Safe Learning and Control using Meta-Learning

Thomas Lew, James Harrison, Apoorva Sharma, Marco Pavone

I. Introduction

When deploying autonomous systems in uncertain environments or for extended durations, mismatch between a model of the system dynamics and the true dynamics is inevitable. For example, an autonomous free-flying spacecraft's thrusters may deviate from nominal behavior due to damage or interference due to debris, modeled on our hardware testbed in Fig. 1. For an autonomous agent to perform tasks in such settings, it must control a system for which the system dynamics are *uncertain*, and do so safely. Furthermore, this initial uncertainty might be too high to carry out the desired task safely, in which case autonomous agents must be able to *learn*, using data observed online to reduce the uncertainty about the system dynamics.

This problem of maintaining safety constraints while controlling uncertain systems is well-suited to model-based control approaches where the uncertain dynamics are represented using a Gaussian process (GP) model [3, 5, 8]. A related line of work has also considerd learning system dynamics while maintaining stability or safety [1, 7, 11]. However, the complexity of nonparametric kernel GPs scales poorly with the amount of observed data (or the choice of the number of inducing variables for sparse GPs [9]) and their ability to incorporate prior knowledge is limited to the choice of kernel and a handful of hyperparameters.

Recent work in *meta-learning* has emerged as a promising, data-driven alternative for online learning, using an offline training phase to imbue learning algorithms with prior knowledge needed to efficiently fit data observed online. Using nonlinear activation functions and a Bayesian output layer which is adapted online, such approaches outperform nonparametric GPs in accuracy and adaptation capabilities [4]. In this work, we present a unified framework for safe learning and adaptive control of an uncertain nonlinear system which leverages the computational and data efficiency gains of a meta-learned dynamics model. This framework combines three key technical contributions: (1) Lipschitz normalization techniques to improve the uncertainty propagation properties of the meta-learned dynamics model; (2) a tractable optimization objective for the exploration phase; (3) formulations of exploration and exploitation tasks as chance-constrained optimal control problems, which are solved using a novel sequential convex programming (SCP) algorithm.

II. Meta-Learning a Dynamics Model

The goal of this work consists in controlling an uncertain nonlinear system from an initial state $\mathbf{x}_0$ to a goal region $\mathcal{X}_{goal}$ while respecting safety constraints $\mathbf{x} \in \mathcal{X}_{free}$. The system is characterized by its unknown discrete-time dynamics $f_{\boldsymbol{\theta}}$ and subject to external disturbances $\boldsymbol{\epsilon}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ as

$$\mathbf{x}_{k+1} = f_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) + \boldsymbol{\epsilon}_k, \tag{1}$$

T. Lew is with the Institute for Dynamic Systems and Control, ETH Zürich, Zürich, Switzerland. lewt@student.ethz.ch.

J. Harrison, A. Sharma, and M. Pavone are with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305. {jharrison, apoorva, pavone}@stanford.edu.

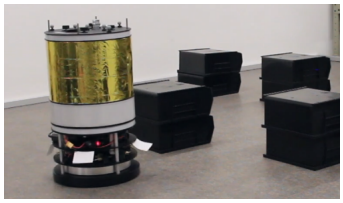

Fig. 1: Perturbed Free-Flyer system avoiding obstacles.

where $\mathbf{x}_k \in \mathbb{R}^n$ is the state, $\mathbf{u}_k \in \mathcal{U} \subset \mathbb{R}^m$ are the control inputs and $\boldsymbol{\theta} \in \boldsymbol{\Theta} \subset \mathbb{R}^{n_{\boldsymbol{\theta}}}$ are unknown latent parameters. Since the true value of $\boldsymbol{\theta}$ is initially unknown, it is necessary to use an approximate model for (1) which includes the uncertainty of its prediction, exploits prior knowledge of $f_{\boldsymbol{\theta}}$ and enables fast online adaptation to fit the true model.

In this work, we leverage the Bayesian meta-learning architecture presented in [4]. Decomposing $f_{\boldsymbol{\theta}}(\cdot)$ into a nominal model $f(\cdot)$ and an uncertain term $\mathbf{K}^T\boldsymbol{\phi}(\cdot)$, where $\boldsymbol{\phi}(\cdot)$ is a neural network and $\mathbf{K} \sim \mathcal{MN}(\bar{\mathbf{K}}, \boldsymbol{\Lambda}^{-1}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ with $\mathcal{MN}(\cdot)$ denoting the matrix normal distribution [4] and $\boldsymbol{\Lambda}$ the precision matrix, an approximate model for (1) is

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{K}^T\boldsymbol{\phi}(\mathbf{x}_k, \mathbf{u}_k) + \boldsymbol{\epsilon}_k. \tag{2}$$

With this model structure, the distribution on $\mathbf{K}$ can be efficiently updated as data is observed online. Specifically, given a measurement $\mathbf{x}_{k+1}$ from a state $\mathbf{x}_k$ using an input $\mathbf{u}_k$, a prior distribution $\mathbf{K}_k \sim \mathcal{MN}(\bar{\mathbf{K}}_k, \boldsymbol{\Lambda}_k^{-1}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ is updated:

$$\boldsymbol{\Lambda}_{k+1}^{-1} = \boldsymbol{\Lambda}_k^{-1} - \frac{1}{1+\boldsymbol{\phi}_k^T\boldsymbol{\Lambda}_k^{-1}\boldsymbol{\phi}_k}(\boldsymbol{\Lambda}_k^{-1}\boldsymbol{\phi}_k)(\boldsymbol{\Lambda}_k^{-1}\boldsymbol{\phi}_k)^T \tag{3a}$$

$$\mathbf{Q}_{k+1} = \boldsymbol{\phi}_k\mathbf{y}_{k+1}^T + \mathbf{Q}_k \qquad \text{with } \mathbf{y}_{k+1} = (\mathbf{x}_{k+1} - f(\mathbf{x}_k, \mathbf{u}_k)) \tag{3b}$$

$$\bar{\mathbf{K}}_{k+1} = \boldsymbol{\Lambda}_{k+1}^{-1}\mathbf{Q}_{k+1}, \tag{3c}$$

where $\mathbf{Q}_0 := \boldsymbol{\Lambda}_0\bar{\mathbf{K}}_0$ and $\boldsymbol{\phi}_k := \boldsymbol{\phi}(\mathbf{x}_k, \mathbf{u}_k)$. Since all the observed data is summarized in the updated distribution parameters, the computational complexity of inference in this model remains constant as a function of data observed online, in contrast to nonparametric GP approaches.

Given the parameters of the posterior distribution $\boldsymbol{\Lambda}^{-1}$ and $\bar{\mathbf{K}}$, an initial state distribution $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, and an action sequence $\mathbf{u}_0, \dots, \mathbf{u}_{N-1}$, one can use a Taylor approximation to approximate the distribution over future states. Denoting $\boldsymbol{\Sigma}_k := \text{Var}\{\mathbf{x}_k\}$, $\bar{f} := f(\boldsymbol{\mu}_k, \mathbf{u}_k)$ and $\bar{\boldsymbol{\phi}} := \boldsymbol{\phi}(\boldsymbol{\mu}_k, \mathbf{u}_k)$, we write

$$\boldsymbol{\mu}_{k+1} = f(\boldsymbol{\mu}_k, \mathbf{u}_k) + \mathbf{K}^T\boldsymbol{\phi}(\boldsymbol{\mu}_k, \mathbf{u}_k) \tag{4a}$$

$$\boldsymbol{\Sigma}_{k+1} = \left(1 + \bar{\boldsymbol{\phi}}^T\boldsymbol{\Lambda}^{-1}\bar{\boldsymbol{\phi}}\right)\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} + \text{Tr}\left(\boldsymbol{\nabla}\bar{\boldsymbol{\phi}}^T\boldsymbol{\Lambda}^{-1}\boldsymbol{\nabla}\bar{\boldsymbol{\phi}}\boldsymbol{\Sigma}_k\right)\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} + \boldsymbol{\nabla}(\bar{f} + \bar{\mathbf{K}}^T\bar{\boldsymbol{\phi}})\boldsymbol{\Sigma}_k\boldsymbol{\nabla}(\bar{f} + \bar{\mathbf{K}}^T\bar{\boldsymbol{\phi}})^T. \tag{4b}$$

Note that since the initial state $\mathbf{x}_0$ is known perfectly, the first prediction of the variance is exact and is given by the first term of (4b), which was derived in [4]. Also, it is possible to reduce the uncertainty propagation by using a nominal linear state-feedback controller as in [5], which is done in this work.

The weights $\mathbf{W}$ of the neural network $\boldsymbol{\phi}$ and the prior parameters $\bar{\mathbf{K}}_0, \boldsymbol{\Lambda}_0$ are learned in an offline meta-training phase: Using a dataset of trajectories obtained by sampling

from a distribution of possible dynamics models in a simulator, we use part of the trajectory together with Eqs. (3) to obtain the posterior on $\mathbf{K}$. Using this posterior and Eqs. (4), we compute a distribution of the predictions for the remainder of the trajectory. The negative log likelihood of the realized trajectory under this distribution serves as the loss function for the offline meta-training phase, thereby allowing us to learn weights for $\boldsymbol{\phi}$ that capture the structure in the problem domain, and a prior on $\mathbf{K}$ that captures the uncertainty of the model.

However, when performing uncertainty propagation in the planning phase, some techniques rely on the Lipschitz constant of the approximate model (e.g., in [7]) which, for a model with $\tanh(\cdot)$ activation functions, is bounded by the product of the maximum singular values of the layer weights. To make the approximate model more amenable to Lipschitz-based uncertainty propagation, we extend the training approach from [4] by constraining the maximum singular values of each layer's weights $\mathbf{W}_j$ during the offline training procedure:

$$\begin{aligned} \min \quad & Loss(\bar{\mathbf{K}}_0, \boldsymbol{\Lambda}_0, \boldsymbol{\phi}) && \text{(5a)} \\ \text{subject to} \quad & \text{meta-learning constraints [4]} && \text{(5b)} \\ & \sigma_{max}(\mathbf{W}_j) \leq \bar{\sigma}_{\boldsymbol{\phi}^j}, \quad \forall j = 1, \dots l && \text{(5c)} \\ & \sigma_{max}(\bar{\mathbf{K}}_0) \leq \bar{\sigma}_{\bar{\mathbf{K}}}. && \text{(5d)} \end{aligned}$$

III. Exploration - Exploitation Algorithm

A. Exploration Objective

The exploration objective can be expressed as the mutual information $I(\cdot)$ between the unknown function $f_{\boldsymbol{\theta}}$ and an observation $\mathbf{x}_{k+1}$, characterizing the *information gain* [10] from observing $\mathbf{x}_{k+1}$. Excluding the derivation due to space constraints, the expected uncertainty reduction resulting from observing $\mathbf{x}_{k+1}$ can be approximated as

$$\mathbb{E}\{I(\mathbf{x}_{k+1}; f_{\boldsymbol{\theta}})\} \approx \frac{1}{2}\left(\log(1+\bar{\boldsymbol{\phi}}^T \boldsymbol{\Lambda}^{-1} \bar{\boldsymbol{\phi}})\right). \qquad (6)$$

Note that the complexity of evaluating the exploration cost above and its gradient are constant. This is an advantage compared to GPs and Sparse GPs [9], where complexity scales with the number of observed data points or inducing variables, respectively.

B. Chance-Constrained Trajectory Optimization

Using this model which captures $f_{\boldsymbol{\theta}}$ with high probability, both the exploration and exploitation tasks can be written as

$$\begin{aligned} \min_{\boldsymbol{\mu}, \mathbf{u}} \quad & l_f(\boldsymbol{\mu}_N) + \sum_{k=0}^{N-1} l(\boldsymbol{\mu}_k, \mathbf{u}_k) && && \text{(7a)} \\ \text{s.t.} \quad & \boldsymbol{\mu}_{k+1} = (f + \bar{\mathbf{K}}\boldsymbol{\phi})(\boldsymbol{\mu}_k, \mathbf{u}_k) && \forall k = 0, \dots, N-1 && \text{(7b)} \\ & \Pr(\mathbf{x}_k \in \mathcal{X}_{free}) \geq p_x && \forall k = 1, \dots, N-1 && \text{(7c)} \\ & \Pr(\mathbf{x}_N \in \mathcal{X}_f) \geq p_x && && \text{(7d)} \\ & \Pr(\mathbf{u}_k \in \mathcal{U}) \geq p_u && \forall k = 0, \dots, N-1 && \text{(7e)} \\ & \mathbf{x}_0 = \mathbf{x}(0), && && \text{(7f)} \end{aligned}$$

where $l(\cdot), l_f(\cdot)$ are positive cost functions, $\mathcal{X}_{free}$ is the set of states satisfying all constraints (velocity bounds, obstacle-free, ...), $\mathcal{X}_f \subset \mathcal{X}_{free}$ is the terminal set, p_x, p_u are probability thresholds and $\mathbf{x}(0) \in \mathcal{X}_0 \subset \mathcal{X}_{free}$, with $\mathcal{X}_0$ the initial region. We assume that all sets, including the positions of the obstacles, are perfectly known, static and given to the planner.

To solve this problem, we use an extension of GuSTO [2]. By propagating the uncertainty and convexifying non-convex terms at each SCP iteration, this problem is solved efficiently and this method can handle different uncertainty propagation methods (e.g., using (4) or leveraging Lipschitz continuity as in [7]). This is in contrast to previous GP-MPC approaches, where the uncertainty of the trajectory is often only approximated along a nominal trajectory.

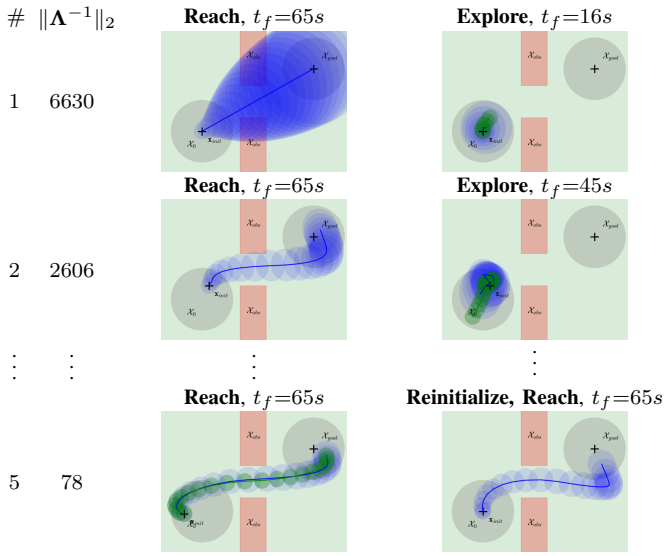

Fig. 2: Sequential Exploration - Exploitation algorithm. In blue: planned trajectory with the p_x-confidence ellipsoidal regions. In green: true trajectory after executing a control input sequence, where the robot is represented as a green cylinder.

C. Dual Algorithm

Starting from an initially highly uncertain prior model $(\bar{\mathbf{K}}_0, \boldsymbol{\Lambda}_0)$, it may be unfeasible to compute a feasible trajectory to the goal. Therefore, we separate the learning and reaching problem into two distinct exploration and exploitation phases that are successively solved until the end region $\mathcal{X}_{goal}$ is reached. The exploitation problem consists in reaching the end region $\mathcal{X}_f = \mathcal{X}_{goal}$ while minimizing the actuation effort $\sum_k \|\mathbf{u}_k\|$. The exploration problem consists in maximizing the sum of information gains in (6) with the end state being in the initial safe region $\mathcal{X}_f = \mathcal{X}_0$. Since $\boldsymbol{\Lambda}$ is kept constant to evaluate (6), this objective is only an approximation for the maximization of the gain of information along the trajectory. At the end of each phase, the resulting trajectory is used to update the model parameters $(\bar{\mathbf{K}}, \boldsymbol{\Lambda})$ using (3). By assuming a Probabilistic Invariant Set for $\mathcal{X}_0$ [6] and an optimization horizon (N, t_f) ensuring the existence of a solution to the chance-constrained problem, we can guarantee that the system remains safe at all times with high probability.

IV. Results and Conclusions

In Fig. 2, we show our exploration / exploitation algorithm. Using a perturbed free-flyer system shown on Fig. 1, the goal consists in traversing a terrain with obstacles. At first, the problem is unfeasible due to high uncertainties. Therefore, it is necessary to explore with a short planning horizon. After executing each control input sequence, the resulting trajectory is used to perform regression and we show that the norm of the covariance matrix $\boldsymbol{\Lambda}^{-1}$ rapidly decreases. At iteration #5, the problem is feasible and $\mathcal{X}_{goal}$ is reached. On the last figure, we re-initialize the system with the inferred model and safely reach the goal region without further exploration, which was initially impossible. All constraints are satisfied at all time, verifying the probabilistic safety properties of our method.

ACKNOWLEDGMENTS

T. Lew was partially supported by the Master's Thesis Grant of the Zeno Karl Schindler Foundation. This work was supported by the Office of Naval Research YIP program (Grant N00014-17-1-2433), by DARPA under the Assured Autonomy program, and by the Toyota Research Institute (TRI). This article solely reflects the opinions and conclusions of its authors and not ONR, DARPA, TRI or any other Toyota entity. James Harrison was supported in part by the Stanford Graduate Fellowship and the National Sciences and Engineering Research Council (NSERC).

REFERENCES

[1] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause. Safe model-based reinforcement learning with stability guarantees. In *Conf. on Neural Information Processing Systems*, 2017.

[2] R. Bonalli, A. Cauligi, A. Bylard, and M. Pavone. GuSTO: Guaranteed sequential trajectory optimization via sequential convex programming. In *Proc. IEEE Conf. on Robotics and Automation*, 2019.

[3] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems, 2017. Available at https://arxiv.org/abs/1705.01292.

[4] J. Harrison, A. Sharma, and M. Pavone. Meta-learning priors for efficient online bayesian regression. In *Workshop on Algorithmic Foundations of Robotics*, 2018. In Press.

[5] Juraj Hewing, Lukas Kabzan and Melanie N. Zeilinger. Cautious Model Predictive Control using Gaussian Process Regression. 2018. Available at https://arxiv.org/abs/1705.10702.

[6] L. Hewing, A. Carron, K. P. Wabersich, and M. N. Zeilinger. On a correspondence between probabilistic and robust invariant sets for linear systems. In *European Control Conference*, 2018.

[7] Torsten Koller, Felix Berkenkamp, Matteo Turchetta, and Andreas Krause. Learning-based model predictive control for safe exploration. 2018.

[8] Chris J. Ostafew, Angela P. Schoellig, and Timothy D. Barfoot. Robust Constrained Learning-based NMPC enabling reliable mobile robot path tracking. *Int. Journal of Robotics Research*, 2016.

[9] Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 2005.

[10] Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Int. Conf. on Machine Learning*, 2010.

[11] Li Wang, Evangelos A. Theodorou, and Magnus Egerstedt. Safe Learning of Quadrotor Dynamics Using Barrier Certificates. In *Proc. IEEE Conf. on Robotics and Automation*. IEEE, 2018.