Chance-Constrained Trajectory Optimization for Safe Exploration and Learning of Nonlinear Systems

Yashwanth Kumar Nakka, Anqi Liu, Guanya Shi, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung

***Abstract*—Learning-based control algorithms require collection of abundant supervision for training. Safe exploration algorithms enable this data collection to proceed safely even when only partial knowledge is available. In this paper, we present a new episodic framework to design a sub-optimal pool of motion plans that aid exploration for learning unknown residual dynamics under safety constraints. We derive an iterative convex optimization algorithm that solves an information-cost Stochastic Nonlinear Optimal Control problem (Info-SNOC), subject to chance constraints and approximated dynamics to compute a safe trajectory. The optimization objective encodes both performance and exploration, and the safety is incorporated as distributionally robust chance constraints. The dynamics are predicted from a robust learning model. We prove the safety of rollouts from our exploration method and reduction in uncertainty over epochs ensuring consistency of our learning method. We validate the effectiveness of Info-SNOC by designing and implementing a pool of safe trajectories for a planar robot.**

I. Introduction

Robotic systems deployed in a real-world scenario often operate in partially-known environments. Modeling the complex dynamic interactions with the environment requires high-fidelity technique that are often computationally expensive. For example, spherical harmonics are used to model the effects of non-uniform gravity field on a spacecraft. Machine-learning models can remedy this difficulty by approximating the dynamics from data [1]–[4]. The learned models typically require off-line training with labeled data, often not available or hard to collect in many applications. Safe exploration is an efficient approach to collect ground truth data by safely interacting with the environment. Recent research on safe exploration [5] uses a deterministic approach in an episodic framework to collect labeled data by querying the domain of interest for informative data.

Planning for safe exploration is challenging when a probabilistic machine learning model is used to approximate unknown dynamics, because: 1) the uncertainties from the learned dynamic model lead to stochastic nonlinear dynamics, 2) the safety constraints are formulated as chance constraints that are generally non-convex, 3) performance objective may be a *min–max* stochastic optimal control problem, e.g. maximum exploration with minimum control effort and 4) the propagation errors and safety violations need to be quantified, when the controller is computed with estimated dynamics.

The authors are with California Institute of Technology. Email:{ynakka, anqiliu, gshi, anima, yyue, sjchung}@caltech.edu

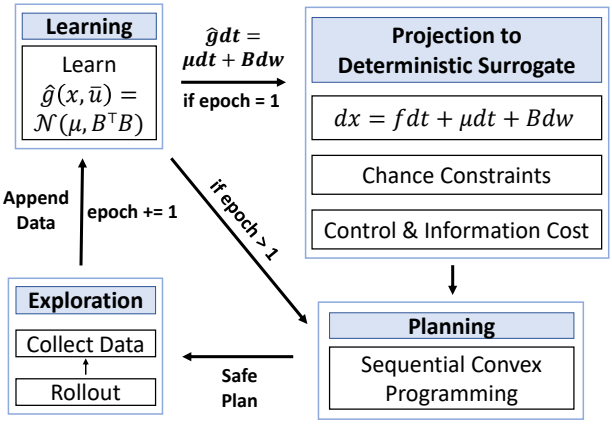


Fig. 1. An end-to-end episodic framework for safe exploration using chance-constrained trajectory optimization. In the framework, an initial estimate of the dynamics is computed using a known safe control policy [5]. A probabilistic safe trajectory and policy that satisfies safety chance-constraints is computed using Info-SNOC for the estimated dynamics. This policy is used for rollout with a stable feedback controller to collect data.

In this paper, we systematically address the aforementioned challenges by proposing an end-to-end episodic framework that unifies learning, planning, and exploration for active and safe data collection for continuous-time dynamical systems, as shown in Fig. 1. The key contributions of the present paper are summarized as follows: a) We use a multivariate robust regression model [6] under a covariate shift constraint to compute the multi-dimensional uncertainty estimates of the unknown dynamics; b) We propose a novel iterative solution method to Information Stochastic Nonlinear Optimal Control (Info-SNOC) problem to plan a pool of sub-optimal safe and informative trajectories with the learned approximation of the dynamics. We build on the recent method [7] to solve the chance-constrained SNOC problem by projecting it to the generalized polynomial chaos (gPC) space; and c) We prove the safety of rollouts from our exploration method and reduction in uncertainty over epochs ensuring consistency of our learning method under mild assumptions. Rollout is defined as executing the computed safe trajectory and policy using a stable feedback controller. To ensure real-time safety, the feedback controller is augmented with a safety filter [8].

Related Work: Safe exploration has been extensively studied in the reinforcement learning domain [9]. For continuous dynamical system, the problem has been studied using the following three frameworks: learning-based model-predictive control (MPC), dual-control, and active dynamics learning.

Learning-based MPC [5], [10]–[12] has been studied extensively for controlling the estimated system. These techniques are also applied for planning an information trajectory to learn online. The effect of uncertainty in the learned model on the propagation of dynamics is estimated using methods such as bounding box and linearization. We use the method of generalized polynomial chaos (gPC) [7] expansion for propagation, which has asymptotic convergence to the original distribution, and provides guarantees on the constraint satisfaction. In MPC, safety is guaranteed by recursively checking feasibility and appending a safe policy to an exploring policy at the end of each rollout. In contrast, we formulate safety as joint chance constraints for given risk of constraint violation.

Estimating unknown parameters while simultaneous optimizing for performance has been studied as a dual control problem [13]. Dual control is an optimal control problem formulation to compute a control policy that is optimized for performance and guaranteed parameter convergence. In some recent work [14], [15], the convergence of the estimate is achieved by using persistence of excitation condition in the optimal control problem. Our method uses Sequential Convex Programming (SCP) [16]–[18] to compute the persistent excitation trajectory. Recent work [19] uses nonlinear programming tools to solve optimal control problems with an upper-confidence bound [20] cost for exploration without safety constraints. We follow a similar approach but formulate the planning problem as an SNOC with distributionally robust linear and quadratic chance constraints for safety. The distributionally robust chance constraints are convexified via projection to the gPC space. The algorithm proposed in this paper can be used in the MPC framework with appropriate terminal conditions for feasibility and to solve dual control problems with high efficiency using the interior point methods.

The paper is organized as follows. We discuss the SNOC problem with results on deterministic approximations of chance constraints along with preliminaries on robust regression in Sec. II. The Info-SNOC algorithm along with exploration policy is presented in Sec. III. In Sec. IV, we derive the end-to-end safety guarantees. In Sec. V, we apply the Info-SNOC algorithm to the nonlinear three degree-of-freedom spacecraft robot model [21]. We conclude the paper in Sec. VI with brief discussion on the results of the analysis and the application of the Info-SNOC method.

II. Preliminaries and Problem Definition

A. Robust Regression For Learning

Learning dynamics is regarded as a regression problem under covariate shift. Covariate shift is a special case of distribution shift between training and testing data distributions, where the conditional output distribution given the input variable remains the same while the input distribution is different between training and testing. We refer to them as the source distribution $\Pr_s(x)$ and the target distribution $\Pr_t(x)$. An exploration step in active data collection for learning dynamics is a covariate shift problem. We use x and y to represent input and output of the learning model. Robust regression is derived from a *min–max* adversarial estimation framework, where the estimator tries to minimize a loss function and the adversary tries to maximize the loss under statistical constraints. The minimax framework derives a model that is robust to the worst-case possible data by generating a conditional distribution that is "compatible" with finite training data, while minimizing a loss function defined on a testing data distribution. Robust regression can handle multivariate outputs and the correlations efficiently by incorporating neural networks and predicting a multivariate Gaussian distribution directly, whereas traditional methods like Gaussian process regression suffer from high-dimensions and require heavy tuning of kernels [22].

The resulting Gaussian distributions provided by this learning framework are given below. For more technical details, we refer the readers to the prior work [6], [22]. The output Gaussian distribution takes the form $N(\mu, \Sigma)$, where

$$\Sigma(x,\theta) = \left(2\frac{\Pr_s(x)}{\Pr_t(x)}\theta_{yy} + \Sigma_0^{-1}\right)^{-1} \tag{1}$$
$$\mu(x,\theta) = \Sigma(x,\theta)\left(-2\frac{\Pr_s(x)}{\Pr_t(x)}\theta_{xy}\phi_f(x) + \mu_0\Sigma_0^{-1}\right)$$

with a non-informative base distribution $N_0(\mu_0, \Sigma_0)$, θ is the model parameter learned from data, and $\phi_f(x)$ is the feature function. The features $\phi_f(x)$ can be learned using neural networks directly from data. The density ratio $\frac{\Pr_s(x)}{\Pr_t(x)}$ is estimated from data beforehand.

B. Optimal and Safe Planning Problem

In this section, we present the finite-time chance-constrained stochastic optimal control problem formulation [7] used to design an informative trajectory. The optimization has control effort and terminal cost as performance objectives, and the safety is modelled as joint chance constraints. The full stochastic optimal control problem is as follows:

$$J^* = \min_{x(t),\bar{u}(t)} \quad \mathbb{E}\left[\int_{t_0}^{t_f} J(x(t),\bar{u}(t))dt + J_f(x(t),\bar{u}(t))\right] \tag{2}$$
$$\text{s.t.} \quad \dot{x}(t) = f(x(t),\bar{u}(t)) + \hat{g}(x(t),\bar{u}(t)) \tag{3}$$
$$\Pr(x(t) \in \mathcal{F}) \geq 1-\epsilon, \quad \forall t \in [t_0, t_f] \tag{4}$$
$$\bar{u}(t) \in \mathcal{U} \quad \forall t \in [t_0, t_f] \tag{5}$$
$$x(t_0) = x_0 \quad \mathbb{E}(x(t_f)) = \mu_{x_f}, \tag{6}$$

where any realization of the random variable $x \in \mathcal{X} \subseteq \mathbb{R}^n$, x_0 and x_f are initial and terminal state distributions respectively, the control $\bar{u} \in \mathcal{U} \subseteq \mathbb{R}^m$ is deterministic, $\hat{g}$ is the learned probabilistic model, and $\mathbb{E}$ is the expectation operator. The modelling assumptions and the problem formulation will be elaborated in the following sections.

1) Dynamical Model: The $\hat{g}$ term of (3) is the estimated model of the unknown g term of the original dynamics:

$$\dot{\bar{p}} = f(\bar{p},\bar{u}) + \underbrace{g(\bar{p},\bar{u})}_{\text{unknown}}, \tag{7}$$

where the state $\bar{p} \in \mathcal{X}$ is now considered deterministic, and the functions $f : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^n$ and $g : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^n$ are Lipschitz with respect to $\bar{p}$ and $\bar{u}$.

Assumption 1. The control set $\mathcal{U}$ is convex and compact.

Remark 1. The maximum entropy distribution with the known mean μ_x and covariance matrix Σ_x of the random variable x is the Gaussian distribution $\mathcal{N}(\mu_x, \Sigma_x)$. We follow this notation for mean and covariance matrix for remainder of the paper.

The learning algorithm computes the mean vector $\mu_g(\mu_x, \bar{u})$ and the covariance matrix $\Sigma_g(\mu_x, \bar{u})$ estimates of $g(x, \bar{u})$ that are functions of mean μ_x of the state x and control $\bar{u}$. Due to Remark 1, the unknown bias term is modeled as a multivariate Gaussian distribution $\hat{g}(\mu_x, \bar{u}) \sim \mathcal{N}(\mu_g, \Sigma_g)$. The estimate $\hat{g}$ in (3) can be expressed as

$$\hat{g} = B(\mu_x, \bar{u})\theta + \mu_g(\mu_x, \bar{u}), \tag{8}$$

where $\theta \sim \mathcal{N}(0, \mathrm{I})$ i.i.d and $B(\mu_x, \bar{u})B^\top(\mu_x, \bar{u}) = \Sigma_g(\mu_x, \bar{u})$. Using (8), (7) can be written in standard Ito stochastic differential equation (SDE) form as given below, where $\theta dt = dW$.

$$dx = f(x, \bar{u})dt + \mu_g(\mu_x, \bar{u})dt + B(\mu_x, \bar{u})dW \tag{9}$$

The existence and uniqueness of a solution to the SDE for a given initial distribution x_0 and control trajectory $\bar{u}$ such that $\Pr(|x(t_0) - x_0)| = 0) = 1$ with measure Pr, is guaranteed by the following assumptions: *(a) Lipschitz Condition:* There exists a constant $K_1 > 0$ such that $\forall t \geq t_0$ any realization $(a, \bar{u}_a)$ and $(b, \bar{u}_b) \in \mathcal{X} \times \mathcal{U}$

$$\begin{aligned} &\|f(a, \bar{u}_a) - f(b, \bar{u}_b)\| + \|\mu_g(a, \bar{u}_a) - \mu_g(b, \bar{u}_b)\| \\ &+\|B(a, \bar{u}_a) - B(b, \bar{u}_b)\|_F \leq K_1\|(a, \bar{u}_a) - (b, \bar{u}_b)\|. \end{aligned} \tag{10}$$

(b) Restriction on Growth: There exists a constant $K_2 > 0$, $\|.\|_F$ is the Frobenius norm such that $\|f(a, \bar{u}_a)\|^2 + \|\mu_g(a, \bar{u}_a)\|^2 + \|B(a, \bar{u}_a)\|_F^2 \leq K_2(1 + \|(a, \bar{u}_a)\|^2)$.

Assumption 2. The approximate system (9) is controllable in the given feasible space.

2) State and Safety Constraints: Safety is defined as a constraint on the state space x, $x(t) \in \mathcal{F}$ at time t. The safe set $\mathcal{F}$ is relaxed by formulating a joint chance constraint with risk of constraint violation as

$$\Pr(x \in \mathcal{F}) \geq 1 - \epsilon. \tag{11}$$

The constant ϵ is called the risk measure of a chance constraint in this paper. We consider the polytopic constraint set $\mathcal{F}_{\text{lin}} = \{x \in \mathcal{X} : \wedge_{i=1}^k a_i^\top x + b_i \leq 0\}$ with k flat sides and a quadratic constraint set $\mathcal{F}_{\text{quad}} = \{x \in \mathcal{X} : x^\top Ax \leq c\}$ for any realization x of the state. The joint chance constraint $\Pr(\wedge_{i=1}^k a_i^\top x + b_i \leq 0) \geq 1 - \epsilon$ can be transformed to the following individual chance constraint:

$$\Pr(a_i^\top x + b_i \leq 0) \geq 1 - \epsilon_i, \tag{12}$$

such that $\sum_{i=1}^k \epsilon_i = \epsilon$. Here we use $\epsilon_i = \frac{\epsilon}{k}$. The individual risk measure ϵ_i can be optimally allocated between the k constraints as discussed in [23]. A quadratic chance constraint is given as

$$\Pr(x^\top Ax \geq c) \leq \epsilon, \tag{13}$$

where A is a positive definite matrix. The chance constrained formulation of the sets in (12,13) is computationally intractable due to the multi-modal probability density function of x and multi-dimensional integrals. We use a tractable conservative approximation by formulating distributionally robust chance constraint [24] for known mean μ_x and variance Σ_x of the random variable x.

Lemma 1. *The linear distributionally robust chance constraint* $\inf_{x\sim(\mu_x,\Sigma_x)} \Pr(a_i^\top x + b_i \leq 0) \geq 1 - \epsilon_\ell$ *is equivalent to the deterministic constraint:*

$$a^\top \mu_x + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top \Sigma_x a} \leq 0. \tag{14}$$

Proof: See [24]. ∎

Lemma 2. *The semi-definite constraint on the variance* Σ_x

$$\frac{1}{c}\mathrm{tr}(A\Sigma_x) \leq \epsilon_q \tag{15}$$

is a conservative deterministic approximation of the quadratic chance-constraint $\Pr((x - \mu_x)^\top A(x - \mu_x) \geq c) \leq \epsilon_q$.

Proof: See [7]. ∎

The risk measures ϵ_ℓ and ϵ_q are assumed to be given.

3) Cost Functional: The integrand cost functional includes two objectives: 1) Exploration: to achieve maximum value of information for learning the unknown dynamics g, and 2) Performance: to achieve fuel optimality. The cost functional $J = J_\mathcal{C} + J_\mathcal{I}$ is split into $J_\mathcal{C}$ and $J_\mathcal{I}$ corresponding to the performance cost and the information cost for exploration, respectively. The cost functional $J_\mathcal{C}$ is of the following form, and is convex in $\bar{u}$.

$$J_\mathcal{C} = \|\bar{u}\|_s \quad \text{where} \quad s \in \{1, 2, \infty\} \tag{16}$$

We use the following Gaussian Upper Confidence Bound (UCB) [20] based information cost for exploration, for each i^{th} element μ_{g_i} in μ_g and i^{th} diagonal element σ_i^2 in Σ_g.

$$J_\mathcal{I} = -\sum_{i=1}^{n}\Big(\mu_{g_i}(\mu_x, \bar{u}) + \gamma^{1/2}\sigma_i(\mu_x, \bar{u})\Big) \tag{17}$$

The information cost $J_\mathcal{I}$ in (17) is a functional of the mean μ_x of the state x and control $\bar{u}$ at time t. The coefficient γ is chosen based on the confidence interval. Minimizing the cost $J_\mathcal{I}$, we maximize the information [20] available in the trajectory x to learn the unknown model g. The cost functional includes the mean estimate μ_g to implicitly trade off between exploration–exploitation during the trajectory design. The terminal cost functional J_f is quadratic in the state x, $J_f = x(t_f)^\top Q_f x(t_f)$, where Q_f is a positive semi-definite function. For rest of the paper we consider the following individual chance-constrained problem,

$$\begin{aligned} (x^*, \bar{u}^*) = &\underset{x(t),\bar{u}(t)}{\text{argmin}} \quad \mathbb{E}\left[\int_{t_0}^{t_f}((1-\rho)J_\mathcal{C} + \rho J_\mathcal{I})dt + J_f\right] \\ &\text{s.t.}\{(9),\ (12),\ (13),\ (5),\ (6)\} \end{aligned} \tag{18}$$

that is assumed to have a feasible solution with $\rho \in [0, 1]$.

C. Generalized Polynomial Chaos (gPC)

The problem (18) is projected into a finite-dimensional space using the generalized polynomial chaos algorithm presented in [7]. We briefly review the ideas and equations. We refer the readers to [7] for details on computing the functions and matrices that are mentioned below. In the gPC expansion,

any $\mathcal{L}_2$ bounded random variable $x(t)$ is expressed as product of the basis functions matrix $\Phi(\theta)$ defined on the random variable θ and the deterministic time varying coefficients $X(t) = \begin{bmatrix} x_{10} & \cdots & x_{1\ell} & \cdots & x_{n0} & \cdots & x_{n\ell} \end{bmatrix}^\top$. The functions ϕ_i $i \in \{0, 1, \ldots, \ell\}$ are chosen to be Gauss-Hermite polynomials for this paper. Let $\Phi(\theta) = \begin{bmatrix} \phi_0(\theta) & \cdots & \phi_\ell(\theta) \end{bmatrix}^\top$.

$$x \approx \bar{\Phi} X; \text{ where } \bar{\Phi} = \mathbb{I}_{n\times n} \otimes \Phi(\theta)^\top \tag{19}$$

A Galerkin projection is used to transform the SDE (9) to the deterministic equation,

$$\dot{X} = \bar{f}(X, \bar{u}) + \bar{\mu}_g(\mu_x, \bar{u}) + \bar{B}(\mu_x, \bar{u}). \tag{20}$$

The exact form of the function $\bar{f}$ is given in [7]. The moments of the random variable x, μ_x and Σ_x can be expressed as polynomial functions of elements of X. The polynomial functions defined in [7] are used to project the distributionally robust linear chance constraint in (14) to a second-order cone constraint in X as follows:

$$(a^\top \otimes S)X + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{X^\top U N N^\top U^\top X} \leq 0, \tag{21}$$

where the matrices S, U, and N are functions of the expectation of polynomials Φ, that are given in [7]. The quadratic chance constraint in (15) is transformed to the following quadratic constraint in X:

$$\sum_{i=1}^{n}\sum_{k=1}^{\ell} a_i \langle \phi_k, \phi_k \rangle x_{ik}^2 \leq \epsilon_q c, \tag{22}$$

given that A is a diagonal matrix with i^{th} diagonal element as a_i, where $\langle ., . \rangle$ is an inner product operation. Similarly, we project the initial and terminal state constraints $X(t_0)$ and $X(t_f)$. In the next section, we present the Info-SNOC algorithm by using the projected optimal control problem.

III. Info-SNOC Main Algorithm

In this section, we present the main algorithm of the paper by projecting (18) to the gPC space. We formulate a deterministic optimal control problem in the gPC space and solve it using Sequential Convex Programming (SCP) method [7], [17], [18]. The gPC projection of (18) is given by the following equation

$$\begin{aligned} (X^*, \bar{u}^*) = \underset{X(t), \bar{u}(t)}{\text{argmin}} \quad & \Big[\int_{t_0}^{t_f} ((1-\rho)J_{\mathcal{C}} + \rho J_{\mathcal{I}})dt + \mathbb{E}J_{df}\Big] \\ \text{s.t. } & \{(20),\ (21),\ (22),\ (5)\} \\ & X(t_0) = X_0, \quad X(t_f) = X_f. \end{aligned} \tag{23}$$

In SCP, the projected dynamics (20) is linearized about a feasible nominal trajectory and discretized to formulate a linear equality constraint. Note that the constraints (21) and (22) are already convex in the states X. The terminal constraint J_f is projected to the quadratic constraint $J_{df} = X_f^\top \bar{\Phi}^\top Q_f \bar{\Phi} X_f$. The information cost functional $J_{\mathcal{I}}$ from (17) is expressed as a function of X by using the polynomial representation [7] of μ_x in terms of X. Let $S = (X, \bar{u})^\top$ and the cost $J_{\mathcal{I}}$ is linearized around a feasible nominal trajectory $S^o = (X^o, \bar{u}^o)^\top$ to derive a linear convex cost functional $J_{d\mathcal{I}}$:

$$\begin{aligned} J_{d\mathcal{I}} = \sum_{i=1}^{n} \Big(& -\mu_{g_i}(S^o) - \gamma^{1/2}\sigma_i(S^o) \\ & - \frac{\partial \mu_{g_i}}{\partial S}\Big|_{S^o}(S - S^o) - \gamma^{1/2}\frac{\partial \sigma_i}{\partial S}\Big|_{S^o}(S - S^o)\Big). \end{aligned} \tag{24}$$

We use the convex approximation $J_{d\mathcal{I}}$ as the information cost in the SCP formulation of the optimal control problem in (23). In the gPC space, we split the problem into two cases: a) $\rho = 0$ that computes a performance trajectory, and b) $\rho \in (0, 1]$ that computes information trajectory to have stable iterations. The main algorithm is outlined below.

Algorithm 1 Info-SNOC using SCP [18] and gPC [7]

1: Initial Safe Set Data, Feasible Nominal Trajectory$(x^o, \bar{u}^o)$
2: gPC Projection as discussed in Sec. II-C
3: Linearize the gPC cost and dynamics, see [7]
4: epoch = 1
5: **while** Learning Criteria Not Satisfied **do**
6: Robust Regression
7: $(x_p, \bar{u}_p)$ = SCP$((x^o, \bar{u}^o), \rho = 0)$, using (23)
8: $(x_i, \bar{u}_i)$ = SCP$((x_p, \bar{u}_p), \rho \in (0, 1])$, using (23)
9: Sample $(x_i, \bar{u}_i)$ for $(\bar{p}_d, \bar{u}_d)$
10: Rollout using sample $(\bar{p}_d, \bar{u}_d)$ and u_c
11: Data collection during rollout
12: epoch $\leftarrow$ epoch + 1
13: **end while**

Algorithm: An initial estimate of the model (8) learned from data generated by a known safe control policy, and a nominal initial trajectory (x^o, u^o) is used to initialize Algorithm 1. The stochastic model and the chance constraints are projected to gPC state space, which is in line 2 of Algorithm 1. The projected dynamics is linearized around the nominal trajectory and used as a constraint in the SCP. The projection step is only needed in the first epoch. The projected system can be directly used for epoch > 1. The current estimated model is used to solve (23) using SCP, in line 7 with ρ = 0, for a performance trajectory. The output $(x_p, \bar{u}_p)$ of this optimization is used as initialization to the Info-SNOC problem obtained by setting $\rho \in (0, 1]$. The information trajectory $(x_i, \bar{u}_i)$ is then sampled for a safe motion plan in line 9, that is used for rollout, in line 10, to collect more data for learning. The SCP step is performed in the gPC space X. After each SCP step, the gPC space coordinates X are projected back to the random variable x space. The algorithm outputs a trajectory of random variable x with finite variance at each epoch.

Convergence and Optimality: The information trajectory $(x_i, \bar{u}_i)$ computed using SCP with the approximate linear information cost $J_{d\mathcal{I}}$ (24) is a sub-optimal solution of (23) with the optimal cost value $J^*_{d\mathcal{I}}$. Therefore, the optimal cost of (23) given by $J^*_{\mathcal{I}}$ is bounded above by $J^*_{d\mathcal{I}}$, $J^*_{\mathcal{I}} \leq J^*_{d\mathcal{I}}$. For the Info-SNOC algorithm, we cannot guarantee the convergence of SCP iterations to a Karush-Kuhn-Tucker point using the method in [16], [18] due to the non-convexity of $J_{\mathcal{I}}$. Due to the non-convex cost function $J_{\mathcal{I}}$, the linear approximation $J_{d\mathcal{I}}$ of the

cost $J_{\mathcal{I}}$ can potentially lead to numerical instability in SCP iterations. Finding an initial performance trajectory, $\rho = 0$, and then optimizing for information, $\rho \in (0, 1]$, is observed to be numerically stable compared to directly optimizing for the original cost functional $J = J_{\mathcal{C}} + J_{\mathcal{I}}$. The two advantages of this approach are: 1) we compute a fuel efficient trajectory that is also informative, and 2) SCP iterations are stable as we are searching around a feasible trajectory.

Feasibility: The initial phases of learning might lead to a large covariance Σ_g due to the insufficient data, resulting in an infeasible optimal control problem. To overcome this, we use two strategies: 1) Explore the initial safe set till we find a feasible solution to the problem, and 2) Use slack variables on the terminal condition to approximately reach the goal accounting for a large variance. The feasibility of SCP iterations is ensured by using the techniques discussed in Sec. IV.E of [17].

A. Rollout Policy Implementation

The information trajectory $(x_i, \bar{u}_i)$ computed using the Info-SNOC algorithm is sampled for a pool of motion plans $(\bar{p}_d, \bar{u}_d)$. The trajectory pool is computed by randomly sampling the multivariate Gaussian distribution θ and transforming it using the gPC expansion $x(\theta) \approx \bar{\Phi}(\theta)X$. For any realization $\bar{\theta}$ of θ, we get a deterministic trajectory $\bar{p}_d = \bar{\Phi}(\bar{\theta})X$ that is ϵ safe with respect to the distributionally robust chance constraints. The trajectory $(\bar{p}_d, \bar{u}_d)$ is executed using the closed-loop control law $u_c = u_c(\bar{p}, \bar{p}_d, \bar{u}_i)$ for rollout, where $\bar{p}$ is the current state. The properties of the control law u_c and the safety during rollout are studied in the following section.

IV. Analysis

In this section, we present the main theoretical results analyzing the following two questions: 1) at any epoch i how do learning errors translate to safety violation bounds during rollout, and 2) consistency of the multivariate robust regression as epoch $\to \infty$.

Assumption 3. The projected problem (23) computes a feasible trajectory to the original problem (18). The assumption is generally true if we choose a sufficient number of polynomials [7], for the projection operation.

Assumption 4. The following bounds are satisfied with high probability for the same input $(\bar{p}, \bar{u})$ to the original model g, and the learned model μ_g. The $\mathrm{tr}(\Sigma_g)$ also satisfies the bounded shown below

$$\|g(\bar{p}, \bar{u}) - \mu_g(\bar{p}, \bar{u})\|_2^2 \leq c_1, \quad \mathrm{tr}\left(\Sigma_g\right) \leq c_2. \tag{25}$$

As shown in [22], the mean predictions made by the model learned using robust regression is bounded by c_1, which depends on the choice of the function class for learning. The variance prediction c_2 is bounded by design. With Assumptions 3 and 4, the analysis is decomposed into the following three subsections.

A. State Error Bounds During Rollout

We make the following assumptions on the nominal system $\dot{\bar{p}} = f(\bar{p}, u)$ to derive the state tracking error bound during rollout.

Assumption 5. There exists a globally exponentially stable (i.e., finite-gain $\mathcal{L}_p$ stable) tracking control law $u_c = u_c(\bar{p}, \bar{p}_d, \bar{u}_d)$ for the nominal dynamics $\dot{\bar{p}} = f(\bar{p}, u_c)$. The control law u_c satisfies the property $u_c(\bar{p}_d, \bar{p}_d, \bar{u}_d) = \bar{u}_d$ for any sampled trajectory $(\bar{p}_d, \bar{u}_d)$ from the information trajectory $(x_i, \bar{u}_i)$. At any time t the state $\bar{p}$ satisfies the following inequality, when the closed-loop control u_c is applied to the nominal dynamics,

$$M(\bar{p}, t)\frac{\partial f}{\partial \bar{p}} + \left(\frac{\partial f}{\partial \bar{p}}\right)^{\top} M(\bar{p}, t) + \frac{d}{dt}M(\bar{p}, t) \leq -2\alpha M(\bar{p}, t), \tag{26}$$

where $f = f(\bar{p}, u_c(\bar{p}, \bar{p}_d, \bar{u}_d))$, $\alpha > 0$, $M(\bar{p}, t)$ is a uniformly positive definite matrix with $(\lambda_{\min}(M)\|\bar{p}\|^2 \leq \bar{p}^{\top} M(\bar{p}, t)\bar{p} \leq \lambda_{\max}(M)\|\bar{p}\|^2)$, and $\lambda_{\max}$ and $\lambda_{\min}$ are the maximum and minimum eigenvalues.

Assumption 6. The unknown model g satisfies the bound $\| \left(g\left(\bar{p}, u_c\right) - g(\bar{p}_d, \bar{u}_d)\right) \|_2^2 \leq c_3$.

Assumption 7. The probability density ratio $\frac{\rho_{x_i(t)}}{\rho_{x_i(0)}} \leq r$ is bounded, where the functions $\rho_{x_i(0)}$,$\rho_{x_i(t)}$ are probability density functions for x_i at time t_0 and t respectively.

Lemma 3. *Given that the estimated model* (8) *satisfies the Assumption 4, and the systems (7) and* (9) *satisfy Assumptions 5, 6, 7, if the control $u_c = u_c(\bar{p}, x_i, \bar{u}_i)$ is applied to the system* (7)*, then the following condition holds at time t*

$$\begin{aligned}\mathbb{E}_{x_i(t)}(\|\bar{p} - x_i\|_2^2) \leq \frac{\lambda_{\max}(M)}{2\lambda_{\min}(M)\alpha_m}(c_1 + c_2 + c_3)r \\ + \frac{\lambda_{\max}(M)r}{\lambda_{\min}(M)}\mathbb{E}\left(\|\bar{p}(0) - x_i(0)\|^2\right)e^{-2\alpha_m t},\end{aligned} \tag{27}$$

where $(x_i, \bar{u}_i)$ is computed from (23) *and $\alpha_m = (\alpha - 1)$. The states $\bar{p} \in \mathcal{X}$, and $x_i \in \mathcal{X}$ are feasible trajectories of the deterministic dynamics (7) and the SDE (9) for the initial conditions $\bar{p}(0) \in \mathcal{X}$ and $x_i(0) \in \mathcal{X}$ respectively at $t \geq t_0$.*

Proof: See [25]. ∎

Lemma 3 states that the expected mean squared error $\mathbb{E}(\|\bar{p} - x_i\|^2)$ is bounded by $\frac{\lambda_{\max}(M)(c_1+c_2+c_3)r}{2\alpha_m\lambda_{\min}(M)}$ as $t \to \infty$ when the control law u_c is applied to the dynamics in (7). The bounded tracking performance leads to constraint violation, which is studied in the next section.

B. Safety Bounds

The safety of the original system (7) for the linear and quadratic chance constraints during rollout with a controller u_c discussed in Sec. IV-A is analyzed in Theorems 1 and 2.

Theorem 1. *Given a feasible solution $(x, \bar{u}_x)$ of (18), with the quadratic chance constraint* $\Pr((x - \mu_x)^{\top} A(x - \mu_x) \geq c) \leq \frac{\mathbb{E}((x-\mu_x)^{\top} A(x-\mu_x))}{c} \leq \epsilon_q$*, the trajectory $\bar{p}$ of the deterministic dynamics* (7) *satisfies the following inequality at any time t:*

$$(\bar{p} - \mu_x)^{\top} A(\bar{p} - \mu_x) \leq \lambda_{\max}(A)\mathbb{E}_x\left(\|\bar{p} - x\|_2^2\right), \tag{28}$$

where $\mathbb{E}\left(\|\bar{p} - x\|_2^2\right)$ is bounded as defined in Lemma 3.

Proof: Consider the expectation of the ellipsoidal set $(\bar{p}-\mu_x)^\top A(\bar{p}-\mu_x)$. Using $\bar{p}-\mu_x = \bar{p}-x+x-\mu_x$, the expectation of the set can be expressed as follows

$$\mathbb{E}\left((\bar{p}-\mu_x)^\top A(\bar{p}-\mu_x)\right) = \mathbb{E}\left((\bar{p}-x)^\top A(\bar{p}-x)\right) \quad (29)$$
$$+\mathbb{E}\left((x-\mu_x)^\top A(x-\mu_x)\right) + 2\mathbb{E}\left((\bar{p}-x)^\top A(x-\mu_x)\right).$$

Using the following equality:

$$\mathbb{E}\left((\bar{p}-x)^\top A(x-\mu_x)\right) = -\mathbb{E}\left((x-\mu_x)^\top A(x-\mu_x)\right),$$

and $\mathbb{E}\left((x-\mu_x)^\top A(x-\mu_x)\right) \geq 0$ in (29), we obtain the constraint bound in (28). Using the propagation bounds (27) in (28), we can show that the constraint violation bound is a function of learning bounds c_1, c_2, and c_3. ∎

Note that if the learning method converges, i.e., $c_1 \to 0$, $c_2 \to 0$, and $c_3 \to 0$, then $\bar{p} \to \mu_x$. The quadratic constraint violation in (28) depends on the tracking error and the size of the ellipsoidal set described by A.

Theorem 2. *Given a feasible solution $(x, \bar{u}_x)$ of (18) with $\inf_{x\sim(\mu_x,\Sigma_x)} \Pr(a^\top x + b \leq 0) \geq 1-\epsilon_\ell$, the trajectory $\bar{p}$ of the deterministic dynamics* (7)*, with control u_c, satisfies the following inequality at any time t:*

$$\inf_{x\sim(\mu_x,\Sigma_x)} \Pr\left(a^\top \bar{p} + b \leq \delta(x)\right) \geq 1-\epsilon_\ell, \quad (30)$$

where $\delta(x) = \|a\|_2\mathbb{E}_x(\|(\bar{p}-x)\|_2) - \|a\|_2\sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{c_4}$, $\mathbb{E}_x(\|(\bar{p}-x\|_2)$ is bounded as defined in (27) *and $\mathrm{tr}(\Sigma_x) = c_4$.*

Proof: From Lemma 1, the feasible solution (x, u_x) satisfies the equivalent condition $\mathcal{P}(\mu_x, \Sigma_x) \leq 0$, where $\mathcal{P}(\mu_x, \Sigma_x) = a^\top\mu_x + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_x a}$ for the risk measure ϵ_ℓ, mean μ_x and covariance Σ_x. Consider the similar condition for the actual trajectory $\bar{p}(t)$, $\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}})$, as shown below.

$$\begin{aligned}\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) &= a^\top\mu_{\bar{p}} + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_{\bar{p}}a} \\ &= a^\top\mu_x + a^\top(\mu_{\bar{p}} - \mu_x) + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_{\bar{p}}a} \\ &+ \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_x a} - \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_x a} \quad (31)\end{aligned}$$

Note that since the system (7) is deterministic, we have $\mu_{\bar{p}} = \bar{p}$ and $\Sigma_{\bar{p}} = 0$. Using $\mathcal{P}(\mu_x, \Sigma_x) \leq 0$, the right hand side of the above inequality reduces to the following:

$$\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) \leq a^\top(\bar{p} - \mu_x) - \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{a^\top\Sigma_x a}. \quad (32)$$

Using the decomposition $\Sigma_x = \tilde{G}^\top\tilde{G}$, Cauchy-Schwarz's inequality, and Jensen's inequality, we have $a^\top(\mu_{\bar{p}} - \mu_x) \leq \|a\|_2\|(\bar{p} - \mu_x)\|_2 \leq \|a\|_2\mathbb{E}_x(\|(\bar{p} - x)\|_2)$. Using the sub-multiplicative property of ℓ_2-norm in the inequality above, we have $\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) \leq \|a\|_2\left(\mathbb{E}(\|(\bar{p}-x)\|_2) - \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\|\tilde{G}\|_F\right)$. Assuming that $\mathrm{tr}(\Sigma_x) = c_4$ in the above inequality, we have

$$\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) \leq \|a\|_2\mathbb{E}(\|(\bar{p}-x)\|_2) - \|a\|_2\sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{c_4}.$$

The above inequality is equivalent to the probabilistic linear constraint in (30). The bound $\delta(x) = \|a\|_2\mathbb{E}(\|(\bar{p}-x)\|_2) - \|a\|_2\sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}}\sqrt{c_4}$ is a function of the learning bounds in (25) by substituting the tracking bound in (27). ∎

The linear constraint is offset by δ leading to constraint violation of the original formulation (12). Note that, if $c_1 \to 0$, $c_2 \to 0$, $c_3 \to 0$, and $c_4 \to 0$ then $\delta \to 0$. In order to ensure real-time safety during trajectory tracking, we use a high gain control for disturbance attenuation with safety filter augmentation for constraint satisfaction.

C. Consistency

Data is collected during the rollout of the dynamical system to learn a new model for next epoch. For epoch e, predictor g_x^e is a multivariate Gaussian distribution $\mathcal{N}(\mu_x^e, \Sigma_x^e)$ and $\tilde{g}_x$ is the empirical true data. We assume that set $\mathcal{X}_e \subset \mathcal{X}$ generated by the optimization problem in (18) for the first e iterations is a discretization of $\mathcal{X}$. Assuming that there exists a global optimal predictor g_x^* in the function class $\mathcal{G}$ that can achieve the best error at each epoch e:

$$\max_{x\in\mathcal{X}_e} \|\tilde{g}_x - g_x^*\|_2 = \epsilon^*, \quad (33)$$

the consistency of the learning algorithm is defined as

$$\forall x \in \mathcal{X}_e, \quad \sum_e \|g_x^e - g_x^*\|_2 \leq \eta \quad (34)$$

and proven in the following theorem.

Theorem 3. *If the maximum prediction error at epoch e is ϵ_e, i.e. $\max_{x\in\mathcal{X}_e} \|\tilde{g}_x - \mu_x^e\|_2 \triangleq \epsilon_e$, the learning error of Algorithm 1 converges with probability $1-\delta$ for any η defined as in* (34) *after $\mathcal{E}$ epochs. Here $\mathcal{E}$ is the smallest integer that satisfy $\sqrt{\mathcal{E}\sum_{e=1}^{\mathcal{E}}\gamma_e} \leq \eta$ with $\gamma_e = \epsilon_{e-1}^2 + dC^2 + \epsilon^{*2}$. δ is defined as*

$$\delta \triangleq |\mathcal{X}_e|(C^d\prod_p^d \Delta_p)^{-1}\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_x^{e-1}|^{\frac{1}{2}}}e^{-\frac{1}{2}C^2\sum_p\Delta_p}, \quad (35)$$

where d is the output dimension, and $\Delta_p \triangleq \sum_{q=1}^d m_{qp} > 0$, $\forall 1 \leq p \leq d$, if $\mathcal{M} = (\Sigma_x^{e-1})^{-1}$ and $\mathcal{M} = (m_{pq})$.

Proof: The squared error of g_x^e from the optimal predictor g_x^* is $r_e^2 = \|g_x^e - g_x^*\|_2^2$. We first decompose g_x^e and bound it using the inequality $\Pr(g_x^e - \mu_x^{e-1} \geq C\mathbf{e}) < \delta_e$, $\forall x \in \mathcal{X}_e$, since $g_x^e \sim \mathcal{N}(\mu_x^{e-1}, \Sigma_x^{e-1})$ as follows:

$$r_e^2 = \|g_x^e - g_x^*\|_2^2 \leq \|\mu_x^{e-1} + C\mathbf{e} - g_x^*\|_2^2, \quad (36)$$

where $\delta_e = (C^d\prod_p^d\Delta_p)^{-1}\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_x^{e-1}|^{\frac{1}{2}}}\exp(-\frac{1}{2}C^2\sum_p\Delta_p)$, Δ_p is defined in [26], and $\mathbf{e}$ is unit vector in d dimensions. This is the tail probabilities inequality of multivariate Gaussian distributions. The error r_e^2 is then bounded using the empirical prediction error $\|\mu_x^{e-1} - \tilde{g}_x\|_2^2$ and the best prediction error (33) as follows:

$$r_e^2 = \|\mu_x^{e-1} - \tilde{g}_x\|_2^2 + dC^2 + \epsilon^{*2} \leq \epsilon_{e-1}^2 + dC^2 + \epsilon^{*2}.$$

For converged robust regression learning in epoch e, we have $x \in \mathcal{X}_e$, $|\|\mu_x^{e-1}\|_2^2 + \mathrm{tr}(\Sigma_x^{e-1}) - \|\tilde{g}_x\|_2^2| \leq \omega_e$ [22]. The bound on r_e is made more concrete as

$$\begin{aligned}r_e^2 &\leq |\|\mu_x^{e-1}\|_2^2 - \|\tilde{g}_x\|_2^2| + dC^2 + \epsilon^{*2} \\ &\leq \omega_e + \mathrm{tr}(\Sigma_x^{e-1}) + dC^2 + \epsilon^{*2}\end{aligned} \quad (37)$$

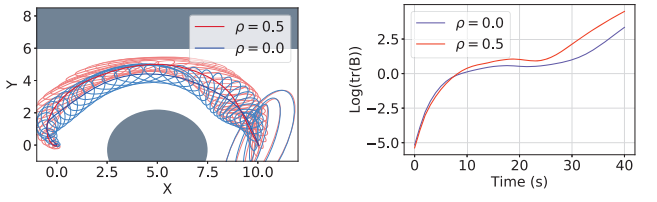
where ω_e is a hyper-parameter that is associated with model selection in robust regression in epoch e [22]. Using Cauchy-Schwartz inequality, we have $(\sum_{e=1}^{\mathcal{E}} r_e)^2 \leq \mathcal{E}\sum_e r_e^2$. If $\sqrt{\mathcal{E}\sum_{e=1}^{\mathcal{E}} r_e^2} \leq \eta$, we have $\sum_{e=1}^{\mathcal{E}} r_e \leq \eta$ achieved. Therefore, to guarantee the learning consistency r_e and $\mathrm{tr}(\Sigma_x^{e-1})$ needs to be upper bounded in planning. The Info-SNOC approach automatically outputs trajectories of finite variance. ■

V. Simulation and Discussion

Problem Setup: We test the framework in Fig. 1 on the three degree-of-freedom robotic spacecraft simulator [21] dynamics with an unknown friction model and an over-actuated thruster configuration that is used for a real spacecraft. The dynamics of $\vec{x} = (x, y, \theta)^T$ with respect to an inertial frame is given as

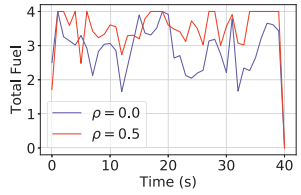
$$\ddot{\vec{x}} = f(\theta, H, u) + g(\dot{x}, \dot{y}, \dot{\theta}), \quad f = \begin{bmatrix} R(\theta) & 0 \\ 0 & 1 \end{bmatrix} Hu, \quad (38)$$

where $R(\theta) \in SO(2)$. The states $(x, y) \in \mathbb{R}^2$ denote position, and $\theta \in [0, 2\pi)$ orientation. The function g is unknown, and assumed to be linear viscous damping in the simulations. The control effort $u \in \mathbb{R}^8$ is constrained to be $0 \leq u \leq 1$, and $H(m, I, l, b) \in \mathbb{R}^{3\times 8}$ is the control allocation matrix where $m = 17$ kg and $I = 2$ kgm^2 are the mass and the inertia matrix, and $l = b = 0.4$ m is the moment arm. The unknown function $g = \mathrm{diag}([-0.02, -0.02, -0.002])\dot{x}$ is modeled as a multivariate Gaussian distribution to learn from data using robust regression. Note that even though the numerical values of the damping function are small, due to small data and propagation of dynamics, the terminal variance of the learned dynamical model is large. To get an initial estimate of the model, we explore a small safe set around the initial condition till the optimal control problem (18) becomes feasible.

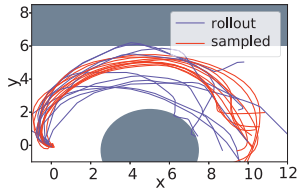


(a) Performance trajectory vs. Information trajectory.

(b) Trace of terminal variance w.r.t time.



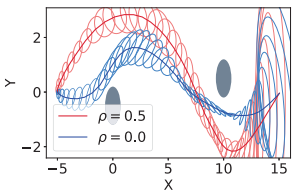
(c) Total fuel used at each time step.



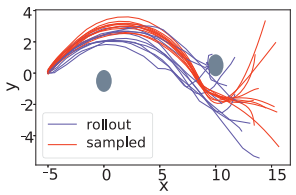
(d) 20 trials of rollout using a safety augmented stable controller.

Fig. 2. Info-SNOC trajectory pool and rollout for Scenario 1.

Info-SNOC Results: The learned dynamics is used to design safe trajectories for fixed time of 40 s using the Info-SNOC algorithm for Scenarios 1 and 2 as shown in Figs. 2a and 3a, respectively. The algorithm is initialized using a feasible solution for nominal dynamics. The obstacles in both scenarios are transformed to linear chance constraints as discussed in [7], with a risk measure of collision $\epsilon_\ell = 0.05$. The Info-SNOC algorithm is applied with $\rho = 0$ with ℓ_1 norm control cost and $\rho = 0.5$ with UCB cost. We compare the mean of the trajectories along with 2σ-confidence ellipse around the mean and observe that for the $\rho = 0.5$ case, the safe trajectory explores more state-space compared to the $\rho = 0$ case, which corresponds to the performance trajectory. In Scenario 2, there are multiple possible paths. The solution is biased towards the nominal trajectory used to initialize the Info-SNOC algorithm. We choose to use this trajectory to demonstrate the possible collision during rollout as discussed later. The total control effort at each time is shown in Fig. 2c. It shows that information trajectory uses more energy compared to the performance trajectory. We also observe that the terminal varianceis large in both scenarios. This is due to the correlation among the multiple dimensions in dynamics that is predicted by the learning algorithm.



(a) Performance trajectory vs. Information trajectory.



(b) 20 trials of rollout using a safety augmented stable controller.

Fig. 3. Info-SNOC trajectory pool and rollout for Scenario 2.

Safe Rollout: The safe trajectories in Figs. 2a and 3a are sampled for exploration following the method discussed in Sec. III-A, using the controller designed in [21] that satisfies Lemma 3. The sample trajectories and rollout trajectories are shown in red and blue respectively in Figs. 2d and 3b. The sampled trajectory is safe with the risk measure of collision $\epsilon_\ell = 0.05$ around the obstacles. The rollout trajectories collide with the obstacles due to the following two reasons: 1) the learning bounds (25) lead to bounded tracking performance and thereby constraint violation as discussed in Theorems 1 and 2, and 2) the state-dependent uncertainty model of the learned dynamics might predict large variance that can saturate the actuators. Saturated actuators cannot compensate for the unmodelled dynamics. In order to ensure safety, we augment the feedback controller with a real-time safety augmentation using barrier function based quadratic program [8]. Using this filter, the blue rollout trajectories are diverted from obstacles, as seen in Figs. 2d and 3b, avoiding constraint violation. Theorems 1 and 2 state that as the learning bounds c_1 and c_2 decrease, the safety of the rollout trajectory increases. Validating the Theorems quantitatively, showing safety in rollout with decreasing c_1 and c_2 is difficult since the desired trajectories for rollout are sampled and c_1 depends on the choice of function class complexity.

Consistency: The data collected during rollout is appended to the earlier data to learn a new model. Figure 4 shows improvement in performance (control cost) and terminal variance

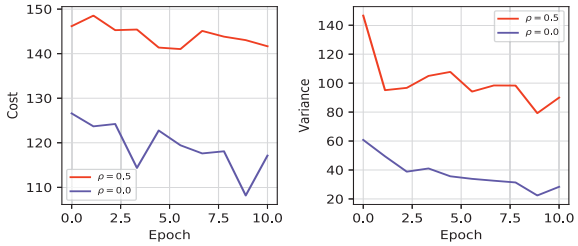


Fig. 4. Learning and performance improvement with epochs for Scenario 1.

reduction with increasing number of epochs. The variance of the performance trajectory is small compared to the information trajectory. The large variance leads to more exploration in the state space to collect informative data. The terminal variance of the information trajectory computed using Info-SNOC decreases from 150 to 84 by applying the framework in Fig. 1 for 10 epochs. The main assumption for consistency, which states that the variance of the trajectory computed using (18) is bounded and decreasing, is satisfied, thereby demonstrating the correctness of Theorem 3.

Note that the Info-SNOC algorithm can be applied to learn any nonlinear dynamical model that satisfies the mentioned regularity assumptions. Although we use the robust regression method, extension to other learning methods is straightforward as the planning algorithm is agnostic to the learning method. The algorithm can be initialized by planning for the nominal model or by using sampling-based methods such as rapidly exploring random trees. The algorithm also naturally extends to the case when learning error is zero, as the problem reduces to a deterministic optimal control problem.

VI. Conclusion

We presented the new Info-SNOC algorithm of trajectory optimization and safe exploration by solving information-cost stochastic optimal control using a partially learned nonlinear dynamical model. The Gaussian upper-confidence bound on the learned model is used as the information cost, while the safety is formulated as distributionally robust chance constraints. The problem is then projected to the generalized polynomial chaos space and solved using sequential convex programming. We used the Info-SNOC method to compute a safe and informative pool of trajectories for rollout using an exponentially stable controller with a safety filter augmentation for safe data collection. We analyzed the constraint violation during rollout and present the probability of violation for both linear and quadratic constraints. We showed that the safety constraints are satisfied for the dynamics with unknown model, as the learned model converges to the unknown model. The consistency of the learning method using the Info-SNOC algorithm was proven under mild assumptions. The episodic framework was applied to the robotic spacecraft model to learn the friction under collision constraints and achieve safe exploration. We designed a pool of safe trajectories using the Info-SNOC algorithm for a learned spacecraft model under collision constraints and discuss an approach for rollout using a stable feedback control law to collect data. In order to ensure real-time safety during rollout, we demonstrated how to augment the feedback control law with a barrier function based safety filter. We also validated the consistency of robust regression method by showing reduction in variance of the estimates over 10 epochs.

Acknowledgement

This work was in part funded by the Jet Propulsion Laboratory, California Institute of Technology and the Raytheon Company. We would like to thank Irene S. Crowell for her contribution to the implementation of the projection method.

References

[1] G. Shi, X. Shi, M. OConnell, R. Yu, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural lander: Stable drone landing control using learned dynamics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 9784–9790.

[2] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "On the sample complexity of the linear quadratic regulator," *Found. of Comput. Math.*, pp. 1–47, 2017.

[3] C. J. Ostafew, A. P. Schoellig, T. D. Barfoot, and J. Collier, "Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking," *J. of Field Robot.*, vol. 33, no. 1, pp. 133–152, 2016.

[4] A. Punjani and P. Abbeel, "Deep learning helicopter dynamics models," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 3223–3230.

[5] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 6059–6066.

[6] X. Chen, M. Monfort, A. Liu, and B. D. Ziebart, "Robust covariate shift regression," in *Artif. Intell. and Stat.*, 2016, pp. 1270–1279.

[7] Y. K. Nakka and S.-J. Chung, "Trajectory optimization for chance-constrained nonlinear stochastic systems," in *Proc. IEEE Conf. Decis. Control*, 2019.

[8] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Euro. Control Conf.*, 2019, pp. 3420–3431.

[9] J. Garcia and F. Fernández, "Safe exploration of state and action spaces in reinforcement learning," *J. of Artif. Intell. Res.*, vol. 45, pp. 515–564, 2012.

[10] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *Automatica*, vol. 49, no. 5, pp. 1216–1226, 2013.

[11] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *Int. J. Robotics Res.*, vol. 35, no. 13, pp. 1547–1563, 2016.

[12] L. Hewing, A. Liniger, and M. N. Zeilinger, "Cautious NMPC with Gaussian process dynamics for autonomous miniature race cars," in *Euro. Control Conf.*, 2018, pp. 1341–1348.

[13] A. A. Feldbaum, "Dual control theory," *Automation and Remote Control*, vol. 21, no. 9, pp. 874–1039, 1960.

[14] A. Mesbah, "Stochastic model predictive control with active uncertainty learning: a survey on dual control," *Ann. Rev. in Cont.*, vol. 45, pp. 107–117, 2018.

[15] Y. Cheng, S. Haghighat, and S. Di Cairano, "Robust dual control MPC with application to soft-landing control," in *Proc. IEEE American Control Conf.*, 2015, pp. 3862–3867.

[16] Q. T. Dinh and M. Diehl, "Local convergence of sequential convex programming for nonconvex optimization," in *Recent Adv. in Opt. and its App. in Engr.* Springer, 2010, pp. 93–102.

[17] D. Morgan, S.-J. Chung, and F. Y. Hadaegh, "Model predictive control of swarms of spacecraft using sequential convex programming," *J. of Guid., Control, and Dyn.*, vol. 37, no. 6, pp. 1725–1740, 2014.

[18] D. Morgan, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *Int. J. Robotics Res.*, vol. 35, no. 10, pp. 1261–1285, 2016.

[19] M. Buisson-Fenet, F. Solowjow, and S. Trimpe, "Actively learning Gaussian process dynamics," *arXiv:1911.09946*, 2019.

[20] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: no regret and experimental design," in *Proc. of Int. Conf. on Mach. Learning*, 2010, pp. 1015–1022.

[21] Y. K. Nakka, R. C. Foust, E. S. Lupu, D. B. Elliott, I. S. Crowell, S.-J. Chung, and F. Y. Hadaegh, "Six degree-of-freedom spacecraft dynamics simulator for formation control research," in *AAS/AIAA Astrodynamics Specialist Conf.*, 2018, pp. 3367–3387.

[22] A. Liu, G. Shi, S.-J. Chung, A. Anandkumar, and Y. Yue, "Robust regression for safe exploration in control," *arXiv:1906.05819*, 2019.

[23] M. Ono and B. C. Williams, "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint," in *Proc. IEEE Conf. Decis. Control*, 2008, pp. 3427–3432.

[24] G. C. Calafiore and L. El Ghaoui, "On distributionally robust chance-constrained linear programs," *J. of Opt. Theory and App.*, vol. 130, no. 1, pp. 1–22, 2006.

[25] A. P. Dani, S.-J. Chung, and S. Hutchinson, "Observer design for stochastic nonlinear systems via contraction-based incremental stability," *IEEE Trans. on Autom. Control*, vol. 60, no. 3, pp. 700–714, 2014.

[26] I. R. Savage, "Mills ratio for multivariate normal distributions," *J. Res. Nat. Bur. Standards Sect. B*, vol. 66, pp. 93–96, 1962.