

# 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 data collection with abundant supervision for training. Safe exploration algorithms ensure the safety of this data collection process even when only partial knowledge is available. We present a new approach for optimal motion planning with safe exploration that integrates chance-constrained stochastic optimal control with dynamics learning and feedback control. We derive an iterative convex optimization algorithm that solves an **Information-cost Stochastic Nonlinear Optimal Control problem (Info-SNOC)**. The optimization objective encodes control cost for performance and exploration cost for learning, and the safety is incorporated as distributionally robust chance constraints. The dynamics are predicted from a robust regression model that is learned from data. The Info-SNOC algorithm is used to compute a sub-optimal pool of safe motion plans that aid in exploration for learning unknown residual dynamics under safety constraints. A stable feedback controller is used to execute the motion plan and collect data for model learning. We prove the safety of rollout from our exploration method and reduction in uncertainty over epochs, thereby guaranteeing the 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. We demonstrate that our approach has higher success rate in ensuring safety when compared to a deterministic trajectory optimization approach.

**Index Terms**—Motion and Path Planning, Model Learning for Control, Machine Learning for Robot Control, Chance Constraints

## I. INTRODUCTION

Robots deployed in the real world often operate in unknown or partially-known environments. Modeling the complex dynamic interactions with the environment requires high-fidelity techniques that are often computationally expensive. 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 that are 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.

We present an episodic learning and control algorithm for safe exploration, as shown in Fig. 1, that integrates learning, stochastic trajectory planning, and rollout for active and safe data collection. *Rollout* is defined as executing the computed

Manuscript received: May, 10, 2020; Revised: August, 11, 2020; Accepted: October, 1, 2020.

This paper was recommended for publication by Editor Nancy Amato upon evaluation of the Associate Editor and Reviewers' comments.

The authors are with California Institute of Technology. Email: {ynakka, anqiliu, gshi, anima, yyue, sjchung}@caltech.edu. This work was funded in part by the Jet Propulsion Laboratory, Caltech and the Raytheon Company. A. Liu is supported by a PIMCO Postdoctoral Fellowship. We acknowledge the contribution of Irene S. Crowell in implementing Info-SNOC.

Digital Object Identifier (DOI): see top of the page.

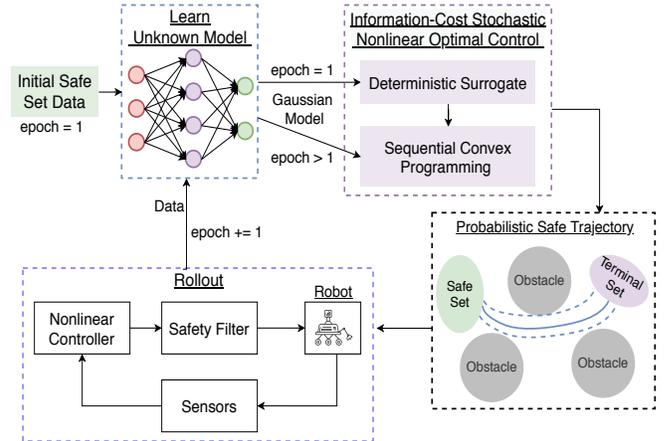


Fig. 1. An episodic framework for safe exploration using chance-constrained trajectory optimization. An initial estimate of the dynamics is computed using a known safe set [8] and control policy. A probabilistic safe trajectory and policy that satisfies safety chance-constraints is computed using Info-SNOC for the estimated dynamics. This policy is used for rollout with a stable feedback controller to collect data.

safe trajectory and policy using a stable feedback controller. The planning problem is formulated as an Information-cost Stochastic Nonlinear Optimal Control (Info-SNOC) problem that maximizes exploration and minimizes the control effort. Safety constraints are formulated as chance constraints.

The propagation of uncertainty in the dynamic model and chance constraints in Info-SNOC are addressed by projecting the problem to the generalized polynomial chaos (gPC) space and computing a distributionally robust [5], [6] convex approximation. By building on [6], we derive a sequential convex optimization solution to the Info-SNOC problem to plan a pool of sub-optimal safe and information-rich trajectories with the learned approximation of the dynamics. A sample of the trajectory pool is used as an input to the rollout stage to collect new data. To ensure real-time safety, the nonlinear feedback controller with a safety filter used in the rollout stage certifies bounded stochastic stability [7]. The new data is used to learn an improved dynamic model.

**Related Work:** Safe exploration for continuous dynamical systems has been studied using the following three frameworks: learning-based model-predictive control (MPC), dual-control, and active dynamics learning. Learning-based MPC [8]–[11] has been studied extensively for controlling the learned system. These deterministic techniques are also applied for planning an information trajectory<sup>1</sup> to learn online. The approach has limited analysis on safety of the motion

<sup>1</sup>An information trajectory is defined as a result of optimal motion planning that has more information about the unknown model compared to a fuel-optimal trajectory.

plans that use recursive feasibility argument and by appending a known safe control policy. In contrast, stochastic trajectory planning [6], [12]–[14] naturally extends to incorporate probabilistic learned dynamic model. The safety constraints formulated as joint chance constraints [12] facilitate a new approach to analyze the safety of the motion plans computed using Info-SNOC. The effect of uncertainty in the learned model on the propagation of dynamics is estimated using the method of generalized polynomial chaos (gPC) [6] expansion for propagation, which has asymptotic convergence to the original distribution, and provides guarantees on the constraint satisfaction.

Estimating unknown parameters while simultaneously optimizing for performance has been studied as a dual control problem [15]. 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 [13], [16], the convergence of the estimate is achieved by using the condition of persistency of excitation in the optimal control problem. Our method uses Sequential Convex Programming (SCP) [17]–[19] to compute the persistent excitation trajectory. Recent work [20] uses nonlinear programming tools to solve optimal control problems with an upper-confidence bound [21] 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 contributions of the paper are as follows: a) we propose a new safe exploration and motion planning method by directly incorporating safety as chance constraints and ensuring stochastic nonlinear stability in the closed-loop control along with a safety filter; b) we derive a new solution method to the Info-SNOC problem for safe and optimal motion planning, while ensuring the consistency and reduced uncertainty of our dynamics learning method; and c) we use a multivariate robust regression model [22] under a covariate shift assumption to compute the multi-dimensional uncertainty estimates of the unknown dynamics used in Info-SNOC.

*Organization:* We discuss the preliminaries on robust regression learning method, the Info-SNOC problem formulation, and results on deterministic approximations of chance constraints in Sec. II. The Info-SNOC algorithm along with the implementation of rollout policy is presented in Sec. III. In Sec. IV, we derive the end-to-end safety guarantees. In Sec. V, we discuss a result of applying the Info-SNOC algorithm to the nonlinear three degree-of-freedom spacecraft robot model [23]. We conclude the paper in Sec. VI.

## II. PRELIMINARIES AND PROBLEM FORMULATION

In this section, we discuss the preliminaries of the learning method used for modelling the unknown dynamics, the formulation of the Info-SNOC problem, and the deterministic surrogate projection using gPC.

### A. Robust Regression For Learning

An exploration step in active data collection for learning dynamics is regarded as a covariate shift problem. Covariate shift is a special case of distribution shift between training and testing data distributions. In particular, we aim to learn the unknown part of the dynamics  $g(x, \bar{u})$  from state  $x$  and control  $\bar{u}$  defined in Sec. II-B. The covariate shift assumption indicates the conditional dynamics distribution given the states and controls remains the same while the input distribution of training ( $\text{Pr}_s(x, \bar{u})$ ) is different from the target input distribution ( $\text{Pr}_t(x, \bar{u})$ ). Robust regression is derived from a *min-max* adversarial estimation framework, where the estimator minimizes a loss function and the adversary maximizes the loss under statistical constraints. The resulting Gaussian distributions provided by this learning framework are given below. For more technical details, we refer the readers to [22], [24]. The output Gaussian distribution takes the form  $\mathcal{N}(\mu_g, \Sigma_g)$  with mean  $\mu_g$  and variance  $\Sigma_g$ :

$$\begin{aligned} \Sigma_g(x, \bar{u}, \theta_2) &= \left( 2 \frac{\text{Pr}_s(x, \bar{u})}{\text{Pr}_t(x, \bar{u})} \theta_2 + \Sigma_0^{-1} \right)^{-1}, \\ \mu_g(x, \bar{u}, \theta_1) &= \Sigma_g(x, \bar{u}, \theta_2) \left( -2 \frac{\text{Pr}_s(x, \bar{u})}{\text{Pr}_t(x, \bar{u})} \theta_1 \phi(x, \bar{u}) + \mu_0 \Sigma_0^{-1} \right), \end{aligned} \quad (1)$$

where  $\mathcal{N}(\mu_0, \Sigma_0)$  is a non-informative (i.e., large  $\Sigma_0$ ) base distribution, and  $\phi(x, \bar{u})$  is the feature function that is learned using neural networks from data. The model parameters  $\theta_1$  and  $\theta_2$  are selected by maximizing the target log likelihood. The density ratio  $\frac{\text{Pr}_s(x, \bar{u})}{\text{Pr}_t(x, \bar{u})}$  is estimated from data beforehand. Robust regression can handle multivariate outputs with correlation efficiently by incorporating neural networks and predicting a multivariate Gaussian distribution, whereas traditional methods like Gaussian process regression suffer from high-dimensions and require heavy tuning of kernels [24].

### B. Optimal and Safe Planning Problem

In this section, we present the finite-time chance-constrained stochastic optimal control problem formulation [6] used to design an information-rich trajectory. The optimization has control effort and terminal cost as performance objectives, and the safety is modelled as a joint chance constraint of a set  $\mathcal{F}$  defined by a polytope or a quadratic constraint. The full stochastic optimal control problem is as follows:

$$J^* = \min_{x(t), \bar{u}(t)} \mathbb{E} \left[ \int_{t_0}^{t_f} J(x(t), \bar{u}(t)) dt + J_f(x(t), \bar{u}(t)) \right] \quad (2)$$

$$\text{s.t.} \quad \dot{x}(t) = f(x(t), \bar{u}(t)) + \hat{g}(x(t), \bar{u}(t)) \quad (3)$$

$$\text{Pr}(x(t) \in \mathcal{F}) \geq 1 - \epsilon, \quad \forall t \in [t_0, t_f] \quad (4)$$

$$\bar{u}(t) \in \mathcal{U} \quad \forall t \in [t_0, t_f] \quad (5)$$

$$x(t_0) = x_0 \quad \mathbb{E}(x(t_f)) = \mu_{x_f}, \quad (6)$$

where  $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$  denotes the state of the dynamics,  $x_0$  and  $x_f$  are the initial and the terminal states 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:

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

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

*Remark 1.* The maximum entropy distribution with the known mean  $\mu_g$  and covariance matrix  $\Sigma_g$  of the random variable  $\hat{g}$  is the Gaussian distribution  $\mathcal{N}(\mu_g, \Sigma_g)$ .

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 the mean  $\mu_x$  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

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

where  $\theta \sim \mathcal{N}(0, I)$  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  $\theta dt = dW$ .

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

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 *Lipschitz condition* and *Restriction on Growth condition* [25]. The approximate system (9) is assumed to be controllable in the given feasible space.

2) *Cost Functional*: The integrand cost functional  $J = J_C + J_{\mathcal{I}}$  includes two objectives: a) exploration cost  $J_{\mathcal{I}}$  for achieving the maximum value of information for learning the unknown dynamics  $g$ , and b) performance cost  $J_C$  (e.g., minimizing the control effort). The integrand cost functional  $J_C$  for fuel optimality, which is convex in  $\bar{u}$ , is given as

$$J_C = \|\bar{u}\|_s \quad \text{where } s \in \{1, 2, \infty\} \quad (10)$$

One example of  $J_{\mathcal{I}}$  is the following variance-based information cost using each  $i^{\text{th}}$  diagonal element  $\sigma_i^2$  in  $\Sigma_g$ .

$$J_{\mathcal{I}} = - \sum_{i=1}^n \sigma_i(\mu_x, \bar{u}) \quad (11)$$

The information cost  $J_{\mathcal{I}}$  in (11) is a functional of the mean  $\mu_x$  of the state  $x$  and control  $\bar{u}$  at time  $t$ . By minimizing the cost  $J_{\mathcal{I}}$ , we maximize the information [21] available in the trajectory  $x$  to learn the unknown model  $g$ . 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.

3) *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

$$\Pr(x \in \mathcal{F}) \geq 1 - \epsilon. \quad (12)$$

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

$$\Pr(a_i^\top x + b_i \leq 0) \geq 1 - \epsilon_i, \quad (13)$$

such that  $\sum_{i=1}^k \epsilon_i = \epsilon$ . Here we use  $\epsilon_i = \frac{\epsilon}{k}$ . The individual risk measure  $\epsilon_i$  can be optimally allocated between the  $k$  constraints [26]. A quadratic chance constraint is given as:

$$\Pr(x^\top A x \geq c) \leq \epsilon, \quad (14)$$

where  $A$  is a positive definite matrix. We use a tractable conservative approximation of sets in (13,14) by formulating distributionally robust chance constraint [5] for known mean  $\mu_x$  and variance  $\Sigma_x$  of the random variable  $x$  at each time  $t$ .

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

$$a^\top \mu_x + b + \sqrt{\frac{1 - \epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_x a} \leq 0. \quad (15)$$

*Proof:* See Theorem 3.1 in [5]. ■

**Lemma 2.** *The semi-definite constraint on the variance  $\Sigma_x$*

$$\text{tr}(A \Sigma_x) \leq \epsilon_q c \quad (16)$$

*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 Proposition 1 in [6]. ■

The risk measures  $\epsilon_\ell$  and  $\epsilon_q$  are assumed to be given. For the rest of the paper, we consider the following individual chance-constrained problem,

$$\begin{aligned} (x^*, \bar{u}^*) = \underset{x(t), \bar{u}(t)}{\text{argmin}} \quad & \mathbb{E} \left[ \int_{t_0}^{t_f} ((1 - \rho)J_C + \rho J_{\mathcal{I}}) dt + J_f \right], \\ \text{s.t.} \quad & \{(9), (15), (16), (5), (6)\} \end{aligned} \quad (17)$$

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

### C. Generalized Polynomial Chaos (gPC)

The problem (17) is projected into a finite-dimensional space using the generalized polynomial chaos algorithm presented in [6]. We briefly review the ideas and equations. We refer the readers to [6] 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  $\theta$  and the deterministic time varying coefficients  $X(t) = [x_{10} \ \cdots \ x_{1\ell} \ \cdots \ x_{n0} \ \cdots \ x_{n\ell}]^\top$ . The functions  $\phi_i$ ,  $i \in \{0, 1, \dots, \ell\}$  are chosen to be Gauss-Hermite polynomials in this paper. Let  $\Phi(\theta) = [\phi_0(\theta) \ \cdots \ \phi_\ell(\theta)]^\top$ .

$$x \approx \bar{\Phi} X; \quad \text{where } \bar{\Phi} = \mathbb{I}_{n \times n} \otimes \Phi(\theta)^\top \quad (18)$$

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

$$\dot{X} = \bar{f}(X, \bar{u}) + \bar{\mu}_g(\mu_x, \bar{u}) + \bar{B}(\mu_x, \bar{u}). \quad (19)$$

The exact form of the function  $\bar{f}$  is given in [6]. The moments of the random variable  $x$ ,  $\mu_x$  and  $\Sigma_x$  can be expressed as polynomial functions of elements of  $X$ . The polynomial functions defined in [6] are used to project the distributionally robust linear chance constraint in (15) to a second-order cone constraint in  $X$  as follows:

$$(a^\top \otimes S)X + b + \sqrt{\frac{1 - \epsilon_\ell}{\epsilon_\ell}} \sqrt{X^\top U N N^\top U^\top X} \leq 0, \quad (20)$$

where the matrices  $S$ ,  $U$ , and  $N$  are functions of the expectation of polynomials  $\Phi$ , that are given in [6]. The quadratic chance constraint in (16) is transformed to the following quadratic constraint in  $X$ :

$$\sum_{i=1}^n \sum_{k=1}^{\ell} a_i \langle \phi_k, \phi_k \rangle x_{ik}^2 \leq \epsilon_q c, \quad (21)$$

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

### III. INFO-SNOC MAIN ALGORITHM

We present the main algorithm for the architecture shown in Fig. 1 that integrates the learning method and the Info-SNOC method with rollout. We discuss an iterative solution method to Info-SNOC by projecting (17) to the gPC space. We formulate a deterministic optimal control problem in the gPC space and solve it using SCP [6], [18], [19] method. The gPC projection of (17) is given by the following equation:

$$\begin{aligned} (X^*, \bar{u}^*) &= \underset{X(t), \bar{u}(t)}{\operatorname{argmin}} \left[ \int_{t_0}^{t_f} ((1 - \rho)J_c + \rho J_{\mathcal{I}}) dt + \mathbb{E} J_{gPC_f} \right] \\ \text{s.t. } &\{(19), (20), (21), (5)\} \\ &X(t_0) = X_0, \quad X(t_f) = X_f. \end{aligned} \quad (22)$$

In SCP, the projected dynamic model (19) is linearized about a feasible nominal trajectory and discretized to formulate a linear equality constraint. Note that the constraints (20) and (21) are already convex in the states  $X$ . The terminal constraint  $J_f$  is projected to the quadratic constraint  $J_{gPC_f} = X_f^\top \bar{\Phi}^\top Q_f \bar{\Phi} X_f$ . The information cost functional  $J_{\mathcal{I}}$  from (11) is expressed as a function of  $X$  by using the polynomial representation [6] of  $\mu_x$  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  $J_{d\mathcal{I}}$ :

$$J_{d\mathcal{I}} = - \sum_{i=1}^n \left( \sigma_i(S^o) + \frac{\partial \sigma_i}{\partial S_i} \Big|_{S^o} (S - S^o) \right). \quad (23)$$

We use the convex approximation  $J_{d\mathcal{I}}$  as the information cost in the SCP formulation of the optimal control problem in (22). 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:* We use an initial estimate of the model (8) learned from data generated by a known safe control policy, and a *nominal trajectory*  $(x^o, \bar{u}^o)$  computed using deterministic SCP [18] with nominal model to initialize Algorithm 1. The stochastic model and the chance constraints are projected to the 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 (22) using SCP, in line 7 with  $\rho = 0$ , for a performance trajectory  $(x_p, \bar{u}_p)$ . The output  $(x_p, \bar{u}_p)$  of this optimization is used as initialization to the Info-SNOC problem obtained by setting  $\rho \in (0, 1]$  to compute the information trajectory  $(x_i, \bar{u}_i)$ . The trajectory  $(x_i, \bar{u}_i)$  is then sampled for a safe motion plan  $(\bar{p}_d, \bar{u}_d)$  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 Info-SNOC problem outputs a trajectory of random variable  $x$  with finite variance at each epoch.

---

#### Algorithm 1 Info-SNOC using SCP [19] and gPC [6]

---

- 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 [6]
  - 4: epoch = 1
  - 5: **while** Learning Criteria Not Satisfied **do**
  - 6:   Learn  $g \sim \mathcal{N}(\mu_g, \Sigma_g)$  using Robust Regression
  - 7:    $(x_p, \bar{u}_p) = \text{SCP}((x^o, \bar{u}^o), \rho = 0)$ , using (22)
  - 8:    $(x_i, \bar{u}_i) = \text{SCP}((x_p, \bar{u}_p), \rho \in (0, 1])$ , using (22)
  - 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**
- 

*Convergence and Optimality:* The information trajectory  $(x_i, \bar{u}_i)$  computed using SCP with the approximate linear information cost  $J_{d\mathcal{I}}$  (23) is a sub-optimal solution of (22) with the optimal cost value  $J_{d\mathcal{I}}^*$ . Therefore, the optimal cost of (22) 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 [17], [19] 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  $J = J_c + J_{\mathcal{I}}$ .

*Feasibility:* The initial phases of learning might lead to a large covariance  $\Sigma_g$  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.

### 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  $\theta$  and transforming it using the gPC expansion  $x(\theta) \approx \bar{\Phi}(\theta)X$ . For any realization  $\bar{\theta}$  of  $\theta$ , we get a deterministic trajectory  $\bar{p}_d = \bar{\Phi}(\bar{\theta})X$  that is  $\epsilon$  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  $u_c = u_c(\bar{p}, \bar{p}_d, \bar{u}_d)$  for rollout, where  $\bar{p}$  is the current state. To ensure real-time safety during the initial stages of exploration, a safe control policy  $u_s$  is computed using the control barrier function-based safety filter. 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) under what assumptions is the multivariate robust regression a consistent learning method as epoch  $\rightarrow \infty$ . The analysis proves that if the Info-SNOC algorithm computes a motion plan with finite variance, then learning is consistent and implies safety during rollout.

*Assumption 1.* The projected problem (22) computes a feasible trajectory to the original problem (17). The assumption is generally true if we choose a sufficient number of polynomials [6], for the projection operation.

*Assumption 2.* The probabilistic inequality  $\Pr(\|g(\bar{p}, \bar{u}) - \mu_g(\bar{p}, \bar{u})\|_2^2 \leq c_1) \geq 1 - \epsilon_{\ell b}$  holds, where  $\epsilon_{\ell b}$  is small, for the same input  $(\bar{p}, \bar{u})$  to the original model  $g$ , and the mean  $\mu_g$  of the learned model. Using this inequality, we can say that the following bounds hold with high probability:

$$\|g(\bar{p}, \bar{u}) - \mu_g(\bar{p}, \bar{u})\|_2^2 \leq c_1, \text{tr}(\Sigma_g) \leq c_2, \quad (24)$$

where  $c_2 = c_1 \epsilon_{\ell b}$ . As shown in [24], 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  $\Sigma_g$  is bounded by design. With Assumptions 1 and 2, the analysis is decomposed into the following three subsections.

### A. State Error Bounds During Rollout

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

*Assumption 3.* 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,

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

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

*Assumption 5.* The probability density ratio  $\frac{\rho_{x_i(t)}}{\rho_{x_i(0)}} \leq r$  is bounded, where the functions  $\rho_{x_i(0)}$  and  $\rho_{x_i(t)}$  are the probability density functions of  $x_i$  at time  $t = 0$  and  $t$  respectively.

**Lemma 3.** *Given that the estimated model (8) satisfies the Assumption 2, and the systems (7) and (9) satisfy Assumptions 3, 4, 5, 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  $t$*

$$\mathbb{E}_{x_i(t)}(\|\bar{p} - x_i\|_2^2) \leq \frac{\lambda_{\max}(M)}{2\lambda_{\min}(M)\alpha_m} (c_1 + c_2 + c_3)r \quad (25) \\ + \frac{\lambda_{\max}(M)r}{\lambda_{\min}(M)} \mathbb{E}(\|\bar{p}(0) - x_i(0)\|_2^2) e^{-2\alpha_m t},$$

where  $(x_i, \bar{u}_i)$  is computed from (22) 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 Lemma 2 in [7]. ■

Lemma 3 states that the expected mean squared error  $\mathbb{E}(\|\bar{p} - x_i\|_2^2)$  is bounded by  $\frac{\lambda_{\max}(M)(c_1+c_2+c_3)r}{2\alpha_m\lambda_{\min}(M)}$  as  $t \rightarrow \infty$  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 (17), 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  $t$ :*

$$(\bar{p} - \mu_x)^\top A(\bar{p} - \mu_x) \leq \lambda_{\max}(A)\mathbb{E}_x(\|\bar{p} - x\|_2^2), \quad (26)$$

where  $\mathbb{E}(\|\bar{p} - x\|_2^2)$  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

$$\mathbb{E}((\bar{p} - \mu_x)^\top A(\bar{p} - \mu_x)) = \mathbb{E}((\bar{p} - x)^\top A(\bar{p} - x)) \quad (27) \\ + \mathbb{E}((x - \mu_x)^\top A(x - \mu_x)) + 2\mathbb{E}((\bar{p} - x)^\top A(x - \mu_x)).$$

Using the following equality:

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

and  $\mathbb{E}((x - \mu_x)^\top A(x - \mu_x)) \geq 0$  in (27), we obtain the constraint bound in (26). Using the feedback tracking bound (25)

in (26), 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.,  $c_1 \rightarrow 0$ ,  $c_2 \rightarrow 0$ , and  $c_3 \rightarrow 0$ , then  $\bar{p} \rightarrow \mu_x$ . The quadratic constraint violation in (26) 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 (17) 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  $t$ :*

$$\inf_{x \sim (\mu_x, \Sigma_x)} \Pr(a^\top \bar{p} + b \leq \delta_\ell(x)) \geq 1 - \epsilon_\ell, \quad (28)$$

where  $\delta_\ell(x) = \|a\|_2 \mathbb{E}_x(\|(\bar{p} - x)\|_2) - \|a\|_2 \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{c_4}$ ,  $\mathbb{E}_x(\|(\bar{p} - x)\|_2)$  is bounded as defined in (25) and  $\text{tr}(\Sigma_x) = c_4$ .

*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  $\epsilon_\ell$ , mean  $\mu_x$ , and covariance  $\Sigma_x$ . Consider the similar condition for the actual trajectory  $\bar{p}(t)$ ,  $\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}})$ :

$$\begin{aligned} \mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) &= a^\top \mu_{\bar{p}} + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_{\bar{p}} a} \\ &= a^\top \mu_x + a^\top (\mu_{\bar{p}} - \mu_x) + b + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_{\bar{p}} a} \\ &\quad + \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_x a} - \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_x a} \end{aligned} \quad (29)$$

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:

$$\mathcal{P}(\mu_{\bar{p}}, \Sigma_{\bar{p}}) \leq a^\top (\bar{p} - \mu_x) - \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{a^\top \Sigma_x a}. \quad (30)$$

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  $\ell_2$ -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  $\text{tr}(\Sigma_x) = c_4$  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 (28). The bound  $\delta_\ell(x) = \|a\|_2 \mathbb{E}(\|(\bar{p} - x)\|_2) - \|a\|_2 \sqrt{\frac{1-\epsilon_\ell}{\epsilon_\ell}} \sqrt{c_4}$  is a function of the learning bounds in (24) by substituting the feedback tracking bound in (25). ■

The linear constraint is offset by  $\delta_\ell$  leading to constraint violation of the original formulation (13). Note that, if  $c_1 \rightarrow 0$ ,  $c_2 \rightarrow 0$ ,  $c_3 \rightarrow 0$ , and  $c_4 \rightarrow 0$ , then  $\delta_\ell \rightarrow 0$ . 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 nonlinear system to learn a new model for the next epoch. For epoch  $e$ , the predictor  $\hat{g}^e$  follows a multivariate Gaussian distribution  $\mathcal{N}(\mu_g^e, \Sigma_g^e)$  and  $g$  is the empirical true data. We assume that set  $\mathcal{X}_e \subset \mathcal{X}$  generated by the optimization problem in (17) for the first  $e$  iterations is a discretization of  $\mathcal{X}$ . Assuming that there exists a global optimal predictor  $\hat{g}^*$  in the function class  $\mathcal{G}$  that can achieve the best error  $\epsilon^*$  at each epoch  $e$ :

$$\max_{x \in \mathcal{X}_e} \|g - \hat{g}^*\|_2 = \epsilon^*, \quad (31)$$

the consistency of the learning algorithm is given by the convergence of the regret  $r_e$ , which is defined as the Euclidean distance between the predictor  $\hat{g}^e$  and the optimal predictor  $\hat{g}^*$ :

$$r_e = \|\hat{g}^e - \hat{g}^*\|_2 \rightarrow 0, \text{ as } e \rightarrow \infty. \quad (32)$$

We prove the consistency of Algorithm 1 in Theorem 3.

**Theorem 3.** *If Assumption 2 holds with the maximum prediction error  $\max_{x \in \mathcal{X}_e} \|g - \mu_g^e\|_2 \triangleq c_1^e$  at epoch  $e$ , then the regret  $r_e = \|\hat{g}^e - \hat{g}^*\|_2$  of Algorithm 1 achieves the bound:*

$$r_e^2 \leq c_1^e + dC^2 + \epsilon^{*2} \text{ with probability } 1 - |\mathcal{X}_e| \delta_e, \quad (33)$$

$$\text{where } \delta_e \triangleq (C^d \prod_p \Delta_p)^{-1} \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_g^e|^{\frac{1}{2}}} e^{-\frac{1}{2} C^2 \Sigma_{p=1}^d \Delta_p} \quad (34)$$

with the output dimension  $d$ ,  $\Delta_p \triangleq \sum_{q=1}^d m_{qp} > 0$ ,  $\forall p \in \{1, 2, \dots, d\}$ , and  $\mathcal{M} = (\Sigma_g^e)^{-1}$  with its  $(q, p)$ -th element  $(m_{qp})$ . Furthermore, the regret  $r_e \rightarrow 0$  as  $e \rightarrow \infty$ .

*Proof:* Using the inequality  $\Pr(\hat{g}^e - \mu_g^e \geq Ce) < \delta_e$ ,  $\forall x \in \mathcal{X}_e$ , since  $\hat{g}^e \sim \mathcal{N}(\mu_g^e, \Sigma_g^e)$ , we can bound  $r_e$  as follows:

$$r_e^2 \leq \|\mu_g^e + Ce - \hat{g}^*\|_2^2, \quad (35)$$

where  $\delta_e$  is defined in (34), and  $e$  is unit vector in  $d$  dimensions. Hence,  $C$  is a function of  $|\mathcal{X}_e| \delta_e$ . This is the tail probabilities inequality [27] of multivariate Gaussian distributions. The error  $r_e^2$  is then bounded using the empirical prediction error  $\|\mu_g^e - g\|_2^2$  and the best prediction error (31) as  $r_e^2 \leq \|\mu_g^e + Ce - g\|_2^2 + \|g - \hat{g}^*\|_2^2$ . By using the triangular inequality, we get the bound in (33) as follows:

$$r_e^2 \leq \|\mu_g^e - g\|_2^2 + dC^2 + \epsilon^{*2} \leq c_1^e + dC^2 + \epsilon^{*2}. \quad (36)$$

At any epoch  $e$ , we have  $x \in \mathcal{X}_e$ ,  $\|\mu_g^e\|_2^2 + \text{tr}(\Sigma_g^e) - \|g\|_2^2 \leq \omega_e$  [24], where  $\omega_e$  is a hyper-parameter that is associated with model selection in robust regression. Using  $\text{tr}(\Sigma_g^e) \leq c_2^e$ , (35) becomes  $r_e^2 \leq \omega_e + c_2^e + dC^2 + \epsilon^{*2}$ .

Assuming that the ground truth model is in the function class  $\mathcal{G}$ , we have  $\epsilon^* = 0$ . With infinite data, the hyper parameter  $\omega_e = 0$  and  $c_2^e \rightarrow 0$  at each data point. For achieving the same value of  $\delta_e$ ,  $C \rightarrow 0$  with shrinking  $\Sigma_g^e$ . This implies that for large data, i.e., as  $e \rightarrow \infty$ , we have  $r_e \rightarrow 0$ . The regret  $r_e$  is bounded as shown in (33). Therefore, Algorithm 1 is consistent with high probability. Note that the Info-SNOC approach outputs trajectories of finite variance at epoch  $e$ , given that  $\text{tr}(\Sigma_g^e)$  is bounded. ■

During the initial exploration phase, the finite dimensional approximation in the gPC projection might add to the error

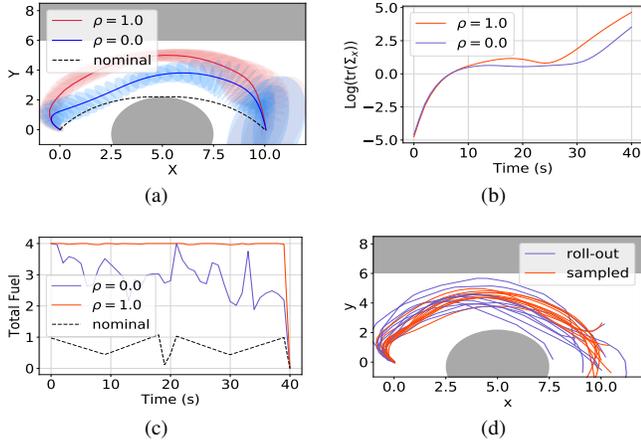


Fig. 2. Info-SNOC applied to Scenario 1. In Fig. (a), we show the motion plan along with the  $2\sigma$  confidence in position of the performance trajectory ( $\rho = 0$ ) and the information trajectory ( $\rho = 1$ ) computed using Info-SNOC, and the nominal trajectory computed using SCP under nominal dynamics. In Fig. (b), we show the trace of  $\Sigma_x$  w.r.t time. The information trajectory ( $\rho = 1$ ) has higher  $\Sigma_x$  compared to the performance trajectory ( $\rho = 0$ ). We compare total open-loop fuel computed at each time step in Fig. (c), and in Fig. (d) we demonstrate collision avoidance during exploration for 20 trials of rollout using a safety augmented stable controller.

bounds discussed in Theorems 1 and 2 leading to safety violation. This approximation error is automatically incorporated as residual dynamics in the stochastic dynamics when collecting data during rollout and learned using robust regression. Thus ensuring that safety is guaranteed with increasing data.

## V. SIMULATION AND DISCUSSION

We test the Info-SNOC framework shown in Fig. 1 on the three degree-of-freedom robotic spacecraft simulator [23] 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:

$$\ddot{\vec{x}} = f(\theta, H, u) + g(\dot{x}, \dot{y}, \dot{\theta}), \quad f = \text{blkdiag}(R(\theta), 1)Hu,$$

where  $R(\theta) \in SO(2)$ . The states  $(x, y) \in \mathbb{R}^2$ ,  $\theta \in [0, 2\pi)$  denote position and orientation respectively. 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  $m = 17 \text{ kg}$  and  $I = 2 \text{ kg m s}^{-2}$  are the mass and the inertia matrix, and  $l = b = 0.4 \text{ m}$  is the moment arm. The unknown function  $g = \text{diag}([-0.02, -0.02, -0.002])\dot{x}$  is modeled as a multivariate Gaussian distribution to learn from data using robust regression. To get an initial estimate of the model, we explore a small safe set around the initial condition and collect data. For the following test cases, we collect 40 data points to have a feasible optimal control problem (17) in the planning stage as discussed in Sec. III. The algorithm is initialized with a *nominal trajectory* computed using deterministic SCP under nominal dynamics.

*Info-SNOC Results:* The learned dynamics is used to design safe trajectories for 40s using the Info-SNOC algorithm for Scenarios 1 and 2 as shown in Figs. 2a and 3a, respectively.

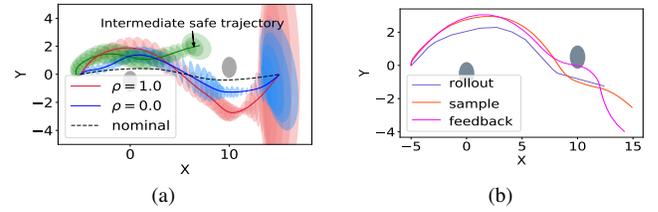


Fig. 3. Info-SNOC applied to Scenario 2. In Fig. (a), we show a comparison of the performance trajectory ( $\rho = 0$ ), the information trajectory ( $\rho = 1$ ), and an intermediate safe trajectory (green) computed using Info-SNOC and the nominal trajectory computed using deterministic SCP under nominal dynamics. In Fig. (b), we compare a sampled trajectory with the trajectories generated by feedback tracking and rollout with a safety filter.

Scenario 1 has a wall at  $y = 6 \text{ m}$  and a circular obstacle of radius  $2.5 \text{ m}$  at  $(5, -0.3)$ , and scenario 2 has two circular obstacles of radius  $0.6 \text{ m}$  at  $(5, -0.5)$  and  $(10, 0.5)$  as collision constraints respectively. The obstacles in both scenarios are transformed to linear chance constraints and the terminal constraint is transformed to quadratic chance constraint as discussed in [6], with a risk measure of  $\epsilon_\ell = \epsilon_q = 0.05$ . The Info-SNOC algorithm is applied with  $\rho = 0$  ( $\mathcal{L}_1$ -norm control cost) and  $\rho = 1$  (information cost). We compare the mean  $\mu_x$  and  $2\sigma$ -confidence ellipse around  $\mu_x$  of the trajectories with the nominal trajectory. We observe that for the  $\rho = 1$ , the safe trajectory explores larger state-space compared to the  $\rho = 0$  case, which corresponds to the fuel-optimal trajectory. 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. The extra fuel is used to explore the domain for improving the model. The terminal variance  $\Sigma_x(t_f)$  in both scenarios is large due to the correlation among the multiple dimensions of  $g$  that are predicted by the learning algorithm.

*Safe Rollout:* The safe trajectories in Figs. 2a and 3a are sampled following the method discussed in Sec. III-A and the rollout is performed using the controller designed in [23] 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. Rollout trajectories, with a feedback controller, collide with the obstacles due to the following two reasons: 1) the learning bounds (24) lead to constraint violation as discussed in Theorems 1 and 2, and 2) the state-dependent uncertainty model  $\mathcal{N}(\mu_g, \Sigma_g)$  might predict large  $\Sigma_g$  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 [28]. Using this filter, the blue rollout trajectories are diverted from obstacles, as seen in Figs. 2d and 3b, avoiding constraint violation.

*Consistency:* The data collected during rollout is appended to the earlier data to learn a new model. Figure 4 shows improvement in control performance (control cost for  $\rho = 0.0$ ) with increasing number of epochs. The prediction variance  $\Sigma_g$  decreases from  $1e-2$  to  $0.3e-2$  over 15 epochs. The main assumption for consistency, which states that the variance of the

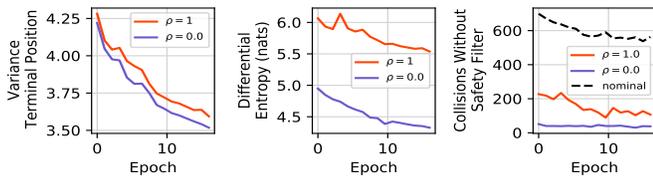


Fig. 4. Performance over epochs for Scenario 1. Left: we show decrease in the terminal position variance over epochs demonstrating improved goal reaching with epoch. Center: the differential entropy of the prediction variance  $\Sigma_g$  for information trajectory ( $\rho = 1$ ) is larger compared to the performance trajectory ( $\rho = 0$ ). Right: the number of collisions during rollout for 1000 trials decrease as the learning converges, validating Theorem 2.

trajectory computed using (17) is bounded and decreasing, is satisfied, thereby demonstrating the correctness of Theorem 3. We observe that the differential entropy of the information trajectory is higher (i.e., contains more information about  $g$ ) than the performance trajectory by design.

We observed that  $\Sigma_x(t_f)$  of the information trajectory ( $\rho = 1$ ) computed using Info-SNOC decreases from 104.11 to 61.46 over 15 epochs, by applying the framework in Fig. 1 for Scenario 1, demonstrating the increase in probability of reaching the quadratic terminal set. This validates the Theorem 1. The number of collisions of the rollout trajectory without a safety filter over a 1000 trials at each epoch is shown in Fig. 4. The number of collisions decrease from 227 to 107 for the information trajectory, from 51 to 38 for the performance trajectory, and from 699 to 563 for the nominal trajectory over 15 epochs. This increase in the probability of the linear chance constraint satisfaction with epoch validates the Theorem 2. In the first epoch, we observed that the rollout using the nominal trajectory leads to 69.9% collisions over 1000 trials, using the information trajectory leads to 22% collisions, and using the performance trajectory leads to less than 5.1% collisions, demonstrating the effectiveness of Info-SNOC.

## VI. CONCLUSION

We present a new method of learning-based safe exploration and motion planning by solving information-cost stochastic optimal control using a partially learned nonlinear dynamical model. The variance prediction of 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 use the Info-SNOC method to compute a safe and information-rich pool of trajectories for rollout using an exponentially stable controller with a safety filter augmentation for safe data collection. We analyze the probability of constraint violation for both linear and quadratic constraints. We show that the safety constraints are satisfied for rollout under learned dynamics, as the learned model converges to the optimal predictor over epochs. The consistency of the learning method using the Info-SNOC algorithm is proven under mild assumptions.

The episodic learning framework was applied to the robotic spacecraft model to explore the state space and learn the friction under collision constraints. We compute a pool of safe and optimal 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 for learning. We validate the consistency of robust regression method and the safety guarantees by showing the reduction in variance of the learned model predictions and the number of collisions over 15 epochs respectively.

## REFERENCES

- [1] 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 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. 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. 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] G. C. Calafiore and L. El Ghaoui, “On distributionally robust chance-constrained linear programs,” *J. Opt. Theory App.*, vol. 130, no. 1, pp. 1–22, 2006.
- [6] Y. K. Nakka and S.-J. Chung, “Trajectory optimization for chance-constrained nonlinear stochastic systems,” in *Proc. IEEE Conf. Decis. Control*, 2019.
- [7] A. P. Dani, S.-J. Chung, and S. Hutchinson, “Observer design for stochastic nonlinear systems via contraction-based incremental stability,” *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 700–714, 2014.
- [8] 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.
- [9] 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.
- [10] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, “Robust constrained learning-based NMPC enabling reliable mobile robot path tracking,” *Int. J. Robot. Res.*, vol. 35, no. 13, pp. 1547–1563, 2016.
- [11] 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.
- [12] L. Blackmore, M. Ono, and B. C. Williams, “Chance-constrained optimal path planning with obstacles,” *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1080–1094, 2011.
- [13] 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.
- [14] A. Wang, X. Huang, A. Jasour, and B. Williams, “Fast risk assessment for autonomous vehicles using learned models of agent futures,” in *Robotics: Sci. Systems*, 2020.
- [15] A. A. Feldbaum, “Dual control theory,” *Autom. Remote Control*, vol. 21, no. 9, pp. 874–1039, 1960.
- [16] Y. Cheng, S. Haghghat, and S. Di Cairano, “Robust dual control MPC with application to soft-landing control,” in *Proc. Amer. Control Conf.*, 2015, pp. 3862–3867.
- [17] Q. T. Dinh and M. Diehl, “Local convergence of sequential convex programming for nonconvex optimization,” in *Recent Adv. Opt. & its App. Engr.* Springer, 2010, pp. 93–102.
- [18] D. Morgan, S.-J. Chung, and F. Y. Hadaegh, “Model predictive control of swarms of spacecraft using sequential convex programming,” *J. Guid. Control Dyn.*, vol. 37, no. 6, pp. 1725–1740, 2014.
- [19] 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. Robot. Res.*, vol. 35, no. 10, pp. 1261–1285, 2016.
- [20] M. Buisson-Fenet, F. Solowjow, and S. Trimpe, “Actively learning Gaussian process dynamics,” *arXiv:1911.09946*, 2019.
- [21] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, “Gaussian process optimization in the bandit setting: no regret and experimental design,” in *Proc. Int. Conf. Mach. Learning*, 2010, pp. 1015–1022.
- [22] X. Chen, M. Monfort, A. Liu, and B. D. Ziebart, “Robust covariate shift regression,” in *Artif. Intell. Stat.*, 2016, pp. 1270–1279.

- [23] 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 Astrodyn. Specialist Conf.*, 2018, pp. 3367–3387.
- [24] A. Liu, G. Shi, S.-J. Chung, A. Anandkumar, and Y. Yue, "Robust regression for safe exploration in control," *arXiv:1906.05819*, 2019.
- [25] L. Arnold, *Stochastic Differential Equations*, 1974.
- [26] 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.
- [27] I. R. Savage, "Mills' ratio for multivariate normal distributions," *J. Res. Nat. Bur. Standards Sect. B*, vol. 66, pp. 93–96, 1962.
- [28] 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.