Linear System Identification Under Multiplicative Noise from Multiple Trajectory Data

Yu Xing[1], Ben Gravell[2], Xingkang He[3], Karl Henrik Johansson[3], and Tyler Summers[2]

***Abstract*— The study of multiplicative noise models has a long history in control theory but is re-emerging in the context of complex networked systems and systems with learning-based control. We consider linear system identification with multiplicative noise from multiple state-input trajectory data. We propose exploratory input signals along with a least-squares algorithm to simultaneously estimate nominal system parameters and multiplicative noise covariance matrices. The asymptotic consistency of the least-squares estimator is demonstrated by analyzing first and second moment dynamics of the system. The results are illustrated by numerical simulations.**

I. Introduction

The study of stochastic systems with noise which multiplies with the state and input i.e. multiplicative noise has a long history in control theory [1], but is re-emerging in the context of complex networked systems and systems with learning-based control. In contrast with the well-known additive noise setting, multiplicative noise has the ability to capture dependence of the noise on the state and/or control input. This situation occurs in modern control systems as diverse as robotics with distance-dependent sensor errors [2], networked systems with noisy communication channels [3], [4], modern power networks with high penetration of intermittent renewables [5], turbulent fluid flow [6], and neuronal brain networks [7]. Linear systems with multiplicative noise are particularly attractive as a stochastic modeling framework because they remain simple enough to admit closed-form expressions for stability and stabilization via generalized Lyapunov equations (e.g. [8]), optimal control via the solution of generalized Riccati equations [1], [9] and state estimation. Additionally, recent results show that the optimal control of this class of systems can be learned strictly from sample data without constructing a model via the reinforcement learning technique of policy gradient [10]. As a complementary perspective, here we tackle the problem from a model-based perspective where the goal is to learn and construct a model from sample data, which can then be used e.g. for optimal control design.

The first issue that must be addressed is that a complete multiplicative noise system model requires accurate estimates not only of the nominal linear system matrices, but also the noise covariance structure. This stands in stark contrast to the additive noise case where the noise covariance structure has no bearing on the control design and can thus be ignored during system identification. For the identification of a nominal linear system, recursive algorithms have been developed in the control literature, such as the recursive least-squares algorithm [11]. These can be utilized for linear systems with multiplicative noise provided that certain assumptions on the noise and on system stability hold. For the estimation of noise covariances, both recursive and batch estimation methods have been proposed over the last few decades (see [12] for a review), but these focus nearly exclusively on additive noise. In order to estimate multiplicative noise covariances, the maximum-likelihood approach was introduced in [13], and the Bayesian framework was utilized in [14], for example assuming Gaussian or known distributions with unknown parameters. These methods, however, require prior assumptions on the noise distributions whose incorrectness may worsen the performance of the concerned algorithms for optimal control. Our paper concentrates on jointly estimating the nominal system parameters and the multiplicative noise covariances without imposing any prior assumptions on the distribution of the noises, other than being independent and identically distributed (i.i.d.) with finite first and second moments, which complicates the problem. Both state- and control-dependent noise in the system leads to coupling, which also makes the identification task more difficult.

The second issue we address is that of performing system identification based on multiple state-input trajectory data rather than a single trajectory. Multiple trajectory data arises in two broad situations: 1) episodic tasks where a single system is reset to an initial state after a finite run time, as encountered in iterative learning control and reinforcement learning problems [15] and 2) collecting data from multiple identical systems in parallel, for example, physical experiments [16] and snapshots of social interaction processes [17]. For multiple trajectory data the duration of each trajectory sample may be small, but a large sample size can be obtained by virtue of repetition in the case of episodic tasks and parallel execution in the case of multiple identical systems. Thus, there is a growing interest in system identification based on multiple trajectory data, along with their applications in machine learning literature [18], [19].

[1]Y. Xing is with Key Lab of Systems and Control, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, and School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing, P. R. China, yxing@amss.ac.cn. His work was supported by National Key R&D Program of China (2016YFB0901900) and National Natural Science Foundation of China (61573345).

[2]B. Gravell and T. Summers are with Department of Mechanical Engineering, The University of Texas at Dallas, Richardson, TX, USA, {Benjamin.Gravell,Tyler.Summers}@utdallas.edu. Their work was supported by the Air Force Office of Scientific Research under award number FA2386-19-1-4073.

[3]X. He and K. H. Johansson are with Division of Decision and Control Systems, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden, {xingkang, kallej}@kth.se. Their work was supported by Knut & Alice Wallenberg Foundation, and Swedish Research Council.

In this paper we consider linear system identification with multiplicative noise from multiple trajectory data. Our contributions are two-fold:

- We propose a least-squares estimation algorithm to jointly estimate the nominal system matrices and multiplicative noise covariances from sample averages of multiple finite-horizon trajectory rollouts (Algorithm 1). A two-stage algorithm based on first and second moment dynamics that separate the nominal parameters from the noise variances is utilized, where a stochastic input design, from Gaussian and Wishart distributions, is used for exciting the moment dynamics. The algorithm does not need prior knowledge for the multiplicative noise or stability conditions for the system, except that the noises are i.i.d. among different trajectories, with finite first and second moments so it may be applied to a wide range of scenarios.
- The asymptotic consistency of our proposed algorithm is demonstrated. First, it is shown that dynamics defined by the first and second moments of states can generate a well-defined closed-form expression of the parameters, provided sufficiently exciting input sequences and certain controllability conditions hold (Theorems 1 and 2). Then by assuming the multiple trajectory data are i.i.d., the consistency of the estimator, i.e., convergence to the true value as the number of trajectory samples grows to infinity, is obtained by combining the former result and the law of large numbers (Theorem 3).

The remainder of the paper is organized as follows: we formulate the problem in Section II, then in Section III the algorithm is introduced and theoretical results are given, numerical simulation results are presented in Section IV, and in Section V we conclude. All proofs are omitted here and can be found in [20].

Notation. We denote the n-dimensional Euclidean space by $\mathbb{R}^n$, and the set of $n \times m$ real matrices by $\mathbb{R}^{n\times m}$. We use $\|\cdot\|$ to denote the Euclidean norm for vectors and the Frobenius norm for matrices. The expectation of a random vector X is represented by $\mathbb{E}\{X\}$. The Kronecker product of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is represented by $A \otimes B$, and the vectorization of A is represented by $\mathrm{vec}(A) = (a_{11}\ a_{21}\ \cdots\ a_{m1}\ a_{12}\ a_{22}\ \cdots\ a_{mn})^\intercal$. For a block matrix

$$B = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1n} \\ \vdots & \vdots & & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mn} \end{bmatrix} \in \mathbb{R}^{mp\times nq},$$

where $B_{ij} \in \mathbb{R}^{p\times q}$, we define the following matrix reshaping operator $F : \mathbb{R}^{mp\times nq} \to \mathbb{R}^{mn\times pq}$:

$$\begin{aligned} F(B,m,n,p,q) :=& [\mathrm{vec}(B_{11})\ \ \mathrm{vec}(B_{21})\ \ \cdots\ \ \mathrm{vec}(B_{m1})\ \cdots \\ & \mathrm{vec}(B_{12})\ \ \mathrm{vec}(B_{22})\ \ \cdots\ \ \mathrm{vec}(B_{mn})]^\intercal. \end{aligned}$$

Then we have that $F(A\otimes A, m, n, m, n) = \mathrm{vec}(A)\,\mathrm{vec}(A)^\intercal$ for $A \in \mathbb{R}^{m\times n}$, which demonstrates the relation between the entries of $A \otimes A$ and those of $\mathrm{vec}(A)\,\mathrm{vec}(A)^\intercal$. Note when $p = q = 1$, $F(\cdot)$ degenerates to $\mathrm{vec}(\cdot)$.

II. Problem Formulation

We consider linear systems with multiplicative noise

$$x_{t+1} = (A + \bar{A}_t)x_t + (B + \bar{B}_t)u_t \tag{1}$$

where $x_t \in \mathbb{R}^n$ is the system state and $u_t \in \mathbb{R}^m$ is the control input to be designed. The dynamics are described by a nominal dynamics matrix $A \in \mathbb{R}^{n\times n}$ and nominal input matrix $B \in R^{n\times m}$ and incorporate multiplicative noise terms modeled by the i.i.d. and mutually independent random matrices $\bar{A}_t$ and $\bar{B}_t$ which have zero mean and covariance matrices $\Sigma_A := \mathbb{E}\{\mathrm{vec}(\bar{A}_t)\,\mathrm{vec}(\bar{A}_t)^T\} \in \mathbb{R}^{n^2\times n^2}$ and $\Sigma_B := \mathbb{E}\{\mathrm{vec}(\bar{B}_t)\,\mathrm{vec}(\bar{B}_t)^T\} \in \mathbb{R}^{nm\times nm}$. Note that if $\bar{A}_t$ has non-zero mean $\bar{A}$, then we can consider a system with nominal matrix $(A + \bar{A}, B)$, as well as noise terms $\bar{A}_t - \bar{A}$ and $\bar{B}_t$, which satisfies the above zero-mean assumption. This also holds for cases with $\bar{B}_t$ non-zero mean. The term multiplicative noise refers to the fact that noises $\bar{A}_t$ and $\bar{B}_t$ enter the system as multipliers of x_t and u_t, rather than as additions. In the latter case, the noises are called additive ones, resulting in much simpler system dynamics.

As an example of system (1), consider the following system studied in the optimal control literature [8], [10].

$$x_{t+1} = (A + \sum_{i=1}^{r} A_i p_{i,t})x_t + (B + \sum_{j=1}^{s} B_j q_{j,t})u_t, \tag{2}$$

where $\{p_{i,t}\}$ and $\{q_{i,t}\}$ are mutually independent i.i.d. scalar random variables, with $\mathbb{E}\{p_{i,t}\} = \mathbb{E}\{q_{j,t}\} = 0$, $\mathbb{E}\{p_{i,t}^2\} = \sigma_i^2$, and $\mathbb{E}\{q_{j,t}^2\} = \delta_j^2$, $\forall i \in [1,r], j \in [1,s], t \geq 0$. It can be seen that $\bar{A}_t = \sum_{i=1}^r A_i p_{i,t}$ and $\bar{B}_t = \sum_{j=1}^s B_j q_{j,t}$ where σ_i and δ_j are the eigenvalues of Σ_A and Σ_B, and A_i and B_j are the reshaped eigenvectors of Σ_A and Σ_B. These parameters are necessary for optimal controller design, as [10] showed. However, for new systems with unknown parameters, the key problem is to identify them in the first place, stated as follows. Another example of system (1) is interconnected systems, where the nominal part captures relations among different subsystems, and multiplicative noises characterize randomly varying topologies [21].

Problem. Suppose that the system parameters A, B, Σ_A, and Σ_B are unknown, but state-input trajectories are available for system identification. Our goal in this paper is to estimate A, B, Σ_A, and Σ_B based on multiple trajectory data $\{x_t^{(k)}, 0 \leq t \leq \ell, k \in \mathbb{N}^+\}$, by appropriately designing the input sequence $\{u_t^{(k)}, 0 \leq t \leq \ell - 1, k \in \mathbb{N}^+\}$ and initial states $x_0^{(k)}$, where $\{x_t^{(k)}, 0 \leq t \leq \ell\}$, is the k-th trajectory sample, and ℓ is the final time-step for every trajectory.

III. Least-Squares Algorithm Based on Multiple Trajectory Data

A. Algorithm Design

In this section, we propose our exploratory input sequence design and least-squares algorithm to estimate the system parameters from multiple trajectory data. We assume that the sampled trajectory data are collected independently, and refer to each trajectory sample as a *rollout*. Because every rollout

is affected by the multiplicative noise, we will use least-squares on the first and second moment dynamics averaged over multiple trajectories to solve the system identification problem. Also, we assume inputs of arbitrary magnitude may be executed perfectly.

Taking the expectation of both sides of (1) we obtain the first-moment dynamics of states, i.e., the dynamics of $\mathbb{E}\{x_t\}$,

$$\mu_{t+1} = A\mu_t + B\nu_t, \tag{3}$$

where $\mu_t := \mathbb{E}\{x_t\}$ and $\nu_t := \mathbb{E}\{u_t\}$.

Likewise, denote the vectorization of the instantaneous second moment matrices of state, state-input, and input at time t by $X_t := \mathrm{vec}(\mathbb{E}\{x_t x_t^\intercal\})$, $W_t := \mathrm{vec}(\mathbb{E}\{x_t u_t^\intercal\})$, and $U_t := \mathrm{vec}(\mathbb{E}\{u_t u_t^\intercal\})$. Note that the second moment matrix we used here, namely $\mathbb{E}\{XY^\intercal\}$ for two random vectors X and Y, is different from the covariance matrix, which is $\mathbb{E}\{(X-\mathbb{E}\{X\})(Y-\mathbb{E}\{Y\})^\intercal\} = \mathbb{E}\{XY^\intercal\} - \mathbb{E}\{X\}\mathbb{E}\{Y\}^\intercal$.

From the independence of $\bar{A}_t$ and $\bar{B}_t$, as well as vectorization, the second moment dynamics of (1) are

$$\begin{aligned} X_{t+1} &= (A\otimes A)X_t + (B\otimes A)W_t + (A\otimes B)W_t^\intercal \\ &\quad + (B\otimes B)U_t + \mathbb{E}\left\{(\bar{A}_t\otimes\bar{A}_t)\,\mathrm{vec}(x_t x_t^\intercal)\right\} \\ &\quad + \mathbb{E}\left\{(\bar{B}_t\otimes\bar{B}_t)\,\mathrm{vec}(u_t u_t^\intercal)\right\} \\ &= (A\otimes A + \Sigma'_A)X_t + (B\otimes B + \Sigma'_B)U_t \\ &\quad + (B\otimes A)W_t + (A\otimes B)W_t^\intercal \end{aligned} \tag{4}$$

where we denote $\Sigma'_A = \mathbb{E}\{\bar{A}_t\otimes\bar{A}_t\} \in \mathbb{R}^{n^2\times n^2}$ and $\Sigma'_B = \mathbb{E}\{\bar{B}_t\otimes\bar{B}_t\} \in \mathbb{R}^{n^2\times m^2}$. The relation between (Σ_A, Σ_B) and (Σ'_A, Σ'_B) can be illustrated by $F(\Sigma'_A, n, n, n, n) = \Sigma_A$ and $F(\Sigma'_B, n, m, n, m) = \Sigma_B$, where the reshaping operator $F(\cdot)$ is defined in the notation section.

The first and second moment dynamics (3) and (4) are linear in the dynamic model parameters to be estimated. It is natural to consider a two-stage least-squares procedure, where first the nominal system matrices (A,B) are estimated from (3), and then these estimates are plugged in to obtain estimates for the variances (Σ_A, Σ_B) from (4). If we had access to the exact first and second moment dynamics, this procedure would produce exact estimates.However, we must estimate the first and second moment dynamics from rollout data, and we propose to take a sample average over multiple independent rollouts. To obtain persistently exciting inputs, we randomly generate the first and second moment of the input sequence from standard Gaussian and Wishart[1] distributions [22], respectively. Likewise, the initial states are assumed to be randomly drawn from a distribution $\mathcal{X}$ with finite second moment (see Sec. III-B.2). The overall algorithm is shown in Algorithm 1, where the superscript (k) represents the k-th rollout.

Remark 1: Since Σ_A and Σ_B are the covariance matrices of $\bar{A}_t$ and $\bar{B}_t$ they must be positive semidefinite. Hence a positive semidefinite constraint must be imposed on the optimization problem in line 15 of Algorithm 1 which can be easily achieved by generic convex optimization parser-solvers such as CVX in MATLAB [23]. Note that Σ'_A and Σ'_B are related to Σ_A and Σ_B via one-to-one mappings by inverse of the $F(\cdot)$ operator. However, if the estimator is consistent (as we will prove later it is), then as the amount of sample data increases the estimated covariances will become arbitrarily close to the true values and the semidefinite constraint will naturally become ineffective.

[1]The Wishart distribution $W_p(V, n)$ is the probability distribution of the matrix $X = GG^\intercal$ where each column of the matrix G is drawn from the p-variate Gaussian distribution $\mathcal{N}_p(0, V)$. Clearly Wishart distributions are supported on the set of positive semidefinite matrices.

Algorithm 1
Multiple-trajectory averaging least-squares (MALS)

1: **for** t from 0 to ℓ **do**
2: Generate ν_t and $\bar{U}_t$ independently from zero-mean Gaussian and Wishart [22] distributions, respectively. Both ν_t and $\bar{U}_t$ are fixed after generation
3: **end for**
4: **for** k from 1 to n_r **do**
5: Generate $x_0^{(k)}$ independently from the distribution $\mathcal{X}$
6: **for** t from 0 to $\ell-1$ **do**
7: Generate $u_t^{(k)}$ independently from the Gaussian distribution $\mathcal{N}(\nu_t, \bar{U}_t)$
8: $x_{t+1}^{(k)} = (A+\bar{A}_t^{(k)})x_t^{(k)} + (B+\bar{B}_t^{(k)})u_t^{(k)}$
9: **end for**
10: **end for**
11: **for** t from 0 to ℓ **do**
12: Compute

$$\begin{aligned} \hat{\mu}_t &:= \frac{1}{n_r}\sum_{k=1}^{n_r} x_t^{(k)}, \\ \hat{X}_t &:= \frac{1}{n_r}\,\mathrm{vec}\left(\sum_{k=1}^{n_r} x_t^{(k)}(x_t^{(k)})^\intercal\right), \\ \hat{W}_t &:= \frac{1}{n_r}\,\mathrm{vec}\left(\sum_{k=1}^{n_r} x_t^{(k)}\nu_t^\intercal\right) = \mathrm{vec}(\hat{\mu}_t\nu_t^\intercal), \\ U_t &:= \mathrm{vec}(\bar{U}_t + \nu_t\nu_t^\intercal) \end{aligned}$$

13: **end for**
14: $(\hat{A}, \hat{B}) = \mathrm{argmin}_{(A,B)}\{\frac{1}{2}\sum_{t=0}^{\ell-1}\|\hat{\mu}_{t+1} - (A\hat{\mu}_t + B\nu_t)\|_2^2\}$
15: $(\hat{\Sigma}'_A, \hat{\Sigma}'_B) = \underset{\Sigma'_A\succeq 0, \Sigma'_B\succeq 0}{\mathrm{argmin}}\{\frac{1}{2}\sum_{t=0}^{\ell-1}\|\hat{X}_{t+1} - [(\hat{A}\otimes\hat{A})\hat{X}_t + (\hat{B}\otimes\hat{A})\hat{W}_t + (\hat{A}\otimes\hat{B})\hat{W}_t^\intercal + (\hat{B}\otimes\hat{B})U_t + \Sigma'_A\hat{X}_t + \Sigma'_B U_t]\|_2^2\}$
16: $\hat{\Sigma}_A = F(\hat{\Sigma}'_A, n, n, n, n)$, $\hat{\Sigma}_B = F(\hat{\Sigma}'_B, n, m, n, m)$

B. Theoretical Consistency Analysis

In this section we analyze the consistency of Algorithm 1 by investigating the moment dynamics (3) and (4), which motivated the least-squares approach in Algorithm 1.

1) Moment Dynamics: Note again if we know μ_t and X_t, then it is possible to recover the parameters via least-squares

as in lines 14-15 in Algorithm 1. Let

$$\begin{aligned}\mathbf{Y} &:= \begin{bmatrix}\mu_\ell & \cdots & \mu_1\end{bmatrix}, \quad \mathbf{Z} := \begin{bmatrix}\mu_{\ell-1} & \cdots & \mu_0 \\ \nu_{\ell-1} & \cdots & \nu_0\end{bmatrix}, \\ \mathbf{C} &:= \begin{bmatrix}C_\ell & \cdots & C_1\end{bmatrix}, \quad \mathbf{D} := \begin{bmatrix}X_{\ell-1} & \cdots & X_0 \\ U_{\ell-1} & \cdots & U_0\end{bmatrix},\end{aligned} \tag{5}$$

where $C_t = X_t - [(A \otimes A)X_{t-1} + (B \otimes A)W_{t-1} + (A \otimes B)W_{t-1}^\intercal + (B \otimes B)U_{t-1}]$, $1 \le t \le \ell$. Then closed-form solutions of the least-squares problems are

$$\begin{aligned}(\hat{A}, \hat{B}) &= \mathbf{Y}\mathbf{Z}^\intercal(\mathbf{Z}\mathbf{Z}^\intercal)^\dagger, \\ (\hat{\Sigma}'_A, \hat{\Sigma}'_B) &= \mathbf{C}\mathbf{D}^\intercal(\mathbf{D}\mathbf{D}^\intercal)^\dagger,\end{aligned}$$

where $\mathbf{C}$, $\mathbf{D}$, $\mathbf{Y}$, and $\mathbf{Z}$ are defined in (5) above, and the sign $\dagger$ represents the pseudoinverse. When the inverse matrices exist, the solutions are identical to true values, that is, $(\hat{A}, \hat{B}) = (A, B)$ and $(\hat{\Sigma}'_A, \hat{\Sigma}'_B) = (\Sigma'_A, \Sigma'_B)$. Hence, the first question towards the consistency of the algorithm is whether the matrices $\mathbf{Z}\mathbf{Z}^\intercal$ and $\mathbf{D}\mathbf{D}^\intercal$ are invertible, which is necessary for the consistency of the algorithm. As to be shown, this invertibility can be obtained by designing a proper input sequence, if systems (A, B) and $(A \otimes A + \Sigma'_A, B \otimes B + \Sigma'_B)$ are controllable, and the final time-step ℓ is large enough. In fact, in this paper we randomly generalized the first and second moments of inputs to ensure the invertibility. As a consequence, we need to demonstrate the following results in a probability sense, intuitively saying that random generation of input statistics results in the expected invertibility.

Theorem 1: Suppose that $\ell \ge \frac{1}{2}mn^2 + \frac{1}{2}mn + m + 1$ and (A, B) is controllable. The matrix $\mathbf{Z}$ has full row rank with probability one, and consequently $\mathbf{Z}\mathbf{Z}^\intercal$ is invertible, if the entries of ν_t, $0 \le t \le \ell - 1$, are generated i.i.d. from a non-degenerate Gaussian distributions.

Remark 2: The above theorem shows that for large enough time step of each rollout, the full row rank condition of $\mathbf{Z}$ can be guaranteed with probability one if the mean of the input at each time step is generated randomly and independently. In the proof, the controllability of (A, B) plays a key role. In addition, although the lower bound in the theorem is relatively small, one may conjecture that $\ell \ge n + m$ is a sharp lower bound for the invertibility of $\mathbf{Z}\mathbf{Z}^\intercal$, which will be a future work.

Theorem 2: Suppose that $\ell \ge \frac{1}{2}m^2n^4 + \frac{1}{2}m^2n^2 + m^2 + 1$ and $(A \otimes A + \Sigma'_A, B \otimes B + \Sigma'_B)$ is controllable. The matrix $\mathbf{D}$ has full row rank with probability one, and consequently $\mathbf{D}\mathbf{D}^\intercal$ is invertible, if ν_t have been fixed and the entries of $\bar{U}_t$ are generated i.i.d. from a non-degenerate Wishart distributions, $0 \le t \le \ell - 1$, where $\bar{U}_t$ is defined in line 2 of Algorithm 1.

Remark 3: The controllability condition in Theorem 2 reflects the nature of the multiplicative noise, i.e., coupling between $\bar{A}_t$ and x_t, and that between $\bar{B}_t$ and u_t. It also indicates that a controllability condition on (4), the dynamics of the second moments of states, is necessary to ensure the successful identification of Σ_A and Σ_B.

Corollary 1: Suppose that $\ell \ge \frac{1}{2}m^2n^4 + \frac{1}{2}m^2n^2 + m^2 + 1$, and both (A, B) and $(A \otimes A + \Sigma'_A, B \otimes B + \Sigma'_B)$ are controllable. The matrices $\mathbf{Z}\mathbf{Z}^\intercal$ and $\mathbf{D}\mathbf{D}^\intercal$ are invertible, if first the entries of ν_t are generated i.i.d. from a non-degenerate Gaussian distribution and then $\bar{U}_t$ is generated i.i.d. from a non-degenerate Wishart distribution, $0 \le t \le \ell - 1$, where $\bar{U}_t$ is defined in line 2 of Algorithm 1.

Remark 4: From the proof of Theorems 1 and 2, we know that the existence of the inverses of $\mathbf{Z}\mathbf{Z}^\intercal$ and $\mathbf{D}\mathbf{D}^\intercal$ can in fact be guaranteed with probability one, as long as ν_t and $\bar{U}_t$, the mean and vectorized second moment matrix of the input at time t, are generated independently from a distribution that is absolutely continuous with respect to Lebesgue measure. Also note the random generation of the first and second moments of inputs leads to non-stationarity of the input sequence. Critically this provides sufficient excitation of both the first and second moments of the state and makes it possible to estimate all model parameters in the presence of multiplicative noise.

2) Consistency: After the discussion in the previous section, we now assume that the means and second moments of the input sequences have been generated in Algorithm 1, and both $\mathbf{Z}\mathbf{Z}^\intercal$ and $\mathbf{D}\mathbf{D}^\intercal$ have been designed to be invertible. The closed-form estimates generated by Algorithm 1 are

$$(\hat{A}, \hat{B}) = \hat{\mathbf{Y}}\hat{\mathbf{Z}}^\intercal(\hat{\mathbf{Z}}\hat{\mathbf{Z}}^\intercal)^\dagger, \tag{6}$$

$$(\hat{\Sigma}'_A, \hat{\Sigma}'_B) = \hat{\mathbf{C}}\hat{\mathbf{D}}^\intercal(\hat{\mathbf{D}}\hat{\mathbf{D}}^\intercal)^\dagger, \tag{7}$$

where

$$\begin{aligned}\hat{\mathbf{Y}} &:= \begin{bmatrix}\hat{\mu}_\ell & \cdots & \hat{\mu}_1\end{bmatrix}, \quad \hat{\mathbf{Z}} := \begin{bmatrix}\hat{\mu}_{\ell-1} & \cdots & \hat{\mu}_0 \\ \nu_{\ell-1} & \cdots & \nu_0\end{bmatrix}, \\ \hat{\mathbf{C}} &:= \begin{bmatrix}\hat{C}_\ell & \cdots & \hat{C}_1\end{bmatrix}, \quad \hat{\mathbf{D}} := \begin{bmatrix}\hat{X}_{\ell-1} & \cdots & \hat{X}_0 \\ U_{\ell-1} & \cdots & U_0\end{bmatrix},\end{aligned}$$

and $\hat{C}_t = \hat{X}_t - [(\hat{A} \otimes \hat{A})\hat{X}_{t-1} + (\hat{B} \otimes \hat{A})\hat{W}_{t-1} + (\hat{A} \otimes \hat{B})\hat{W}_{t-1}^\intercal + (\hat{B} \otimes \hat{B})U_{t-1}]$, $1 \le t \le \ell$. Here $\hat{A}$ and $\hat{B}$ are obtained by Algorithm 1. The estimates above depend on the number of rollouts n_r, but we omit it for convenience. Before stating the consistency result, we present the following assumption for the system and data:

Assumption 1: For all rollouts, the below conditions hold.
(i) The final time-step is fixed to be $\ell \ge \frac{1}{2}m^2n^4 + \frac{1}{2}m^2n^2 + m^2 + 1$.
(ii) The initial state $x_0^{(k)}$, $1 \le k \le n_r$, is generated independently from the same distribution with $\mathbb{E}\{\|x_0^{(k)}\|^2\} < \infty$, and is independent of the subsequent process.
(iii) $\{\bar{A}_t^{(k)}\}$ and $\{\bar{B}_t^{(k)}\}$, $0 \le t \le \ell$, $1 \le k \le n_r$, have zero mean and finite second moments, i.e., $\mathbb{E}\{\bar{A}_t\} = \mathbb{E}\{\bar{B}_t\} = \mathbf{0}$ and $\|\Sigma_A\|, \|\Sigma_B\| < \infty$. Also, these two sequences are i.i.d. and mutually independent.
(iv) The input signals are generated according to Line 6 of Algorithm 1.

Under Assumption 1 the rollouts $x_0^{(k)}, \ldots, x_l^{(k)}$, $1 \le k \le n_r$, are i.i.d., so consistency can be established from Kolmogorov's strong law of large numbers.

Theorem 3: (Consistency) Suppose that Assumption 1 holds, and both $\mathbf{Z}\mathbf{Z}^\intercal$ and $\mathbf{D}\mathbf{D}^\intercal$ are invertible. Then the estimators (6)-(7) are asymptotically consistent, i.e.,

$$(\hat{A}, \hat{B}) \to (A, B), \quad \text{and} \quad (\hat{\Sigma}'_A, \hat{\Sigma}'_B) \to (\Sigma'_A, \Sigma'_B),$$

with probability one as the number of rollouts $n_r \to \infty$.

Remark 5: This theorem indicates that despite the relatively small final time-step for each trajectory, an increasing number of rollouts compensates for this deficiency and guarantees asymptotic estimation performance.

IV. Numerical Simulations

To empirically validate our theoretical consistency result, we simulated our least-squares estimator[2] on two example systems. The first is a simple 2-state, 1-input system where we use a large amount of data to show *asymptotic trends*, while the second is an 8-state, 8-input system representing lossy diffusion dynamics on a network for a more *practical application*. Python code which implements the algorithms and performs the simulated experiments described here is available on GitHub at `https://github.com/TSummersLab/sysid-multinoise`.

A. Simple example

We consider a simple example system with $n = 2$, $m = 1$, $A = \begin{bmatrix} -0.2 & 0.3 \\ -0.4 & 0.8 \end{bmatrix}$, $B = \begin{bmatrix} -1.8 \\ -0.8 \end{bmatrix}$, and noise covariances

$$\Sigma_A = \frac{1}{100}\begin{bmatrix} 8 & -2 & 0 & 0 \\ -2 & 16 & 2 & 0 \\ 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 8 \end{bmatrix}, \Sigma_B = \frac{1}{100}\begin{bmatrix} 5 & -2 \\ -2 & 20 \end{bmatrix}.$$

We performed a simulated experiment where rollout data of length $\ell = \frac{1}{2}m^2n^4 + \frac{1}{2}m^2n^2 + m^2 + 1 = 12$, i.e., according to the bound prescribed by our theoretical result, was collected for $n_r = 10,000,000$. We used control inputs distributed as $u_t \sim \mathcal{N}(\nu_t, \bar{U}_t)$, where ν_t and $\bar{U}_t$ are generated from $\mathcal{N}(0, I_n)$ and $\mathcal{W}_n(0.1I_n, n)$, respectively, and then are fixed. Model estimates were computed at 100 increasing logarithmically spaced numbers of rollouts between 1 and n_r. This experiment was repeated 50 times and the results are plotted in Fig. 1.

It is well known that least-squares estimation of finite-impulse response (FIR) models yields estimates whose error decreases as $\mathcal{O}(N^{-1/2})$ where N is the number of samples [24]. Empirically we observed a similar trend on the convergence rate; we conjecture that $\hat{A}$ and $\hat{B}$ converge to A and B as $\mathcal{O}(n_r^{-1/2})$ while $\hat{\Sigma}_A$ and $\hat{\Sigma}_B$ converge to Σ_A and Σ_B at a slower rate, perhaps $\mathcal{O}(n_r^{-\frac{1}{2n}})$ and $\mathcal{O}(n_r^{-\frac{1}{2m}})$ respectively. This can be observed in the similarity between the dashed red reference curves and the median empirical model estimate curves. However it is difficult, even for small example systems like this one, to deduce the asymptotic convergence rate from empirical observation.

Although our consistency result implies that theoretically the estimates converge regardless of the size of the variance of the random inputs, in practice the design of the inputs has a large impact on the (transient) quality of the estimates. In particular, it is important to strike a balance between exciting the system modes and multiplicative noises and not overwhelming the state- and input-dependent noises by the (random) control inputs i.e. the magnitude of the control input means ν_t and covariances $\bar{U}_t$ should be chosen in a "sweet spot". We used values that gave good results empirically for the example system considered.

[2]For computational efficiency we solve the unconstrained least-squares problem for $\hat{\Sigma}_A$ and $\hat{\Sigma}_B$ then project onto the positive semidefinite cone; for all but the smallest numbers of rollouts this projection is ineffective.

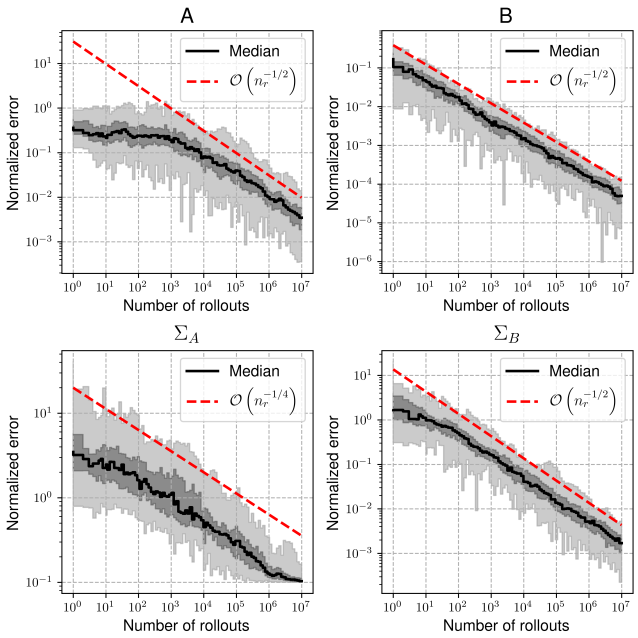

Fig. 1. Normalized error vs number of rollouts for an example system. Each subplot gives the normalized Frobenius norm error in the specified parameter matrix e.g. $\frac{\|A-\hat{A}\|}{\|A\|}$. The minimum and maximum of all experiments are bounded in the light grey region, the interquartile range in the dark grey region, and the median at the bold black line. Reference curves are plotted as dashed red lines.

B. Network example

Many practical networked systems can be approximated by diffusion dynamics with loss; examples include heat flow through uninsulated pipes, hydraulic flow through leaky pipes, information flow between processors with packet loss, electrical power flow between generators with resistant electrical power lines, etc. These dynamics in continuous-time act on an undirected graph with no self-loops with symmetric weighted adjacency matrix A_c, degree matrix $D_c = \text{diag}(A_c\mathbf{1}_{n\times 1})$, graph Laplacian $L = D_c - A_c$, diagonal loss matrix F_c, and diagonal input matrix B_c:

$$\dot{x} = -(L_c + F_c)x + B_u u \tag{8}$$

Discretizing these dynamics using the forward Euler method with a step size T yields $x_{t+1} = Ax_t + Bu_t$ where $A = I - T(L_c + F_c)$ and $B = TB_u$. Uncertainty on an edge weight of the graph i.e. on entry (j,k) of A_c manifests as a noise matrix with entries

$$[A_i]_{p,q} = \begin{cases} +1 \text{ if } \{j = p \text{ and } k = p\} \text{ or } \{j = q \text{ and } k = q\}, \\ -1 \text{ if } \{j = q \text{ and } k = p\} \text{ or } \{j = p \text{ and } k = q\}, \\ 0 \text{ otherwise.} \end{cases}$$

Uncertainty on an input strength i.e. entry (k,k) of B_u manifests as a noise matrix with entries

$$[B_j]_{p,q} = \begin{cases} +1 \text{ if } k = q = p, \\ 0 \text{ otherwise.} \end{cases}$$

For computational tractability we estimated only the noise variances while giving the estimator knowledge of the noise directions A_i and B_j. To formulate this setting mathematically it is easier to work with the eigendecomposition of the noises as in (2). The least-squares estimation for this case is a simple modification to the full covariance estimator:

$$(\hat{\sigma}^2, \hat{\delta}^2) = \hat{\mathbf{C}}\hat{\mathbf{D}}^\intercal(\hat{\mathbf{D}}\hat{\mathbf{D}}^\intercal)^\dagger, \tag{9}$$

where $(\hat{\sigma}^2, \hat{\delta}^2)$ are vectors of the noise variances and

$$\hat{\mathbf{C}} := \operatorname{vec}\big(\begin{bmatrix} \hat{C}_\ell & \cdots & \hat{C}_1 \end{bmatrix} \big),$$

$$\hat{\mathbf{D}} := \begin{bmatrix} \operatorname{vec}\big((A_1 \otimes A_1)\hat{X}_{\ell-1} & \cdots & (A_1 \otimes A_1)\hat{X}_0\big) \\ \vdots & & \\ \operatorname{vec}\big((A_r \otimes A_r)\hat{X}_{\ell-1} & \cdots & (A_r \otimes A_r)\hat{X}_0\big) \\ \operatorname{vec}\big((B_1 \otimes B_1)U_{\ell-1} & \cdots & (B_1 \otimes B_1)U_0\big) \\ \vdots & & \\ \operatorname{vec}\big((B_s \otimes B_s)U_{\ell-1} & \cdots & (B_s \otimes B_s)U_0\big) \end{bmatrix},$$

and $\hat{C}_t = \hat{X}_t - [(\hat{A} \otimes \hat{A})\hat{X}_{t-1} + (\hat{B} \otimes \hat{A})\hat{W}_{t-1} + (\hat{A} \otimes \hat{B})\hat{W}_{t-1}^\intercal + (\hat{B} \otimes \hat{B})U_{t-1}]$, $1 \le t \le \ell$. As before, $\hat{A}$ and $\hat{B}$ are obtained by Algorithm 1.

We chose a network with $n = 8$ nodes and edges placed via the Erdos-Renyi random graph generation with random integer weights. The graph was selected to be connected so that the system would be controllable. We used rollout data of length $\ell = \frac{1}{2}m^2n^4 + \frac{1}{2}m^2n^2 + m^2 + 1 = 133185$ and collected 7 rollouts; more rollouts could be used, but empirically this amount of data was sufficient to give good estimates. Table I shows the averages and maximums of the normalized noise variance estimation errors

$$\overline{\sigma}_i^2 = \frac{|\sigma_i^2 - \hat{\sigma}_i^2|}{\sigma_i^2}, \quad \text{and} \quad \overline{\delta}_j^2 = \frac{|\delta_j^2 - \hat{\delta}_j^2|}{\delta_j^2}.$$

TABLE I

$\frac{1}{r}\sum_i^r \overline{\sigma}_i^2$	$\max_i \overline{\sigma}_i^2$	$\frac{1}{s}\sum_j^s \overline{\delta}_j^2$	$\max_j \overline{\delta}_j^2$
0.0358	0.132	0.0517	0.141

V. Conclusions

In this paper we proposed a system identification scheme for linear systems with multiplicative noise based on multiple trajectory data. By designing appropriate persistently exciting input signals, a least-squares algorithm was proposed for the joint estimation of nominal system and multiplicative noise covariances. The asymptotic consistency of the algorithm was proved, and illustrated by numerical simulations. Ongoing and future research directions include studying the convergence rate and non-asymptotic behavior of the proposed algorithm, problems of optimal input design, identification from single-trajectory data, and sparsity-promoting regularization for identification of networked systems with prior knowledge of sparsity levels.

References

[1] W. M. Wonham, "Optimal stationary control of a linear system with state-dependent noise," *SIAM Journal on Control*, vol. 5, no. 3, pp. 486–500, 1967.

[2] N. E. Du Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, 2011.

[3] P. Antsaklis and J. Baillieul, "Special issue on technology of networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 5–8, 2007.

[4] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 138–162, 2007.

[5] Y. Guo and T. H. Summers, "A performance and stability analysis of low-inertia power grids with stochastic system inertia," in *2019 American Control Conference, ACC 2019, Philadelphia, PA, USA, July 10-12, 2019*. IEEE, 2019, pp. 1965–1970. [Online]. Available: http://ieeexplore.ieee.org/document/8814402

[6] J. L. Lumley, *Stochastic Tools in Turbulence*. Courier Corporation, 2007.

[7] M. Breakspear, "Dynamic models of large-scale brain activity," *Nature neuroscience*, vol. 20, no. 3, p. 340, 2017.

[8] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Siam, 1994, vol. 15.

[9] D. Kleinman, "Optimal stationary control of linear systems with control-dependent noise," *IEEE Transactions on Automatic Control*, vol. 14, no. 6, pp. 673–677, 1969.

[10] B. Gravell, P. M. Esfahani, and T. Summers, "Learning robust control for LQR systems with multiplicative noise via policy gradient," arXiv:1905.13547, 2019.

[11] H.-F. Chen and L. Guo, *Identification and Stochastic Adaptive Control*. Springer Science & Business Media, 2012.

[12] J. Duník, O. Straka, O. Kost, and J. Havlík, "Noise covariance matrices in state-space models: A survey and comparison of estimation methods, part I," *International Journal of Adaptive Control and Signal Processing*, vol. 31, no. 11, pp. 1505–1543, 2017.

[13] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Automatica*, vol. 47, no. 1, pp. 39–49, 2011.

[14] G. Kitagawa, "A self-organizing state-space model," *Journal of the American Statistical Association*, pp. 1203–1215, 1998.

[15] N. Matni, A. Proutiere, A. Rantzer, and S. Tu, "From self-tuning regulators to reinforcement learning and back again," *arXiv preprint arXiv:1906.11392*, 2019.

[16] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3389–3396.

[17] M. T. Schaub, S. Segarra, and J. N. Tsitsiklis, "Blind identification of stochastic block models from dynamical observations," *arXiv preprint arXiv:1905.09107*, 2019.

[18] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "On the sample complexity of the linear quadratic regulator," *arXiv preprint arXiv:1710.01688*, 2017.

[19] N. Matni and S. Tu, "A tutorial on concentration bounds for system identification," *arXiv preprint arXiv:1906.11395*, 2019.

[20] Y. Xing, B. Gravell, X. He, K. H. Johansson, and T. Summers, "Linear system identification under multiplicative noise from multiple trajectory data," *arXiv preprint arXiv:2002.06613*, 2020.

[21] A. Haber and M. Verhaegen, "Subspace identification of large-scale interconnected systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2754–2759, 2014.

[22] A. K. Gupta and D. K. Nagar, *Matrix Variate Distributions*. Chapman and Hall/CRC, 2018.

[23] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[24] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1986.