Control-Oriented Identification via Stochastic Optimization

Sean Anderson, João P. Hespanha, *Fellow, IEEE*

***Abstract*— Data-driven control benefits from rich datasets, but constructing such datasets becomes challenging when gathering data is limited. We consider an offline experiment design approach to gathering data where we design a control input to collect data that will most improve the performance of a feedback controller. We consider a setting in which the dynamics are modeled parametrically and formulate a control-oriented identification procedure by way of a stochastic optimization problem that explicitly optimizes the post-experiment closed-loop control performance. We propose solving this problem via stochastic gradient descent by first constructing a gradient estimator of our stochastic objective. We then focus on a particular setting with linear dynamics and quadratic objective, which benefits from a numerically tractable gradient estimator. We show our formulation numerically outperforms an A- and L-optimal experiment design approach, illustrate the effects of scaling the system state and input dimensions, and compare against a recent robust dual control approach.**

***Index Terms*— identification for control, data driven control, uncertain systems, optimization**

I. Introduction

MODEL-BASED control methods benefit from accurate models of the controlled system. Consider a setting in which there is uncertainty in the model parameters and there is an opportunity to collect experimental data to learn more about the system. This motivates the following control-oriented experiment design problem: select a control input for a data-collection experiment so that the feedback controller designed using the data acquired will lead to improving the control performance as much as possible.

This paper includes two key contributions: First, we propose an experiment design formulation that explicitly optimizes the post-experiment closed-loop control performance. Notably, this formulation sidesteps the classical exploration-exploitation tradeoff through a unique optimization and achieves "optimal" exploration by construction. Second, we derive a gradient estimator to solve the resulting nonconvex optimization through stochastic gradient descent. In the setting with linear dynamics and quadratic objective function, we observe that our method generally leads to closed-loop controllers that exhibit higher performance than what would be achieved by 1) classical forms of experiment design such as A- [1] and L-optimal design and 2) a robust dual control method that minimizes the worst-case system cost.

In Section II we present a general formulation for experiment design that aims to minimize the expected post-experiment control performance by taking into account 1) *a-priori* parameter uncertainty in discrete-time dynamics and 2) process disturbances that occur during the experiment. Our approach is general in terms of the control design procedure used to generate the controller from the experimental data collected. However, in this paper we focus our attention on controllers generated through certainty equivalence, which in this context means constructing an *a-posteriori* estimate for the process and designing a controller for this estimate.

We present the solution method to the experiment design problem in Section III. We use a first-order approach by designing a pathwise gradient estimator for the purposes of stochastic gradient descent. For the remainder of the paper, we focus on the linear quadratic regulator (LQR) setting presented in Section IV. We address the system identification step and how to handle exploding trajectories during (simulated) experiments. For the data-driven controller, we use an LQR controller with certainty equivalence in the parameters [2]. The solutions available in LQR are amenable to fast computations, which is conducive to the gradient estimator presented in Section III. In Section V, we compare our method against A- and L-optimal experiment design in a car string setting and show how our approach scales numerically. We also consider how our approach can be compared against a recent robust dual control formulation.

Related work: The issue of how to gather data through well-planned experiments has traditionally been addressed through the framework of optimal experiment design. Modern optimal experiment design is often attributed to Gustav Elfving, who designed experiments to minimize measures of parameter error covariance [3]. Later on, researchers worked on aligning experiment design with particular criteria, including control objectives [4], [5]. Recent work in this area includes [6], which proposes a stochastic gradient descent approach to designing experiments that minimizes a post-experiment optimal control objective. Work in experiment design for control in the statistical learning community includes [7], [8], which emphasize theoretical aspects of learning linear systems. As an alternative

Submitted for review on March 8, 2024. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 2139319 and by the U.S. Office of Naval Research MURI grant No. N00014-23-1-2708. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the National Science Foundation or U.S. Office of Naval Research.

S. Anderson and J. P. Hespanha are with the Electrical and Computer Engineering Department, University of California, Santa Barbara, CA 93106 USA (e-mail: seananderson@ucsb.edu, hespanha@ece.ucsb.edu).

paradigm, online learning [9] or adaptive control [10] allow for improvement during the experiment trial.

In the control community, recent work in the linear quadratic setting includes [11], [12] in which the authors propose a robust dual control approach that minimizes the control cost associated with a worst-case system. The authors in [13] account for changes in parameter estimates in the setting in [12] while [14] proposes a robust experiment design method for virtual reference feedback tuning. Our method differs from these in that we consider the expected value over the prior while [11]–[13] consider the worst-case system given a pre-specified confidence level, thus requiring a different approach to solve and yielding different results (as in Section V-B).

Gradient estimation has seen particular attention in the machine learning community [15], and two dominant methods are via the pathwise gradient and the score function (or log score) method [16]. Score function estimators benefit from only taking the gradient of the density function, but the structure of our problem is not amenable: as such, we focus on the pathwise gradient estimate.

II. Experiment Design for Data-Driven Control

We consider a discrete-time system with dynamics of the form

$$x_{t+1} = f(x_t, u_t, w_t; \theta), \tag{1}$$

with state $x_t \in \mathbb{R}^{n_x}$, control input $u_t \in \mathbb{R}^{n_u}$, and an unmeasured stochastic disturbance $w_t \in \mathbb{R}^{n_w}$ independent and identically distributed across time. The dynamics depend on parameters θ that are unknown, but for which we have an *a-priori* distribution. As such, we treat θ as a random variable in the sense that we do not know its value during an experiment realization.

An experiment will be performed to provide additional information about the parameter θ. State and input measurements are collected throughout the experiment providing a sequence of M triples $\mathcal{D} := \{(x_i^+, x_i, u_i) : i = 1, ..., M\}$ that satisfy the basic model of the dynamical system:

$$x_i^+ = f(x_i, u_i, w_i; \theta), \quad \forall i \in \{1, \dots, M\} \tag{2}$$

with the w_i independent across indices i and with the same distribution as the disturbance. If the experiment consists of a single run of (1) over a time horizon $t = 1$ through $t = T$, then the index i is simply time and $M = T$. However, in general, "an experiment" may include multiple runs of (1) over different time horizons, in which case (2) include all the data collected. To simplify notation we collect all the column vectors $x_i^+, x_i \in \mathbb{R}^{n_x}, u_i \in \mathbb{R}^{n_u}$ into matrices with M columns that we denote by X^+, $X \in \mathbb{R}^{n_x \times M}, U \in \mathbb{R}^{n_u \times M}$, respectively.

Our goal is to design a controller π that optimizes a given cost function $J(\pi; \theta)$ that depends both on the controller and θ, which at the time of control synthesis we only have an a-posteriori distribution for, given $\mathcal{D}$. We also take as given a control design procedure that maps the experiment design data $\mathcal{D}$ to a specific controller π, with the goal of minimizing the cost $J(\pi; \theta)$. In this work we consider a controller $\pi = K(\mathcal{D})$ that minimizes $J(\pi; \hat{\theta})$ where $\hat{\theta} := \mathbb{E}_\theta[\theta \,|\, \mathcal{D}]$, but our approach allows for general control design procedures $K : \mathcal{D} \mapsto \pi$.

The experiment design problem arises from the observation that the data $\mathcal{D}$ collected depends on the realization of the parameter θ, control inputs U used during the experiment, as well as on the realizations of the random disturbances w_t such that the state trajectory X is a random variable. We use the notation $\mathcal{D}_{U,X}$ to express the dependence of the dataset on these variables. The optimal experiment design problem can then be formulated as

$$\min_{U \in \mathcal{U}} \mathop{\mathbb{E}}_{X,\theta} \big[J\big(\pi; \theta\big) \big], \quad \pi := K\big(\mathcal{D}_{U,X}\big), \tag{3}$$

where $\mathbb{E}_{X,\theta} \big[J\big(\pi; \theta\big) \big] := \int J\big(\pi; \theta\big) p(X, \theta; U) dX d\theta$ refers to an integration over (i) the *a-priori* distribution $p(\theta)$ of the parameter θ, and (ii) the realization of the state trajectory during the experiment of length T (such that $M = T$). The minimization is performed over a set of admissible controls that we denote generically by $\mathcal{U}$.

III. Experiment Design via Gradient Descent

In order to solve the experiment design optimization (3), we take a gradient-descent approach:

$$U_{j+1} = \text{Proj}_{\mathcal{U}}(U_j - \eta_j \hat{\nabla}_U), \tag{4}$$

where $\text{Proj}_{\mathcal{U}}(U)$ projects U onto the set of admissible inputs $\mathcal{U}$, η_j is step size, and we estimate the true gradient ∇_U with a pathwise gradient estimator to produce $\hat{\nabla}_U$.

We recall that for a general function $F(y)$ that is differentiable with respect to a random variable y with probability density function $p(y; U)$ that depends on a parameter U, the Monte Carlo pathwise gradient estimator of

$$\nabla_U \mathop{\mathbb{E}}_{y}[F(y)] := \nabla_U \int F(y) p(y; U) dy \tag{5}$$

is defined by

$$\hat{\nabla}_U := \frac{1}{L} \sum_{l=1}^{L} \nabla_U F(g(\epsilon^{(l)}; U)), \quad \epsilon^{(l)} \sim p(\epsilon), \tag{6}$$

where L is the number of Monte Carlo samples, and g is a differentiable sampling path such that y is distributed the same as $z := g(\epsilon; U)$, where ϵ is a random variable with continuous distribution $p(\epsilon)$ [15]. Pathwise gradient estimators are unbiased, typically low variance, and computationally efficient [15]. The variance has been shown to be bounded by the square of the Lipschitz constant of F [17].

Assumption 1 (Regularity): Assume that the controller π is parameterized by S scalar parameters $(\pi_1, \pi_2, \dots, \pi_S)$; J is differentiable with respect to π_s such that infinitesimal perturbations in π_s lead to infinitesimal perturbations in J, where π_s comes from the control design procedure $K(\mathcal{D})$; the procedure $K(\mathcal{D})$ is differentiable with respect to X and U such that infinitesimal perturbations in the data lead to infinitesimal perturbations in the controller; and, similarly, $f(\cdot)$ in (1) is differentiable with respect to x_t and u_t.

Theorem 1: Assume that the process noise at each time step is distributed according to a continuous distribution $p(w_t)$ such that $p(W) := \Pi_{t=1}^{T-1} p(w_{t-1})$ and Assumption 1 holds.

Then, the ijth element of the pathwise gradient estimator of $\mathbb{E}_{X,\theta}[J(\pi;\theta)]$ in (3) is given by

$$\hat{\nabla}_{U_{ij}} = \frac{1}{L}\sum_{l=1}^{L}\sum_{s,m,n}\frac{\partial J}{\partial \pi_s}\left(\frac{\partial K_s}{\partial X_{nm}}\frac{\partial g_{nm}}{\partial U_{ij}} + \frac{\partial K_s}{\partial U_{ij}}\right), \quad (7)$$

with the understanding that the gradient is evaluated at U, and the inner sum is over all S parameters and the elements of X. The sampling path is given by:

$$g(W,\theta;U) = \begin{bmatrix} x_0^T \\ g_0(x_0,u_0,w_0;\theta)^T \\ \vdots \\ g_{T-2}(x_0,u_{0:T-2},w_{0:T-2};\theta)^T \end{bmatrix}^T, \quad (8a)$$

such that we sample $W^{(l)}$ from $p(W)$ and $\theta^{(l)}$ from $p(\theta)$, and $u_{0:k}$ are the first $k+1$ columns of U, $g_k(x_0,u_{0:k},w_{0:k},\theta) := f(g_{k-1}(x_0,u_{0:k-1},w_{0:k-1}),u_k,w_k;\theta)\ \forall k \in \{1,...,T-1\}$, and $g_0(x_0,u_0,w_0;\theta) = f(x_0,u_0,w_0,\theta)$.

Proof: Consider the integral in the experiment design criteria in (3):

$$\int J\big(K(\mathcal{D}_{U,X});\theta\big)p(X,\theta;U)dXd\theta. \quad (9a)$$

By Bayes' rule for probability density functions $p(X,\theta;U) = p(X\,|\,\theta;U)p(\theta;U)$ and the joint distribution of the states can be recursively expanded as

$$p(X\,|\,\theta;U) = \Pi_{t=1}^{T-1}p(x_t\,|\,x_0,...,x_{t-1},\theta;U), \quad (9b)$$

where x_0 is known. From (1), we have Markovian dynamics such that $p(x_t\,|\,x_0,...,x_{t-1},\theta;U) = p(x_t\,|\,x_{t-1},\theta;u_{t-1})$ and

$$p(X\,|\,U,x_0,\theta) = \Pi_{t=1}^{T-1}p(x_t\,|\,x_{t-1},\theta;u_{t-1}). \quad (9c)$$

We can further decompose this by integrating over the process noise in (1):

$$= \Pi_{t=1}^{T-1}\int p(x_t\,|\,x_{t-1},w_{t-1},\theta;u_{t-1})p(w_{t-1})dw_{t-1}. \quad (9d)$$

Since $p(x_t\,|\,x_{t-1},w_{t-1},\theta;u_{t-1})$ occurs with probability one when the state at time t equals x_t, we express this using a delta function:

$$= \Pi_{t=1}^{T-1}\int \delta(x_t - f(x_{t-1},u_{t-1},w_{t-1}))p(w_{t-1})dw_{t-1}. \quad (9e)$$

Substituting into (9a) yields

$$\int J\big(K(\mathcal{D}_{U,X});\theta\big)\Pi_{t=1}^{T-1}\int \delta\big(x_t - f(x_{t-1},u_{t-1},w_{t-1})\big) p(w_{t-1})dw_{t-1}dx_{t-1}p(\theta)d\theta. \quad (9f)$$

Integrating with respect to any x_t leads to

$$\int J\big(K(\mathcal{D}_{U,X});\theta\big)\delta\big(x_t - f(x_{t-1},u_{t-1},w_{t-1})\big)dx_t = J\big(K(\mathcal{D}_{U,[x_0,...x_t=g_{t-1}(x_0,u_{0:t-1},w_{0:t-1};\theta),...]};\theta\big). \quad (9g)$$

If we integrate out x_t for all t and define $g(W,\theta;U) := [x_0,g_0,...,g_{T-1}]$, then (9a) equals

$$\int J\big(K(\mathcal{D}_{U,g(W,\theta;U)});\theta\big)\Pi_{t=1}^{T-1}p(w_{t-1})dw_{t-1}p(\theta)d\theta, \quad (9h)$$

such that the probability density functions are independent of U. Thus, under the change of variable g, $\mathbb{E}_{X,\theta}\big[J\big(K(\mathcal{D}_{U,X};\theta)\big)\big] = \mathbb{E}_{W,\theta}\big[J\big(K(\mathcal{D}_{U,g(W,\theta;U)})\big)\big]$. Now, differentiating (9h) can be achieved under the differentiability assumptions on J and K, and g inherits differentiability from f. Differentiating J with respect to the ijth element of U at the current input U and $X = g(W,\theta;U)$ yields

$$\nabla_{U_{ij}}J = \sum_{s,m,n}\frac{\partial J}{\partial \pi_s}\left(\frac{\partial K_s}{\partial X_{nm}}\frac{\partial g_{nm}}{\partial U_{ij}} + \frac{\partial K_s}{\partial U_{ij}}\right). \quad (9i)$$

We then take a Monte Carlo sample average to obtain (7). ■

A. Algorithm for Experiment Design Problem

In the pathwise gradient estimator (7) for each sample, l, we obtain a single experiment trajectory under sampled noise W for a sampled system θ under the candidate input U. For this realization, we compute an *a-posteriori* system estimate, $\hat{\theta}$, and compute the control, π. The step size η_j decays exponentially, and Algorithm 1 terminates when the moving average of the norm of the gradient is sufficiently small or hits the max allowable iterations. Because the experiment design problem is solved offline, it remains practical to use a conservative stopping condition (and a large max number of iterations).

The main hurdles in this general setting are obtaining a computationally efficient form of 1) $K(\mathcal{D})$ since this requires estimating the system and computing a controller with respect to the system estimate and 2) $J(\cdot)$ due to computing the value function and $K(\mathcal{D})$ for each sample.

Algorithm 1 Control-Oriented Experiment Design

Input $p(\theta)$ (prior on θ), U_0 (initialization), L (batch size), $\mathcal{U}$ (feasible set), $p(W)$ (noise dist.)
Output U^*

while not converged **do**
  for i=1 to L **do**
    $\theta^{(i)} \sim p(\theta), W^{(i)} \sim p(W)$
    $X \leftarrow g(W^{(i)},\theta^{(i)};U_j)$
    $\mathcal{D} \leftarrow X, U_j$
    $\pi \leftarrow K(\mathcal{D})$
    $\nabla_U J_i \leftarrow$ Compute gradient of $J(\pi;\theta^{(i)})$
  end for
  $U_{j+1} \leftarrow \text{Proj}_{\mathcal{U}}\big(U_j - \eta_j\frac{1}{L}\sum_{l=1}^{L}\nabla_U J_l\big)$
end while

IV. Experiment Design for the Linear Quadratic Regulator

We now specialize the general setup described above to the finite-horizon linear quadratic regulator setup. Specifically, we consider the process

$$x_{t+1} = Ax_t + Bu_t + w_t, \quad (10)$$

such that θ contains the elements of A and B, and w_t is Gaussian noise identically distributed across time with zero mean and covariance Σ_w.

We consider a quadratic optimization criterion of the form:

$$J(\pi;\theta) := \mathbb{E}_W\left[x_N^T Q_N x_N + \sum_{t=0}^{N-1} x_t^T Q x_t + u_t^T R u_t \,|\, \theta\right], \tag{11}$$

where the expectation refers to an integration over the disturbances, $W := [w_0, ..., w_{N-1}]$, encountered by the controller π; Q_N, Q are positive semidefinite matrices; and R a positive definite matrix.

We consider a common option for control design generally known as *certainty equivalence (CE)*: certainty equivalence design $K_{CE}(\mathcal{D})$ computes the *a-posteriori* expected value of the unknown parameters $\hat{\theta} := \mathbb{E}_\theta[\theta \,|\, \mathcal{D}]$ and computes the linear optimal controller $u_t = K_t x_t$ that minimizes (11), *assuming that the estimate $\hat{\theta}$ is correct*.

1) System identification: In order to generate the *a-posteriori* estimate of the system $\hat{\theta}$ for $K_{CE}(\mathcal{D})$, we employ weighted Bayesian estimation on a dataset $\mathcal{D}$, which in our case will be the dataset generated under the experiment decision variable U. For identification, we express (10) as:

$$X^+ = \Theta Z + W \tag{12}$$

with $Z = [X; U] \in \mathbb{R}^{(n_x+n_u)\times M}$, and $\Theta := [A, B] \in \mathbb{R}^{n_x \times (n_x+n_u)}$. For ease of notation, we use $\theta \in \mathbb{R}^{n_x(n_x+n_u)}$ to denote the vectorized version of Θ via stacking its columns.

We consider a Gaussian prior on the parameters with mean $\Theta_0 \in \mathbb{R}^{n_x\times(n_x+n_u)}$ and covariance of the (i,j)th element with the (k,l)th element of Θ given by $\mathbb{E}_\Theta[(\hat{\Theta}-\Theta)_{ij}(\hat{\Theta}-\Theta)_{kl}] = (\Sigma_w)_{ki}(\Lambda_0^{-1})_{jl}$, where Σ_w is the known noise covariance and $\Lambda_0^{-1} \in \mathbb{R}^{(n_x+n_u)\times(n_x+n_u)}$ is a prior on the parameter covariance. The weighted Bayesian estimator for Θ is

$$\hat{\Theta} = (\Theta_0\Lambda_0 + X^+ S Z^T)\Lambda_n^{-1}, \tag{13a}$$

and the error covariance of the estimate $\hat{\Theta}$ is

$$\mathop{\mathbb{E}}_{\Theta}\left[(\hat{\Theta}-\Theta)_{ij}(\hat{\Theta}-\Theta)_{kl}\right] = (\Sigma_w)_{ki}(\Lambda_n^{-1})_{jl}, \tag{13b}$$

where $\Lambda_n := \Lambda_0 + ZSZ^T$, and $S \in \mathbb{R}^{M\times M}$ is the weight matrix. While a different prior and estimator could be used, this closed-form solution allows for efficient computations for our proposed experiment design. For derivation see e.g [18].

The weight matrix S improves the numerics of the regression problem, particularly since simulating unstable systems can lead to exponential growth in the state that, due to large numbers, lead to deleterious performance in the inversion of Λ_n. In particular, let

$$S(X) := \text{diag}([s(x_0), ..., s(x_N)]) \tag{14}$$

where $s(x) \in [0,1]$ ensures that the weight matrix assigns zero weight to points on trajectories that are numerically too large. For this work, we choose $s(x) := \arctan(\|x_t - \alpha_1\|\alpha_2)/\pi + 0.5$, and α_1, α_2 are design parameters.

2) Certainty equivalent control: Given an estimate of the parameters θ from (13) with means $\hat{A}$ and $\hat{B}$, respectively, we construct our controller $K_{CE}(\mathcal{D})$ by recursively solving the Riccati difference equations given by

$$K_t = -(R + \hat{B}^T P_{t+1}\hat{B})^{-1}\hat{B}^T P_{t+1}\hat{A}, \tag{15a}$$

$$P_t = Q + K_t^T R K_t + (\hat{A} + \hat{B}K_t)^T P_{t+1}(\hat{A} + \hat{B}K_t), \tag{15b}$$

with $P_N = Q_N$; Q, Q_N are positive semidefinite matrices and R a positive definite matrix.

Corollary 1: [19] For a sequence of linear feedback gains, $\pi := \{K_0, ..., K_{N-1}\}$ from $K_{CE}(\mathcal{D})$, we can express the finite-horizon LQR cost (11) for the system in (10) parameterized by θ as

$$J(\pi;\theta) = x_0^T P_0 x_0 + \sum_{t=0}^{N-1} \text{tr}(P_{t+1}\Sigma_w), \tag{16a}$$

where

$$P_t = Q + K_t^T R K_t + (A + BK_t)^T P_{t+1}(A + BK_t), \tag{16b}$$

with boundary condition $P_N = Q_N$.

Corollary 2: For the LQR experiment design pathwise gradient estimate, the sampling path $g(W, \theta; U)$ is given by

$$g(W,\theta;U) = \begin{bmatrix} x_0^T \\ (Ax_0 + Bu_0 + w_0)^T \\ \vdots \\ \left(A^{T-1}x_0 + \sum_{l=0}^{T-2} A^{T-2-l}(Bu_l + w_l)\right)^T \end{bmatrix}^T, \tag{17}$$

and in this problem the controller can be parameterized by the controller gains or the certainty equivalent estimate from which the gains are constructed.

Proof: See technical report [20] for proof. ■

In practice, the gradient can be computed efficiently using automatic differentiation.

V. Numerical Experiments

A. Car String

We consider the problem of maintaining a fixed distance, $\bar{L}$, between n cars at a desired velocity v. We adapt the continuous-time dynamics for relative position as given in [21] to discrete-time dynamics with sampling time T_s:

$$\Delta v_{t+1}^{(n)} = \left(-\frac{\alpha^{(n)}T_s}{m^{(n)}} + 1\right)\Delta v_t^{(n)} + \frac{T_s}{m^{(n)}}\Delta u_t^{(n)}, \tag{18}$$

$$\Delta w_{t+1}^{(n)} = T_s(\Delta v_t^{(n)} - \Delta v_t^{(n+1)}) + \Delta w_t^{(n)}, \tag{19}$$

where $\Delta v^{(n)}$ is the deviation from the reference velocity at car n and $\Delta w^{(n)}$ is the deviation of the gap between cars $n+1$ and n from the desired gap L. Δu is a change in force input for each car. This leads to an n car state-vector $x_{t+1} := [\Delta v_{t+1}^{(1)}, \Delta w_{t+1}^{(1)}, \Delta v_{t+1}^{(2)}, ..., \Delta v_{t+1}^{(n)}]^T$. While there is a specific structure to the resulting (A, B) matrices, we assume we do not know the structure and estimate all $(2n-1)(3n-1)$ entries as our method does not require *a-priori* knowledge of structure. We specify the noise covariance in the dynamics (10) as $\Sigma_w = 1\text{e}{-2} \times I_5$. The prior on the parameters (13) is $\Theta_0 = [A, B]$ with $m^{(1)} = m^{(2)} = m^{(3)} = 1$, $\alpha^{(1)} = \alpha^{(2)} = \alpha^{(3)} = 1$, $T_s = 0.1$; $\Lambda_0^{-1} = \text{diag}([0.1, 0.01, 0.05, 0.1, 0.01, 0.05, 0.1, 0.05])$, motivated by having high uncertainty in the velocity evolution and the influence of the input. The full expressions for A and B in the problem are shown in [20]. Horizon $N = 30$.

1) Experiment Design Setup: In the results that follow, we use an experiment horizon of $T = 20$ time steps, and batch size $L = 1000$ in Algorithm 1. As in [21], for the criteria in (11) Q includes penalties of magnitude 10 on the positions Δw and zero on the velocity Δv. R is the identity matrix. The weight matrix S has parameters $\alpha_1 = 10^3, \alpha_2 = 10^6$. U is initialized with $u_t \sim U[10^{-3}, 10^{-2}]$ and is fixed across experiments. We initialize $\eta_0 = 0.01$ from a small hyperparameter grid search.

We compare with A-optimality and L-optimality:

$$\min_{U \in \mathcal{U}} \mathbb{E}_{X,\theta} \left[(\hat{\theta} - \theta)^T H (\hat{\theta} - \theta) \right], \tag{20}$$

where $\hat{\theta}$ is a function of X and θ as in (13) and H is a positive semi-definite weight matrix that is the identity matrix in A-optimal design. For the L-optimal design, we use H inspired from [22] which considers the parameter sensitivity of the optimality gap $\mathcal{R}(\pi; \theta) := J(\pi; \theta) - \inf_\pi J(\pi; \theta)$ under a policy π such that $H = \nabla^2_\theta \mathcal{R}(\pi; \theta)|_{\theta=\hat{\theta}} + \mu I$, with μ chosen to ensure positive semi-definiteness. We solve this using a gradient estimator of the same form as (6).

2) Results and Discussion: We compare the performance of our method against A- and L- optimal design (20) in terms of post-experiment LQR control performance (11). For the experiment design (3), we consider a feasible input set $\mathcal{U} = \{U \,|\, \|U\|_F \leq \beta\}$, with β a design parameter. We vary the allowed magnitude, β, in Figure 1 and observe our method outperforms the alternative designs uniformly. More notably, the input budget needed to achieve the same cost as our method is significantly more for most values of β. For any experiment design, if we knew the values of (A, B), we would achieve the lowest possible control cost such that this is a lower bound on achievable performance. We also show the expected control performance associated with using a controller that uses the *a-priori* system estimate, which indicates the gap for improvement via experimentation.

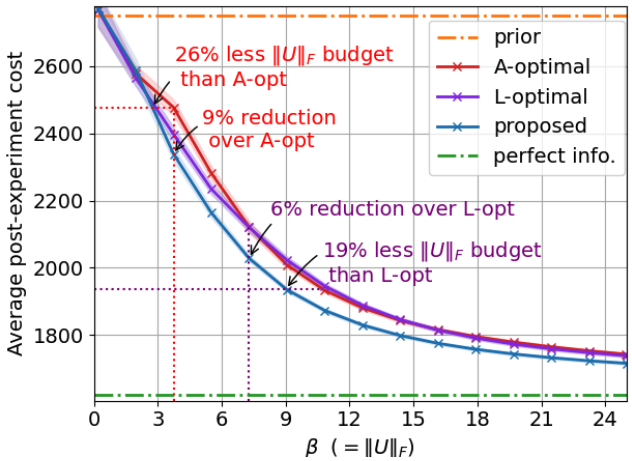

Fig. 1. We compare the performance of our control-oriented system identification against A-optimal experiment design for a system with five states and three inputs, and known initial condition $\boldsymbol{x_0} = [\mathbf{0.}, \mathbf{-4.3}, \mathbf{0.}, \mathbf{2.1}, \mathbf{2.5}]^T$ as in [21]. The value of $\boldsymbol{\beta}$ is varied and this constraint is active in all cases. We include 95% confidence intervals using $\mathbf{10^5}$ samples.

Figure 2 shows how the problem scales with the system dimension. In the first subplot we see the convergence of the experiment criteria in (3) as a function of iterations. The criteria is normalized by the lower bound (given by the performance if we knew A, B). The number of iterations until the criteria stabilizes is roughly constant across problem dimension suggesting that the number of iterations required is independent of the system size though the variance tends to grow with system dimension. Since the convergence rate of SGD is closely tied to the Lipschitz constant, this would suggest that the Lipschitz constant is roughly the same as this car string problem scales. In the second subplot, the time to compute each gradient sample is shown as a function of the state dimension. A- and L-optimal design avoid computing the post-experiment optimal control and control cost, such that the computation time should roughly be the red band in Figure 2. However, the offline experiment design setting reduces the necessity of fast computation time.

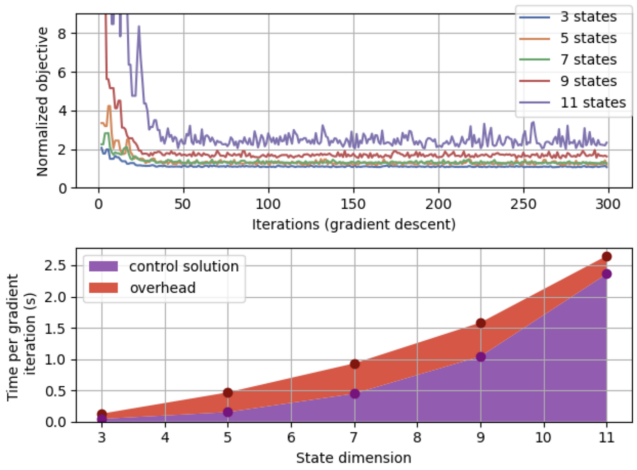

Fig. 2. In the upper subplot, the number of iterations to converge for the car string problem is essentially the same regardless of system size suggesting good scaling properties of our method. In the lower subplot, we observe the average time to compute a single gradient iteration on an Apple M1 Pro 10 core CPU with 32GB RAM in JAX [23]. The time is dominated by solving the control problem and "overhead" refers to tasks such as automatic differentiation, initial compile time, etc.

B. Dual control as experiment design

We also compare our method against a dual control approach [11] termed "robust reinforcement learning" (RRL) that approximately solves a problem of the form:

$$\min_{\pi_i^{RRL}} \sum_{i=1}^{N_{epochs}} \sup_{\theta \in \boldsymbol{\Theta}_i} \mathbb{E}_{w_t, e_t \forall t} \left[\sum_{t=t_{i-1}}^{t_i} x_t^T Q x_t + u_t^T R u_t \right] \tag{21}$$

where the dynamics (10) are driven by $u_t^{RRL} = \pi^{RRL}(x_t) = Kx_t + \Sigma^{\frac{1}{2}} e_t, e_t \sim \mathcal{N}(0, I)$, with optimization variables (K, Σ). The set $\boldsymbol{\Theta}_i$ contains system parameters such that $P(\theta_{tr} \in \boldsymbol{\Theta}_i \,|\, \mathcal{D}_i) = 1 - \delta$ for a particular system θ_{tr}. The dataset is initialized with $\mathcal{D}_1 = \mathcal{D}_{prior}$, an initial dataset gathered from N_{traj} trajectories of the system θ_{tr}. In this setup our prior is Gaussian with mean $\mathbb{E}_\theta[\theta \,|\, \mathcal{D}_{prior}]$ and variance $Var(\theta \,|\, \mathcal{D}_{prior})$ obtained from least squares estimation in RRL. π^{RRL} can be considered an alternative experiment input

signal to ours and we consider the application of π^{RRL} and our method for a single epoch ($N_{epochs} = 1$). We use the code and problem setup from [11] with θ_{tr} given by:

$$A = \begin{bmatrix} 1.1 & 0.5 & 0 \\ 0 & 0.9 & 0.1 \\ 0 & -0.2 & 0.8 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 \\ 0.1 & 0 \\ 0 & 2 \end{bmatrix} \tag{22}$$

and $Q = I, R = \text{blkdiag}(0.1, 1), \sigma_w = 0.5, \delta = 0.05$. The only parameters we change are the control and experiment horizon $T = N = t_i - t_{i-1} = 20$, which improves the computation of sample averages for both approaches.

Because our design requires a bound, β, on the experiment input and [11] does not include one, we first run the RRL experiments and then bound our design such that $\beta = \frac{1}{S}\sum_{s=1}^{S} \left\| U_{(s)}^{RRL} \right\|_F$ similar to [7], [22] where $U_{(s)}^{RRL}$ is an input sequence realization under RRL.

In Figure 3 we vary the information in the prior–measured as the trace of the prior covariance–by varying N_{traj} from 500 down to 200. RRL suffers from a very small percentage of systems causing the average to be very large, and we also see the range of the realized costs from the 5th and 95th percentiles is smaller for our method. Finally, since a particular system (22) generates the dataset for RRL, we note that across the datasets our method improves on RRL by an average of 1% and up to 4%. In terms of computation time, RRL takes 1.5 seconds and ours 30.0 seconds, so RRL is more than an order of magnitude faster.

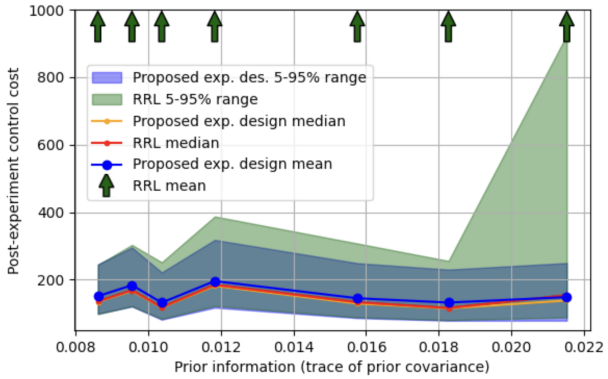

Fig. 3. We show the performance of our proposed method against RRL by varying the prior information, which is achieved by varying N_{traj} from 500 trajectories used in [11] down to 200 in increments of 50. The mean value for our method is shown in blue where the post-experiment cost remains below 200 for all priors. For RRL, we observe that a few very large samples move the RRL mean to be very large such that we use arrows to indicate the values lie outside the axis limits.

VI. Conclusion

We proposed a control-oriented identification approach that in expectation improves data-driven controllers by construction. Our solution method via SGD is numerically shown in the LQR setting to outperform relevant benchmarks. Establishing results on the convergence rate and sample complexity of the stochastic gradient descent is important future work.

VII. Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

[1] H. Chernoff, "Locally Optimal Designs for Estimating Parameters," *The Annals of Mathematical Statistics*, vol. 24, no. 4, pp. 586–602, Dec. 1953, publisher: Institute of Mathematical Statistics.

[2] H. A. Simon, "Dynamic Programming Under Uncertainty with a Quadratic Criterion Function," *Econometrica*, vol. 24, no. 1, pp. 74–81, 1956, publisher: [Wiley, Econometric Society].

[3] G. Elfving, "Optimum Allocation in Linear Regression Theory," *The Annals of Mathematical Statistics*, vol. 23, no. 2, pp. 255–262, Jun. 1952, publisher: Institute of Mathematical Statistics.

[4] K. Lindqvist and H. Hjalmarsson, "Identification for control: adaptive input design using convex optimization," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)*, vol. 5. Orlando, FL, USA: IEEE, 2001, pp. 4326–4331.

[5] M. Gevers, "Identification for Control: From the Early Achievements to the Revival of Experiment Design," in *Proceedings of the 44th IEEE Conference on Decision and Control*, Dec. 2005, pp. 12–12.

[6] S. Anderson, K. Byl, and J. P. Hespanha, "Experiment design with Gaussian process regression with applications to chance-constrained control," in *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023, pp. 3931–3938.

[7] B. D. Lee, I. Ziemann, A. Tsiamis, H. Sandberg, and N. Matni, "The fundamental limitations of learning linear-quadratic regulators," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 4053–4060.

[8] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "On the Sample Complexity of the Linear Quadratic Regulator," *Foundations of Computational Mathematics*, vol. 20, no. 4, pp. 633–679, Aug. 2020.

[9] M. Simchowitz and D. Foster, "Naive Exploration is Optimal for Online LQR," in *Proceedings of the 37th International Conference on Machine Learning*. PMLR, Nov. 2020, pp. 8937–8948, iSSN: 2640-3498.

[10] K. J. Åström and B. Wittenmark, *Adaptive control*. Courier Corp., 2008.

[11] J. Umenberger, M. Ferizbegovic, T. B. Schön, and H. Hjalmarsson, "Robust exploration in linear quadratic reinforcement learning," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019.

[12] M. Ferizbegovic, J. Umenberger, H. Hjalmarsson, and T. B. Schön, "Learning robust LQ-controllers using application oriented exploration," *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 19–24, 2020.

[13] J. Venkatasubramanian, J. Köhler, J. Berberich, and F. Allgöwer, "Robust dual control based on gain scheduling," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2270–2277.

[14] G. Rallo, S. Formentin, C. R. Rojas, and S. M. Savaresi, "Robust experiment design for virtual reference feedback tuning," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 2271–2276.

[15] S. Mohamed, M. Rosca, M. Figurnov, and A. Mnih, "Monte Carlo Gradient Estimation in Machine Learning," Sep. 2020.

[16] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Networks*, vol. 21, no. 4, pp. 682–697, 2008.

[17] K. Fan, Z. Wang, J. Beck, J. Kwok, and K. A. Heller, "Fast second order stochastic backpropagation for variational inference," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[18] P. E. Rossi, G. M. Allenby, and R. McCulloch, *Bayesian Statistics and Marketing*. Wiley, Oct. 2006, pp. 31–34.

[19] D. Bertsekas, *Dynamic programming and optimal control: Volume I*. Athena scientific, 2012, vol. 4, pp. 110–112.

[20] S. Anderson and J. P. Hespanha, "Control-oriented identification via stochastic optimization: Technical report," Santa Barbara, Mar. 2024. [Online]. Available: https://arxiv.org/abs/2403.05455

[21] W. Levine and M. Athans, "On the optimal error regulation of a string of moving vehicles," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 355–361, Jul. 1966.

[22] A. J. Wagenmaker, M. Simchowitz, and K. Jamieson, "Task-Optimal Exploration in Linear Dynamical Systems," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, Jul. 2021, pp. 10 641–10 652, iSSN: 2640-3498.

[23] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2018.