Abstract
A data-driven identification of dynamical systems requiring only minimal prior knowledge is promising whenever no analytically derived model structure is available, e.g., from first principles in physics. However, meta-knowledge on the system’s behavior is often given and should be exploited: Stability as fundamental property is essential when the model is used for controller design or movement generation. Therefore, this paper proposes a framework for learning stable stochastic systems from data. We focus on identifying a state-dependent coefficient form of the nonlinear stochastic model which is globally asymptotically stable according to probabilistic Lyapunov methods. We compare our approach to other state of the art methods on real-world datasets in terms of flexibility and stability.

1. Introduction
An accurate identification of the system dynamics is the first and very crucial step to many modern control methods. Although reinforcement learning also allows model-free search for optimal policies, it is known to be less efficient and difficult to analyze. Therefore, classical control engineers employ system identification techniques to obtain parametric model descriptions of dynamical systems from observation data, e.g., in the linear case ARX and ARMAX models. The identification focuses on model selection, i.e., finding the model structure and the corresponding set of parameters. But often this set of model candidates is difficult to find, especially for complex, possibly non-deterministic, systems (Ljung, 1998). Therefore, the need for data-driven models has emerged recently as control engineering is increasingly applied in areas without analytic description of the dynamical system. We consider the following two application scenarios: First, assume a set of trajectories for a robotic task is given through human demonstrations, e.g., object grasping. The goal is to represent the motion with a dynamical system. To ensure the reproduction terminates at the desired final point (object to grasp), we introduce the stability constraint. Second, consider a dynamical system which is known to be stable, e.g., a pendulum which rests in hanging position. The goal is to identify the dynamics precisely without further physical insights.

Bayesian non-parametric methods, more particularly Gaussian Processes (GPs) where successfully employed by Kocijan et al. (2005) and Wang et al. (2005) for system identification. Other approaches focus on learning switching linear systems (Fox et al., 2009) or employ an EM algorithm (Ghahramani & Roweis, 1999) for nonlinear systems. However, these approaches neglect the prior assumption that the dynamical system is stable, which becomes crucial when the learned model is used as a generative process such as in movement generation for robotics (Ijspeert et al., 2002). If stability is not considered during learning, the identified model suffers from spurious attractors which are not part of the true dynamics or instability.

Only little work has merged the extensive knowledge on stability theory from control engineering with the powerful data-driven approaches for system identification: For example Boots et al. (2008) and Chiuso & Pillonetto (2010) take stability constraints for learning dynamical systems into account but are limited to linear systems. The work by Khansari-Zadeh & Billard (2011) ensures stability of the system by constraining the optimization of a Gaussian Mixture Model (GMM) to stability conditions derived from Lyapunov methods. The work by Paraschos et al. (2013) relies on a phase variable to ensure stability, which makes the approach time-dependent and therefore less robust. Control Lyapunov functions are used by Khansari-Zadeh & Billard (2014) to ensure global stability for the learned system. These approaches partially employ probabilistic models (GP, GMM), but limit the analysis to the deterministic part by only considering the mean regression. By discarding the true underlying probability distribution, information regarding reliability of the model provided by the data is lost. This leads to overconfident conclusions regarding performance or safety on the real system.

Therefore, this work proposes a framework for learning
probabilistic nonlinear dynamical systems from observation, which takes the prior assumption of stability into account. The required stochastic stability conditions of the discrete-time Markov processes are derived from Lyapunov theory. We provide simulation results to validate the proposed approach and compare it to previously mentioned methods for identifying dynamical systems.

2. Problem Formulation

We consider an autonomous, dynamical, discrete-time system with continuous-valued state $x_k \in \mathcal{X} = \mathbb{R}^d$. The state evolves according to an unknown stochastic process:

$$x_{k+1} = \tilde{f}(x_k, \omega_k),$$

with initial value $x_0 \in \mathcal{X}$ and $\omega_k$ is a random variable from the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with sample space $\Omega$, the corresponding $\sigma$-algebra $\mathcal{F}$ and the probability measure $\mathbb{P}$. Since $x_k \in \mathcal{X}$ is fixed at each step, (1) describes a state-dependent distribution over $x_{k+1}$. A realization of $\omega_k \in \Omega$, is drawn at every time step, yielding a realization of the next step. As the distribution for $x_{k+1}$ only depends on the state at time step $k$, $\tilde{f}$ is a Markov process, denoted by $\{x_k\}$.

We assume that consecutive measurements of the state are available, thus $N$ data pairs are given in the trainings set $\mathcal{D} = \{\bar{x}_n, x_{n+1}\}_{n=1}^N$. Based on these measurements, we model the unknown dynamics $\tilde{f}$ including the distribution $\omega_k$ using the prior knowledge, that the stochastic process (1) converges to the origin $x_k = 0$. The model consists of the mapping $f_\psi$ and an encoding of the random variable $\omega$ defined by a finite parameter vector $\psi \in \Psi$. As the model $f_\psi$ must best possibly explain the data $\mathcal{D}$, the problem is formulated as constrained likelihood maximization:

$$\psi^* = \arg \max_{\psi} \sum_{n=1}^N \log \mathbb{P}(\bar{x}_{n+1}|\bar{x}_n, f_\psi),$$

(2a)

s.t. $\{x_k\}$ converges to the origin for $k \to \infty$.  

(2b)

As different stochastic stability concepts exist, the convergence in (2b) is defined as convergence with probability one (w.p.1) (Kushner, 1971):

**Definition 1** (Convergence w.p.1). $\{x_k\}$ converges to the origin w.p.1 if for each $\epsilon > 0$, $\|x_k\| \geq \epsilon$ only finitely often.

---

**Notation:** Bold symbols denote vectors or multivariate functions, capital letters matrices and $I_p$ the $p \times p$ identity matrix. $A > 0$ denotes positive definiteness of the matrix $A$, $\mathbb{E}[\cdot]$ the expected value, $\mathbb{V}[\cdot]$ the variance of a random variable and $\mathbb{C}[\cdot, \cdot]$ the covariance between two random variables, where $\mathbb{C}[a] = \mathbb{C}[a, a]$. $X$ denotes a realization of the random variable $\mathcal{X}$. Imitating Matlab indexing, $A(i, :)$ denotes the $i$-th column, $A(j, :)$ the $j$-th row and $A(1, 2, :)$ the first and second element in the $i$-the column of $A$. The $i$-th entry of the vector $x_k$ is denoted $x_{k,i}$.

This also implies the following type of convergence, which might be more intuitive to the reader.

**Definition 2** (Convergence in probability). The chain $\{x_k\}$ converges to the origin in probability if $\mathbb{P}(\|x_k\| \geq \epsilon) \to 0$, for each $\epsilon > 0$.

We do not consider any control input here, thus the identification takes place for the closed-loop system for an existing controller or an uncontrolled system.

3. Stability Conditions for the Model

3.1. The Model

Consider the state-dependent coefficient form of $f_\psi$

$$x_{k+1} = A(x_k)x_k,$$

(3)

where, for a fixed $x_k$, $A$ is a random variable from the probability space $(\Omega_A, \mathcal{F}_A, \mathbb{P}_A)$ with the sample space $\Omega_A \subset \mathbb{R}^{d \times d}$. The probability density function of $A$ is specified by the vector $\theta \in \Theta$, which is state dependent through $\theta_\psi : \mathcal{X} \to \Theta$. This mapping is itself parametrized by a vector $\psi$. At each step, a realization of $A$, denoted by $\hat{A}$, is drawn and multiplied by the state $x_k$ to proceed by one step. This is visualized in Figure 1 along with the two-layer model structure: The first layer maps current state $x_k \in \mathcal{X}$ onto the parameter $\theta \in \Theta$, denoted by $\theta_\psi : \mathcal{X} \to \Theta$. The mapping is parametrized by $\psi$. The second layer is the probability distribution on $A$ which assigns to each element in the sample space $\Omega_A$ a probability based on $\theta$.

**Figure 1.** Illustration of the two layer design of the proposed model and the process of simulation.

To illustrate this multilayer design, we give a brief example in the scalar case $d = 1$: Assume $A(x_k)$ follows, for a given $x_k$, a Gaussian distribution $A \sim \mathcal{N}(\mu, \sigma)$. Therefore, the parameter vector is $\theta_\psi = [\mu, \sigma]$, with $\mu \in \mathbb{R}, \sigma \in \mathbb{R}^+$, thus $\Theta_\psi \subset \mathcal{R} \times \mathcal{R}$. The dependency of these parameters on the current state $x_k$ is expressed in $\theta_\psi(x_k)$, e.g.,

$$\begin{bmatrix} \mu(x_k) \\ \sigma(x_k) \end{bmatrix} = \theta_\psi(x_k) = \begin{bmatrix} \omega x_k^2 \\ z x_k^2 \end{bmatrix},$$

(4)
where linear dependency of the mean on the state and a quadratic relation between variance and the state is assumed. The parameters defining $\theta^\psi_X$ here are $\psi = [w z]^T$. Generally, the first layer $\theta^\psi : \mathcal{X} \rightarrow \Theta$ can be any state of the art parametric regression method which is parametrized by $\psi$. For layer two, any probability distribution with a fixed set of parameters is applicable for $A$.

Leaving the stochastic aspect aside, model (3) is the state-dependent coefficient (SDC) form which is reached by factorizing a nonlinear system into a linear-like structure. It was shown, that for any continuous differentiable function $f$ with $f(0) = 0$, there exists a matrix-valued function $A(x)$ such that $f(x) = A(x)x$, see Cimen (2008). Thus, the SDC form is not limiting the expressive power of our model. It also reflects the setup of many real-world systems, e.g., consider an actuator whose output is generally noisy. Let it also encode also nonlinear dynamics. By modeling the temperature as a state, the model (3) allows to capture this varying precision of the actuator.

The structure of the model (3) combines two important criteria. First, it provides more flexibility than a linear system with random parameters, so it encodes also nonlinear dynamics. Second, it is simple enough to allow a quadratic Lyapunov function analysis and therefore the derivation of analytic constraints for convergence as needed for the optimization in (2).

### 3.2. Stability Analysis

For approaching the problem as formulated in Section 2, an analytic condition for the constraint in the optimization problem (2b), given that $f_\psi$ is of the form (3), is needed. The literature on stability criteria for dynamical systems is very rich and for nonlinear systems Lyapunov type methods are often used. They are based on the following idea: If there is a function representing the “energy” in the system (called Lyapunov function) which constantly decreases over time, the state will converge to “zero energy” state, the origin. More precise, the Lyapunov function must be positive definite and it must be strictly decreasing over time, except in the origin. Using the stochastic discrete-time version of Lyapunov methods and the Borel-Cantelli Lemma leads to the following conditions for exponential stability (which implies convergence w.p.1 as defined in Definition 1).

**Theorem 1** (Exponential Stability, (Kushner, 1971)).

*Given a positive definite function $V(x_k) \geq 0$ for which*

\[
\mathbb{E}[V(x_{k+1})|x_k] - V(x_k) \leq -\alpha V(x_k), \forall x_k \in \mathcal{X} \setminus 0, \tag{5}
\]

*for some $\alpha > 0$ then*

\[
\mathbb{E}[V(x_{k+m})|x_k] \leq (1-\alpha)^m V(x_k) \quad and \tag{6}
\]

\[
V(x_{k+m}) \rightarrow 0 \text{ for } m \rightarrow \infty (\text{w.p.1}). \tag{7}
\]

For the class of systems (3) a quadratic function $V(x_k)$ is a proper Lyapunov function to derive sufficient stability constraints for arbitrary distributions on $A$ as shown in the following proposition:

**Proposition 1** (Stability of the model (3)). Consider a stochastic process generated from (3) where in each step of realization of $A$ is drawn from sample space $\Omega_A \subset \mathbb{R}^{d \times d}$. The process is globally exponentially stable at $x_k = 0$ if there exists a $P > 0$ such that

\[
\mathbb{E}[A^T(x_k)] P \mathbb{E}[A(x_k)] + Q - (1-\alpha)P \preceq 0, \forall x_k \in \mathcal{X}, \tag{8}
\]

*for some $\alpha > 0$, where $Q$ is defined as*

\[
Q_{i,j}(x_k) = \sum_i P_{(i,:)} C [A_{(i,i)}(x_k), A_{(l,j)}(x_k)], \tag{9}
\]

*for any $x_0 \in \mathcal{X}$.*

**Proof.** Considering a quadratic Lyapunov function $V(x_k) = x_k^T P x_k$ with $P \succeq 0$, the inequality from Theorem 1 in (5) is given as

\[
\mathbb{E}[x_{k+1}^T P x_{k+1}|x_{k+1}] - x_k^T P x_k \leq -\alpha x_k^T P x_k, \tag{10}
\]

which yields for the stochastic process $x_{k+1} = A(x_k)x_k$

\[
x_k^T \mathbb{E}[A^T] P \mathbb{E}[A] x_k + \text{Tr}(P \mathbb{E}[A x_k]) - (1-\alpha) x_k^T P x_k \leq 0, \forall x_k \in \mathcal{X}. \tag{11}
\]

Now, an expression for the trace is derived as follows

\[
\text{Tr}(P \mathbb{E}[A x_k]) = \text{Tr}(P \mathbb{E} \left[ \sum_i A_{(i,i)} x_{k,i} \right])
= \text{Tr}(P \sum_{i,j} A_{(i,i)} A_{(i,j)} x_{k,i} x_{k,j})
= \sum_{i,j} P_{(i,:)} C [A_{(i,i)}, A_{(j,j)}] x_{k,i} x_{k,j} = x_k^T Q x_k
\]

where definition of $Q$ in (9) was substituted. Using this simplification, (10) is rewritten as

\[
x_k^T \left( \mathbb{E}[A^T] P \mathbb{E}[A] + Q - (1-\alpha)P \right) x_k \leq 0,
\]

which must hold for $\forall x_k \in \mathcal{X}$. To ensure this, the matrix $\mathbb{E}[A^T] P \mathbb{E}[A] + Q - (1-\alpha)P$ must be negative semi-definite, which concludes the proof. \(\square\)
The interpretation of Proposition 1 is analogue to the linear deterministic case $x_{k+1} = Ax_k$ which is stable if there exists a matrix $P$ for which $A^TPA - P < 0$: In the nonlinear case in (3) the negative definiteness must be fulfilled for $A(x_k)$, $\forall x_k \in \mathcal{X}$. The probabilistic nature of the system (3) in addition requires "a buffer", which here is $Q$. The deterministic case is reconstructed if $A$ has zero variance. The scalar case, considered in the following remark, also allows an intuitive insight to the Proposition 1: There is a trade-off between the magnitude of the expected value and the variance of $A$ as follows:

**Remark 1.** In the scalar case\(^3\), i.e. $d = 1$ in (3), with $Q = P \mathbb{V} [A(x_k)]$ condition (8) simplifies for any $P > 0$ to

$$\mathbb{E} [A(x_k)]^2 + \mathbb{V} [A(x_k)] \leq 1 - \alpha, \ \forall x_k \in \mathcal{X}. \quad (11)$$

### 4. Stable Learning with Various Distributions

Our learning framework consists of three major steps:

1. Chose any probability distribution for the random variable $A$ in (3) which is given by a fixed set of parameters $\theta \in \Theta$ and whose first two moments are available. It is assumed that subset $\Theta^* \subseteq \Theta$ for which (8) is fulfilled is non-empty, thus $\Theta^* \neq \emptyset$.

2. Chose any parametric regression method to represent the mapping $\theta^\psi : \mathcal{X} \to \Theta$. The parameters of this mapping are denoted by $\psi \in \Psi$. The set of all $\psi$ for which all $x_k \in \mathcal{X}$ map to $\Theta^*$ is denoted by $\Psi^*$.

3. The likelihood maximization under constraints

$$\psi^* = \arg \max_{\psi \in \Psi^*} \sum_{n=1}^{N} \log \mathcal{P} (x_{n+1}|x_n, \theta^\psi), \quad (12)$$

is solved, where $\psi \in \Psi^*$ is equivalent to constraint (8) with $P > 0$ and $\alpha > 0$.

The optimization (12) is a general constrained nonlinear program in a rather high dimensional space (depending on number of parameters of the regression method in step 2). However, independent of the optimality, the model $f_{\psi^*}$ of the form (3) is exponentially stable, thus any sample path of the system converges. For computational simplicity, we focus on two types of distribution which naturally fulfill the constraints as explained in the next sections.

#### 4.1. Stability with Beta Distribution

For certain choices of distributions, constraint (8) is fulfilled for all possible parameter $\theta$, thus $\Theta^* = \Theta$, which makes the optimization unconstrained. One example of such a distribution is the Beta distribution as given in the following corollary.

**Corollary 1.** The scalar system $x_{k+1} = A(x_k)x_k$ where $A(x_k) = \kappa (A(x_k) - \eta)$ with Beta distributed $A(x_k) \sim \mathcal{B}(a(x_k), b(x_k))$ and $\kappa = 2$, $\eta = 0.5$ with state dependent parameters $[a(x_k), b(x_k)]^\top = \theta^\psi_B(x_k)$, with any $\theta^\psi_B : \mathcal{X} \to \Theta_B = \mathbb{R}_+^2$ is exponentially stable.

**Proof.** Applying the affine transformation to mean and variance leads to\(^4\)

$$\mathbb{E} [A(x_k)] = \kappa \left( \mathbb{E} [\tilde{A}(x_k)] - \eta \right) = \kappa \left( \frac{a}{a+b} - \eta \right),$$

$$\mathbb{V} [A(x_k)] = \kappa^2 \mathbb{V} [\tilde{A}(x_k)] = \frac{ab}{(a+b)^2(a+b+1)}.$$

Condition (11) is rewritten to

$$\kappa (\mathbb{E} [A(x_k)] - \eta))^2 + \kappa^2 \mathbb{V} [A(x_k)] \leq 1 - \alpha, \ \forall x_k \in \mathcal{X},$$

where the best possible choice for $\eta$ minimizes $(\mathbb{E} [A(x_k)] - \eta)^2$, because it leaves the largest possible range for $\kappa$. As $\mathbb{E} [A]$ is in the interval $[0,1]$ the minimization is achieved with the choice $\eta = \frac{1}{2}$. Then, condition (11), divided by $\kappa^2$ on both sides, evaluates to

$$\frac{a^2}{(a+b)^2} - \frac{a}{a+b} + \frac{1}{4} + \frac{ab}{(a+b)(a+b+1)} = \frac{ab}{(a+b)(a+b+1)} + \frac{1}{4} \leq \frac{1 - \alpha}{\kappa^2}.$$

As $\alpha > 0$ can be chosen arbitrarily small this condition holds for every $|\kappa| \leq 2$. Hence, according to Theorem 1 the system $x_{n+1} = A(x_n)x_k$ is exponentially stable. $\Box$

To ensure maximal flexibility of the model, $\kappa = 2$ is set for further considerations. This leads to the conclusion that $\Theta^*_B = \Theta_B = \mathbb{R}_+^2$. Therefore, in the optimization, no constraints on $\psi$ must be considered, thus $\Psi = \Psi^*$.

#### 4.2. Stability with Dirichlet Distribution

Constructing $A$ from a Dirichlet distribution also allows for unconstrained optimization as it also leads to stable behavior as shown in the following corollary.

**Corollary 2.** The $d$-dimensional system $x_{k+1} = A(x_k)x_k$ where each row of $A(x_k)$ consists of the first $d$ elements of a $d + 1$ dimensional Dirichlet distributed vector, thus,

$$A_{(i,:)} = a^{(i)}(1:d) \sim \mathcal{D} \left( \theta^\psi_D(x_k) \right), \quad \forall i = 1 \ldots d,$$

with any $\theta^\psi_D : \mathcal{X} \to \Theta_D, \forall i$, is asymptotically stable w.p. 1.

---

\(^3\)Even though $A, Q, P$ are scalars here, we keep them capitalized for notational consistency.

\(^4\)The state dependency of $a, b$ is dropped for notational convenience.
Proof. By construction the sample space of $A$, $Ω_Α$ contains only elements for which
\[
A_{(i,j)} > 0, \quad A_{(i,j)} < 1, \quad \forall i, j = 1 \ldots d
\]
and
\[
\sum_{j=1}^{d} A_{(i,j)} < 1, \quad \forall i = 1 \ldots d.
\] (13)

Consider now a realization of $A(x_k)$ denoted by $\hat{A}$ and since the following statements hold for any realization in the sample space, we omit writing $\forall \hat{A} \in Ω_Α$. It follows
\[
\sum_{j=1}^{d} \hat{A}_{(i,j)} < 1 \forall i \Rightarrow \max_{i=1:d} \sum_{j=1}^{d} \hat{A}_{(i,j)} < 1
\]
\[
\Rightarrow \|\hat{A}\|_∞ = \max_{i=1:d} \sum_{j=1}^{d} |\hat{A}_{(i,j)}| < 1,
\]
where the last inequality holds because all elements of $\hat{A}$ are strictly positive and $\|\hat{A}\|_∞$ denotes the Maximum Absolute Row Sum Norm. Consider now $M$ consecutive realizations $\hat{Α}^{(i)}$ with $i = 1, \ldots, M$. For the maximum norm of state in the $M$-th step holds
\[
\|x_{k+M}\|_∞ = \left\| \prod_{m=1}^{M} \hat{Α}^{(m)} x_k \right\|_∞ \leq \prod_{m=1}^{M} \|\hat{Α}^{(m)}\|_∞ \|x_k\|_∞
\]
\[
\leq \left( \max_m \|\hat{Α}^{(m)}\|_∞ \right)^M \|x_k\|_∞ \xrightarrow{M \to \infty} 0,
\]
where the submultiplicativity property of induced matrices (Horn & Johnson, 2013) is used. As convergence towards the origin holds for each element in the sample space, the system is stable with probability one. Therefore, the parameter space is unrestricted $Θ^*_D = \Theta_D = \mathbb{R}_{+}^{d+1}$.

**Remark 2.** Note that this approach only allows to represent the special class of positive systems. Nevertheless, positive systems play an important role in control engineering for modeling the evolution of strictly positive quantities as shown in (Farina & Rinaldi, 2011).

**Remark 3.** An affine transformation, as shown for Beta distribution is not possible here because absolute values are taken in the row sum. Therefore, from $\sum_{j=1}^{d} |\hat{Α}_{(i,j)}| < 1$ one cannot conclude $\sum_{j=1}^{d} |κ(\hat{Α}_{(i,j)} - 0.5)| < 1$ for any $κ > 1$.

## 5. Simulations

### 5.1. Setup

We validate our approach, labeled LeSSS (for Learning Stable Stochastic Systems), using synthetic and human motion data and the simulation of a chemical reactor. For the Beta distribution, Gaussian Mixture Regression (GMR) is used for the mapping from the state to the parameters $θ_B : X → Θ_B$. Thus, the parameter vector $ψ$, is the concatenation of the prior $π_l$, the means $μ_l$ and the covariances $Σ_l$ for $l = 1, \ldots, L$. The code (based on Calinon (2009)) includes $k$-means clustering initialization and a transformation of $Σ_l$ and $π_l$ to make it an unconstrained optimization. To evaluate the likelihood function for each training point $\{x_{n+1}, \ldots, x_n\}$, the Beta distribution parameters are computed $[a_n, b_n]^T = θ_B(\bar{x}_n)$ using GMR. Then, the log likelihood of $A_n = \bar{x}_{n+1}/\bar{x}_n$ given the parameters $[a_n, b_n]$ is evaluated using the density function of the Beta distribution. As all possible parameter $θ_B = [a, b]^T \in \mathbb{R}^2_d$ lead to stability, finding $ψ^*$ is an unconstrained optimization problem.

For the Dirichlet distribution, the mapping from the state to the parameters $θ_D^ψ : X → Θ_D$ uses a nearest neighbor approach for computational simplicity. The $2d = 4$ closest data points are considered for fitting the training parameters of the Dirichlet distribution locally. Then a training point is placed at the center of these four points. At reproduction, the closest such training point and its Dirichlet parameters are taken for regression. This does not necessarily maximize the likelihood, but shows accurate results for reproduction. We compare the following models from literature regarding reproduction precision and convergence properties:

- The approach introduced by Boots et al. (2008) learns stable linear dynamical system (stable LDS) from data. It constraints the search of the deterministic dynamic matrix $A$ to ensure the stability.
of \(x_{k+1} = Ax_k\).

- Gaussian Process Dynamical Models (GPDM) (Wang et al., 2005) represent dynamical system in the general form \(x_{k+1} = f(x_k)\), with Gaussian Process \(f \sim \mathcal{GP}(0, k(x_k, x_k'))\). We employ a zero prior mean function and a squared exponential kernel. The hyper-parameters of the kernel are optimized using the likelihood as described by Rasmussen & Williams (2006). In reproduction, this method can either be used in deterministic setting by only taking the posterior mean prediction \(\mu_{\text{GP}}(x_k)\) thus \(x_{k+1} = \mu_{\text{GP}}(x_k)\) or the stochastic setting \(x_{k+1} \sim \mathcal{N}(\mu_{\text{GP}}(x_k), \Sigma_{\text{GP}}(x_k))\), where \(\Sigma_{\text{GP}}(x_k)\) is the posterior variance. GPDMs are bounded (Beckers & Hirche, 2016a;b) but not stable.

- The Stable Estimator of Dynamical Systems (SEDS) as introduced by Khansari-Zadeh & Billard (2011) constrains the likelihood optimization of GMR parameters to a class of mean stable dynamical time-continuous systems. The GMR maps from current state \(x\) to the time derivative \(\dot{x}\). It focuses on deterministic systems by only considering stability criteria for the mean prediction of the GMR, \(\mu_{\text{GMM}}(x)\), while ignoring the stochastic nature of GMMs, (its variance prediction \(\Sigma_{\text{GMM}}\)). We also run this method in a stochastic setting, where \(\dot{x} \sim \mathcal{N}(\mu_{\text{GMM}}(x), \Sigma_{\text{GMM}}(x))\). For our simulations, five mixtures are employed.

Before starting the comparison to existing approaches, LeSSS is demonstrated on a synthetic dataset.

### 5.2. Simulation 1: Synthetic Data

For the first simulation, the task is to identify the stable nonlinear stochastic system given by

\[
x_{k+1} = A(x_k)x_k,
\]

where \(A(x_k) \sim \mathcal{B}((x_k - 5)^2, (x_k + 5)^2)\).

The learning algorithm is given 100 training points \(\{\tilde{x}, \tilde{x}_{n+1}\}_{n=1}^{100}\) equally spaced in the state space interval \([-8, 8]\) which are drawn from the state dependent Beta distribution (14). Here \(L = 3\) was chosen for the number of mixtures in the GMR for the mapping \(\theta^B_\mathcal{X} : \mathcal{X} \rightarrow \Theta_B\).

Figure 2 compares the mean and variance of the original system (14) to the one inferred by our model. It clearly shows that the model offers sufficient flexibility to reconstruct the original system. Note: It is also possible to verify the parameter functions \(a(x_k), b(x_k)\) as given in (14), but we directly look at the mean and variance functions as there exists a unique mapping and it is more intuitive for interpretation. It must be omitted, that the data was generated from the same model which the algorithm is learning. This explains the good fitting, but is of course not often the case in practical application. Therefore, we continue with a real world dataset in the following.

### 5.3. Simulation 2: Human Motion Data

For the next simulation, we use the data set for letter-shaped motions provided by Khansari-Zadeh & Billard (2011). The 225 trainings points of 3 trajectories of the two dimensional Z-shaped motion are projected on the y-axis. The GMR for \(\theta^B_\mathcal{X} : \mathcal{X} \rightarrow \Theta_B\) is trained with two mixtures.

Figure 3 shows the training data along with the fit of the mean and variance functions. The mean function shows a smoothed estimate of the training data. The model identifies properly that the training data has higher variability (around \(x_k = 0\)) and captures this in its variance function.

Figure 4 compares the reproduction of the models stable LDS, GPDM, SEDS and LeSSS if taking the deterministic (mean) output of each model (all starting from the same
initial point). The stable LDS approach leads to a converging trajectory, but fails to capture the complexity of the dynamic (as the true dynamic is nonlinear). The GPDM converges to a spurious attractor at $x \approx -9.3$ which is undesired but not surprising. SEDS and LeSSS both lead to asymptotic stable reproductions of the movement. Since the data does not contain the full state (due to the projection on the $y$-axis), it is not possible to reproduce the movement precisely with a dynamical system model.

Figure 5 compares the reproduction of the three stochastic dynamical models GPDM, SEDS and LeSSS based on three sample paths drawn from each model. The GPDM again converges to the spurious attractor. SEDS clearly shows that convergence of the mean is not sufficient for converging trajectories of a stochastic system, as the drawn sample paths are strongly oscillating around the origin without tendency to converge. In the stochastic case only LeSSS generates converging trajectories.

Figure 6 shows an example for the human motion imitation in the 2D case on a different training data set. It shows the deterministic trajectory and 5 sample path realizations, where all of them show high reproduction precision and convergence to the origin.

5.4. Simulation 3: Chemical Reactor Simulation

For the last validation, we utilize simulated data from a simplified chemical reactor (Einarsson, 1998). The closed-loop reactor is modeled by a piecewise affine system with two states: the fluid level $x_1$ and the temperature $x_2$. Both states are physically positive quantities, therefore the approach in Section 4.2 is suitable. The switching between different dynamic matrices is state dependent and occurs at $x_1 = 3$ and $x_2 = 50$, which corresponds to a discrete change of the control inputs. The training data consists of 8 trajectories of 15 steps each, which are pairwise initialized at the 4 different regions of the dynamics and perturbed with white noise with $\sigma = 0.01$ for both states.

Figure 7 shows the training data along with the reproduction using stable LDS, GPDM, SEDS and LeSSS in the deterministic setting. The initial points in the test case were set close to the one in the training data. The stable LDS is not capable to capture the varying behavior in the different regions of the piecewise affine system and therefore fails in accuracy of the reproduction. GPDM leads again to convergence outside the origin, which is undesirable. SEDS and LeSSS are both converging as it is enforced by design. Figure 8 shows that similar to the 1D case GPDM and SEDS fail to converge in the stochastic case while LeSSS is stable in all sample paths. Table 1 compares the methods with regard to the reproduction precision quantitatively. It shows that LeSSS outperforms other methods in this measure while providing the necessary guarantees regarding convergence.

5.5. Discussion

The simulations show that LeSSS is powerful enough to represent various nonlinear dynamics, while capturing the probabilistic nature of the process. The incorporation of the prior knowledge on goal convergence ensures that the learned model is stable in probability.

The computational complexity for learning the parameters of the model using interior-point methods, is mainly determined by the employed mapping in the first layer $\theta^{(0)}$. The
Stable Stochastic Nonlinear Dynamical Systems

Figure 7. Training data (black) of chemical reactor and the deterministic simulation for stable LDS, GPDM, SEDS and LeSSS approach.

Figure 8. Training data (black) of chemical reactor and the stochastic simulation for GPDM, SEDS and LeSSS approach.

<table>
<thead>
<tr>
<th>Simulation</th>
<th>LDS</th>
<th>GPDM</th>
<th>SEDS</th>
<th>LeSSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulation 2</td>
<td>0.016s</td>
<td>8.53s</td>
<td>2.47s</td>
<td>9.88s</td>
</tr>
<tr>
<td>Time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulation 3</td>
<td>4.54s</td>
<td>11.2s</td>
<td>22.3s</td>
<td>18.2s</td>
</tr>
</tbody>
</table>

Table 2. Computation times for model learning and simulation for all compared approaches.

computation times on a i5 CPU 2.30GHz, 2 Cores and 8GB RAM are given for Simulation 2 and 3 in Table 2. Since the GPDM, SEDS and LeSSS all solve non-convex optimization problems, their commutation times are in the same order of magnitude. The linear model has advantage here. Regarding the scalability with more training points, the parameter fitting performs similarly to other approaches requiring likelihood computation since this is the major factor. However, the scalability strongly depends on the employed distribution and the mapping in the first layer $\theta^\psi$.

6. Conclusion

This work proposes a framework for learning nonlinear stable stochastic dynamical systems from data. We introduce a flexible model, which builds on the state-dependent coefficient form and derive exponential stability conditions based on stochastic Lyapunov methods. The criteria is applicable to various probability distributions, while we focus to investigate the application to Beta and Dirichlet distributions. Simulation results verify sufficient flexibility of the model and the correct identification of the system’s uncertainty. In comparison to existing approaches it showed advantages in reproduction precision and convergence properties on human motion data and simulated data from a real system.

Acknowledgment

The research leading to these results has received funding from the European Research Council under the European Union Seventh Framework Program (FP7/2007-2013) ERC Starting Grant “Control based on Human Models (con-humo)” agreement number 337654. We also would like to thank the reviewers for very constructive feedback.
References


