Learning Partially Known Stochastic Dynamics with Empirical PAC Bayes

Manuel Haußmann[1*] **Sebastian Gerwinn**[2*] **Andreas Look**[2]
Barbara Rakitsch[2] **Melih Kandemir**[2]
[1]HCI/IWR, Heidelberg University [2]Bosch Center for Artificial Intelligence
manuel.haussmann@iwr.uni-heidelberg.de firstname.lastname@de.bosch.com
Heidelberg, Germany Renningen, Germany

Abstract

Neural Stochastic Differential Equations model a dynamical environment with neural nets assigned to their drift and diffusion terms. The high expressive power of their nonlinearity comes at the expense of instability in the identification of the large set of free parameters. This paper presents a recipe to improve the prediction accuracy of such models in three steps: i) accounting for epistemic uncertainty by assuming probabilistic weights, ii) incorporation of partial knowledge on the state dynamics, and iii) training the resultant hybrid model by an objective derived from a PAC-Bayesian generalization bound. We observe in our experiments that this recipe effectively translates partial and noisy prior knowledge into an improved model fit.

1 INTRODUCTION

In many engineering applications, it is often easy to model dominant characteristics of a dynamical environment by a system of differential equations with a small set of state variables. In contrast, black-box machine learning methods are often highly accurate but less interpretable. Pushing the model towards high fidelity by capturing intricate properties of the environment, however, usually requires highly flexible, e.g. over-parameterized models. Fitting these models to data can, in turn, result in over-fitting and hence poor generalization ability due to their high capacity.

Our work combines the benefits of both types of models by hybrid modeling: We set up the learning task as a non-linear system identification problem with partially known system characteristics. It assumes to have access to a differential equation system that describes the dynamics of the target environment with low fidelity, e.g. by describing the vector field on a reduced dimensionality, by ignoring detailed models of some system components, or by avoiding certain dependencies for computational feasibility. We incorporate the ODE system provided by the domain expert into a non-linear system identification engine, which we choose to be a *Bayesian Neural Stochastic Differential Equation* (BNSDE) to cover a large scope of dynamical systems, resulting in a *hybrid model*.

We propose a new algorithm for stable and effective training of such a hybrid BNSDE that combines the strengths of two statistical approaches: i) Bayesian model selection (Williams and Rasmussen, 2006), and ii) Probably Approximately Correct (PAC) Bayesian bounds (McAllester, 1999; Seeger, 2002). We improve the theoretical links between these two approaches (Germain et al., 2016b) by demonstrating how they can co-operate *during* training. To this end, we propose a novel training objective that suits SDE inference and derive a PAC-Bayesian generalization bound. Further, we provide a proof that this bound is upper bounded by the marginal likelihood of the BNSDE hyperparameters and a complexity penalizer. Gradients of this upper bound are *tied* to the actual PAC bound, hence tightening the upper bound also tightens the PAC bound. Consequently, optimizing this bound amounts to Empirical Bayes stabilized by a regularizer developed from first principles. We refer to using this objective for training as *Empirical PAC-Bayes*.

We demonstrate that our method can translate coarse descriptions of the actual underlying dynamics into a consistent forecasting accuracy increase. We first show the necessity of each of the multiple steps that

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

[*] Equal contribution.

comprise our method in an ablation study. Finally, we demonstrate in a real-world motion capture modelling task, that our method outperforms black-box system identification approaches (Chen et al., 2018; Hegde et al., 2019; Look and Kandemir, 2019) and alternative hybridization schemes that incorporate second-order Newtonian mechanics (Yildiz et al., 2019).

2 BACKGROUND

Our contribution combines approaches from stochastic differential equations, PAC-Bayes, and Empirical Bayes. Hence, we first introduce each of these concepts.

Stochastic Differential Equations. Stochastic differential equations (SDEs) are an extension of ordinary differential equations (ODEs) to include stochastic fluctuations in the dynamics (Oksendal, 1992). If we let $\mathbf{h}_t \in \mathbb{R}^P$ denote the P-dimensional state, the dynamics can be written in the following form:

$$d\mathbf{h}_t = f(\mathbf{h}_t, t)dt + G(\mathbf{h}_t, t)dW_t, \tag{1}$$

where the drift term is given by an arbitrary non-linear function $f(\cdot,\cdot) : \mathbb{R}^P \times \mathbb{R}_+ \to \mathbb{R}^P$ and the matrix valued function $G(\cdot,\cdot) : \mathbb{R}^P \times \mathbb{R}_+ \to \mathbb{R}^{P\times P}$ governs the diffusion dynamics. Finally, W_t denotes a P-dimensional Wiener Process determining the stochastic fluctuations. The solution to the SDE is a stochastic process $\mathbf{h}_t$.

As analytical solutions of SDEs are not available except for specific choices of f and G, one has to resort to numerical approximation methods. Analogous to the practice for ODEs, a common approach which we follow is to use the Euler-Maruyama (EM) method (Särkkä and Solin, 2019), which discretizes the SDE in time steps $t_1, \dots, t_K$, resulting in the following sample-based approximation to the joint distribution:

$$\begin{aligned} \mathbf{h}_{t_{k+1}} &= \mathbf{h}_{t_k} + f(\mathbf{h}_{t_k}, t_k)\Delta t_k + G(\mathbf{h}_{t_k}, t_k)\Delta W_k, \\ \Delta W_k &\sim \mathcal{N}(0, \Delta t_k \mathbb{1}_P), \quad \Delta t_k := t_{k+1} - t_k, \end{aligned} \tag{2}$$

where $\mathbb{1}_P$ is a P dimensional identity matrix. Using this sampling scheme, we obtain an approximation to the joint distribution $p(\mathbf{h}_{t_1}, \dots \mathbf{h}_{t_K})$ for the given (fixed) drift and diffusion functions.

PAC-Bayes. Probably approximately correct (PAC) bounds quantify a model's generalization capabilities from a training set to the true data distribution. To this end, a risk $R(h) = \mathbb{E}_x\left[l(x, h(x))\right]$ of a hypothesis h is defined via a loss function $l(x, h(x))$ that measures the loss of the hypothesis evaluated at a data point x. Particularly, we build upon the PAC-Bayesian formulation (McAllester, 1999, 2003), in which the generalization performance of a posterior, i.e. a distribution Q over hypotheses, is characterized by the following bound which holds with probability greater than $1 - \delta$:

$$\forall Q: \quad \mathbb{E}_{h\sim Q}\left[R(h)\right] \leq \mathbb{E}_{h\sim Q}\left[R_{\mathcal{D}}(h)\right] + \mathcal{C}(P, Q, \delta, N).$$

In the inequality above, $\mathbb{E}_Q\left[R(h)\right]$ is the expected risk across all hypotheses under the true data distribution, which is not accessible in practice, and $\mathbb{E}_Q\left[R_{\mathcal{D}}(h)\right] = \mathbb{E}_Q\left[\frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}} l\big(x, h(x)\big)\right]$ is its empirical counterpart in which the risk is averaged across the observed data $\mathcal{D}$. A distribution P over the hypotheses referred to as the prior determines the complexity term $\mathcal{C}(P, Q, \delta, N)$. This term additionally depends on the number of observed data points N and a confidence variable δ specifying the probability with which the bound holds (McAllester, 1999; Maurer, 2004).

Empirical Bayes. Bayesian models define a prior distribution $p_\phi(\theta)$ over parameters θ with hyperparameter ϕ, which together with the likelihood $p(\mathcal{D}|\theta)$ defines the full model. The standard approach consists of learning a posterior over these parameters $p(\theta|\mathcal{D})$ keeping the hyperparameters ϕ fixed and marginalizing over θ in a second step to get the posterior predictive. An alternative, known as Empirical Bayes or Type-II maximum likelihood (Bishop, 2006), directly marginalizes over the prior, and optimizes the resulting marginal likelihood with respect to the hyperparameters ϕ,

$$\phi^* = \arg\max_\phi \int p(\mathcal{D}|\theta)p_\phi(\theta)d\theta. \tag{3}$$

3 THE PROPOSED METHOD

In this section, we describe how to combine these tools into a coherent whole for effective inference. We first construct a BNSDE and equip it with domain-specific prior knowledge. Then, we derive a PAC-Bayesian objective to fit it to data and conclude with results on the proposed approach's convergence.

3.1 A Hybrid BNSDE

Application of deep learning to differential equation modelling paves the way to high-capacity predictors for capturing complex dynamics (Chen et al., 2018; Rackauckas et al., 2020). Neural Stochastic Differential Equations (NSDEs) (Look and Kandemir, 2019; Tzen and Raginsky, 2019) are SDEs as defined in (1) where the drift function, and potentially also the diffusion function are modelled as neural nets. As an initial step towards effective training, we introduce a prior distribution $p_\phi(\theta_f)$, parameterized by ϕ on the weights θ_f of an NSDE drift network, and arrive at

$$d\mathbf{h}_t = f_{\theta_f}(\mathbf{h}_t, t)dt + G(\mathbf{h}_t, t)dW_t, \quad \theta_f \sim p_\phi(\theta_f), \tag{4}$$

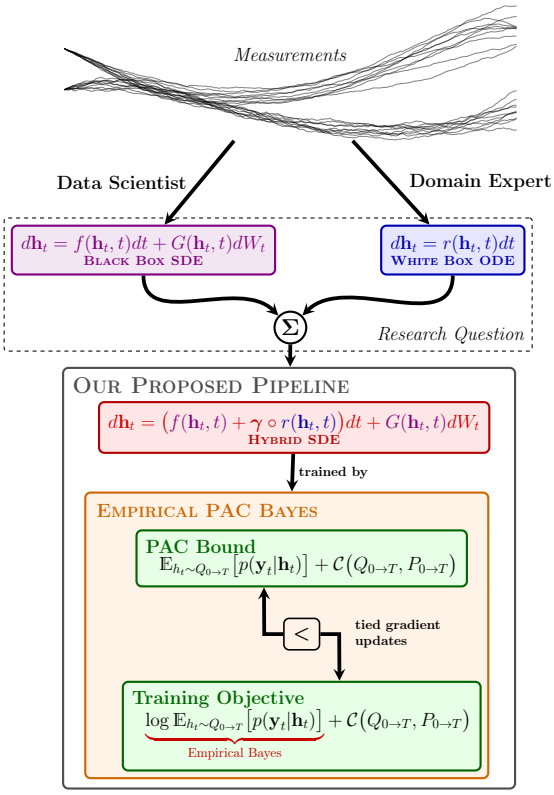

Figure 1: Illustration of the research question we pose (above) and our proposed solution (below).

which we refer to as a *Bayesian Neural Stochastic Differential Equation (BNSDE)*. The epistemic uncertainty introduced on the network weights allows the model to quantify the model uncertainty, i.e. the knowledge of which synaptic map fits best to data, in addition to the aleatoric uncertainty that the Wiener Process models. For technical reasons to be clarified below, we assume $f_{\theta_f}(\cdot,\cdot)$ and $G(\cdot,\cdot)$ to be L-Lipschitz-continuous, and $G(\cdot,\cdot)$ not to have any learnable parameters.

A coarse description of the environment dynamics is sometimes available as an incomplete set of differential equations in real-world applications. For instance, the dynamics of a three-dimensional volume might be modelled as a flow through a single point, such as the center of mass. Alternatively, a model on a subset of the system components might be provided. We assume this prior knowledge to be available as an ODE

$$d\mathbf{h}_t = r_\xi(\mathbf{h}_t, t)dt, \tag{5}$$

where $r_\xi(\cdot,\cdot) : \mathbb{R}^P \times \mathbb{R}_+ \to \mathbb{R}^P$ is an arbitrary non-linear function parameterized by a fixed set of parameters ξ. We can incorporate these known dynamics into the BNSDE by adding them to the drift as follows:

$$d\mathbf{h}_t = \big(f_{\theta_f}(\mathbf{h}_t, t) + \boldsymbol{\gamma} \circ r_\xi(\mathbf{h}_t, t)\big)dt + G(\mathbf{h}_t, t)dW_t, \tag{6}$$

which can be viewed as a hybrid SDE with the free parameter vector $\boldsymbol{\gamma} \in [0,1]^P$ governing the relative importance of prior knowledge on the learning problem and $\circ$ referring to element-wise multiplication. Although we specified (5) within the same dimensional state space as (6), $\boldsymbol{\gamma}$ allows us to provide only partial information. When prior knowledge is available only for a subset of the state space dimensions, the remaining dimensions d can be filled in by simply setting $\gamma_d = 0$.

We define a prior stochastic process representing solely the prior knowledge of the dynamics as

$$d\mathbf{h}_t = \big(\boldsymbol{\gamma} \circ r_\xi(\mathbf{h}_t, t)\big)dt + G(\mathbf{h}_t, t)dW_t. \tag{7}$$

This prior SDE will be used as a reference distribution for complexity penalization as part of the final PAC training objective of our hybrid SDE. Note that we have used the same diffusion term as in (6) for specifying the prior SDE, which makes the complexity term within the PAC-formulation tractable, as we will show later. Also note that $\boldsymbol{\gamma}$ is a free parameter of the prior.

3.2 Learning via Empirical Bayes

Solving the SDE in (6) even for fixed parameters θ_f over an interval $[0, T]$ is analytically intractable for basically all practically interesting use cases. While our method is applicable to any discretization scheme, we demonstrate its use with the straightforward EM for simplicity, which gives us the discrete-time version of the hybrid BNSDE below

$$\begin{aligned}
&\theta_f \sim p_\phi(\theta_f), \quad \mathbf{h}_0 \sim p(\mathbf{h}_0),\\
&\mathbf{h}_{k+1}|\mathbf{h}_k, \theta_f \sim \mathcal{N}\left(\mathbf{h}_{k+1}\middle|\mathbf{h}_k + d(\mathbf{h}_k, t_k)\Delta t_k, \Sigma_k\right),\\
&d(\mathbf{h}_k, t_k) = f_{\theta_f}(\mathbf{h}_k, t_k) + \boldsymbol{\gamma} \circ r_\xi(\mathbf{h}_k, t_k),
\end{aligned}$$

with $\Sigma_k := \mathbf{J}_k \Delta t_k, \mathbf{J}_k := G(\mathbf{h}_k, t_k)G(\mathbf{h}_k, t_k)^\top$, $\Delta t_k := t_{k+1} - t_k$, and $p(\mathbf{h}_0)$ defined on the initial state.

Analogously to latent state space models, we assume that the observations of the dynamics described in (4), (6), and (7) are linked via a likelihood $p(\mathbf{y}_k|\mathbf{h}_k)$. Specifically, we observe these dynamics as a time series $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ consisting of K observations $\mathbf{y}_k \in \mathbb{R}^D$, collected at irregular time points $\mathbf{t} = \{t_1, \ldots, t_K\}$.

Given an observed set of N such time series trajectories $\mathcal{D} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_N\}$, the classical approach (MacKay, 2003; Gelman et al., 2013) would now require as a first step the inference of the posterior over both the global variables θ_f as well as the local variables $\mathbf{H}_n = \{\mathbf{h}_1^n, ..., \mathbf{h}_K^n\}$, i.e. of $p(\theta_f, \mathbf{H}_1, \ldots, \mathbf{H}_N|\mathcal{D})$, and as a second step a marginalization over this posterior to get the posterior predictive. As an analytical solution is intractable, approximate solutions such as Markov Chain Monte Carlo (MCMC) methods or Variational Inference (VI) are required. Application of either of these approaches to BNSDEs is prohibitive, the former

computationally, the latter in terms of expressiveness since existing work makes strong independence and structural assumptions on the approximate posterior.

We propose in the following to apply model selection as an alternative path to BNSDE inference. Instead of performing the posterior inference on the latent variables, we marginalize them out and learn those hyperparameters ϕ from data that provide the highest log marginal likelihood (Williams and Rasmussen, 2006). That is our BNSDE learns the optimal ϕ^* via

$$\arg\max_{\phi} \int p(\mathcal{D}|\mathbf{H})p(\mathbf{H}|\theta_f)p_\phi(\theta_f)d(\mathbf{H},\theta_f). \quad (8)$$

An advantage of this construction is that the marginal likelihood has the identical functional form to the predictive distribution, which is the quantity of interest in a typical prediction task. Marginal likelihood learning has also been applied before in the context of neural networks (Sensoy et al., 2018; Malinin and Gales, 2018; Garnelo et al., 2018). Fitting the hyperparameters of an SDE to data via marginal likelihood maximization can also be viewed as an instance of the simulated likelihood method (Särkkä and Solin, 2019).

Marginalizing over θ_f in (8) is intractable for most practical use cases. However, it can be approximated by Monte Carlo integration without constructing chains on the global parameters. Sampling directly from the prior, we get for a single observation n and $s = 1, \dots S$

$$\begin{aligned} &\theta_f^s \sim p_\phi(\theta_f), \quad \mathbf{H}^s \sim p(\mathbf{H}|\theta_f^s), \\ &\phi^* := \arg\max_{\phi} \log\Big(\frac{1}{S}\sum_{s=1}^{S} p(\mathcal{D}|\mathbf{H}^s)\Big). \end{aligned} \quad (9)$$

In order to maximize this objective, we require an efficient computation of gradients w.r.t. the hyperparameters ϕ. Access to ϕ is only given via samples from the distribution it is parameterizing. In our experiments, we assume this distribution $p_\phi(\theta_f)$ to be normal, allowing us to make use of a standard reparameterization. We separate the sampling process into a parameter-free source of randomness and a parametric transformation, i.e. we have $\varepsilon \sim p(\varepsilon)$, $\theta_f = g_\phi(\varepsilon)$, for a suitable $g_\phi(\cdot)$. In order to further reduce the variance noise introduced to the gradients due to this sampling step, we also use the *local reparameterization trick* (Kingma et al., 2015) in the drift, i.e. we sample the layer outputs during the forward propagation instead of individual layer weights.

The objective (9) is agnostic to the specific SDE employed. Therefore, we refer to the discretized black-box SDE in (4) governing $p(\mathbf{H}|\theta_f)$ and trained w.r.t. ϕ via this objective as *E-Bayes* throughout the experiments. Analogously, we refer to training a hybrid SDE as in (6) with the same method as *E-Bayes-Hybrid*.

3.3 A Trainable PAC Bound

A major downside of the objective in (9), when applied to BNSDEs, is that it optimizes a large set of hyperparameters, i.e. means and variances of drift network weights, without a proper regularization aside from the implicit regularization inherent in the chosen architecture and the marginalization itself. While the hybrid approach already allows us to incorporate prior expert knowledge, it remains a guiding signal without an explicit model capacity regularizer. Next, we address this problem by developing a training objective derived from a PAC-Bayesian bound objective that combines the benefits from the results we arrived at so far with a proper regularization scheme.

The proposed approach is still agnostic to the chosen discretization scheme. Consequently, we refer for any time horizon $T > 0$ to all local latent variables by $\mathbf{h}_{0\to T}$. To distinguish the density given by the hybrid SDE in (6) from the prior SDE in (7), we further refer to the two densities induced by them respectively as $p_{\text{hyb}}(\mathbf{h}_{0\to T}|\theta_f)$ and $p_{\text{pri}}(\mathbf{h}_{0\to T})$. We define two distributions Q and P over $(\mathbf{h}_{0\to T}, \theta_f)$. For the former, we have the joint distribution of the hybrid process

$$Q_{0\to T}(\mathbf{h}_{0\to T}, \theta_f) = p_{\text{hyb}}(\mathbf{h}_{0\to T}|\theta_f)p_\phi(\theta_f), \quad (10)$$

while the latter stands for the joint of the prior process

$$P_{0\to T}(\mathbf{h}_{0\to T}, \theta_f) = p_{\text{pri}}(\mathbf{h}_{0\to T})p_{\text{pri}}(\theta_f). \quad (11)$$

Although the prior process is independent of the drift parameters θ_f, we specify a fixed prior distribution $p_{\text{pri}}(\theta_f)$, which we choose to be a standard normal within our experiments. To be compliant with the notational practice in the PAC-Bayesian literature, we denote the prior distribution as P and the posterior distribution that is fit to data as Q.[1]

As both Q and P share the same diffusion term, the Kullback-Leibler (KL) divergence between these processes can be calculated by extending the proof of Archambeau et al. (2008). The following Lemma holds for any choice of diffusion $G(\cdot,\cdot)$. You can find the proofs for it and the following Theorems in the appendix.

Lemma 1. *For the process distributions $Q_{0\to T}$ and $P_{0\to T}$, it holds that*

$$\begin{aligned} D_{KL}\big(Q_{0\to T}||P_{0\to T}\big) = \\ \frac{1}{2}\int_0^T \mathbb{E}_{Q_{0\to T}}\Big[f_{\theta_f}(\mathbf{h}_t,t)^\top \mathbf{J}_t^{-1} f_{\theta_f}(\mathbf{h}_t,t)\Big]dt \\ + D_{KL}\big(p_\phi(\theta_f)||p_{\text{pri}}(\theta_f)\big), \end{aligned}$$

[1] In the PAC-Bayesian framework, P and Q do not have to be linked to each other via application of the Bayes rule on an explicitly defined likelihood.

for some $T > 0$, *where* $\mathbf{J}_t = G(\mathbf{h}_t, t)G(\mathbf{h}_t, t)^\top$.

This Lemma[2] provides one of the main ingredients for deriving a PAC-Bayesian bound on the generalization performance of a learned distribution $Q_{0\to T}$. To derive such a bound, we additionally specify the risk via a loss function measuring the model mismatch. We assume the likelihood function $p(\mathbf{y}_t|\mathbf{h}_t)$ to be uniformly bounded everywhere.[3] We then define the true risk of a draw from $Q_{0\to T}$ on an i.i.d. sampled trajectory $\mathbf{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_K\}$ at discrete and potentially irregular time points $t_1, \dots, t_K$ drawn from an unknown ground-truth stochastic process $\mathfrak{G}(t)$ as the expected model misfit on the sample. Specifically, we define the risk over hypotheses $H = (\mathbf{h}_{0\to T}, \theta_f)$ as follows:

$$R(H) = 1 - \mathbb{E}_{\mathbf{Y}\sim\mathfrak{G}(t)}\Big[\prod_{k=1}^{K} p(\mathbf{y}_k|\mathbf{h}_k)/\overline{B}\Big], \tag{12}$$

for time horizon $T > 0$ and the corresponding empirical risk on a data set $\mathcal{D} = \{\mathbf{Y}_1, \dots, \mathbf{Y}_N\}$ as

$$R_{\mathcal{D}}(H) = 1 - \frac{1}{N}\sum_{n=1}^{N}\Big[\prod_{k=1}^{K} p(\mathbf{y}_k^n|\mathbf{h}_k^n)/\overline{B}\Big]. \tag{13}$$

Here, $\overline{B} := \max_{\mathbf{y}_k,\mathbf{h}_k} p(\mathbf{y}_k|\mathbf{h}_k)$ is a uniform bound to guarantee a $[0, 1]$-valued loss.

Next, we develop a PAC-Bayesian generalization bound building on these risk definitions. Furthermore, we upper bound it with a trainable objective.

Theorem 1. *The expected true risk is bounded above with probability* $\mathbb{P} \geq 1 - \delta$, *for* $\delta \in (0, 1]$ *by:*

$$\begin{aligned}
&\mathbb{E}_{H\sim Q_{0\to T}}\left[R(H)\right] \\
&\leq \mathbb{E}_{H\sim Q_{0\to T}}\left[R_{\mathcal{D}}(H)\right] + \mathcal{C}_\delta(Q_{0\to T}, P_{0\to T}) && (14)\\
&\leq -\frac{1}{N}\sum_{n=1}^{N}\log\left(\frac{1}{S}\sum_{s=1}^{S}\prod_{k=1}^{K} p(\mathbf{y}_k^n|\mathbf{h}_k^{n,s})\right) && (15)\\
&\quad + \underbrace{\mathcal{C}_{\delta/2}(Q_{0\to T}, P_{0\to T}) + \sqrt{\frac{\log(2N/\delta)}{2S}} + K\log\overline{B}}_{=:C}, \\
&\leq -\frac{1}{SN}\sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{k=1}^{K}\log\Big(p(\mathbf{y}_k^n|\mathbf{h}_k^{n,s})\Big) + C && (16)
\end{aligned}$$

with S *the sample count taken independently for each observed sequence, and the complexity functional:*

$$\mathcal{C}_\delta(H_{0\to T}, P_{0\to T}) = \sqrt{\frac{D_{KL}\big(Q_{0\to T}||P_{0\to T}\big) + \log(2\sqrt{N}) - \log(\delta/2)}{2N}}$$

where $D_{KL}\big(Q_{0\to T}||P_{0\to T}\big)$ *as in Lemma 1.*

As the complexity term in (16) vanishes for large sample sizes (N, S), the first term converges to the expected log-likelihood for a given time resolution K. Although the bound loosens as K increases, note that the gradient of the bound w.r.t. model parameters remains unaffected. Theorem 1 can be used to learn a posterior distribution $Q_{0\to T}$ from data by adjusting ϕ. Additionally, we can also learn the importance of the prior by fitting the γ parameter to data. While directly learning γ by optimizing the PAC-bound violates the generalization guarantee, we can define a collection of prior distributions $P_{0\to T}$ for a set Γ of discretized values of γ and employ the same union bound as Reeb et al. (2018). The resulting PAC-bound differs by a constant accounting for the number of distinct γ values within the collection. Therefore, we can use the same gradient based optimization to learn γ and quantize the value to the closest point within Γ to evaluate the PAC bound.

3.4 The Training Algorithm

The first term in (14) does not correspond to the Empirical Bayes objective as it averages over likelihoods, and not log-likelihoods (Germain et al., 2016a). However, the first term in (15) provides a sampling based approximation to the empirical Bayes objective. By defining the risk in such a way and employing the PAC-Bayesian framework, we obtain a regularized version of empirical Bayes. Although placing the $\log(\cdot)$ function into its summands loosens the bound on the true risk, it improves numerical robustness and optimizing (16) still tightens the original PAC-Bayesian bound, i.e. (14), as stated in the following corollary.

Corollary 1. *For Lipschitz-continuous risk and likelihood, a gradient step that reduces* (16) *also tightens the PAC bound in* (14).

Minimizing (16) hence closes the loop as the Empirical Bayes objective derived in (9) reappears in (15) but

[2] We assume p_{pri} and p_ϕ to be Gaussians. However, to express the KL divergence between prior and posterior processes analytically, it is sufficient for them to share the same diffusion. Following the established definition of a stochastic process on which the Itô calculus has been built, we assume Gaussian diffusion noise across time increments. In our case, these Gaussian increments are warped by non-linear drift functions (neural nets) in the subsequent time steps. Hence, they are capable of expressing arbitrarily complicated marginal process densities. In effect, one can attain a Gaussian distributed marginal process only from a linear time-invariant SDE. The integral could be analogously defined also for other consistent increment choices (e.g., Levy-Flights instead of Brownian motions).

[3] In our experiments, we ensure this condition by choosing the likelihood to be a normal density with bounded variance, i.e. bounded mass on the mode.

is now combined in a principled way with the regularization term $\mathcal{C}_\delta$. We can ignore the terms that do not depend on ϕ and adopt the remaining expression bound as our final objective and learn ϕ^* via

$$\phi^* := \arg\max_\phi \frac{1}{SN}\sum_{n=1}^{N}\sum_{s=1}^{S}\sum_{k=1}^{K}\log\Big(p(\mathbf{y}_k^n|\mathbf{h}_k^{n,s})\Big) + \sqrt{\Big(D_{KL}\big(Q_{0\to T}||P_{0\to T}\big) + \log(4\sqrt{N}/\delta)\Big)/2N}. \tag{17}$$

In this training procedure, we only train w.r.t. ϕ which determine the drift term. To also learn the diffusion, one could represent G also by a BNN. However, the corresponding training procedure would invalidate the PAC statement. Nevertheless, the diffusion term could be learnt on a held-out data set and then incorporated as fixed to the bound (16). As Theorem 1 applies to any diffusion term, we keep the genericness of its statement. However, in the experiments, we stick to a constant diffusion term for practical reasons.

Although we require i.i.d. observations of time series in the theory, we can in practice use mini-batches of trajectories provided that the batches are sufficiently far apart so that they become essentially independent. The objective (15) differs from the Empirical Bayes one in (9) only by the complexity term. The only complicated calculation step in this term is the integral through the process, which can be made more implementation friendly using Fubini's theorem:

$$\int_0^T \mathbb{E}_{Q_{0\to T}}\Big[f_{\theta_f}(\mathbf{h}_t,t)^\top \mathbf{J}_t^{-1} f_{\theta_f}(\mathbf{h}_t,t)\Big]dt = \mathbb{E}_{Q_{0\to T}}\Big[\int_0^T f_{\theta_f}(\mathbf{h}_t,t)^\top \mathbf{J}_t^{-1} f_{\theta_f}(\mathbf{h}_t,t)dt\Big].$$

A pseudo-code description of the procedure is given in Algorithm 1. Our sampling-based method naturally couples with the EM approximation and inherits its convergence properties. We show strong convergence to the true solution with shrinking step size by extending the plain EM proof (Kloeden and Platen, 2011).

Theorem 2 (strong convergence). *Let $\mathbf{h}_t^\theta$ be an Itô process as in (4) with drift and diffusion parameters θ and $\widetilde{\mathbf{h}}_t^\theta$ its Euler-Maruyama approximation for some regular step size $\Delta t > 0$. For some coefficient $R > 0$ and any $T > 0$, the below inequality holds as $S \to \infty$*

$$\mathbb{E}\left[\sup_{0\le t\le T}\left|\mathbb{E}_\theta[\mathbf{h}_t^\theta] - \frac{1}{S}\sum_{s=1}^{S}\widetilde{\mathbf{h}}_t^{\theta^{(s)}}\right|\right] \le R\Delta t^{1/2},$$

where $\theta^{(s)}$ are i.i.d. draws from a prior $p_\phi(\theta)$.

Algorithm 1: E-PAC-Bayes-Hybrid Loss

Input: set of N trajectories $\mathcal{D}$, prior drift $r_\xi(\cdot,\cdot)$, time points $\mathbf{t}$, drift $f_{\theta_f}(\cdot,\cdot)$, diffusion $G(\cdot,\cdot)$, weight distribution $p_\phi(\theta_f)$, number of samples S, prior parameter γ
Output: training objective loss

```
// init. marginal log-likelihood (mll) and kl
mll ← 0; kl ← 0
for n ∈ {1,...,N} do // for each trajectory
    for s ∈ {1,...,S} do // and each sample
        // sample initial state and weights
        h_0^{n,s} ~ p(h_0); θ_f^{n,s} ~ p_φ(θ_f)
        // for each of the K steps
        for k ∈ {1,...,K} do
            // get drift,prior,diffusion output
            f_k^{n,s} ← f_{θ_f^{n,s}}(h_{k-1}^{n,s}, t_{k-1})
            r_k^{n,s} ← r_ξ(h_{k-1}^{n,s}, t_{k-1})
            G_k^{n,s} ← G(h_{k-1}^{n,s}, t_{k-1})
            // sample stochasticity
            Δt_k ← t_k − t_{k-1}
            W_k^{n,s} ~ N(0, Δt_k 𝟙)
            // update state
            h_k^{n,s} ←
              h_{k-1}^{n,s} + (f_k^{n,s} + γ r_k^{n,s})Δt_k + G_k^{n,s} W_k^{n,s}
            // and update mll and kl
            mll ← mll + (1/SN) log p(y_k^n | h_k^{n,s})
            kl ← kl + (1/2S) f_k^{n,s⊤} (G_k^{n,s} G_k^{n,s⊤})^{-1} f_k^{n,s} Δt_k
        end
    end
end
// add penalty for modified drift distribution
kl ← kl + D_KL( p_φ(θ_f) || p_pri(θ_f) )
// and assign final loss
loss ← −mll + sqrt( (kl + log(4√N/δ)) / (2N) )
// to be returned and optimized
return loss
```

4 RELATED WORK

Empirical Bayes as PAC Learning. Germain et al. (2016b) propose a learnable PAC-Bayesian bound that provides generalization guarantees as a function of a marginal log-likelihood. Our method differs from this work in two main lines. First, Germain et al. (2016b) define risk as $-\log p(\mathbf{Y}|\mathbf{H}) \in (-\infty, +\infty)$ and compensate for the unboundedness by either truncating the support of the likelihood function or introducing assumptions on the data distribution, such as sub-Gaussian or sub-Gamma. Our risk defined in (12) assumes uniform boundedness, yet can be incorporated into a PAC-Bayesian bound without further restrictions. Second,

Germain et al. (2016b)'s bound is an unparameterized rescaling of the marginal log-likelihood. Hence, it is not linked to a capacity penalizer, which can be used at *training time* for regularization. Applying this method to hybrid sequence modelling boils down to performing plain Empirical Bayes, i.e. *E-Bayes* in our experiments.

Differential GPs. Hegde et al. (2019) model the dynamics of the activation maps of a *feed-forward* learner by the predictive distribution of a GP. This method allocates the mean of a GP as the drift and covariance as the diffusion. It infers the resultant model using variational inference. While direct application of this method to time series modeling is not straightforward, we represent it in our experiments by sticking to our generic non-linear BNSDE design in (4), and inferring it by maximizing the ELBO: $\mathcal{L}(\phi) = \mathbb{E}_{\mathbf{H},\theta}\big[\log\ p(\mathbf{Y}|\mathbf{H})\big] - D_{KL}\big(p_\phi(\theta)||p(\theta)\big)$, applying the local reparameterization trick on θ. Although variational inference can be seen from a PAC-perspective by choosing the log-likelihood as the loss (Knoblauch et al., 2019), the ELBO does not account for the deviation of variational posterior over latent dynamics from the prior latent dynamics. We refer to this baseline in the experiments as *D-BNN (VI)*. The approximate posterior design here closely follows the PR-SSM approach (Doerr et al., 2018), which represents state of the art in state-space modelling.

Differential BNNs with SGLD. The learning algorithm of Look and Kandemir (2019) shares our BNSDE modeling assumptions, however, it uses Stochastic Gradient Langevin Dynamics (SGLD) to infer θ. The algorithm is equivalent to performing MAP estimation of the model parameters in (4) while distorting the gradient updates with decaying normal noise that also determines the learning rate.

Black-box identification of dynamic systems. There are various approaches to identify a dynamical system that differ in the model class used for fitting the right-hand side of the differential equation and may also allow for transitional noise (e.g. Brunton et al., 2016; Durstewitz, 2016). These approaches could be incorporated into ours, using their transition likelihood and prior over parameters. Our black-box neural SDE can be seen as one instance of such a black-box identification of dynamical systems (*E-Bayes*). As we are mainly interested in incorporating prior knowledge into such black-box models, we chose one such competitor (Hegde et al., 2019), with reported results on the CMU Motion capture data set (Tab. 2).

Table 1: Ablation study on the Lorenz attractor to evaluate the contributions of the prior knowledge on the predictive performance measured in Mean Squared Error (MSE) with standard error over fifty repetitions. The hybrid models ((iii), (iv)) consistently improve on the black box models ((i),(ii)). The last row (v) shows the performance for the case the model has full access to the true dynamics with noisy parameters in (5).

Prior Knowledge	Model	Test MSE
None	(i)	29.20 ± 0.19
	(ii)	29.05 ± 0.23
$\boldsymbol{\gamma} = [1, 0, 0], \quad \zeta \sim \mathcal{N}(10, 1)$	(iii)	27.58 ± 0.17
	(iv)	27.42 ± 0.16
$\boldsymbol{\gamma} = [0, 1, 0], \quad \kappa \sim \mathcal{N}(2.67, 1)$	(iii)	15.87 ± 0.46
	(iv)	15.06 ± 0.35
$\boldsymbol{\gamma} = [0, 0, 1], \quad \rho \sim \mathcal{N}(28, 1)$	(iii)	27.82 ± 0.26
	(iv)	28.37 ± 0.21
$\boldsymbol{\gamma} = [1, 1, 1]$, $(\zeta, \kappa, \rho)^\top \sim \mathcal{N}\big((10, 2.67, 28)^\top, \mathbb{1}_3\big)$	(v)	16.40 ± 2.31

5 EXPERIMENTS

We evaluate the following four variants of our method:

(i) *E-Bayes.* Empirical Bayes without prior knowledge, i.e. training (9) with $p(\mathbf{h}_{0\to T})$ given by (4).

(ii) *E-PAC-Bayes.* Empirical PAC Bayes on the BNDSE using the objective in (16) with an uninformative prior drift, i.e. $r_\xi(\mathbf{h}_t, t) = 0$.

(iii) *E-Bayes-Hybrid.* Same training objective as (i), however with the hybrid model as proposed in (6).

(iv) *E-PAC-Bayes-Hybrid.* The hybrid model (6) with the same loss as *E-PAC-Bayes*, which is the combination we propose.

We extend the Empirical Bayes objective in (9) by PAC-Bayes to tune many hyperparameters without overfitting and incorporate prior domain knowledge in a principled way. We evaluate the first motivation as *E-PAC-Bayes*, i.e. objective (16) but without a prior SDE, and the complete model including a prior SDE as *E-PAC-Bayes-Hybrid*. See the appendix for a detailed discussion of each of these methods' computational cost and further experiments.

Lorenz Attractor. This chaotic non-linear system has the the following inherently unsolvable dynamics

$$\begin{aligned} dx_t &= \zeta(y_t - x_t)dt + dW_t, \\ dy_t &= \big(x_t(\kappa - z_t) - y_t\big)dt + dW_t, \\ dz_t &= (x_t y_t - \rho z_t)dt + dW_t, \end{aligned}$$

Table 2: Benchmarking of our method on the CMU Motion Capture Data Set. Mean Squared Error (MSE) and Negative Log-Likelihood (NLL) on 300 future frames is averaged over ten repetitions (± standard deviation).

Method	Reference	Bayesian	Hybrid	+KL	Test MSE	Test NLL
DTSBN-S	(Gan et al., 2015)	No	No	No	34.86 ± 0.02	Not Applicable
npODE	(Heinonen et al., 2018)	No	No	No	22.96	Not Applicable
Neural-ODE	(Chen et al., 2018)	No	No	No	22.49 ± 0.88	Not Applicable
ODE^{2}VAE	(Yildiz et al., 2019)	Yes	Yes	No	10.06 ± 1.40	Not Reported
ODE^{2}VAE-KL	(Yildiz et al., 2019)	Yes	Yes	Yes	8.09 ± 1.95	Not Reported
D-BNN (SGLD)	(Look and Kandemir, 2019)	Yes	No	No	13.89 ± 2.56	747.92 ± 58.49
D-BNN (VI)	(Hegde et al., 2019)	Yes	No	Yes	9.05 ± 2.05	452.47 ± 102.59
E-Bayes	Baseline	Yes	No	No	8.68 ± 1.56	433.76 ± 77.78
E-PAC-Bayes	Ablation	Yes	No	Yes	9.17 ± 1.20	489.82 ± 67.06
E-Bayes-Hybrid	Ablation	Yes	Yes	No	9.25 ± 1.99	462.82 ± 99.61
E-PAC-Bayes-Hybrid	Proposed	Yes	Yes	Yes	$\mathbf{7.84 \pm 1.41}$	$\mathbf{415.38 \pm 80.37}$

where $\zeta = 10, \kappa = 28, \rho = 2.67$, and W_t is a random variable following a Wiener process with unit diffusion. We generate 1920 observations from the above dynamics initiating the system at $x_0, y_0, z_0 = (1, 1, 28)$, use the first half for training and the rest for testing. We split both the training and the test data into 20 sequences of length 24, which can be interpreted as i.i.d. samples of the system with different initial states. Table 1 presents the 24-step ahead forecasting error in MSE on the test set for our model variants. In each experiment repetition, E-Bayes-Hybrid and E-PAC-Bayes-Hybrid are provided one equation after distorting the corresponding parameter by normal distributed noise. The other equations are hidden by being hard assigned to zero. To set up the corresponding prior and model, we used a constant diffusion with $G = \mathbb{1}$. Despite the imprecision of the provided prior knowledge, the largest performance leap comes from the hybrid models. The complexity term on the PAC-Bayesian bound restricts the model capacity for black-box system identification, while it improves the hybrid setup.

Figure 2 visualizes the predicted trajectories on the test sequence for prior knowledge on dz_t. Even with weak prior knowledge, the proposed model is stable longer than the baseline and shows a proper increase in the predictive variance over time.

CMU Walking Data Set. We benchmark against state of the art on this motion capture data set following the setup of Yildiz et al. (2019). We train an *E-PAC-Bayes* model on the `MOCAP-1` data set consisting of 43 motion capture sequences measured from 43 different subjects. The drift net of the learned BNSDE is then treated as weak and broad prior knowledge of human walking dynamics. We use `MOCAP-2` with 23 walking sequences from Subject 35 to represent a high-fidelity subject-specific modelling task. As reported in Table 2 of Yildiz et al. (2019), the state of the art

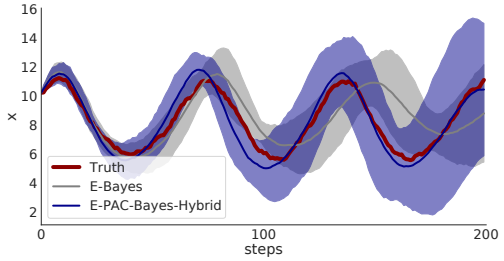

Figure 2: Predicted mean trajectory starting at $T = 10$ on one dimension of the Lorenz data. The shaded areas give ±2 standard deviations over 21 trajectories.

of subject-independent mocap dynamic modelling has twice as high prediction error as subject-specific dynamics (MSE of 15.99 versus 8.09). Analogously to the Lorenz attractor experiment, we fixed the PAC-variants' prior diffusion term to be constant. We report the test MSE and negative log-likelihoods in Table 2. Our method delivers the best prediction accuracy and model fit when all its components are active.

6 CONCLUSION

We have shown that our method incorporates vague prior knowledge into a flexible Bayesian black-box modelling approach for learning SDEs resulting in a robust learning scheme guided by generalization performance via a PAC-Bayesian bound. The method is easily adaptable to other solvers. For example, the training loss derived in (9) can also be optimized using a closed-form normal assumed density scheme applied over a stochastic Runge-Kutta variant (Li et al., 2019). Independent from the sampling scheme and model used, our tied gradient update procedure allows training on the loose, yet numerically stable, bound while providing an improvement w.r.t. the generalization guarantees on its tighter counterpart. Our stochastic approximation of the data

log-likelihood currently relies on samples obtained from the prior, yet could be improved by incorporating a more sophisticated sampling scheme, e.g. using particle filtering (Kantas et al., 2015). Finally, the bound in (16) has the potential to be vacuous for certain drift nets, incorporating a Hoeffding assumption (Alquier et al., 2016) could further tighten it.

References

P. Alquier, J. Ridgway, and N. Chopin. On the properties of variational approximations of gibbs posteriors. *The Journal of Machine Learning Research*, 17(1): 8374–8414, 2016.

C. Archambeau, M. Opper, Y. Shen, D. Cornford, and J.S. Shawe-Taylor. Variational Inference for Diffusion Processes. In *NIPS*. 2008.

C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

S. L Brunton, J.L. Proctor, and J.N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937, 2016.

R. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud. Neural Ordinary Differential Equations. In *NeurIPS*, 2018.

A. Doerr, C. Daniel, M. Schiegg, D. Nguyen-Tuong, S. Schaal, M. Toussaint, and S. Trimpe. Probabilistic Recurrent State-Space Models. In *ICML*, 2018.

D. Durstewitz. A state space approach for piecewise-linear recurrent neural networks for reconstructing nonlinear dynamics from neural measurements. *arXiv preprint arXiv:1612.07846*, 2016.

Z. Gan, C. Li, R. Henao, D.E. Carlson, and L. Carin. Deep Temporal Sigmoid Belief Networks for Sequence Modeling. In *NIPS*, 2015.

M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y.W. Teh, D. Rezende, and S. M.A. Eslami. Conditional Neural Processes. ICML, 2018.

Andrew Gelman, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. CRC press, 2013.

P. Germain, F. Bach, A. Lacoste, and S. Lacoste-Julien. PAC-Bayesian Theory Meets Bayesian Inference. In *NIPS*. 2016a.

P. Germain, F. Bach, A. Lacoste, and S. Lacoste-Julien. PAC-Bayesian Theory Meets Bayesian Inference. In *NIPS*, 2016b.

P. Hegde, M. Heinonen, H. Lähdesmäki, and S. Kaski. Deep Learning with Differential Gaussian Process Flows. In *AISTATS*, 2019.

M. Heinonen, C. Yildiz, H. Mannerström, J. Intosalmi, and H. Lähdesmäki. Learning Unknown ODE Models with Gaussian Processes. In *ICML*, 2018.

N. Kantas, A. Doucet, S.S. Singh, J. Maciejowski, N. Chopin, et al. On particle methods for parameter estimation in state-space models. *Statistical science*, 30(3):328–351, 2015.

D.P. Kingma, T. Salimans, and M. Welling. Variational Dropout and The Local Reparameterization Trick. In *NIPS*, 2015.

P.E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer-Verlag, 2011.

J. Knoblauch, J. Jewson, and T. Damoulas. Generalized variational inference. *arXiv preprint arXiv:1904.02063*, 2019.

X. Li, Y. Wu, L. Mackey, and M.A. Erdogdu. Stochastic Runge-Kutta accelerates Langevin Monte Carlo and beyond. In *NeurIPS*. 2019.

A. Look and M. Kandemir. Differential Bayesian Neural Nets. In *4th NeurIPS Workshop on Bayesian Deep Learning*, 2019.

D. JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

A. Malinin and M. Gales. Predictive uncertainty estimation via prior networks. In *NeurIPS*, 2018.

A. Maurer. A note on the PAC Bayesian theorem. *arXiv preprint cs/0411099*, 2004.

D. McAllester. PAC-Bayesian Model Averaging. In *COLT*, 1999.

D. McAllester. PAC-Bayesian Stochastic Model Selection. *Machine Learning*, 51:5–21, 2003.

B. Oksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer-Verlag, 1992.

C. Rackauckas, Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, and A. Ramadhan. Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*, 2020.

D. Reeb, A. Doerr, S. Gerwinn, and B. Rakitsch. Learning Gaussian Processes by Minimizing PAC-Bayesian Generalization Bounds. In *NeurIPS*. 2018.

S. Särkkä and A. Solin. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

M. Seeger. PAC-Bayesian Generalisation Error Bounds for Gaussian Process Classification. *Journal of Machine Learning Research*, 3:233–269, 2002.

M. Sensoy, L. Kaplan, and M. Kandemir. Evidential Deep Learning to Quantify Classification Uncertainty. In *NeurIPS*, 2018.

B. Tzen and M. Raginsky. Neural Stochastic Differential Equations: Deep Latent Gaussian Models in the Diffusion Limit. *ArXiv*, abs/1905.09883, 2019.

C. KI Williams and C. E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

C. Yildiz, M. Heinonen, and H. Lahdesmaki. ODE2VAE: Deep Generative Second Order ODEs with Bayesian Neural Networks. In *NeurIPS*. 2019.