Safe Sequential Exploration and Exploitation

Thomas Lew, Apoorva Sharma, James Harrison, Marco Pavone
Stanford University, Stanford, CA, 94305
{thomas.lew,apoorva,jharrison,pavone}@stanford.edu

Abstract

To safely deploy learning-based systems in highly uncertain environments, one must ensure that they will always satisfy constraints. This work proposes **SEELS**: a model-based meta-reinforcement learning framework to tackle this problem. By opting for a Bayesian meta-learning model with linear uncertainty, we derive confidence sets for the parameters which hold at all times with high probability. We then propose an algorithm consisting of distinct exploration and exploitation phases, to tackle problems with high dynamics uncertainty, for which reaching a goal safely is initially unfeasible. By leveraging a new uncertainty propagation technique rooted in random set theory, and by deriving a new regularizer for our Bayesian model, our approach scales to higher dimensional systems than previous work. Under reasonable assumptions, we prove that our framework provides strong probabilistic safety guarantees in the form of a single joint chance constraint.

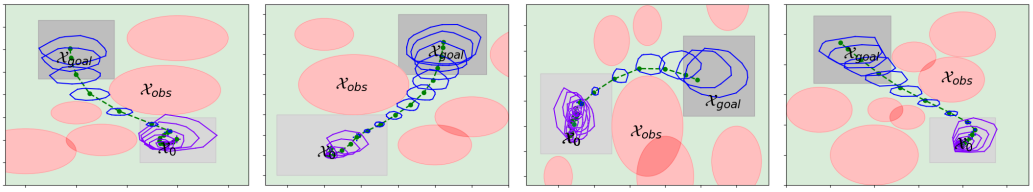

Figure 1: Initially, uncertainty is too high to safely reach the goal, Instead, we plan safe information-gathering trajectories to infer the dynamics and reduce uncertainty. Once planning to $\mathcal{X}_{\text{goal}}$ is feasible, the robot can safely reach the goal while satisfying all constraints with high probability.
Color legend: true trajectory, reachable sets for exploration (6), reachable sets for exploitation.

1 Introduction

Deployment of truly autonomous robotic systems in changing and unpredictable environments requires agents that are capable of learning during operation and safely adapting to new environments. Reinforcement learning (RL) can be an effective approach to controlling uncertain systems (Hwangbo et al., 2019), and model-based methods in particular enable an agent to consider its uncertainty over dynamics when choosing actions (Deisenroth et al., 2015). However, standard model-based reinforcement learning (MBRL) methods do not provide guarantees on maintaining safety during operation. Existing work on safety in MBRL has developed algorithms with strong theoretical guarantees, but has either been limited to linear systems (Dean et al., 2019), or utilized kernel-based dynamics models which struggle to scale with state dimension, and uncertainty propagation schemes that can be too conservative or too slow for practical use (Koller et al., 2018; Lew and Pavone, 2020).

We tackle this problem by leveraging Bayesian meta-learning and sampling-based reachability analysis to develop a framework for nonlinear MBRL that is practically useful and probabilistically safe. To handle high levels of initial uncertainty, our approach decouples online learning to reducing dynamics uncertainty (the exploration phase) and executing the desired task (the exploitation phase).

NeurIPS 2020 Workshop on Robot Learning: Grounding Machine Learning Development in the Real World

2 Problem Formulation: Safe Navigation to a Goal

The goal of this work is to enable robots to safely navigate from an initial state $\boldsymbol{x}(0)$ to a goal region $\mathcal{X}_{\text{goal}}$ despite highly uncertain dynamics, while minimizing a chosen cost $l(\cdot)$ (e.g., fuel consumption). We denote the state of the agent as $\boldsymbol{x}_k \in \mathbb{R}^n$, and $\boldsymbol{u}_k \in \mathbb{R}^m$ denotes the control inputs. The system follows dynamics $\boldsymbol{x}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}) + \boldsymbol{\epsilon}_k$, where $\boldsymbol{h}(\cdot)$ is known, $\boldsymbol{g}$ is unknown, $\boldsymbol{\theta}$ are parameters, and $\boldsymbol{\epsilon}_k$ are stochastic (bounded sub-Gaussian) disturbances. We assume the (unobserved) parameters are sampled $\boldsymbol{\theta}_j \sim p(\boldsymbol{\theta})$ at the beginning of each episode j, and fixed throughout its duration. They correspond to uncertain properties of the system, e.g. the inertia of a payload.

Critically, this algorithm should guarantee safety *at all times* by respecting system constraints ($\boldsymbol{x}_k \in \mathcal{X}_{\text{free}}$, $\boldsymbol{u}_k \in \mathcal{U}$, where $\mathcal{X}_{\text{free}}, \mathcal{U}$ are feasible state and control spaces). Due to the stochasticity of the system and the uncertain dynamics, strictly enforcing all constraints for all times may be challenging without further assumptions, e.g., bounded model mismatch. Instead, we enforce all constraints with a *single joint chance constraint* at probability level $(1-\delta) \in (0, 1)$. The problem is

$$\min_{\boldsymbol{x}, \boldsymbol{u}} \ \mathbb{E}\Big(\sum_{k=0}^{N} l(\boldsymbol{x}_k, \boldsymbol{u}_k) \Big), \qquad \boldsymbol{x}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}) + \boldsymbol{\epsilon}_k, \quad \boldsymbol{x}_0 = \boldsymbol{x}(0), \tag{1a}$$

$$\mathbb{P}\Big(\bigwedge_{k=1}^{N} \big(\boldsymbol{x}_k \in \mathcal{X}_{\text{free}}\big) \ \cap \ \bigwedge_{k=0}^{N-1} \big(\boldsymbol{u}_k \in \mathcal{U}\big) \ \cap \ \big(\boldsymbol{x}_N \in \mathcal{X}_{\text{goal}}\big) \Big) \geq (1-\delta), \tag{1b}$$

where N is the total duration of the problem (possibly infinite). Satisfying safety constraints with unknown dynamics at all times is extremely difficult without further information (Koller et al., 2018):

Assumption 1 (A1). $\boldsymbol{x}(0) \in \mathcal{X}_0 \subset \mathcal{X}_{\text{free}}$, where $\mathcal{X}_0$ is a control invariant set and we have a feedback controller $\boldsymbol{\pi}(\cdot) : \mathcal{X}_0 \to \mathcal{U}$ under which it is possible to remain in $\mathcal{X}_0$ for all $\boldsymbol{\theta}$ and $\boldsymbol{\epsilon}$.

This assumption reflects that the system is initially stable and satisfies all constraints under a nominal controller (e.g., regulated to a stable linearization point using a simple feedback law such as LQR).

Further, we assume that we have access to a dataset of plausible trajectories generated from sampled parameters $\boldsymbol{\theta}_j$. Such information may come from, for example, previous operation of a robot in similar environments, or data generated from simulations with different parameters. This motivates our use of meta-learning to encode this information and characterize the uncertainty over dynamics.

3 Bayesian Meta-Learning and Adaptation Guarantees

Bayesian Meta-Learning: Our approach leverages a model for the unknown portion of system dynamics $\boldsymbol{g}$. To this end, we employ the architecture presented in (Harrison et al., 2018a,b), which the authors refer to as ALPaCA. It models the unknown dynamics as $\hat{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{K}\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})$, where $\boldsymbol{\phi}$ is a feed-forward neural network, and $\boldsymbol{K}$ is an uncertain matrix. This linear structure allows for efficient online updates whose behavior is well understood. Given transitions $\{(\boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{x}_1), \ldots, (\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{x}_{t+1})\}$, we update the parameters of each i-th row of $\boldsymbol{K}$ using linear regression as

$$\boldsymbol{\Lambda}_{i,t} = \Phi_{t-1}^T \Phi_{t-1} + \boldsymbol{\Lambda}_{i,0}, \qquad \bar{\boldsymbol{k}}_{i,t} = \boldsymbol{\Lambda}_{i,t}^{-1}(\Phi_{t-1}^T \boldsymbol{G}_{i,t} + \boldsymbol{\Lambda}_{i,0}\bar{\boldsymbol{k}}_{i,0}), \qquad i = 1, \ldots, n, \tag{2}$$

where $\boldsymbol{G}_t^T = [\boldsymbol{x}_1 - \boldsymbol{h}(\boldsymbol{x}_0, \boldsymbol{u}_0), \ldots, \boldsymbol{x}_t - \boldsymbol{h}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1})]$, and $\Phi_{t-1}^T = [\boldsymbol{\phi}(\boldsymbol{x}_0, \boldsymbol{u}_0), \ldots, \boldsymbol{\phi}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1})]$.

Offline, this model is meta-trained on a dataset of trajectories corresponding to different system dynamics sampled from the distribution over possible systems. By backpropagating the posterior predictive distribution to learn $\boldsymbol{\phi}$ and the prior parameters $(\bar{\boldsymbol{k}}_{i,0}, \boldsymbol{\Lambda}_{i,0})$, this model translates a dataset of uncertain trajectories into a learned feature space and a calibrated uncertainty characterization.

Online, only the last layer $\boldsymbol{K}$ is adapted, which enables the derivation of strong safety guarantees.

Probabilistic Adaptation Guarantees Our first contribution consists of providing strong probabilistic adaptation guarantees for this model in the form of uniformly calibrated confidence sets:

Theorem 1. *Consider the true system* (1a)*, with σ_ϵ-subgaussian bounded noise, modeled using the meta-learning model $\hat{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{K}\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})$, which is sequentially updated with online data from* (1a) *using* (2)*, leading to the updated parameters $(\bar{\boldsymbol{k}}_{i,t}, \boldsymbol{\Lambda}_{i,t})$ for each dimension $i = 1, \ldots, n$. Let $\delta_i \in (0, 1)$ a confidence threshold. Assume that the following conditions hold:*

For all $\boldsymbol{\theta}$, there exists $\boldsymbol{k}_i^*$ such that $\boldsymbol{k}_i^* \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{g}_i(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\theta})\ \forall \boldsymbol{x} \in \mathcal{X}, \boldsymbol{u} \in \mathcal{U}, i = 1, \ldots, n.$ ***(A2)***

For $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$, and any $i = 1, \ldots, n$, $\mathbb{P}\big(\|\boldsymbol{k}_i^* - \bar{\boldsymbol{k}}_{i,0}\|^2_{\boldsymbol{\Lambda}_{i,0}} \leq \sigma^2_{\epsilon_i} \chi^2_d(1-\delta_i)\big) \geq (1-\delta_i).$ ***(A3)***

Let

$$\beta_i(\boldsymbol{\Lambda}_{i,t}, \delta_i) = \sigma_{\epsilon_i} \left(\sqrt{2 \log \left(\frac{1}{\delta_i} \frac{\det(\boldsymbol{\Lambda}_{i,t})^{1/2}}{\det(\boldsymbol{\Lambda}_{i,0})^{1/2}} \right)} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_{i,0})}{\lambda_{\min}(\boldsymbol{\Lambda}_{i,t})} \chi^2_d(1-\delta_i)} \right), \tag{3}$$

and

$$\mathcal{C}^{\delta}_{i,t}(\bar{\boldsymbol{k}}_{i,t}, \boldsymbol{\Lambda}_{i,t}) = \{\boldsymbol{k}_i \mid \left\| \boldsymbol{k}_i - \bar{\boldsymbol{k}}_{i,t} \right\|_{\boldsymbol{\Lambda}_{i,t}} \leq \beta_i(\boldsymbol{\Lambda}_{i,t}, \delta_i)\}. \tag{4}$$

Then,

$$\mathbb{P}\left(\boldsymbol{k}_i^* \in \mathcal{C}^{\delta}_{i,t}(\bar{\boldsymbol{k}}_{i,t}, \boldsymbol{\Lambda}_{i,t}) \quad \forall t \geq 0\right) \geq (1 - 2\delta_i). \tag{5}$$

This result relies on two key assumptions on the quality of the *offline* meta-learning process: **A2** states that the meta-learning model is capable of fitting the true dynamics, and **A3** that the prior uncertainty characterization is conservative. If the dataset has adequate coverage of the state and action spaces and the dynamics distribution $p(\boldsymbol{\theta})$, the offline meta-learning procedure proposed in (Harrison et al., 2018a) can approach satisfaction of these assumptions, which we discuss in detail in the Appendix.

Derived using results from the literature on linear contextual bandits (Abbasi-Yadkori et al., 2011), Theorem 1 provides confidence set over model parameters which hold *uniformly over all future times*. This is critical to ensure satisfaction of (1b), despite an unknown final time N. This scaling factor β_i is closely related to that used for kernel Gaussian Processes, for which the value of β_i is often too large for practical use and set to a lower value for experiments (Berkenkamp et al., 2017). In contrast, we directly use this theoretical bound in our framework, and can regularize properties that influence β_i during offline meta-learning to obtain better performance without compromising safety.

4 Sequential Exploration and Exploitation for Learning Safely (SEELS)

In order to ensure overall safety, i.e. by satisfying (1b) until $\mathcal{X}_{\text{goal}}$ is reached, we require confidence *tubes* over trajectories. Indeed, enforcing a chance constraint at each timestep, as in (Hewing et al., 2018; Polymenakos et al., 2020; Lew et al., 2020; Khojasteh et al., 2020; Cheng et al., 2020), does not guarantee safety of the whole trajectory. We construct these tubes using the confidence sets from Theorem 1: given the control inputs $\boldsymbol{u} = (\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N-1})$[1], we define the sequence of reachable sets

$$\mathcal{X}_k^{t,\delta}(\boldsymbol{u}) = \left\{ \boldsymbol{x}_k = \boldsymbol{f}(\cdot, \boldsymbol{u}_{k-1}, \boldsymbol{K}, \boldsymbol{\epsilon}_{k-1}) \circ \ldots \circ \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{K}, \boldsymbol{\epsilon}_0) \;\middle|\; \begin{array}{l} \boldsymbol{x}_0 = \boldsymbol{x}(t),\ \boldsymbol{k}_i \in \mathcal{C}^{\delta}_{i,t},\ \boldsymbol{\epsilon}_j^i \in \mathcal{E}_i, \\ j = 1, \ldots, k-1,\ \ i = 1, \ldots, n \end{array} \right\}, \tag{6}$$

where $k = 1, \ldots, N$, and $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{K}, \boldsymbol{\epsilon}) = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{K}\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\epsilon}$. The construction of these confidence sets enables one to consider $\boldsymbol{x}_k \in \mathcal{X}_k^{t,\delta}$ only, and relax the generally intractable chance-constrained stochastic problem in (1). We follow this approach and transcribe **(CC-OCP)** into a deterministic problem that can be efficiently solved by a general purpose non-convex solver:

$$\min_{\boldsymbol{\mu}, \boldsymbol{u}} \sum_{k=0}^{N} l(\boldsymbol{\mu}_k, \boldsymbol{u}_k), \quad \text{s.t.} \bigwedge_{k=1}^{N} \mathcal{X}_k^{t,\delta} \subset \mathcal{X}_{\text{free}}, \quad \bigwedge_{k=0}^{N-1} \boldsymbol{u}_k \in \mathcal{U}, \quad \mathcal{X}_N^{t,\delta} \subset \mathcal{X}_{\text{f}}, \quad \mathcal{X}_0^{t,\delta} = \{\boldsymbol{x}(t)\}, \tag{7}$$

where $\boldsymbol{\mu} = (\boldsymbol{\mu}_0, \ldots, \boldsymbol{\mu}_N)$ are the centers of the reachable sets $\{\mathcal{X}_k^{t,\delta}\}_{k=1}^N$, see (Lew and Pavone, 2020).

SEELS: Due to high dynamics uncertainty, tight control constraints, and long planning horizons, (7) may be infeasible. This motivates a safe learning-based exploration-exploitation framework to sequentially reduce uncertainty, and eventually reach $\mathcal{X}_{\text{goal}}$. Our approach is based on a repeated two-phase approach: when (7) is feasible with $\mathcal{X}_{\text{f}} = \mathcal{X}_{\text{goal}}$, we enter the *exploitation* phase, and plan a safe trajectory to $\mathcal{X}_{\text{goal}}$ with the current model uncertainty. In the *exploration* phase, we instead strictly perform safe exploration, planning an information-gathering trajectory that returns with high probability to the initial safe invariant set $\mathcal{X}_0$. This split yields a tractable sequence of trajectory optimization problems, although it induces sub-optimality relative to the computationally intractable problem of simultaneously trading off exploration and exploitation (Bar-Shalom and Tse, 1974). Our information cost is derived from the mutual information between the unknown dynamics and the

[1] Accounting for a nominal feedback controller can be used to reduce the size of this tube and is a simple extension. In this work, we omit feedback to better demonstrate the adaptation capabilities of the meta-training model, the tightness of the confidence sets, and to better verify safety claims of the framework.

observations, leveraging the current predictive uncertainty of the model. Notably, its computational complexity does not scale with the amount of data, as is the case for similar objectives for kernel Gaussian processes (Koller et al., 2018; Williams and Rasmussen, 2006). **SEELS** guarantees that the agent is always able to find feasible trajectories and ensure safety at all times with high probability:

Theorem 2. *Under assumption **A1-3**, apply **SEELS** to sequentially explore. Then, there exists an horizon N ensuring the feasibility of each exploration phase at all times[2] with probability $(1-\delta)$.*

Further, assuming that the exploitation problem is feasible at some time[3], the system is guaranteed to satisfy (1b)*, i.e., to be safe at all times and eventually reach $\mathcal{X}_{goal}$ with probability $(1-\delta)$.*

Practical considerations: Implementation of **SEELS** is complicated by challenges in reachability analysis and nonconvex optimization. First, evaluating (6) over multiple timesteps is difficult due to the nonconvexity of ϕ. Further, the updates to the parameters $\boldsymbol{k}_i$ preclude exact offline methods (Bansal et al., 2017; Fan et al., 2020), and methods using Lipschitz continuity to conservatively propagate these sets (Koller et al., 2018) are too conservative in practice. In this work, we leverage **randUP**, a recently derived sampling-based uncertainty propagation scheme for approximate reachability analysis (Lew and Pavone, 2020). We follow this method to formulate (7). This nonconvex optimization problem is then solved through sequential convex programming, which entails solving a sequence of convex reformulations. As the feasibility of (7) depends on the planning horizon N, we also perform a search over a predefined range of values. Further details are provided in the Appendix.

Results: We verify our proposed approach on a nonlinear six-dimensional planar free-flyer robot with tight control constraints navigating in a cluttered environment. The goal consists of safely transporting an uncertain payload, which causes a change in mass and inertial properties (including the location of the center of mass), to a goal region.

We validate the safety and reliability of our framework on a batch of 250 problems with randomized dynamics, obstacle configurations, and initial and final conditions. We compare the sensitivity to the noise magnitude, to the number of samples for reachability analysis, to δ, and to the regularization of β_i. Figure 1 shows illustrative experiments. Results in Figure 2 show that **SEELS** reliably solves this problem for multiple obstacle fields. In particular, (1b) is conservatively satisfied in practice, and the system reaches the goal safely after a few exploration phases. In comparison, a naive approach which only considers uncertainty in $\boldsymbol{\epsilon}_k$ deems reaching $\mathcal{X}_{\text{goal}}$ directly to be safe, and violates safety constraints 80% of the time. This demonstrates the need for sequential online learning to reliably solve this problem. Further, we observe that increasing M does lead to increased success rate and probability of safety. Therefore, by Theorem 2, success is guaranteed as long as the number of samples for reachability analysis M is high enough. Moreover, the conservatism of the algorithm can be tuned by choosing a different value for δ: by opting for lower probability of safety, $\mathcal{X}_{\text{goal}}$ is reached faster in average. Finally, regularizing β_i reduces conservatism, while still guaranteeing probabilistic safety in practice. This correlates both with less conservatism, and faster time to reach $\mathcal{X}_{\text{goal}}$.

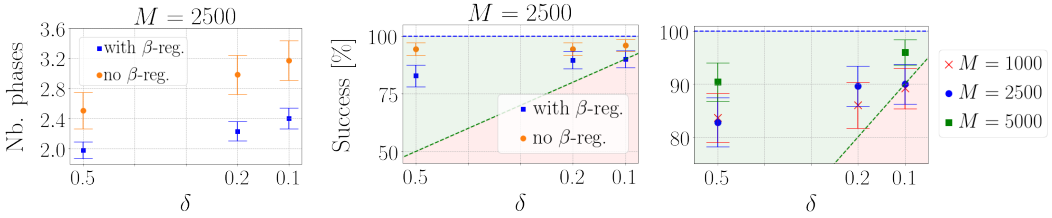

Figure 2: Results for 250 randomized experiments. On plots showing success percentages (all constraints are satisfied and $\boldsymbol{x}_N \in \mathcal{X}_{\text{goal}}$), the green region denotes results with a success percentage at or above the desired success probability $(1-\delta)$ (and vice versa for the red region). Error bars correspond to 95% confidence intervals.

Conclusion: **SEELS** provides a principled MBRL framework to reliably learn and perform tasks while guaranteeing safety at all times with high probability. Our combination of meta-learning with assumptions **A2-3** motivates two directions of future work: safety analysis with feature mismatch, and finite sample guarantees for reachability analysis of uncertain meta-learning models.

[2]This is in contrast to related work on model predictive control which provides probabilistic feasibility over a finite horizon (Ono, 2012). The key consists of exploiting confidence sets which hold jointly for all times.

[3]Assuming that the original problem is feasible with perfect dynamics knowledge, this condition holds if the objective used for exploration leads to actions that continually reduce uncertainty, related to conditions on observability and persistence of excitation (Berberich et al., 2020; Coulson et al., 2018; Mania et al., 2020).

Acknowledgements This work was supported in part by NASA under the University Leadership Initiative and the Early Stage Innovations program, by the Office of Naval Research YIP program, and by DARPA under the Assured Autonomy program.

References

J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26), 2019.

M. Deisenroth, D. Fox, and C. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 37(2):408–423, 2015.

S. Dean, S. Tu, N. Matni, and B. Recht. Safely learning to control the constrained linear quadratic regulator. In *American Control Conference*, 2019.

T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause. Learning-based model predictive control for safe exploration. In *Proc. IEEE Conf. on Decision and Control*, 2018.

T. Lew and M. Pavone. Sampling-based reachability analysis: A random set theory approach with adversarial sampling, 2020. Available at `https://arxiv.org/abs/2008.10180`.

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

J. Harrison, A. Sharma, R. Dyro, X. Wang, R. Calandra, and M. Pavone. Control adaptation via meta-learning dynamics. In *NeurIPS Workshop on Meta-Learning*, 2018b.

Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In *Conf. on Neural Information Processing Systems*, 2011.

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.

L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious Model Predictive Control using Gaussian Process Regression. *IEEE Transactions on Control Systems Technology*, 2018. Early Access.

K. Polymenakos, L. Laurenti, A. Patane, J. P. Calliess, L. Cardelli, M. Kwiatkowska, A. Abate, and S. Roberts. Safety guarantees for planning based on iterative Gaussian processes, 2020. Available at `https://arxiv.org/abs/1912.00071`.

T. Lew, R. Bonalli, and M. Pavone. Chance-constrained sequential convex programming for robust trajectory optimization. In *European Control Conference*, 2020.

M. J. Khojasteh, V. Dhiman, M. Franceschetti, and N. Atanasov. Probabilistic safety constraints for learned high relative degree system dynamics. In *2nd Annual Conference on Learning for Dynamics & Control*, 2020.

R. Cheng, M. J. Khojasteh, A. D. Ames, and J. W. Burdick. Safe multi-agent interaction through robust control barrier functions with learned uncertainties, 2020. Available at `https://arxiv.org/abs/2004.05273`.

Y. Bar-Shalom and E. Tse. Dual effect, certainty equivalence, and separation in stochastic control. *IEEE Transactions on Automatic Control*, 19(5), 1974.

C. K. Williams and C. E. Rasmussen. *Gaussian processes for machine learning*. MIT press, 2006.

M. Ono. Joint chance-constrained model predictive control with probabilistic resolvability. In *American Control Conference*, 2012.

J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer. Robust constraint satisfaction in data-driven MPC, 2020. Available at `https://arxiv.org/abs/2003.06808`.

J. Coulson, J. Lygeros, and F. Dörfler. Data-enabled predictive control: In the shallows of the DeePC. 2018.

H. Mania, M. I. Jordan, and B. Recht. Active learning for nonlinear system identification with guarantees, 2020. Available at https://arxiv.org/abs/2006.10277.

S. Bansal, S. L. Chen, M. Herbert, and C. J. Tomlin. Hamilton-Jacobi reachability: A brief overview and recent advances. In *Proc. IEEE Conf. on Decision and Control*, 2017.

D. D. Fan, A. Agha-mohammadi, and E. A. Theodorou. Deep learning tubes for tube MPC. In *Robotics: Science and Systems*, 2020.

N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Int. Conf. on Machine Learning*, 2010.

D. J. C. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992.

A. Chowdhury, S. R. Gopalan. On kernelized multi-armed bandits. In *Int. Conf. on Machine Learning*, 2017.

S. Kakade, A. Krishnamurthy, K. Lowrey, M. Ohnishi, and W. Sun. Information theoretic regret bounds for online nonlinear control, 2020. Available at https://arxiv.org/abs/2006.12466.

R. Amit and R. Meir. Meta-learning by adjusting priors based on extended pac-bayes theory. In *Int. Conf. on Machine Learning*, 2018.

V. Kuleshov, N. Fenner, and S. Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Int. Conf. on Machine Learning*, 2018.

F. Berkenkamp. *Safe Exploration in Reinforcement Learning: Theory and Applications in Robotics*. PhD thesis, Institute for Machine Learning, ETH Zürich, 2018.

B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1):253–279, 2019.

S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17:1–40, 2016.

J. Coulson, J. Lygeros, and F. Dörfler. Distributionally robust chance constrained data-enabled predictive control, 2020. Available at https://arxiv.org/abs/2006.01702.

G. Shi, X. Shi, M. O'Connell, R. Yu, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung. Neural lander: Stable drone landing control using learned dynamics. In *Proc. IEEE Conf. on Robotics and Automation*, 2019.

A. Nagabandi, I. Clavera, L. Simin, R. S. Fearing, P. Abbeel, S. Levine, and C. Chelsea Finn. Learning to adapt in dynamic real-world environments through meta-reinforcement learning. In *Int. Conf. on Learning Representations*, 2019.

L. Fluckiger, K. Browne, B. Coltin, J. Fusco, T. Morse, and A. Symington. Astrobee robot software: Enabling mobile autonomy on the iss. In *Int. Symp. on Artificial Intelligence, Robotics and Automation in Space*, 2018.

M. Ekal and R. Ventura. On the accuracy of inertial parameter estimation of a free-flying robot while grasping an object. *Journal of Intelligent & Robotic Systems*, pages 1–11, 2019.

A Algorithm, β-Regularization, and Information Objective

A.1 SEELS: Full Algorithm, and Exploration-Exploitation Problems

SEELS is a two-phase approach: when the problem is feasible, we enter the *exploitation* phase; when the problem is infeasible, we instead enter the *exploration* phase. In the exploitation phase, we solve the trajectory optimization problem with the current model uncertainty. In the exploration phase, we instead strictly perform safe exploration, planning an information-gathering trajectory that returns with high probability to the safe invariant set. This split yields a tractable sequence of trajectory optimization problems. Concretely, we write the problems associated with each phase as:

(Explore-OCP)

$$\min_{\boldsymbol{\mu},\boldsymbol{u}} \quad \sum_{k=0}^{N} l_{\text{info}}(\boldsymbol{\mu}_k, \boldsymbol{u}_k) \quad \text{s.t.} \quad \mathcal{X}_N^{t,\delta} \subset \mathcal{X}_0,$$
$$\bigwedge_{k=0}^{N} \begin{pmatrix} \mathcal{X}_k^{t,\delta} \subset \mathcal{X}_{\text{free}} \\ \cap\, \boldsymbol{u}_k \in \mathcal{U} \end{pmatrix}, \quad \mathcal{X}_0^{t,\delta} = \{\boldsymbol{x}(t)\},$$

(Reach-OCP)

$$\min_{\boldsymbol{\mu},\boldsymbol{u}} \quad \sum_{k=0}^{N} l_{\text{reach}}(\boldsymbol{\mu}_k, \boldsymbol{u}_k) \quad \text{s.t.} \quad \mathcal{X}_N^{t,\delta} \subset \mathcal{X}_{\text{goal}},$$
$$\bigwedge_{k=0}^{N} \begin{pmatrix} \mathcal{X}_k^{t,\delta} \subset \mathcal{X}_{\text{free}} \\ \cap\, \boldsymbol{u}_k \in \mathcal{U} \end{pmatrix}, \quad \mathcal{X}_0^{t,\delta} = \{\boldsymbol{x}(t)\},$$

where $\{\mathcal{X}_k^{t,\delta}\}_{k=1}^N$ satisfy (6), and are computed using the confidence sets (4). (**Reach-OCP**) uses the cost function associated with the task, and $\mathcal{X}_{\text{goal}}$ as the desired goal set. (**Explore-OCP**) is similar, but instead uses $\mathcal{X}_0$ as the goal set, thus ensuring the system will be safe for the next phase, and uses an information gathering cost l_{info} to encourage visiting states which reduce remaining uncertainty in the dynamics. In this work, we derive l_{info} from the mutual information between the unknown dynamics and the observations, leveraging the current predictive uncertainty of the model. The specific formula and derivation for our meta-learning model are provided in A.3. Notably, this loss does not suffer from computational complexity that scales with the amount of data, as is the case for similar objectives derived for kernel Gaussian processes (Koller et al., 2018; Williams and Rasmussen, 2006; Srinivas et al., 2010).

Our approach **SEELS**, summarized in Algorithm 1, consists of sequentially learning a model of the dynamics by solving (**Explore-OCP**), before reaching $\mathcal{X}_{\text{goal}}$ whenever (**Reach-OCP**) admits a feasible solution. Due to uncertainty, the feasibility of each problem depends on the optimization horizon N. For this reason, we perform a search over a predefined range of planning horizons. For exploitation, we select the first feasible solution if one exists, although other criteria could be used, e.g., minimal control cost. For exploration, we select the trajectory which leads to the largest expected information gain. Indeed, due to tight control constraints and safety constraints, a larger horizon does not necessarily lead to higher information gain. This heuristic works well in practice, and future work will consist of adopting a continuous time problem formulation with free final time, which is an active field of research.

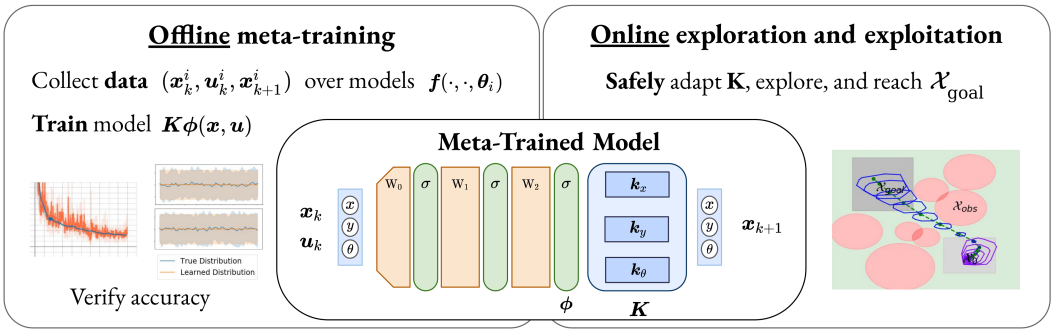

Figure 3: To guarantee safety at all times and reach a goal region $\mathcal{X}_{\text{goal}}$ despite uncertain dynamics $\boldsymbol{f}(\cdot,\cdot,\boldsymbol{\theta}_i)$, our framework consists of an offline phase, where a dataset over multiple models is used to meta-train an uncertain Bayesian meta-learning model of the system. Then, it is deployed and the system safely adapts the last layer $\boldsymbol{K}$ of the model. Using **SEELS**, the agent autonomously explores the environment to decrease its uncertainty, and safely reaches $\mathcal{X}_{\text{goal}}$ with high probability.

We leverage a sampling-based reachability analysis approach to compute approximations of the reachable sets (6), which the authors refer to as **randUP** (Lew and Pavone, 2020). By sampling parameters $(\boldsymbol{k}_i, \boldsymbol{\epsilon}_k)$ within their confidence sets and bounds, evaluating the resulting reachable states $\boldsymbol{x}$ for these parameters, and taking their convex hull to approximate the reachable sets in (6), it provides a scalable approach to efficiently compute these tubes with no assumptions on the system apart from differentiability. Although this method lacks finite time guarantees of conservatism, asymptotic guarantees can be derived using random set theory, and finite-time approximations are generally sufficient to ensure empirical safety, as demonstrated in the results. (**Explore-OCP**) and (**Reach-OCP**) are nonconvex optimal control problems which we solve through a direct method based on sequential convex programming (SCP). In this work, we always initialize SCP with an infeasible straight-line trajectory.

A.2 Regularizing Meta-training for Safe Online Learning

The size of the confidence sets for the model parameters $\boldsymbol{k}_i$ is controlled by the term β_i, which depends on the structure of the problem. Specifically, by relying on the expressiveness of the meta-learned features $\boldsymbol{\phi}(\cdot,\cdot)$, parameterized by a feed-forward neural network, different set of weights for $\boldsymbol{\phi}$ and prior parameters $(\bar{\boldsymbol{k}}_{i,0}, \boldsymbol{\Lambda}_{i,0})$ could be used to parameterize the unknown dynamics, while satisfying Assumptions 2 and 3. Therefore, it is possible to modify the meta-training procedure to obtain a model with lower values of β_i, and improve performance without compromising safety.

Specifically, we note from (3) that the value of β_i depends on the ratio between the maximum and minimum eigenvalues of the prior and posterior precision matrices $\boldsymbol{\Lambda}_i$. If $\lambda_{\max}(\boldsymbol{\Lambda}_{i,0}) \leq 1$ as is typically the case in our experiments, then it holds that $\lambda_{\max}(\boldsymbol{\Lambda}_{i,0})/\lambda_{\min}(\boldsymbol{\Lambda}_{i,t}) = \lambda_{\max}(\boldsymbol{\Lambda}_{i,t}^{-1})/\lambda_{\min}(\boldsymbol{\Lambda}_{i,0}^{-1}) \leq \lambda_{\max}(\boldsymbol{\Lambda}_{i,t}^{-1})\lambda_{\min}(\boldsymbol{\Lambda}_{i,0}^{-1}) \leq \lambda_{\max}(\boldsymbol{\Lambda}_{i,t}^{-1})\lambda_{\max}(\boldsymbol{\Lambda}_{i,0}^{-1})$. Furthermore, $\lambda_{\max}(\boldsymbol{\Lambda}) \leq \sqrt{\mathrm{Tr}(\boldsymbol{\Lambda}^T\boldsymbol{\Lambda})}$. Combining with the above, we propose to regularize an upper bound of the ratio $\lambda_{\max}(\boldsymbol{\Lambda}_{i,0})/\lambda_{\min}(\boldsymbol{\Lambda}_{i,t})$ during offline meta-training:

$$\mathcal{L}_{\mathrm{reg}}(\boldsymbol{\Lambda}_{i,0}) = \alpha_{\mathrm{reg}} \sum_{i=1}^{n} \mathrm{Tr}(\boldsymbol{\Lambda}_{i,t}^{-T}\boldsymbol{\Lambda}_{i,t}^{-1})\mathrm{Tr}(\boldsymbol{\Lambda}_{i,0}^{-T}\boldsymbol{\Lambda}_{i,0}^{-1}) \tag{10}$$

where the scalar α_{reg} controls the strength of this regularization, and is selected using a validation dataset. As the meta-training model is directly parameterized by the inverse of the precision matrices $\boldsymbol{\Lambda}_i$ (Harrison et al., 2018b), this regularizer can easily be added to the standard training loss.

From (3), we observe that β_i also depends on the ratio of determinants of the prior and posterior precision matrices $\big(\det(\boldsymbol{\Lambda}_{i,t})/\det(\boldsymbol{\Lambda}_{i,0})\big)$. Although a convex regularizer for this term can be derived, we found that including it did not lead to performance improvements. This ratio can be interpreted as capturing the amount of information that the model has gathered online, which is independent of the structure of the prior model. Before learning, this ratio is 1, so the other term

Algorithm 1 Sequential Exploration and Exploitation for Learning Safely (**SEELS**)

Input: Meta-training model satisfying A.2 and A.3

while $\boldsymbol{x}_0 \notin \mathcal{X}_{\mathrm{goal}}$ **do**
  for $N_i \in \{\underline{N}_{\mathrm{reach}}, \ldots, \overline{N}_{\mathrm{reach}}\}$ **do** ▷ *Try reaching*
    $(\boldsymbol{\mu}, \boldsymbol{u}) \leftarrow$ Solve (**Reach-OCP**)
    if (**Reach-OCP**) feasible **then**
      Apply $\boldsymbol{u}_{0:N-1}$ to true system ▷ *Reach*
      Break
  for $N_i \in \{1, \ldots, N_{\mathrm{info}}\}$ **do** ▷ *Explore*
    $(\boldsymbol{\mu}^i, \boldsymbol{u}^i) \leftarrow$ Solve (**Explore-OCP**)
    if (**Explore-OCP**) feasible **then**
      Compute $l^i_{\mathrm{info}}(\boldsymbol{\mu}^i, \boldsymbol{u}^i)$
  $i_{\mathrm{best}} \leftarrow \arg\max_i l^i_{\mathrm{info}}(\boldsymbol{\mu}^i, \boldsymbol{u}^i)$ ▷ *Get best N*
  Apply $\boldsymbol{u}^{i_{\mathrm{best}}}$ to true system
  Update $(\boldsymbol{k}, \boldsymbol{\Lambda})$ with $\{(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{x}_{k+1})\}_{k=0}^{N-1}$
  $\boldsymbol{x}_0 \leftarrow \boldsymbol{x}_N$

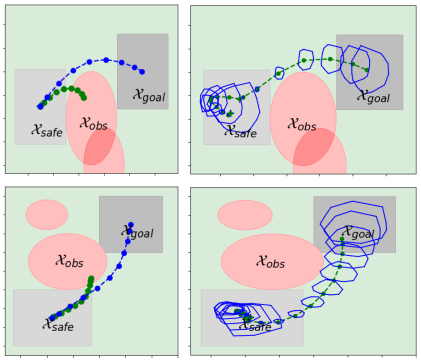

Figure 4: Rollouts on the system considered in experiments: **Left**: Due to high uncertainty, attempting to reach $\mathcal{X}_{\mathrm{goal}}$ is initially unsafe, violating velocity and final constraints. **Right**: Using **SEELS**, the system safely reaches the goal after safely learning its dynamics.

composed of the ratio of eigenvalues dominates β_i. We observed that it is during these early stages that the meta-training model and its bounds β_i are most conservative, which could explain the importance of the regularizer in (10), whereas regularizing the ratio of determinants appears to make little difference.

A.3 Information cost

During the exploration phase, we perform trajectory optimization with an objective function that encourages visiting states and taking actions that reduce uncertainty over the unknown dynamics. To do so, a natural objective function to maximize is the mutual information between the unknown function $\boldsymbol{g}(\cdot, \cdot, \boldsymbol{\theta})$ and the observations $\tilde{\boldsymbol{x}}^+ = \boldsymbol{x}_t - \boldsymbol{h}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1})$. This cost characterizes the *information gain* (MacKay, 1992; Srinivas et al., 2010; Chowdhury, 2017) from observing $\tilde{\boldsymbol{x}}^+$.

We derive this objective for the linear-Gaussian Bayesian model assumed by the meta-learning formulation in (Harrison et al., 2018a). For this formulation, which assumes that observations are corrupted with Gaussian noise, the mutual information can be computed in closed form. While in this work we assume bounded (non-Gaussian) noise corrupting our measurements, we find that making this approximation works well in practice to encourage exploration.

Let the posterior distribution over models be specified by $\boldsymbol{k}_i \sim \mathcal{N}(\bar{\boldsymbol{k}}_i, \sigma_{\epsilon_i}^2 \boldsymbol{\Lambda}_i)$, with Gaussian-distributed observation noise ϵ_i of variance $\sigma_{\epsilon_i}^2$. In this setting, the marginal distribution over observations $\boldsymbol{x}_i^+ = \boldsymbol{k}_i \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u}) + \epsilon_i$ given an arbitrary state $\boldsymbol{x}$ and control input $\boldsymbol{u}$ is also normally distributed as $\mathcal{N}(\boldsymbol{k}_i \boldsymbol{\phi}, (1 + \boldsymbol{\phi}^T \boldsymbol{\Lambda}_i^{-1} \boldsymbol{\phi}) \sigma_{\epsilon_i}^2)$, where $\boldsymbol{\phi} = \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})$.

Next, we define the mutual information $\mathcal{I}$ between the observation $\boldsymbol{x}^+$, and the true model $\boldsymbol{g}(\cdot, \cdot, \boldsymbol{\theta})$, as a function of the current state $\boldsymbol{x}$ and control input $\boldsymbol{u}$, and assuming that Assumption 2 holds. This quantity denotes the information gain from applying the control input $\boldsymbol{u}$ to the true system from $\boldsymbol{x}$, and observing $\boldsymbol{x}^+$ to update our model. The mutual information is defined using the entropy $\mathcal{H}(\cdot)$, which for a Gaussian-distributed random variable $\boldsymbol{x}^+ \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ evaluates to $\mathcal{H}(\boldsymbol{x}) = (1/2)\log(\det(2\pi e \boldsymbol{\Sigma}))$. Hence, the information gain from observing the scalar random variable $\boldsymbol{x}_i^+$ can be expressed as $\mathcal{I}(\boldsymbol{x}_i^+; \boldsymbol{g}_{\boldsymbol{\theta}}) = \mathcal{H}(\boldsymbol{x}_i^+) - \mathcal{H}(\boldsymbol{x}_i^+ | \boldsymbol{g}_{\boldsymbol{\theta}}) = \frac{1}{2}(\log(\text{var}(\boldsymbol{x}_i^+)) - \log(\text{var}(\boldsymbol{x}_i^+ | \boldsymbol{g}_{\boldsymbol{\theta}})) = \frac{1}{2}(\log((1 + \boldsymbol{\phi}^T \boldsymbol{\Lambda}_i^{-1} \boldsymbol{\phi}) \sigma_{\epsilon_i}^2)) - \log(\sigma_{\epsilon_i}^2))) = \frac{1}{2}(\log(1 + \boldsymbol{\phi}^T \boldsymbol{\Lambda}_i^{-1} \boldsymbol{\phi}))$.

For our problem formulation, this quantity approximately expresses the information gain from observing each dimension i of the state (which are modeled independently in our formulation). Intuitively, we would like to design exploration trajectories that visit states and take actions where this quantity is high for all dimensions of the state, as these observations would be the most informative in terms of reducing uncertainty over the underlying model. Thus, we use this term to guide the exploration phases, and optimize for the objective

$$l_{\text{info}}(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{\Lambda}_{1,t}, \ldots, \boldsymbol{\Lambda}_{n,t}) = \frac{1}{2} \sum_{i=1}^{n} \log(1 + \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})^T \boldsymbol{\Lambda}_{i,t}^{-1} \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})). \tag{11}$$

Note that this is a function of the current information state of the model, specified by the updated precision matrices $\boldsymbol{\Lambda}_{1,t}, \ldots, \boldsymbol{\Lambda}_{n,t}$. This provides an objective which encourages exploring states in the feature space spanned by $\boldsymbol{\phi}(\cdot, \cdot)$ which have highest variance, to quickly reduce uncertainty.

Note that the expected information gain along a trajectory is not simply the sum of the expected information gains per transition, as expressed in (**Explore-OCP**) when summing (11) over $k = 0, \ldots, N$. However, correctly computing the expected information gain along the trajectory would require factoring in model updates along the trajectory; we find that considering the sum of single-transition information gain with the current precision matrices $\boldsymbol{\Lambda}_{i,t}$ is sufficient in guiding exploration for our work. The problem of optimal exploration is beyond the scope of this framework.

B Discussion of Assumptions

The linear uncertainty representation of the meta-learned dynamics model enables construction of confidence sets over dynamics models that hold throughout the online learning process, by leveraging results from the literature on linear contextual bandits. These finite-sample *online* learning bounds rely on two critical assumptions on the results of the *offline* meta-learning process: (1) that the meta-learning model is capable of fitting the true system dynamics online, and (2) the uncertainty

estimates that are meta-learned represent a conservative prior over the true dynamics functions. We restate these assumptions below:

Assumption 2 (Capacity of meta-learned dynamics model)**.** For all $\boldsymbol{\theta}$, there exists $\boldsymbol{k}_i^* \in \mathbb{R}^d$ such that $\langle \boldsymbol{k}_i^* \, ; \, \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u}) \rangle = \boldsymbol{g}_i(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\theta})$ for all $\boldsymbol{x} \in \mathcal{X}$, $\boldsymbol{u} \in \mathcal{U}$, and $i = 1, \ldots, n$.

Assumption 3 (Calibration of meta-learned prior)**.** For $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$, each $i = 1, \ldots, n$, and $\delta_i = \delta/(2n)$, with probability at least $(1 - \delta_i)$, $\|\boldsymbol{k}_i^* - \bar{\boldsymbol{k}}_{i,0}\|^2_{\boldsymbol{\Lambda}_{i,0}} \leq \sigma^2_{\epsilon_i} \chi^2_d (1 - \delta_i)$.

These assumptions state that the true dynamics can be represented as a linear combination of finite dimensional nonlinear features, which applies to a plethora of physical dynamical systems (Mania et al., 2020; Kakade et al., 2020). Further, it assumes that the meta-learning model learns appropriate features for such a representation. Formally verifying these assumptions requires making generalization claims on the meta-learning process, perhaps through a PAC-Bayes analysis (Amit and Meir, 2018), and is beyond the scope of this paper. If the dataset has adequate coverage of the state and action spaces and the dynamics distribution $p(\boldsymbol{\theta})$, the offline meta-learning procedure proposed in (Harrison et al., 2018a) can approach satisfaction of these assumptions. The validity of these assumptions can be empirically verified through predictive performance on a validation dataset, and techniques such as temperature scaling can be used to ensure calibration in a post-hoc manner (Kuleshov et al., 2018). Assumption 2 in particular is comparable to asymptotic representation results in Gaussian process-based methods (Berkenkamp, 2018). We believe our combination of meta-learning with these assumptions motivates two directions of future work: safety analysis with feature mismatch, and finite sample guarantees for meta-learning models.

C Related Work

In contrast to model-free approaches to reinforcement learning, model-based methods (generally) provide better sample efficiency while enabling guarantees on constraint satisfaction and stability (Recht, 2019; Deisenroth et al., 2015). These model-based methods rely on the choice of dynamics model parameterization—for example, neural networks (Levine et al., 2016), Gaussian processes (GPs) (Deisenroth et al., 2015), or linear models (Coulson et al., 2018)—each with associated strengths and weaknesses. Recent work in the controls community has leveraged behavioral systems theory to guarantee stability and probabilistic constraints satisfaction of a non-parametric MPC scheme (Coulson et al., 2020; Berberich et al., 2020). Although such methods have been shown to perform well for nonlinear systems (Coulson et al., 2018), their guarantees currently do not extend beyond time invariant linear systems. Moreover, these approaches rely on linear models, limiting their expressiveness and potentially reducing their applicability in diverse scenarios as well as generalization across scenarios.

Nonlinear controllers leveraging a neural network model of the system can provide stability guarantees (Shi et al., 2019), under smoothness and other assumptions. However, these methods require collecting a dataset for a single system (i.e., already being able to solve the task), and would need total retraining if the environment or the system change. Training a neural network dynamics model from scratch for each environment is prohibitively expensive in terms of data requirements. Our approach combines neural network features with linear online adaptation to obtain the best of both models: the linear learning is sample efficient and enables strong guarantees on performance, while the neural network features are highly expressive and enable generalization across environments. While prior work has leveraged meta-learning for fast online adaptation (Nagabandi et al., 2019), such approaches are difficult to provide safety guarantees for, as they typically adapt using online gradient descent in non-convex problems. In contrast, our linear online adaptation enables construction of confidence sets for model parameters that hold throughout the learning process.

Gaussian processes have been widely used for safe learning-based control and exploration, as they can represent any nonlinear function in a bounded reproducing kernel Hilbert space (RKHS). GPs are nonlinear, Bayesian models that obtain sample efficiency through exact conditioning and reasonably expressive features through the choice of kernel (Williams and Rasmussen, 2006). While bounds providing similar guarantees to Theorem 1 can be derived for GPs, such bounds are generally too conservative, in which case the authors usually set these constants to arbitrary values in experiments (Berkenkamp et al., 2017; Koller et al., 2018). Alternatively, assuming that the RKHS is known, and that any function in this space lies in the span of finite-dimensional features $\boldsymbol{\phi}$ is common in practice

(Mania et al., 2020; Kakade et al., 2020). Importantly, this linear structure enables the derivation of bounds over models (Abbasi-Yadkori et al., 2011), which we use directly in this work. In contrast with prior work, we explicitly learn features and quantify prior uncertainty in an offline meta-training procedure (Harrison et al., 2018a,b), enabling us to design a model which is calibrated and accurate enough to represent possible systems, and allows verifying that representation error is small *offline*, before deploying this system.

D Proofs

D.1 Proof of Theorem 1: Uniformly Calibrated Confidence Sets

The proof of Theorem 1 follows from the proof of (Abbasi-Yadkori et al., 2011, Theorem 2), by making substitutions accordingly for our meta-learning model. To do so, we use the following lemma, which follows from (Abbasi-Yadkori et al., 2011, Theorem 1), by considering each dimension $i = 1, \ldots, n$ of the meta-learning model independently.

Lemma 1 (Self-Normalized Bound for Vector-Valued Martingales)**.** *Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration. Define $\{\epsilon_t^i\}_{t=1}^{\infty}$, a real-valued stochastic process such that ϵ_t^i is $\mathcal{F}_t$-measurable, and conditionally σ_{ϵ_i}-subgaussian. Let $\{\boldsymbol{\phi}_t\}_{t=1}^{\infty}$ be a $\mathbb{R}^d$-valued stochastic process such that $\boldsymbol{\phi}_t$ is $\mathcal{F}_{t-1}$-measurable.*

Let $\boldsymbol{\Lambda}_{i,0}$ be a $d \times d$ positive definite matrix, and define $\boldsymbol{\Lambda}_{i,t}$ as in (2)*. Further, for any $t \geq 0$, define $\mathbf{S}_t = \sum_{s=1}^{t} \epsilon_s^i \boldsymbol{\phi}_s$. Then, for any $\delta_i > 0$, with probability at least $(1 - \delta_i)$, for all $t \geq 0$,*

$$\|\mathbf{S}_t\|^2_{\boldsymbol{\Lambda}_{i,t}^{-1}} \leq 2\sigma_{\epsilon_i}^2 \log\left(\frac{1}{\delta}\frac{\det(\boldsymbol{\Lambda}_{i,t})^{1/2}}{\det(\boldsymbol{\Lambda}_{i,0})^{1/2}}\right) \tag{12}$$

Proof. The filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$ is defined by considering the σ-algebra $\mathcal{F}_t = \sigma(\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_{t+1}, \boldsymbol{\epsilon}_0, \ldots, \boldsymbol{\epsilon}_t)$, where $\boldsymbol{\phi}_t = \boldsymbol{\phi}(\boldsymbol{x}_t, \boldsymbol{u}_t)$, and the $\boldsymbol{x}_t$ are given by (1a). Then, this result follows by direct application of (Abbasi-Yadkori et al., 2011, Theorem 1), substituting $(X, \eta, \theta, \bar{V}_t, V)$ with $(\boldsymbol{\phi}, \epsilon^i, \boldsymbol{k}_i, \boldsymbol{\Lambda}_{i,t}, \boldsymbol{\Lambda}_{i,0})$. □

We stress that (12) holds jointly for all times $t \geq 0$, such that $\mathbb{P}((12)) \geq (1 - \delta_i)$. This result is key to ensure joint chance constraint satisfaction, and guarantee safety and feasibility of our framework.

Next, we prove Theorem 1, which we restate here for completeness.

Theorem 1 (Uniformly Calibrated Confidence Sets)**.** *Consider the true system* (1a)*,*

$$\boldsymbol{x}_{k+1} = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}) + \boldsymbol{\epsilon}_k,$$

where $\boldsymbol{\epsilon}_k$ is σ_ϵ-subgaussian and bounded. Consider the meta-learning model, given as $\hat{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{K}\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{u})$, where $\boldsymbol{\phi} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^d$, and $\boldsymbol{K}$ is an $n \times d$ matrix, with n rows $\boldsymbol{k}_i$. Starting from $(\bar{\boldsymbol{k}}_{i,0}, \boldsymbol{\Lambda}_{i,0})$, with $\bar{\boldsymbol{k}}_{i,0} \in \mathbb{R}^d$, and $\boldsymbol{\Lambda}_{i,0}$ a $d \times d$ positive definite matrix, define the sequence $\{(\bar{\boldsymbol{k}}_{i,s}, \boldsymbol{\Lambda}_{i,s})\}_{s=0}^{t}$, where $(\bar{\boldsymbol{k}}_{i,t}, \boldsymbol{\Lambda}_{i,t})$ is computed with online data from (1a) *using* (2) *as*

$$\boldsymbol{\Lambda}_{i,t} = \Phi_{t-1}^T \Phi_{t-1} + \boldsymbol{\Lambda}_{i,0}, \qquad \bar{\boldsymbol{k}}_{i,t} = \boldsymbol{\Lambda}_{i,t}^{-1}(\Phi_{t-1}^T \boldsymbol{G}_{i,t} + \boldsymbol{\Lambda}_{i,0}\bar{\boldsymbol{k}}_{i,0}), \qquad i = 1, \ldots, n,$$

$\boldsymbol{G}_t^T = [\boldsymbol{x}_1 - \boldsymbol{h}(\boldsymbol{x}_0, \boldsymbol{u}_0), \ldots, \boldsymbol{x}_t - \boldsymbol{h}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1})] \in \mathbb{R}^{n \times t}$, and $\Phi_{t-1}^T = [\boldsymbol{\phi}(\boldsymbol{x}_0, \boldsymbol{u}_0), \ldots, \boldsymbol{\phi}(\boldsymbol{x}_{t-1}, \boldsymbol{u}_{t-1})] \in \mathbb{R}^{d \times t}$. Further, define $\delta_i = \delta/(2n)$, and

$$\beta_i(\boldsymbol{\Lambda}_{i,t}, \delta_i) = \sigma_{\epsilon_i}\left(\sqrt{2\log\left(\frac{1}{\delta_i}\frac{\det(\boldsymbol{\Lambda}_{i,t})^{1/2}}{\det(\boldsymbol{\Lambda}_{i,0})^{1/2}}\right)} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_{i,0})}{\lambda_{\min}(\boldsymbol{\Lambda}_{i,t})}\chi_d^2(1-\delta_i)}\right).$$

Then, under Assumptions 2 and 3,

$$\mathbb{P}\left(\|\boldsymbol{k}_i^* - \bar{\boldsymbol{k}}_{i,t}\|_{\boldsymbol{\Lambda}_{i,t}} \leq \beta_i(\boldsymbol{\Lambda}_{i,t}, \delta_i) \quad \forall t \geq 0\right) \geq (1 - 2\delta_i).$$

Proof. This proof is a straightforward extension of (Abbasi-Yadkori et al., 2011, Theorem 2), where we use Assumption 3 to provide a probabilistic error bound for the model missmatch over the prior for $\boldsymbol{k}_i^*$, Lemma 1 to bound the estimation error due to $\boldsymbol{\epsilon}_k$, and Boole's inequality to obtain β_i.

Define $\boldsymbol{\epsilon}^i = (\epsilon_1^i, \ldots, \epsilon_t^i)^T$. For any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$, and $\mathbf{A}$ a $d \times d$ positive definite matrix, define the weighted norm $\|\mathbf{a}\|_{\mathbf{A}}^2 = \mathbf{a}^T\mathbf{A}\mathbf{a}$, and weighted inner product $\langle \mathbf{a}, \mathbf{b} \rangle_{\mathbf{A}} = \mathbf{a}^T\mathbf{A}\mathbf{b}$. For conciseness, we drop the indices i and t, and denote $(\boldsymbol{k}^*, \bar{\boldsymbol{k}}, \boldsymbol{\Lambda}, \bar{\boldsymbol{k}}_0, \boldsymbol{\Lambda}_0, \Phi, \boldsymbol{\epsilon}) = (\boldsymbol{k}_i^*, \bar{\boldsymbol{k}}_{i,t}, \boldsymbol{\Lambda}_{i,t}, \bar{\boldsymbol{k}}_{i,0}, \boldsymbol{\Lambda}_{i,0}, \Phi_{t-1}, \boldsymbol{\epsilon}^i)$.

Under Assumption 2, we can write $\boldsymbol{G}_{i,t} = \Phi \boldsymbol{k}^* + \boldsymbol{\epsilon}$. Then, we rewrite the mean estimate $\bar{\boldsymbol{k}}$ of $\boldsymbol{k}^*$ at time t, as

$$\begin{aligned}\bar{\boldsymbol{k}} &= (\boldsymbol{\Lambda}_0 + \Phi^T\Phi)^{-1}(\boldsymbol{\Lambda}_0\bar{\boldsymbol{k}}_0 + \Phi^T(\Phi\boldsymbol{k}^* + \boldsymbol{\epsilon})) \\ &= (\boldsymbol{\Lambda}_0 + \Phi^T\Phi)^{-1}\Phi^T\boldsymbol{\epsilon} + (\boldsymbol{\Lambda}_0 + \Phi^T\Phi)^{-1}(\boldsymbol{\Lambda}_0 + \Phi^T\Phi)\boldsymbol{k}^* - (\boldsymbol{\Lambda}_0 + \Phi^T\Phi)^{-1}\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0) \\ &= \boldsymbol{\Lambda}^{-1}\Phi^T\epsilon + \boldsymbol{k}^* - \boldsymbol{\Lambda}^{-1}\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0),\end{aligned}$$

from which we obtain, for any $\mathbf{a} \in \mathbb{R}^d$, that $\mathbf{a}^T(\bar{\boldsymbol{k}} - \boldsymbol{k}^*) = \langle \mathbf{a}, \Phi^T\boldsymbol{\epsilon}\rangle_{\boldsymbol{\Lambda}^{-1}} - \langle \mathbf{a}, \boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0)\rangle_{\boldsymbol{\Lambda}^{-1}}$. Note that $\boldsymbol{\Lambda}_0 \succ 0$, so $\boldsymbol{\Lambda} \succ 0$, and these inner products are well defined. With this result, by the Cauchy-Schwarz inequality,

$$\begin{aligned}|\mathbf{a}^T(\bar{\boldsymbol{k}} - \boldsymbol{k}^*)| &\leq \|\mathbf{a}\|_{\boldsymbol{\Lambda}_t^{-1}} \left(\left\|\Phi^T\epsilon\right\|_{\boldsymbol{\Lambda}^{-1}} + \left\|\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0)\right\|_{\boldsymbol{\Lambda}^{-1}}\right) \\ &\leq \|\mathbf{a}\|_{\boldsymbol{\Lambda}^{-1}} \left(\left\|\Phi^T\epsilon\right\|_{\boldsymbol{\Lambda}^{-1}} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}}\left\|\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0\right\|_{\boldsymbol{\Lambda}_0}\right), \qquad (13)\end{aligned}$$

where the second inequality is obtained as

$$\begin{aligned}\left\|\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0)\right\|^2_{\boldsymbol{\Lambda}^{-1}} &\leq \frac{\lambda_{\max}(\boldsymbol{\Lambda}^{-1})}{\lambda_{\min}(\boldsymbol{\Lambda}_0^{-1})}\left\|\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0)\right\|^2_{\boldsymbol{\Lambda}_0^{-1}} = \frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}\left\|\boldsymbol{\Lambda}_0(\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0)\right\|^2_{\boldsymbol{\Lambda}_0^{-1}} \\ &= \frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}\left\|\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0\right\|^2_{\boldsymbol{\Lambda}_0}.\end{aligned}$$

By Lemma 1, for any $\delta_i \geq 0$, with probability at least $(1 - \delta_i)$, we have

$$\left\|\Phi^T\epsilon\right\|^2_{\boldsymbol{\Lambda}^{-1}} \leq 2\sigma^2_{\epsilon_i}\log\left(\frac{1}{\delta_i}\frac{\det(\boldsymbol{\Lambda})^{1/2}}{\det(\boldsymbol{\Lambda}_0)^{1/2}}\right) \quad \forall t \geq 0. \qquad (14)$$

By Assumption 3, for $\delta_i = \delta/(2n)$, with probability at least $(1 - \delta_i)$,

$$\|\boldsymbol{k}^* - \bar{\boldsymbol{k}}_0\|^2_{\boldsymbol{\Lambda}_0} \leq \sigma^2_{\epsilon_i}\chi^2_d(1 - \delta_i). \qquad (15)$$

From (13), by Boole's inequality[4], we have that with probability at least $(1 - 2\delta_i)$, for all $t \geq 0$, and any $\mathbf{a} \in \mathbb{R}^d$,

$$|\mathbf{a}^T(\bar{\boldsymbol{k}} - \boldsymbol{k}^*)| \leq \|\mathbf{a}\|_{\boldsymbol{\Lambda}^{-1}}\sigma_{\epsilon_i}\left(\sqrt{2\log\left(\frac{1}{\delta_i}\frac{\det(\boldsymbol{\Lambda})^{1/2}}{\det(\boldsymbol{\Lambda}_0)^{1/2}}\right)} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}\chi^2_d(1 - \delta_i)}\right).$$

Let $\mathbf{a} = \boldsymbol{\Lambda}(\bar{\boldsymbol{k}} - \boldsymbol{k}^*)$ in the expression above, to obtain

$$\left\|\bar{\boldsymbol{k}} - \boldsymbol{k}^*\right\|^2_{\boldsymbol{\Lambda}} \leq \left\|\boldsymbol{\Lambda}(\bar{\boldsymbol{k}} - \boldsymbol{k}^*)\right\|_{\boldsymbol{\Lambda}^{-1}}\sigma_{\epsilon_i}\left(\sqrt{2\log\left(\frac{1}{\delta_i}\frac{\det(\boldsymbol{\Lambda})^{1/2}}{\det(\boldsymbol{\Lambda}_0)^{1/2}}\right)} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}\chi^2_d(1 - \delta_i)}\right).$$

Since $\left\|\boldsymbol{\Lambda}(\bar{\boldsymbol{k}} - \boldsymbol{k}^*)\right\|_{\boldsymbol{\Lambda}^{-1}} = \left\|\bar{\boldsymbol{k}}_t - \boldsymbol{k}^*\right\|_{\boldsymbol{\Lambda}}$, we divide both sides by $\left\|\bar{\boldsymbol{k}} - \boldsymbol{k}^*\right\|_{\boldsymbol{\Lambda}}$ and obtain

$$\left\|\bar{\boldsymbol{k}} - \boldsymbol{k}^*\right\|_{\boldsymbol{\Lambda}} \leq \sigma_{\epsilon_i}\left(\sqrt{2\log\left(\frac{1}{\delta_i}\frac{\det(\boldsymbol{\Lambda})^{1/2}}{\det(\boldsymbol{\Lambda}_0)^{1/2}}\right)} + \sqrt{\frac{\lambda_{\max}(\boldsymbol{\Lambda}_0)}{\lambda_{\min}(\boldsymbol{\Lambda})}\chi^2_d(1 - \delta_i)}\right) \quad \forall t \geq 0, \qquad (16)$$

which holds with probability at least $(1 - 2\delta_i)$. As this is the expression for β_i, this concludes our proof. □

[4] $\mathbb{P}((14) \cap (15)) = 1 - \mathbb{P}((14)^C \cup (15)^C)$, where A^C denotes the negation of A. Then, by Boole's inequality, $1 - \mathbb{P}((14)^C \cup (15)^C) \geq 1 - \mathbb{P}((14)^C) - \mathbb{P}((15)^C) = -1 + \mathbb{P}((14)) + \mathbb{P}((15))$. Finally, using the lower bounds on the probabilities that (14) and (15) occur, we obtain $\mathbb{P}((14) \cap (15)) \geq -1 + (1 - \delta_i) + (1 - \delta_i) = 1 - 2\delta_i$.

D.2 Proof of Theorem 2, Part 1: Probabilistic Feasibility

We restate the first part of Theorem 2 for ease of reading.

Theorem 2 (Probabilistic Feasibility)**.** *Under Assumptions 1, 2 and 3, there exists an optimization horizon N ensuring the feasibility of (**Explore-OCP**) at all times with probability $(1-\delta)$.*

Proof. Let n_{info} the number of exploration phases before (**Reach-OCP**) becomes feasible[5].

Also, let N^j_{info}, and $t_j = \sum_{l=0}^{j-1} N^l_{\text{info}}$ be, respectively, the planning horizon, and the start time index of each $(\textbf{Explore-OCP})_j$.

For conciseness, define $(\textbf{EOCP})_j$ for $\{(\textbf{Explore-OCP})_j \text{ is feasible}\}$, i.e., the event that the j-th exploration problem is feasible.

Then, by the law of total probability,

$$\begin{aligned}
\mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j\Big) &= \mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0\Big) + \mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \notin \mathcal{X}_0\Big) \\
&\geq \mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0,\ \Big) \\
&= \mathbb{P}\Big((\textbf{EOCP})_{n_{\text{info}}} \mid \bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0\Big)\mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0\Big).
\end{aligned}$$

By Assumption 1, given that $\boldsymbol{x}_{t_j} \in \mathcal{X}_0$, $(\textbf{Explore-OCP})_j$ is feasible for any j-th exploration phase. Indeed, choose $N^j_{\text{info}} = 1$ for $(\textbf{Explore-OCP})_j$. Then, $\boldsymbol{u}^j_0 = \boldsymbol{\pi}(\boldsymbol{x}_{t_j})$ is a feasible solution to $(\textbf{Explore-OCP})_j$. Thus, the event $\{(\textbf{EOCP})_j \mid \boldsymbol{x}_{t_j} \in \mathcal{X}_0\}$ holds with probability one.

In particular, $\{(\textbf{EOCP})_{n_{\text{info}}} \mid \bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j, \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0\}$ holds with probability one.

Next, we use the law of total probability to leverage our confidence sets over parameters:

$$\begin{aligned}
\mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j\Big) &\geq \mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0\Big) \\
&\geq \mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j,\ \boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0,\ \boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_{n_{\text{info}}-1}}\ \forall i\Big) \\
&= \mathbb{P}\Big(\boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0 \mid \bigwedge_j(\textbf{EOCP})_j,\ \boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_{n_{\text{info}}-1}} \forall i\Big)\,\mathbb{P}\Big(\bigwedge_j(\textbf{EOCP})_j,\ \boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_{n_{\text{info}}-1}} \forall i\Big).
\end{aligned}$$

By construction of the reachable sets $\{\mathcal{X}^{t_j,\delta}_k\}_{k=1}^{N^j_{\text{info}}}$, and by definition of $(\textbf{Explore-OCP})_j$ (since $\mathcal{X}^{t_j,\delta}_{N^j_{\text{info}}} \subset \mathcal{X}_0$), we have that $\boldsymbol{x}_{t_{j+1}} \in \mathcal{X}_0$ given that $(\textbf{Explore-OCP})_j$ is feasible and that $\boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_j}\ \forall i$, for any j-th exploration problem.

Thus, the first term $\{\boldsymbol{x}_{t_{n_{\text{info}}}} \in \mathcal{X}_0 \mid \bigwedge_j(\textbf{EOCP})_j,\ \boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_{n_{\text{info}}-1}} \forall i\}$ holds with probability one.

Thus,

$$\mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j\Big) \geq \mathbb{P}\big(\bigwedge_{j=0}^{n_{\text{info}}-1}(\textbf{EOCP})_j,\ \boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_{n_{\text{info}}-1}}\ \forall i\big).$$

Since $(\textbf{EOCP})_0$ is feasible with probability one since $\boldsymbol{x}_0 \in \mathcal{X}_0$, and by reasoning by induction for all $j = n_{\text{info}}, \ldots, 0$, we obtain that

$$\mathbb{P}\Big(\bigwedge_{j=0}^{n_{\text{info}}}(\textbf{EOCP})_j\Big) \geq \mathbb{P}\big(\bigwedge_{j=0}^{n_{\text{info}}-1}\boldsymbol{k}^*_i \in \mathcal{C}^\delta_{i,t_j}\ \forall i\big) \geq (1-\delta),$$

where the last inequality comes from Theorem 1, which concludes this proof. □

[5]This result also holds if **(CC-OCP)** is not feasible, and the algorithm can never solve (**Reach-OCP**) to reach $\mathcal{X}_{\text{goal}}$ (e.g., if $\mathcal{X}_{\text{goal}}$ is surrounded by obstacles). Indeed, if the algorithm is stuck in an infinite number of exploration steps, the last inequality of this proof still holds for $n_{\text{info}} \to \infty$, by Theorem 1.

D.3 Proof of Theorem 2, Part 2: Probabilistic Safety

Next, we prove our result of probabilistic safety. First, we restate second part of Theorem 2 for ease of reading.

Theorem 3 (Probabilistic Safety)**.** *Compute confidence sets for model parameters using* (3) *and* (4)*. Using these confidence sets, compute probabilistic reachable sets* $\{\mathcal{X}_k^{t,\delta}\}_{k=1}^N$ *satisfying* (6)*. Using these sets, apply Algorithm 1 and sequentially solve (**Explore-OCP**) and (**Reach-OCP**).*

*Then, assuming that (**Reach-OCP**) is feasible at some time t, and under Assumptions 2 and 3, the system is guaranteed to satisfy* (1b) *i.e., to be safe at all times and eventually reach* $\mathcal{X}_{goal}$ *with probability* $(1-\delta)$*.*

Proof. We use a proof by construction. First, let N_{info}^j, and $t_j = \sum_{l=1}^{j-1} N_{\text{info}}^l$ be, respectively, the planning horizon, and the start time index of each (**Explore-OCP**)$_j$, where $j=1,\ldots,n_{\text{info}}$, with n_{info} the number of exploration phases. Similarly, define N_{reach}, and t_f to be, respectively, the planning horizon, and the start time index of (**Reach-OCP**). For conciseness, define $\boldsymbol{x}_k^{t_j} = \boldsymbol{x}_{t_j+k}$, corresponding to the state at time (t_j+k) in the j-th phase. Note that without feedback, open-loop controls satisfy $\boldsymbol{u}_k \in \mathcal{U}\ \forall k$. Further, define the event that the trajectory during the j-th exploration phase (or exploitation phase) satisfies all constraints as

$$\{\boldsymbol{x}_{\text{info}}^j \in \mathcal{X}_{\text{info}}^j\} = \left\{ \bigwedge_{k=1}^{N_{\text{info}}^j} \left(\boldsymbol{x}_k^{t_j} \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_{N_{\text{info}}^j}^{t_j} \in \mathcal{X}_0\right) \right\}, \quad j=1,\ldots,n_{\text{info}}, \tag{17}$$

$$\{\boldsymbol{x}_{\text{reach}} \in \mathcal{X}_{\text{reach}}\} = \left\{ \bigwedge_{k=1}^{N_{\text{reach}}} \left(\boldsymbol{x}_k^{t_f} \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_{N_{\text{reach}}}^{t_f} \in \mathcal{X}_{\text{goal}}\right) \right\}. \tag{18}$$

With this notation, we rewrite the safety condition of the original problem we are solving (which is the one we want to prove in this theorem) as

$$\begin{aligned}
\text{(1b)} &= \mathbb{P}\Bigg(\bigwedge_{k=1}^{N_{\text{info}}^1} \left(\boldsymbol{x}_k \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_{N_{\text{info}}^1} \in \mathcal{X}_0\right) \cap \cdots \cap \bigwedge_{k=\sum_i N_{\text{info}}^i}^{\sum_i N_{\text{info}}^i + N_{\text{reach}}} \left(\boldsymbol{x}_k \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_N \in \mathcal{X}_{\text{goal}}\right) \Bigg) \\
&= \mathbb{P}\Bigg(\bigwedge_{k=1}^{N_{\text{info}}^1} \left(\boldsymbol{x}_k^{t_1} \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_{N_{\text{info}}^1}^{t_1} \in \mathcal{X}_0\right) \cap \cdots \cap \bigwedge_{k=1}^{N_{\text{reach}}} \left(\boldsymbol{x}_k^{t_f} \in \mathcal{X}_{\text{free}}\right) \cap \left(\boldsymbol{x}_{N_{\text{reach}}}^{t_f} \in \mathcal{X}_{\text{goal}}\right) \Bigg) \\
&= \mathbb{P}\Bigg(\bigwedge_{j=1}^{n_{\text{info}}} \{\boldsymbol{x}_{\text{info}}^j \in \mathcal{X}_{\text{info}}^j\} \cap \{\boldsymbol{x}_{\text{reach}} \in \mathcal{X}_{\text{reach}}\} \Bigg) := \mathbb{P}\Big(\{\text{Safely Reached}\} \Big).
\end{aligned}$$

Next, using the above, and by the law of total probability, we note that

$$\begin{aligned}
\text{(1b)} &= \mathbb{P}\Big(\{\text{Safely Reached}\} \,|\, \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big) \cdot \mathbb{P}\Big(\boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big) + \\
&\qquad \mathbb{P}\Big(\{\text{Safely Reached}\} \,|\, \boldsymbol{k}_i^* \notin \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big) \cdot \mathbb{P}\Big(\boldsymbol{k}_i^* \notin \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big) \\
&\geq \mathbb{P}\Big(\{\text{Safely Reached}\} \,|\, \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big) \cdot \mathbb{P}\Big(\boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\ \forall t,\ \forall i \Big),
\end{aligned} \tag{19}$$

where $t = t_1,\ldots,t_{n_{\text{info}}}, t_f$, and $i=1,\ldots,n$.

By Assumption 2, our meta-learning model can fit the true dynamics. Hence, if the true parameters are within the confidence sets $\mathcal{C}_{i,t}^\delta$, then, the reachable sets $\mathcal{X}_k^{t,\delta}$ necessarily contain the state trajectory on the true system, by definition (6). Using this fact, we can reformulate the constraints using the reachable sets, since

$$\Big\{ \mathcal{X}_k^{t,\delta} \subset \mathcal{X}_{\text{free}} \Big\} = \Big\{ \boldsymbol{x}_k(\boldsymbol{K}^*) \in \mathcal{X}_{\text{free}} \,|\, \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta,\ \forall i \Big\}. \tag{20}$$

By definition of (**Explore-OCP**) and (**Reach-OCP**), the reachable sets are subsets of the safe set, and the solution satisfies constraints. Hence, given a solution to these problems, we obtain

$$\mathbb{P}\Big(\boldsymbol{x}_k^t(\boldsymbol{K}^*) \in \mathcal{X}_{\text{free}},\ k=1,\ldots,N \,|\, \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta,\ i=1,\ldots,n \Big) = \mathbb{P}\Big(\mathcal{X}_k^{t,\delta} \subset \mathcal{X}_{\text{free}}, k=1,\ldots,N \Big) = 1,$$

which also holds for the final constraints $\boldsymbol{x}_N^t \in \mathcal{X}_0$, and $\boldsymbol{x}_N^{t_f} \in \mathcal{X}_{\text{goal}}$. Thus,

$$\mathbb{P}\Big(\{\text{Safely Reached}\} \,|\, \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta \; \forall t, \; \forall i\Big) = 1.$$

Combining this result with (19), we obtain that

$$\text{(1b)} \geq \mathbb{P}\Big(\boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta \; \forall t, \; \forall i\Big). \tag{21}$$

This last term holds with probability greater than $(1-\delta)$. Indeed, using (a) Boole's inequality, and (b) Theorem 1, we obtain

$$\begin{aligned}
\mathbb{P}\Big(\boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta \; \forall t, \; \forall i\Big) &= \mathbb{P}\Big(\bigwedge_{i=1}^n \bigwedge_t \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\Big) = 1 - \mathbb{P}\Big(\bigvee_{i=1}^n \bigvee_t \boldsymbol{k}_i^* \notin \mathcal{C}_{i,t}^\delta\Big) \\
&\overset{(a)}{\geq} 1 - \sum_{i=1}^n \mathbb{P}\Big(\bigvee_t \boldsymbol{k}_i^* \notin \mathcal{C}_{i,t}^\delta\Big) = 1 - \sum_{i=1}^n \Big(1 - \mathbb{P}\Big(\bigwedge_t \boldsymbol{k}_i^* \in \mathcal{C}_{i,t}^\delta\Big)\Big) \\
&\overset{(b)}{\geq} 1 - \sum_{i=1}^n \big(1 - (1-2\delta_i)\big) = 1 - \sum_{i=1}^n \big(2\delta_i\big) = (1-\delta).
\end{aligned}$$

Since $\delta_i = \delta/(2n)$, combined with (21), this concludes this proof. □

E Experimental Details and Further Results

Problem formulation and implementation: We evaluate our approach on a planar free-flying space robot. This system behaves (approximately) as a double integrator, controlled with gas thrusters and a reaction wheel. We consider the problem of cargo transport, in which the robot is attached to a payload that results in changes to the inertial properties of the system, resulting in nonlinear dynamics. This system mimics a cargo unloading scenario that is one plausible near-term application of autonomous robots on-board the International Space Station (Fluckiger et al., 2018; Ekal and Ventura, 2019).

The state of the system is given by $\boldsymbol{x} = [\mathbf{p}, \theta, \mathbf{v}, \omega] \in \mathbb{R}^6$, with $\mathbf{p}, \mathbf{v} \in \mathbb{R}^2$ the planar position and velocity, and $\theta, \omega \in \mathbb{R}$ the heading and angular velocity, respectively. For safety, we constrain $|v_i| \leq 0.2$ m/s, and $|\omega| \leq 0.25$ rad/s. The control inputs are $\boldsymbol{u} := [\mathbf{F}, M] \in \mathcal{U} \subset \mathbb{R}^3$, where $\mathcal{U} = [-\bar{u}_i, \bar{u}_i]$ represent the limited control authority from the gas thrusters. We set $\bar{u}_{1,2} = 0.15$ N, and $\bar{u}_3 = 0.01$ Nm. The payload causes a change in mass, inertia properties and causes the center of mass to be offset at $\mathbf{p}_0 \in \mathbb{R}^2$. The continuous time nonlinear dynamics of the system (which we write as $\dot{\boldsymbol{x}} = \mathbf{f}_t(\cdot)$) are

$$\dot{\mathbf{p}} = \mathbf{v}, \;\; \dot{\theta} = \omega, \;\; \dot{\mathbf{v}} = \frac{1}{m}\left(\mathbf{F} - \dot{\omega}\begin{bmatrix} -p_{oy} \\ p_{ox} \end{bmatrix} + \omega^2 \mathbf{p}_o\right), \;\; \dot{\omega} = \frac{1}{J}\Big(M - p_{ox}F_y + p_{oy}F_x\Big). \tag{22}$$

We randomize the mass m, inertia J and center of mass offset $\mathbf{p}_0$ according to

$$m \sim \text{Unif}(25, 60)\,\text{kg}, \;\; J \sim \text{Unif}(0.30, 0.70)\,\text{kg·m}^2, \;\; p_{oi} \sim \text{Unif}(-7.5, 7.5)\,\text{cm}, \; i \in \{x, y\}. \tag{23}$$

Using a zero-order hold on the controls and a forward Euler discretization scheme, we discretize (22) as

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \Delta t \cdot \mathbf{f}_t(\boldsymbol{x}_k, \boldsymbol{u}_k, m, \mathbf{J}, \mathbf{p}_o) + \boldsymbol{\epsilon}_k, \tag{24}$$

where the $\epsilon_{k,i}$ are σ_{ϵ_i}-subgaussian, each bounded as $|\epsilon_{k,i}| \leq (\sigma_{\epsilon_i}^2 \chi_1^2(0.95))^{1/2}$. We use this discrete time nonlinear system in simulation, and to collect training data for offline meta-learning.

We use a nominal model of the system $\boldsymbol{h}(\cdot,\cdot)$ using (22) with $(\bar{m}, \bar{J}, \bar{\mathbf{p}}_0) = (35, 0.4, \mathbf{0})$, which corresponds to a double integrator model. To represent the unknown model missmatch $\boldsymbol{g}(\cdot,\cdot,\boldsymbol{\theta})$, we train an ALPaCA model as described in (Harrison et al., 2018b) for 6000 iterations for all experiments.

For trajectory optimization, we use standard linear-quadratic final and step costs on states and controls to minimize control cost and deviation to $\mathcal{X}_0$ or $\mathcal{X}_{\text{goal}}$ depending on the phase. Specifically, we maximize the information cost defined in (11) while minimizing control effort, penalizing high velocities, and minizing the final distance to $\boldsymbol{x}_g$, the center of either $\mathcal{X}_0$, or $\mathcal{X}_{\text{goal}}$. We obtain

$$\max_{\boldsymbol{\mu}, \boldsymbol{u}} \sum_{k=0}^{N-1} \Big(- \alpha_{\text{info}} l_{\text{info}}(\boldsymbol{\mu}_k, \boldsymbol{u}_k) + \boldsymbol{\mu}_k^T \boldsymbol{Q} \boldsymbol{\mu}_k + \boldsymbol{u}_k^T \boldsymbol{R} \boldsymbol{u}_k \Big) + (\boldsymbol{\mu}_N - \boldsymbol{x}_g)^T \boldsymbol{Q}_N (\boldsymbol{\mu}_N - \boldsymbol{x}_g). \tag{25}$$

small σ_{ϵ_i}	δ	# Explore	$\boldsymbol{x} \notin \mathcal{X}_{\text{obs}}$	$\boldsymbol{x} \in \mathcal{X}_{\text{min/max}}$	$\boldsymbol{x}_N \in \mathcal{X}_{\text{goal}}$	$\boldsymbol{x} \in \mathcal{X}_{\text{all}}$
SEELS	0.1	2.3 ± 0.01	$97.6 \pm 1.9\%$	$97.6 \pm 1.9\%$	$96.8 \pm 2.2\%$	$93.2 \pm 3.1\%$
SEELS	0.2	2.43 ± 0.19	$95.6 \pm 2.5\%$	$98.8 \pm 1.3\%$	$98.8 \pm 1.3\%$	$93.6 \pm 3.0\%$
SEELS	0.5	2.22 ± 0.18	$94.8 \pm 2.7\%$	$98.8 \pm 1.3\%$	$97.2 \pm 2.0\%$	$93.2 \pm 3.1\%$
Mean-Equivalent	-	0	$39.6 \pm 6.0\%$	$99.6 \pm 0.8\%$	$22.8 \pm 5.2\%$	$19.6 \pm 4.9\%$

Table 1: Results for 250 randomized experiments for different values of δ, with low noise levels $\boldsymbol{\epsilon}_k$, and $M = 1000$. For each experiment, we report the number of exploration phases, check constraints satisfaction, and report the percentage of experiments for which all constraints are satisfied ($\boldsymbol{x} \in \mathcal{X}_{\text{all}}$), with 95% confidence intervals. We run a mean-equivalent version of **SEELS** (Algorithm 1) which accounts for the disturbances $\boldsymbol{\epsilon}_k$, but does not consider model uncertainty. Our framework is guaranteed to simultaneously respect all constraints $(1 - \delta)$ fraction of the time, which is verified in practice.

high σ_{ϵ_i}	δ	# Explore	$\boldsymbol{x} \notin \mathcal{X}_{\text{obs}}$	$\boldsymbol{x} \in \mathcal{X}_{\text{min/max}}$	$\boldsymbol{x}_N \in \mathcal{X}_{\text{goal}}$	$\boldsymbol{x} \in \mathcal{X}_{\text{all}}$
SEELS	0.1	2.4 ± 0.14	$92.4 \pm 3.3\%$	$99.2 \pm 1.1\%$	$95.6 \pm 2.5\%$	$90.0 \pm 3.7\%$
SEELS	0.2	2.32 ± 0.13	$91.6 \pm 3.4\%$	$100 \pm 0\%$	$95.6 \pm 2.5\%$	$89.6 \pm 3.8\%$
SEELS	0.5	1.98 ± 0.11	$87.6 \pm 4.1\%$	$99.2 \pm 1.1\%$	$90.8 \pm 3.6\%$	$82.8 \pm 4.7\%$
Mean-Equivalent	-	0	$58.8 \pm 6.1\%$	$99.6 \pm 0.8\%$	$39.2 \pm 6.0\%$	$37.2 \pm 6.0\%$

Table 2: Results for 250 randomized experiments for different values of δ, with high noise levels $\boldsymbol{\epsilon}_k$, and $M = 2500$. Our safety guarantees are verified, and the need for exploration is evident, from the low success rate of an approach neglecting dynamics uncertainty.

In these experiments, we set $\alpha_{\text{info}} = 0.025$ for exploration, whereas $\alpha_{\text{info}} = 0$ when reaching $\mathcal{X}_{\text{goal}}$, and $\boldsymbol{Q} = \text{diag}([0, 0, 0, 1, 1, 10])$, $\boldsymbol{R} = \text{diag}([10, 10, 10])$, and $\boldsymbol{Q}_N = 10^3 \text{diag}([1, 1, 0.1, 10, 10, 10])$ for both (**Explore-OCP**) and (**Reach-OCP**).

Outline of results: We evaluate our framework on multiple problems (250) with different parameters $\boldsymbol{\theta}$. Specifically, we consider two different sets of σ_{ϵ_i}, and four environments with different boundary conditions and obstacles. For the scenarios shown in Fig. 5, we evaluate the sensitivity to δ, to the number of samples M for reachability analysis with **randUP**, and the effect of the β-regularizer.

Sensitivity to the magnitude of $\boldsymbol{\epsilon}$: We consider two different noise levels:

1. $\sigma_{\epsilon_i}^2 = 10^{-7}$ for i=1, 2, 4, 5, and $\sigma_{\epsilon_i}^2 = 10^{-6}$ for i=3, 6.
2. $\sigma_{\epsilon_i}^2 = 10^{-6}$ for i=1, 2, $\sigma_{\epsilon_i}^2 = 10^{-5}$ for i=3, 6, and $\sigma_{\epsilon_i}^2 = 10^{-7}$ for i=4, 5.

Results for these different noise levels for different δ are reported in Tables 1 and 2. From Table 1, we see that the performance and overall probability of safety for small σ_{ϵ_i} is not sensitive to the chosen value of δ. We speculate that failures are mostly due to under-approximations from the approximate computation of the reachable sets with **randUP**. For higher noise levels, it is evident that the conservatism of the algorithm can be tuned by choosing a different value for δ, since failures come from statistical errors from updating the model with noisy data (see Theorem 1). We also observe faster times to reach $\mathcal{X}_{\text{goal}}$ when opting for lower probability of safety. In all scenarios, **SEELS** provides safety with high probability, verifying the theoretical guarantees of our framework.

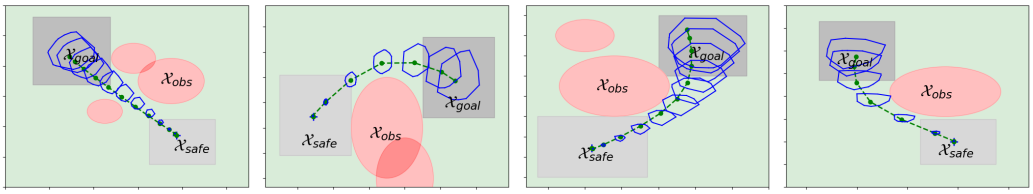

Figure 5: Scenarios considered for the ablation study (Fig. 2). Results on harder scenarios (shown in Fig. 1) showed a similar trend, where we also performed 250 randomized experiments and verified success rate and safety at 93.2% for $\delta = 0.1$, 90.5% for $\delta = 0.2$, and 88.8% for $\delta = 0.5$, with $M = 2500$ and β_i-regularization.