlinear dynamics.** The broadest and most consequential extension is to nonlinear systems, through repeated linearisation that reuses the structured-estimation machinery developed here. This also sets up a contrast with the deep-learning frontier: architectures such as SkyJEPA (Rao et al., 2026) learn high-capacity latent world models for real-time control and recover an interpretable state only through a learned probe, whereas our approach has interpretability and regret guarantees by construction – but only for linear dynamics. Whether structure buys the same sample efficiency once the dynamics are nonlinear, and how much of the interpretability and the guarantees survive, is the open question we find most compelling.

These directions all rest on the premise with which the thesis began. As agents increasingly learn from their own experience rather than from stored data, it is the interactional cost of that experience, not its availability, that becomes the binding constraint. Our answer has been that structure is a lever against this cost. When the system to be controlled is sparse, as the physical systems we most want to control are, exploiting that sparsity lets an agent learn to act well from fewer and cheaper interactions. Crucially, the advantage grows with the very dimension that makes the problem hard. Sparsity is only one form of structure, and linear-quadratic control only one setting, but the principle we have sought to establish is more general: prior knowledge of structure is what makes learning from experience affordable, precisely where experience is most expensive.

Chapter A

Appendix A

A.1 IEEE 39-Bus Grid Construction

This appendix gives the explicit, reproducible construction of the IEEE 39-bus swing network used as the real-world example in Section 5.3.2. In particular, we specify the data sources and the exact dynamics formulation.

Data Sources

Topology and reactances. The network graph and per-unit line reactances are taken from the standard MATPOWER `case39` case file (Zimmerman et al., 2011). Each transmission line is one row of its `branch` matrix, from which we read the two terminal buses and the series reactance x_{ij} (the F_BUS, T_BUS and BR_X columns). We retain the 46 in-service lines and transformers with nonzero reactance and discard series resistance and line charging, as the swing-network coupling depends only on the susceptance $1/x_{ij}$.

Generator inertias. The inertia constants H_i (in seconds, on a 100 Megavolt-Amperes base) are *dynamic* data and are not part of the static MATPOWER case. We use the standard ten-machine New England values (Athay et al., 1979; Pai, 1989), listed in Table A.1. Bus 39 is the large equivalent machine that aggregates the rest of the interconnection ($H_{39} = 500$), and serves as the unactuated angle reference. Load buses carry no published inertia and are assigned a nominal $M_{\text{load}} = 10$.

Dynamics Formulation

Enumerate the $n = 39$ buses and form the susceptance-weighted Laplacian $L \in \mathbb{R}^{n\times n}$,

$$L_{ij} = -\frac{1}{x_{ij}} \quad \text{for a line } (i,j) \quad \text{and} \quad L_{ij} = 0 \quad \text{otherwise}, \qquad L_{ii} = -\sum_{j\neq i} L_{ij}, \tag{A.1}$$

so L inherits the grid's sparse connectivity. Write the inertia of bus i as $M_i = H_i$ for a generator and $M_i = M_{\text{load}}$ otherwise, and use a uniform damping $D_i = D = 3.0$.

Bus	30	31	32	33	34	35	36	37	38	39
H_i [s]	42.0	30.3	35.8	28.6	26.0	34.8	26.4	24.3	34.5	500.0

Table A.1: Generator inertia constants H_i of the ten New England machines (buses 30–39), on a 100 Megavolt-Amperes base. Buses 30–38 are actuated, while bus 39 is the heavy reference machine.

In the interleaved state $\boldsymbol{x} = [\theta_1, \omega_1, \ldots, \theta_n, \omega_n]^\top$ with $d = 2n = 78$, the linearised swing equation (5.4) reads

$$\dot{\theta}_i = \omega_i, \qquad \dot{\omega}_i = -\frac{D}{M_i}\,\omega_i \;-\; \frac{1}{M_i}\sum_j L_{ij}\,\theta_j \;+\; \frac{\mathbb{1}_{\{i \text{ actuated}\}}}{M_i}\,u_i. \tag{A.2}$$

Reading off the blocks of $\mathbf{A}_\star$: each angle row contributes the single entry $(\mathbf{A}_\star)_{\theta_i,\omega_i} = 1$, and each frequency row contributes the damping entry $(\mathbf{A}_\star)_{\omega_i,\omega_i} = -D/M_i$ together with $(\mathbf{A}_\star)_{\omega_i,\theta_j} = -L_{ij}/M_i$ for every bus j in i's neighbourhood (including $j = i$). The frequency row of bus i therefore carries $\deg(i) + 2$ drift nonzeros, and the largest bus degree fixes the densest row support at $s = 7$.

The $p = 9$ actuated generators (buses 30–38), indexed $a_1, \ldots, a_p$, each enter one frequency equation, so $\mathbf{B}_\star$ has a single nonzero per column,

$$(\mathbf{B}_\star)_{\omega_{a_j},\,j} = \frac{M_{\text{ref}}}{M_{a_j}}, \qquad M_{\text{ref}} = \text{median}\{M_{a_1}, \ldots, M_{a_p}\}. \tag{A.3}$$

The median-inertia factor M_{ref} rescales the input to unit order so that the physically tiny control authority $1/M_i$ is lifted above the process noise floor (Section 5.3.2). Since $\mathbf{A}_\star$ depends only on the ratios L/M and D/M, it is unaffected. The resulting $\boldsymbol{\Theta}_\star = [\mathbf{A}_\star \mid \mathbf{B}_\star] \in \mathbb{R}^{78\times 87}$ is 3.2% dense. We set $\mathbf{Q} \coloneqq \mathbf{I}_d$ and $\mathbf{R} \coloneqq \mathbf{I}_p$.

A.2 Tabulated Results

This appendix tabulates the exact numbers behind every experiment of Chapter 5, so that each quantitative claim can be verified without reading values off logarithmic plots. All entries are computed across the 20 seeds of the respective configuration: a single number is the median, and a bracketed pair $[\cdot, \cdot]$ is the interquartile range. Final cumulative regret R_M is reported in units of 10^3. The dense-to-sparse regret ratio and the basin-entry speedup $m_0^{\text{dense}}/m_0^{\text{sparse}}$ (Section 5.1.5) are formed *per seed* under the matched randomness of Section 5.1.4 and aggregated afterwards, so the median ratio does *not* necessarily equal the ratio of the tabulated medians.

Tables A.2 and A.3 list the dimensional sweeps on the synthetic ensemble (cp. Figure 2) and the spring-mass chain (cp. Figure 13). Table A.4 lists the sparsity sweep (cp. Figure 3), and Table A.5 the IEEE 39-bus benchmark (cp. Figure 15). The ablations follow: the exploration episode (Table A.6, cp. Figure 7), the regularisation constant (Table A.7, cp. Figure 8), the excitation scale (Table A.8, cp. Figure 9), the cost weighting (Table A.9, cp. Figure 10), and the discretisation step (Table A.10, cp. Figure 11).

	final regret R_M ($\times 10^3$, median)				regret ratio		speedup
d	dense-g.	dense-e.	sparse-g.	sparse-e.	greedy	excited	
10	0.194	0.201	0.143	0.145	1.29 [1.07, 1.45]	1.33 [1.16, 1.51]	1.54 [1.23, 1.79]
20	0.738	0.747	0.567	0.576	1.24 [1.19, 1.40]	1.30 [1.20, 1.41]	1.79 [1.52, 1.88]
50	3.64	3.70	2.40	2.43	1.54 [1.40, 1.60]	1.49 [1.41, 1.56]	1.50 [1.47, 1.54]
100	13.6	13.9	8.25	8.34	1.62 [1.58, 1.69]	1.68 [1.64, 1.72]	1.41 [1.36, 1.41]
200	47.9	46.6	27.7	27.5	1.73 [1.71, 1.77]	1.69 [1.68, 1.71]	1.30 [1.29, 1.31]
500	256	258	135	136	1.89 [1.86, 1.92]	1.88 [1.86, 1.90]	1.30 [1.30, 1.30]

Table A.2: Synthetic dimensional sweep (Figure 2). Final cumulative regret per agent, dense-to-sparse regret ratio per pair, and basin-entry speedup, vs. dimension d ($p = d/2$).

	final regret R_M ($\times 10^3$, median)				regret ratio		speedup
d	dense-g.	dense-e.	sparse-g.	sparse-e.	greedy	excited	
10	0.083	0.088	0.051	0.059	1.57 [1.31, 1.65]	1.42 [1.34, 1.54]	1.49 [1.30, 1.68]
20	0.308	0.324	0.213	0.219	1.50 [1.38, 1.67]	1.43 [1.37, 1.68]	1.56 [1.39, 1.68]
50	1.55	1.56	0.998	1.02	1.60 [1.51, 1.69]	1.57 [1.50, 1.64]	1.41 [1.35, 1.45]
100	5.82	5.89	3.31	3.38	1.76 [1.72, 1.79]	1.71 [1.67, 1.76]	1.32 [1.31, 1.35]
200	20.6	21.0	10.4	10.5	1.97 [1.91, 2.01]	2.01 [1.96, 2.07]	1.22 [1.20, 1.23]
500	107	108	47.2	47.7	2.28 [2.25, 2.32]	2.27 [2.24, 2.28]	1.20 [1.20, 1.21]

Table A.3: Spring-chain dimensional sweep (Figure 13). Same quantities as Table A.2, on the chain of Section 5.3.1 (fixed $s = 4$, $p = d/4$).

	R_M ($\times 10^3$, median)		regret ratio		speedup
s	dense-g.	sparse-g.	greedy	excited	
2	9.51	5.64	1.68 [1.66, 1.73]	1.73 [1.67, 1.79]	1.44 [1.44, 1.45]
4	17.8	11.2	1.56 [1.50, 1.59]	1.60 [1.57, 1.67]	1.38 [1.35, 1.39]
6	30.6	20.9	1.46 [1.40, 1.49]	1.49 [1.42, 1.54]	1.37 [1.35, 1.40]
8	44.9	32.2	1.38 [1.33, 1.40]	1.38 [1.37, 1.46]	1.41 [1.38, 1.42]
10	66.8	49.0	1.35 [1.28, 1.44]	1.36 [1.31, 1.43]	1.40 [1.39, 1.44]
12	91.1	67.9	1.35 [1.26, 1.38]	1.36 [1.29, 1.41]	1.44 [1.42, 1.45]
14	126	96.1	1.29 [1.20, 1.36]	1.26 [1.20, 1.36]	1.45 [1.43, 1.47]
16	162	129	1.25 [1.20, 1.33]	1.25 [1.15, 1.34]	1.44 [1.42, 1.47]
18	226	171	1.27 [1.21, 1.35]	1.24 [1.18, 1.29]	1.45 [1.43, 1.46]
20	288	235	1.19 [1.13, 1.26]	1.19 [1.12, 1.26]	1.46 [1.43, 1.48]
22	379	342	1.09 [1.03, 1.21]	1.10 [1.01, 1.20]	1.46 [1.43, 1.47]
24	461	412	1.11 [1.05, 1.15]	1.10 [1.04, 1.25]	1.46 [1.43, 1.48]
26	549	491	1.05 [0.97, 1.13]	1.04 [0.89, 1.10]	1.48 [1.45, 1.53]
28	721	778	0.94 [0.85, 1.02]	0.89 [0.79, 0.93]	1.48 [1.45, 1.53]
30	795	1000	0.80 [0.74, 0.84]	0.81 [0.72, 0.88]	1.51 [1.49, 1.56]
32	938	1170	0.79 [0.73, 0.88]	0.80 [0.71, 0.88]	1.54 [1.49, 1.58]
34	1070	1580	0.71 [0.64, 0.76]	0.69 [0.64, 0.75]	1.63 [1.52, 1.72]
36	1140	1570	0.74 [0.70, 0.78]	0.72 [0.69, 0.77]	1.62 [1.56, 1.71]
38	1260	1870	0.70 [0.65, 0.74]	0.69 [0.66, 0.74]	1.65 [1.58, 1.74]
40	1370	2210	0.64 [0.61, 0.71]	0.66 [0.62, 0.72]	1.74 [1.65, 1.82]
42	1480	2290	0.65 [0.60, 0.73]	0.67 [0.62, 0.71]	1.71 [1.60, 1.81]
44	1540	2490	0.63 [0.60, 0.66]	0.65 [0.62, 0.68]	1.74 [1.66, 1.77]
46	1680	2730	0.62 [0.60, 0.66]	0.62 [0.57, 0.64]	1.66 [1.56, 1.74]
48	1860	3020	0.59 [0.58, 0.64]	0.60 [0.56, 0.65]	1.50 [1.36, 1.63]
50	1850	3140	0.62 [0.57, 0.64]	0.63 [0.57, 0.66]	1.52 [1.40, 1.58]

Table A.4: Sparsity sweep at $d = 100$, $p = 50$ (Figure 3), versus the row support size s.

quantity	median [IQR]
R_M dense-greedy ($\times 10^3$)	7.33 [7.20, 7.57]
R_M dense-excited ($\times 10^3$)	7.47 [7.18, 7.64]
R_M sparse-greedy ($\times 10^3$)	5.49 [5.41, 5.69]
R_M sparse-excited ($\times 10^3$)	5.58 [5.34, 5.73]
regret ratio, greedy pair	1.33 [1.28, 1.39]
regret ratio, excited pair	1.33 [1.29, 1.40]
basin-entry speedup	1.18 [1.16, 1.21]

Table A.5: IEEE 39-bus benchmark (Figure 15; $d = 78$, $p = 9$, $s = 7$).

	speedup		regret ratio (greedy)		$\min_i \lambda_{\min}(\mathbf{Z}_{\mathcal{S}_i}^\top \mathbf{Z}_{\mathcal{S}_i}/N_M)$	
d	$m_{\text{exp}} = 1$	$m_{\text{exp}} = 0$	$m_{\text{exp}} = 1$	$m_{\text{exp}} = 0$	$m_{\text{exp}} = 1$	$m_{\text{exp}} = 0$
10	1.54	1.64	1.29	1.23	2.0×10^{-1}	1.8×10^{-1}
20	1.79	1.71	1.24	1.34	2.1×10^{-1}	2.1×10^{-1}
50	1.50	1.60	1.54	1.39	2.2×10^{-1}	2.3×10^{-1}
100	1.41	1.37	1.62	1.32	2.4×10^{-1}	2.1×10^{-1}
200	1.30	0.47	1.73	0.91	2.9×10^{-1}	1.2×10^{-4}
500	1.30	0.70	1.89	1.03	3.4×10^{-1}	8.1×10^{-5}

Table A.6: Exploration-episode ablation (Figure 7), medians across seeds. The restricted-eigenvalue proxy (Section 5.1.5) is the sparse-greedy agent's value at the final episode.

	$d = 10$		$d = 20$		$d = 50$	
c_λ	median [IQR]	worst	median [IQR]	worst	median [IQR]	worst
0.005	0.17 [0.15, 0.21]	0.35	0.65 [0.57, 0.69]	0.97	3.19 [3.03, 3.42]	4.02
0.01	0.18 [0.14, 0.22]	0.30	0.64 [0.54, 0.73]	0.87	2.91 [2.84, 3.13]	3.67
0.02	0.14 [0.13, 0.18]	0.32	0.57 [0.48, 0.61]	0.76	2.40 [2.29, 2.63]	3.44
0.05	0.10 [0.09, 0.32]	5.48	0.33 [0.29, 0.43]	1.80	1.30 [1.23, 1.50]	2.15
0.1	3.77 [2.60, 5.25]	7.88	7.55 [4.77, 10.5]	15.6	16.5 [14.6, 18.3]	28.4

Table A.7: Regularisation-constant calibration (Figure 8): sparse-greedy final regret R_M ($\times 10^3$) versus c_λ, per dimension. The *worst* column is the single worst seed and exposes the heavy tail at $c_\lambda = 0.05$ that the median conceals.

σ_u	$d = 10$	$d = 20$	$d = 50$
0	0.14 [0.13, 0.18]	0.57 [0.48, 0.61]	2.40 [2.29, 2.63]
0.025	0.14 [0.13, 0.20]	0.54 [0.46, 0.62]	2.40 [2.31, 2.59]
0.05	0.15 [0.14, 0.20]	0.57 [0.50, 0.66]	2.45 [2.32, 2.68]
0.1	0.18 [0.17, 0.21]	0.62 [0.54, 0.71]	2.64 [2.52, 2.82]
0.2	0.32 [0.30, 0.36]	0.88 [0.84, 0.90]	3.17 [3.08, 3.39]
0.4	0.89 [0.88, 0.95]	1.96 [1.94, 1.99]	5.78 [5.63, 5.97]

Table A.8: Excitation-strength sweep (Figure 9): sparse-excited final regret R_M ($\times 10^3$) versus σ_u, per dimension ($\sigma_u = 0$ recovers sparse-greedy).

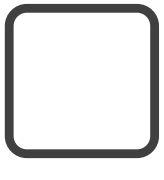
	R_M ($\times 10^3$, median)		regret ratio	
r	dense-g.	sparse-g.	greedy	excited
0.1	13.3	7.79	1.72 [1.64, 1.76]	1.71 [1.62, 1.81]
0.2	12.6	7.92	1.63 [1.60, 1.67]	1.68 [1.63, 1.80]
0.5	12.8	7.90	1.63 [1.58, 1.69]	1.66 [1.64, 1.73]
1	13.6	8.25	1.62 [1.58, 1.69]	1.68 [1.64, 1.72]
2	14.3	8.46	1.61 [1.58, 1.70]	1.70 [1.66, 1.75]
5	15.0	8.25	1.78 [1.75, 1.83]	1.89 [1.77, 1.94]

Table A.9: Cost-weighting sweep at $d = 100$ (Figure 10), versus the control-cost scale r (with $q = 1$).

		R_M ($\times 10^3$, median)		sparse-greedy error	
Δt	H	dense-g.	sparse-g.	drift (err_A)	control (err_B)
0.005	200	12.5	8.54	0.074 [0.072, 0.075]	0.164 [0.161, 0.166]
0.01	100	12.9	8.47	0.075 [0.073, 0.075]	0.165 [0.163, 0.167]
0.02	50	13.6	8.25	0.074 [0.073, 0.076]	0.165 [0.162, 0.168]
0.05	20	13.9	8.00	0.074 [0.072, 0.076]	0.165 [0.162, 0.169]
0.1	10	16.6	10.3	0.073 [0.072, 0.074]	0.349 [0.312, 0.375]
0.2	5	35.1	13.8	0.068 [0.066, 0.069]	0.663 [0.641, 0.693]

Table A.10: Discretisation sweep at $d = 100$ (Figure 11), versus the Euler step Δt (with $H = T/\Delta t$ steps per episode). The errors are the final relative parameter errors of the sparse-greedy agent.

References



Abbasi-Yadkori, Y. and Szepesvári, C. Regret bounds for the adaptive control of linear quadratic systems. In Kakade, S. M. and von Luxburg, U. (eds.), *Proceedings of the 24th annual conference on learning theory*, volume 19 of *Proceedings of machine learning research*, pp. 1–26, Budapest, Hungary, June 2011. PMLR. 4

Agarwal, N., Hazan, E., and Singh, K. Logarithmic regret for online control. In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in neural information processing systems*, volume 32. Curran Associates, Inc., 2019. 4

Athay, T., Podmore, R., and Virmani, S. A Practical Method for the Direct Analysis of Transient Stability. *IEEE Transactions on Power Apparatus and Systems*, PAS-98(2): 573–584, March 1979. 63, 72

Basei, M., Guo, X., Hu, A., and Zhang, Y. Logarithmic regret for episodic continuous-time linear-quadratic reinforcement learning over a finite-time horizon. *Journal of Machine Learning Research*, 23(178):1–34, 2022. 3, 4, 5, 6, 20, 26, 31, 41, 45

Bergen, A. and Hill, D. A Structure Preserving Model for Power System Stability Analysis. *IEEE Transactions on Power Apparatus and Systems*, PAS-100(1):25–35, January 1981. 64

Berner, J., Richter, L., Sendera, M., Rector-Brooks, J., and Malkin, N. From discrete-time policies to continuous-time diffusion samplers: Asymptotic equivalences and faster training. *Transactions on Machine Learning Research*, 2026. 3

Bickel, P. J., Ritov, Y., and Tsybakov, A. B. Simultaneous analysis of Lasso and Dantzig selector. *The Annals of Statistics*, 37(4), August 2009. 33

Bühlmann, P. and Van De Geer, S. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Series in Statistics. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. 4, 17

Carmona, R. and Lauriere, M. Reconciling Discrete-Time Mixed Policies and Continuous-Time Relaxed Controls in Reinforcement Learning and Stochastic Control, 2025. Version Number: 2. 3

Chen, C.-T. *Linear system theory and design*. Oxford University Press, New York, international 4. ed edition, 2014. 12

Coulson, J., Lygeros, J., and Dorfler, F. Data-Enabled Predictive Control: In the Shallows of the DeePC. In *2019 18th European Control Conference (ECC)*, pp. 307–312, Naples, Italy, June 2019. IEEE. 5, 67

Davison, E. and Maki, M. The numerical solution of the matrix Riccati differential equation. *IEEE Transactions on Automatic Control*, 18(1):71–73, February 1973. 16, 47

Dean, S., Mania, H., Matni, N., Recht, B., and Tu, S. On the Sample Complexity of the Linear Quadratic Regulator. *Foundations of Computational Mathematics*, 20(4): 633–679, August 2020. 4, 25

Dizche, A. F., Chakrabortty, A., and Duel-Hallen, A. Sparse Wide-Area Control of Power Systems using Data-driven Reinforcement Learning. In *2019 American Control Conference (ACC)*, pp. 2867–2872, Philadelphia, PA, USA, July 2019. IEEE. 5, 67

Doya, K. Reinforcement Learning in Continuous Time and Space. *Neural Computation*, 12(1):219–245, January 2000. 3

Dörfler, F., Chertkov, M., and Bullo, F. Synchronization in complex oscillator networks and smart grids. *Proceedings of the National Academy of Sciences*, 110(6):2005–2010, February 2013. 64

Dörfler, F., Jovanovic, M. R., Chertkov, M., and Bullo, F. Sparsity-Promoting Optimal Wide-Area Control of Power Networks. *IEEE Transactions on Power Systems*, 29(5): 2281–2291, September 2014. 5, 67

Fattahi, S., Matni, N., and Sojoudi, S. Learning Sparse Dynamical Systems from a Single Sample Trajectory. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 2682–2689, Nice, France, December 2019. IEEE. 4

Feldbaum, A. A. Dual control theory. I. *Automation and Remote Control*, 21(9):874–880, 1960. 19

Friedman, J., Hastie, T., and Tibshirani, R. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1):1–22, 2010. 18, 23

Gao, X. and Zhou, X. Y. Logarithmic Regret Bounds for Continuous-Time Average-Reward Markov Decision Processes. *SIAM Journal on Control and Optimization*, 62 (5):2529–2556, October 2024. 3

Green, M. and Moore, J. B. Persistence of excitation in linear systems. *Systems & Control Letters*, 7(5):351–360, September 1986. 5, 26

Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Xu, H., Ding, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Chen, J., Yuan, J., Tu, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., You, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Zhou, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081): 633–638, September 2025. 1

Haarnoja, T., Moran, B., Lever, G., Huang, S. H., Tirumala, D., Humplik, J., Wulfmeier, M., Tunyasuvunakool, S., Siegel, N. Y., Hafner, R., Bloesch, M., Hartikainen, K., Byravan, A., Hasenclever, L., Tassa, Y., Sadeghi, F., Batchelor, N., Casarini, F., Saliceti, S., Game, C., Sreendra, N., Patel, K., Gwira, M., Huber, A., Hurley, N., Nori, F., Hadsell, R., and Heess, N. Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Science Robotics*, 9(89):eadi8022, April 2024. 1

Hastie, T., Tibshirani, R., and Wainwright, M. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman and Hall/CRC, 0 edition, May 2015. 4, 17

He, J., Zhou, D., and Gu, Q. Logarithmic regret for reinforcement learning with linear function approximation. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th international conference on machine learning*, volume 139 of *Proceedings of machine learning research*, pp. 4171–4180. PMLR, July 2021. 4

Hespanha, J. P. *Linear systems theory*. Princeton University Press, Princeton, 2nd edition edition, 2018. 12

Huang, Y., Jia, Y., and Zhou, X. Y. Sublinear Regret for a Class of Continuous-Time Linear-Quadratic Reinforcement Learning Problems. *SIAM Journal on Control and Optimization*, 63(5):3452–3474, October 2025. 3

Ibrahimi, M., Javanmard, A., and Roy, B. Efficient reinforcement learning for high dimensional linear quadratic systems. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (eds.), *Advances in neural information processing systems*, volume 25. Curran Associates, Inc., 2012. 5

Iten, K., Treven, L., Sukhija, B., Dorfler, F., and Krause, A. Sample-efficient and scalable exploration in continuous-time RL. In *The fourteenth international conference on learning representations*, 2026. 3

Jedra, Y. and Proutiere, A. Minimal expected regret in linear quadratic control. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *Proceedings of the 25th international conference on artificial intelligence and statistics*, volume 151 of *Proceedings of machine learning research*, pp. 10234–10321. PMLR, March 2022. 4

Kloeden, P. E. and Platen, E. *Numerical Solution of Stochastic Differential Equations*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. 22

Kucera, V. A contribution to matrix quadratic equations. *IEEE Transactions on Automatic Control*, 17(3):344–347, June 1972. 49

Kundur, P., Paserba, J., Ajjarapu, V., Andersson, G., Bose, A., Canizares, C., Hatziargyriou, N., Hill, D., Stankovic, A., Taylor, C., Van Cutsem, T., and Vittal, V. Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definitions. *IEEE Transactions on Power Systems*, 19(3): 1387–1401, 2004. 63

Lai, T. Asymptotically efficient adaptive control in stochastic regression models. *Advances in Applied Mathematics*, 7(1):23–45, March 1986. 4

Ljung, L. *System identification: theory for the user*. Prentice Hall information and system sciences series. Prentice Hall PTR, Upper Saddle River, NJ, 2nd ed edition, 1999. 5

Luccioni, S., Jernite, Y., and Strubell, E. Power Hungry Processing: Watts Driving the Cost of AI Deployment? In *The 2024 ACM Conference on Fairness Accountability and Transparency*, pp. 85–99, Rio de Janeiro Brazil, June 2024. ACM. 1

Mania, H., Tu, S., and Recht, B. Certainty equivalence is efficient for linear quadratic control. In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in neural information processing systems*, volume 32. Curran Associates, Inc., 2019. 4, 19, 37

Milano, F., Dörfler, F., Hug, G., Hill, D. J., and Verbič, G. Foundations and Challenges of Low-Inertia Systems (Invited Paper). In *2018 Power Systems Computation Conference (PSCC)*, pp. 1–25, Dublin, Ireland, June 2018. IEEE. 63

Newman, M. E. J. The Structure and Function of Complex Networks. *SIAM Review*, 45(2):167–256, January 2003. 2

Pai, M. A. *Energy Function Analysis for Power System Stability*. Springer US, Boston, MA, 1989. 63, 72

Rao, P., Zhang, W., Balestriero, R., LeCun, Y., and Loianno, G. SkyJEPA: Learning Long-Horizon World Models for Zero-Shot Sim-to-Real Control of Quadrotors, 2026. Version Number: 2. 71

Silver, D. and Sutton, R. S. Welcome to the Era of Experience. 2025. 1

Simchowitz, M. and Foster, D. Naive exploration is optimal for online LQR. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th international conference on machine learning*, volume 119 of *Proceedings of machine learning research*, pp. 8937–8948. PMLR, July 2020. 4, 19, 58

Söderström, T. and Stoica, P. *System identification*. Prentice Hall International series in systems and control engineering. Prentice Hall, New York, 1989. 5

Tang, W. and Zhou, X. Y. Regret of exploratory policy improvement and q-learning, November 2024. arXiv:2411.01302 [cs]. 3

Tibshirani, R. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, January 1996. 17, 23

Treven, L., Hübotter, J., Sukhija, B., Dorfler, F., and Krause, A. Efficient exploration in continuous-time model-based reinforcement learning. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in neural information processing systems*, volume 36, pp. 42119–42147. Curran Associates, Inc., 2023. 3

Treven, L., Sukhija, B., As, Y., Dorfler, F., and Krause, A. When to sense and control? A time-adaptive approach for continuous-time RL. In *The thirty-eighth annual conference on neural information processing systems*, 2024. 3

Tsiamis, A., Ziemann, I., Matni, N., and Pappas, G. J. Statistical Learning Theory for Control: A Finite-Sample Perspective. *IEEE Control Systems*, 43(6):67–97, December 2023. 4

van Erven, T., Grünwald, P. D., Mehta, N. A., Reid, M. D., and Williamson, R. C. Fast rates in statistical and online learning. *Journal of Machine Learning Research*, 16 (54):1793–1861, 2015. 4

Vaughan, D. A negative exponential solution for the matrix Riccati equation. *IEEE Transactions on Automatic Control*, 14(1):72–75, February 1969. 15, 47

Villalobos, P., Ho, A., Sevilla, J., Besiroglu, T., Heim, L., and Hobbhahn, M. Position: Will we run out of data? Limits of LLM scaling based on human-generated data. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st international conference on machine learning*, volume 235 of *Proceedings of machine learning research*, pp. 49523–49544. PMLR, July 2024. 1

Wainwright, M. J. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, 1 edition, February 2019. 4, 17, 33, 40, 42

Wang, H., Zariphopoulou, T., and Zhou, X. Y. Reinforcement learning in continuous time and space: a stochastic control approach. *Journal of Machine Learning Research*, 21(198):1–34, 2020. 3

Wang, J., Lee, B., Ziemann, I., and Matni, N. Logarithmic regret for nonlinear control. In Ozay, N., Balzano, L., Panagou, D., and Abate, A. (eds.), *Proceedings of the 7th annual learning for dynamics & control conference*, volume 283 of *Proceedings of machine learning research*, pp. 552–565. PMLR, June 2025. 5

Wu, T. T. and Lange, K. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, 2(1), March 2008. 18, 23, 47

Yan, Z. and Xu, Y. A Multi-Agent Deep Reinforcement Learning Method for Cooperative Load Frequency Control of a Multi-Area Power System. *IEEE Transactions on Power Systems*, 35(6):4599–4608, November 2020. 5, 67

Yildiz, C., Heinonen, M., and Lähdesmäki, H. Continuous-time model-based reinforcement learning. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th international conference on machine learning*, volume 139 of *Proceedings of machine learning research*, pp. 12009–12018. PMLR, July 2021. 3

Yong, J. and Zhou, X. Y. *Stochastic controls: Hamiltonian systems and HJB equations*. Number 43 in Applications of mathematics. Springer, New York, 1999. 9, 10, 13, 14, 19, 24

Ziemann, I., Tsiamis, A., Lee, B., Jedra, Y., Matni, N., and Pappas, G. J. A Tutorial on the Non-Asymptotic Theory of System Identification. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pp. 8921–8939, Singapore, Singapore, December 2023. IEEE. 4

Zimmerman, R. D., Murillo-Sanchez, C. E., and Thomas, R. J. MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education. *IEEE Transactions on Power Systems*, 26(1):12–19, February 2011. 72

Zolman, N., Lagemann, C., Fasel, U., Kutz, J. N., and Brunton, S. L. SINDy-RL for interpretable and efficient model-based reinforcement learning. *Nature Communications*, 16(1):10714, November 2025. 5