Real-Time Trajectory Generation Algorithms for Uncertain Dynamical Systems Using Covariance Steering

Efstathios Bakolas, Associate Professor

Department of Aerospace Engineering
& Engineering Mechanics

The University of Texas at Austin

Motivation: High-Precision Control of Uncertain Dynamical Systems

Image taken from defense.gov

mars.nasa.gov

Image taken from Raytheon.com

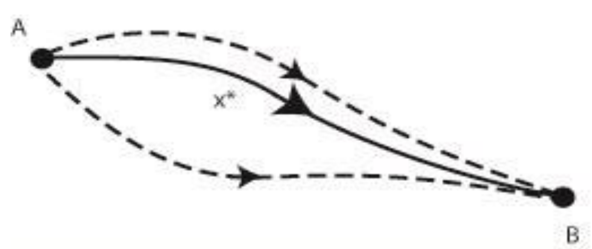
- Modern/future control systems are required to perform control tasks with high precision under extremely uncertain conditions
- **Main challenge**: Accurate models for system dynamics and / or system-environment interactions may not be available a priori
- **Goal:** Design control algorithms for high-precision control of uncertain systems based on the idea of directly controlling the effects of model uncertainty and unknown disturbances on the control system

Motivation: A New Perspective on Trajectory Generation / Optimization Problems

- **Deterministic Trajectory Generation/Optimization**: Steer a system "from point A to point B" while minimizing a relevant performance index without violating certain input and / or state constraints

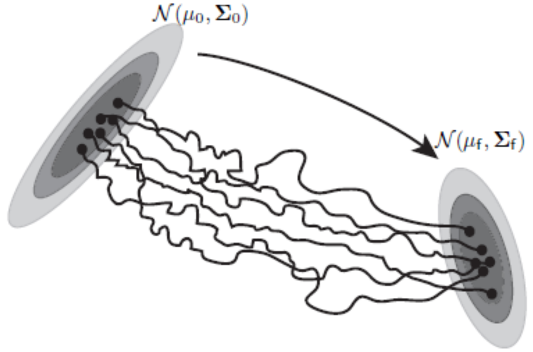
- **Trajectory Optimization for Uncertain Systems**:
  - system model is stochastic and may not be available a priori
  - boundary conditions should be *probabilistic* rather than deterministic

- **Distribution / Covariance Steering**: Special class of stochastic trajectory optimization problems in which the goal is to drive the state mean and state covariance to prescribed quantities



Standard (deterministic) trajectory generation problem

v/s



Covariance Steering Problem

PART I: Model-Based Nonlinear Covariance Steering

Covariance Steering for Discrete-Time Stochastic Nonlinear Systems

- Given the discrete-time stochastic nonlinear system:

$$x(t+1) = f(x(t), u(t)) + w(t), \qquad x(0) = x_0 \qquad (1)$$

- $\mathbb{E}[x_0] = \mu_0$, $\mathrm{cov}(x_0, x_0) = \boldsymbol{\Sigma}_0$, where $\mu_0 \in \mathbb{R}^n$, $\boldsymbol{\Sigma}_0 = {\boldsymbol{\Sigma}_0}' > 0$ (given)
- $\{x(t): t \in [0, N]_d\}$: state process, $\{u(t): t \in [0, N-1]_d\}$: input process,
- $\{w(t): t \in [0, N-1]_d\}$: sequence of i.i.d. (normal) random variables

$$\mathbb{E}\big(w(t)\big) = 0,\ \mathbb{E}(w(t)w(\tau)') = \delta(t,\tau)\mathbf{W},\ \text{where } \mathbf{W} = \mathbf{W}' \geq \mathbf{0}$$

- **Nonlinear Covariance Steering Problem:** Given $\mu_{\mathrm{d}} \in \mathbb{R}^n$, $\boldsymbol{\Sigma}_{\mathrm{d}} = \boldsymbol{\Sigma}_{\mathrm{d}}' > 0$, find a control policy

$$\boldsymbol{\varpi} := \{\varphi_0(x), \varphi_1(x), \dots, \varphi_{N-1}(x)\}$$

that will steer the system (1) to a state $x(N) = x_{\mathrm{f}}$ with

$$\mathbb{E}[x_{\mathrm{f}}] = \mu_{\mathrm{d}}, \qquad \mathrm{cov}(x_{\mathrm{f}}, x_{\mathrm{f}}) = \boldsymbol{\Sigma}_{\mathrm{d}} \qquad (2)$$

while minimizing a relevant performance index: $J(u) = \mathbb{E}[\sum_{t=0}^{N-1} |u(t)|^2]$

Successive Linearization of Dynamics

- At stage $t = k$ the nonlinear system is linearized around (μ_k, ν_k), where μ_k and ν_k are (approximations of) the state mean and input mean at $t = k$

$$z(t+1) = \mathbf{A}_k(z(t) - \mu_k) + \mathbf{B}_k(u(t) - \nu_k) + r_k + w(t),$$
$$\mathbf{A}_k = f_x(\mu_k, \nu_k), \quad \mathbf{B}_k = f_u(\mu_k, \nu_k), \quad r_k = f(\mu_k, \nu_k)$$

- Equivalently,

$$z(t+1) = \mathbf{A}_k z(t) + \mathbf{B}_k u(t) + d_k + w(t), \quad t \in [k, N-1]_d,$$

where $d_k = -\mathbf{A}_k \mu_k - \mathbf{B}_k \nu_k + r_k$

- **Boundary conditions for covariance steering:** $z(k) = z_k$ where $\mathbb{E}[z_k] = \mu_k$, $\mathrm{cov}(z_k, z_k) = \boldsymbol{\Sigma}_k$
- **Remark:** The k-th state space model of the successive linearization approach is computed based on information available at stage $t = k$

Alternative Linearization based on a Reference Trajectory

- Alternatively, linearize around reference state and input trajectories * $\{\bar{z}(t)\colon\ t \in [0, N]_d\}$ and $\{\bar{u}(t)\colon\ t \in [0, N-1]_d\}$, respectively:

$$z(t+1) = \mathbf{A}(t)\big(z(t) - \bar{z}(t)\big) + \mathbf{B}(t)(u(t) - \bar{u}(t)) + r_k + w(t),$$

$$\mathbf{A}(t) = f_x(\bar{z}(t), \bar{u}(t)),$$

$$\mathbf{B}(t) = f_u(\bar{z}(t), \bar{u}(t)),$$

$$r(t) = f(\bar{z}(t), \bar{u}(t)) - \mathbf{A}(t)\bar{z}(t) - \mathbf{B}(t)\bar{u}(t)$$

- The above linearization corresponds to a *single* time-varying state space model and is based on information available at $t = 0$ (computed off-line)
- By contrast, the successive linearization approach relies on the successive computation of time-invariant state space models (computed on-the-fly, i.e., a new time-invariant model at a time)

Ridderhof, J., Okamoto, K. and Tsiotras, P., 2019, December. Nonlinear uncertainty control with iterative covariance steering. In 2019 IEEE 58th Conference on Decision and Control (CDC) (pp. 3484-3490). IEEE

Linear Covariance Steering Problem

Problem: Find a control policy $\pi_k = \{\phi_k(t,z)\colon\ t \in [k, N-1]\}$, where

$$\phi_k(t,z) := v_k(t) + \mathbf{K}_k(t)z,$$

that minimizes the performance index:

$$J(\pi_k) := \mathbb{E}\left[\sum_{t=k}^{N-1} |\phi_k(t,z)|^2\right]$$

subject to the dynamic constraints:

$$z(t+1) = \mathbf{A}_k z(t) + \mathbf{B}_k u(t) + d_k + w(t), \quad t \in [k, N-1]_d$$

and the boundary constraints:

$$\mathbb{E}[z(k)] = \mu_k \qquad \mathrm{cov}(z(k), z(k)) = \mathbf{\Sigma}_\mathrm{k}$$

$$\mathbb{E}[z(N)] = \mu_\mathrm{d} \qquad \mathrm{cov}(z(N), z(N)) = \mathbf{\Sigma}_\mathrm{d}$$

for given $\mu_k, \mu_\mathrm{d} \in \mathbb{R}^n$ and $\mathbf{\Sigma}_k, \mathbf{\Sigma}_\mathrm{d} \in \mathbb{S}_n^{++}$ (positive definite matrices)

Remark 1: Subscript k indicates dependence on information available at stage $t = k$ (i.e., the policy π_k consists of control laws for $t \in [k, N-1]$ based on information available at stage $t = k$)

Solution to Linear Covariance Steering Problem (SDP Convex Optimization Formulation)

- The k-th linearized covariance steering can be reduced (relaxed) to a semi-definite program (SDP)*:

$$\boldsymbol{\Sigma}_{\mathrm{d}} - \mathrm{cov}\big(z(N), z(N)\big) \succcurlyeq \mathbf{0} \quad \text{(LMI constraint)}$$

- The solution to the latter problem will furnish the sequence of gains $\boldsymbol{\mathcal{K}_k} = \{\mathbf{K}_k(t): t \in [k, N-1]_d\}$ and $\boldsymbol{v_k} = \{v_k(t): t \in [k, N-1]_d\}$ that will solve the k-th linearized covariance steering problem

$$\pi_k = \pi_k(\boldsymbol{\mathcal{K}_k}, \boldsymbol{v_k}) = \{\phi_k(k, z), \dots, \phi_k(N-1, z)\},$$

  - $\phi_k(t, z) \coloneqq v_k(t) + \mathbf{K}_k(t) z$ (state feedback)
  - $\phi_k(t, Z) \coloneqq v_k(t) + \sum_{\tau=k}^{t} \mathbf{K}_k(\tau, t) z(\tau)$ where $Z = \{z(k), \dots, z(t)\}$

    (history-based state-feedback)

* Bakolas, E. "Finite-Horizon Covariance Control for Discrete-Time Stochastic Linear Systems Subject to Input Constraints," *Automatica*, vol. 91, no. 5, pp. 61-68, 2018.

Solution to the Linear Covariance Steering Problem (DCP Formulation)

- The computationally tractable SDP formulation of the covariance steering problem relies on the assumption that the latter problem admits a solution for the given (hard) terminal conditions on mean and covariance

- An alternative formulation is based on replacing the hard constraints:

$$\mathrm{cov}\big(z(N), z(N)\big) = \boldsymbol{\Sigma}_{\mathrm{d}} \qquad \mathbb{E}[z(N)] = \mu_{\mathrm{d}}$$

with a terminal cost term measuring the distance between $\mathcal{N}(\mu_{\mathrm{d}}, \boldsymbol{\Sigma}_{\mathrm{d}})$ and the Gaussian approximation $\mathcal{N}(\mu_N, \boldsymbol{\Sigma}_N)$ where $\mu_N = \mathbb{E}[z(N)]$, and $\boldsymbol{\Sigma}_N = \mathrm{cov}(z(N), z(N))$ of the terminal state distribution:

$$J(\pi_k) := \gamma \mathcal{W}^2(\rho_{\mathrm{d}}, \rho_N) + \mathbb{E}\left[\sum_{t=k}^{N-1} |\phi_k(t,z)|^2\right]$$

where $\gamma > 0$ and $\mathcal{W}(\rho_{\mathrm{d}}, \rho_N)$ is the Wasserstein distance between the desired and the actual terminal state distributions*

*Halder, Abhishek, and Eric DB Wendel. "Finite horizon linear quadratic Gaussian density regulator with Wasserstein terminal cost." In 2016 American Control Conference (ACC), pp. 7249-7254. IEEE, 2016.

Solution to k-th Linearized Covariance Steering Problem (Difference of Convex Functions Program Formulation)

- Let the actual terminal state and desired distributions be approximated by Gaussian distributions with densities $\rho_N, \rho_{\mathrm{d}}$, respectively. Then

$$\mathcal{W}^2(\rho_{\mathrm{d}}, \rho_N) = |\mu_{\mathrm{d}} - \mu_N|^2 + \mathrm{tr}\left(\Sigma_{\mathrm{d}} + \Sigma_N - 2\left(\Sigma_N^{1/2}\Sigma_{\mathrm{d}}\Sigma_N^{1/2}\right)^{1/2}\right)$$

- It can be shown that $J(\pi_k)$ can be expressed as a difference of two convex functions*
- The covariance steering problem with terminal cost turns out to be a *Difference of Convex functions Program* (DCP)
- DCPs are computationally tractable (one can solve them by using, for instance, the so-called convex / concave procedure)

* I. Balci and E. Bakolas, "Covariance Steering of Discrete-Time Stochastic Linear Systems Based on Wasserstein Distance Terminal Cost," in *IEEE Control Systems Letters,* doi: 10.1109/LCSYS.2020.3047132

Unscented Transform for One-Stage Prediction of State Mean and Covariance

- Given a feedback control policy π_k, the closed-loop dynamics of the original (nonlinear system) are described by the following equation:

$$x(t+1) = f_{cl}^{k}\big(t, x(t)\big) + w(t), \qquad t \in [k, N-1]_d$$

where $f_{cl}^{k}(t,x) = f\big(x, \phi_k(t,x)\big)$.

- Instead of propagating uncertainty using the linearized model of the nonlinear closed-loop dynamics, we will use the *unscented transform**
- The unscented transform predicts future mean and covariance by only propagating $2n+1$ points known as the *sigma points:*

$$\begin{aligned} \sigma_k^0 &= \mu_k, \\ \sigma_k^i &= \mu_k + \sqrt{(n+\lambda)}\boldsymbol{\Sigma}_k^{1/2} e_i, && \text{if } i \in [1, n]_d \\ \sigma_k^i &= \mu_k - \sqrt{(n+\lambda)}\boldsymbol{\Sigma}_k^{1/2} e_i, && \text{if } i \in [n+1, 2n]_d \end{aligned}$$

* Wan, Eric A., and Rudolph Van Der Merwe. "The unscented Kalman filter for nonlinear estimation." Proc. of the IEEE 2000 Adaptive Systems for Signal Proc., Comm. and Control Symposium, 2000.

Unscented transform for one-stage prediction of state mean and covariance

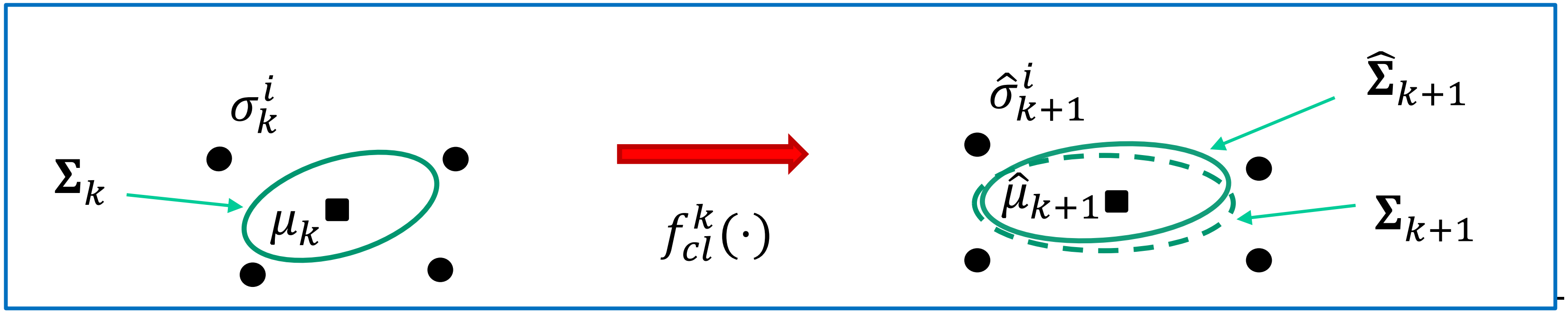
- Propagate the sigma points σ_k^i (stage $t = k$) to obtain a new set of sigma points $\hat{\sigma}_{k+1}^i$ (stage $t = k + 1$):

$$\hat{\sigma}_{k+1}^i = f_{cl}^k(\sigma_k^i), \qquad i \in [0,2n]_d$$

- Compute approximations of the state mean and covariance at stage $t = k + 1$ (one-stage predictions) by using the following equations:

$$\hat{\mu}_{k+1} = \sum_{i=0}^{2n} \gamma_k^i \, \hat{\sigma}_{k+1}^i$$

$$\widehat{\boldsymbol{\Sigma}}_{k+1} = \sum_{i=0}^{2n} \delta_k^i \left(\hat{\sigma}_{k+1}^i - \hat{\mu}_{k+1}\right)\left(\hat{\sigma}_{k+1}^i - \hat{\mu}_{k+1}\right)' + W_k$$



Greedy Covariance Steering Algorithm

The proposed greedy (nonlinear) covariance steering algorithm* consists of the following main steps (repeated until $t = N - 1$)

Step 1: Compute linearization $(\mathbf{A}_k, \mathbf{B}_k, r_k)$ at stage $t = k$ based on known approximations $\hat{\mu}_k$ and $\widehat{\mathbf{\Sigma}}_k$

Step 2: Compute control policy $\pi_k = \{\phi_k(k, z), \dots, \phi_k(N-1, z)\}$ that solves the k-th linearized covariance steering problem from $(\hat{\mu}_k, \widehat{\mathbf{\Sigma}}_k)$ at $t = k$ to $(\mu_{\mathrm{d}}, \mathbf{\Sigma}_{\mathrm{d}})$ at $t = N$

Step 3: Add $\phi_k(k, z)$ (first control law of local control policy π_k) to the "global" control policy ϖ

Step 4: Compute one-stage predictions $\hat{\mu}_{k+1}$ and $\widehat{\mathbf{\Sigma}}_{k+1}$ via unscented transform

Output: Global control policy for nonlinear covariance steering

$$\varpi = \{\varphi_0(x), \varphi_1(x), \dots, \varphi_{N-1}(x),\}, \qquad \varphi_k(x) = \phi_k(k, x)$$

*Bakolas, Efstathios, and Alexandros Tsolovikos. "Greedy finite-horizon covariance steering for discrete-time stochastic nonlinear systems based on the unscented transform," *American Control Conference (ACC)*, 2020.

Numerical Simulations (SDP Formulation)

- Consider a discrete-time nonlinear stochastic system
  $$x(t+1) = x_1(t) + \tau x_2(t)$$
  $$x_2(t+1) = x_2(t) - \tau(\delta x_1(t) + \zeta x_1(t)^3 + \gamma x_2(t)) + \tau u + \sqrt{\tau} w(t)$$
- Boundary conditions: $x_0 \sim N(\mu_0, \boldsymbol{\Sigma}_0)$, $\mu_0 = 0$, $\boldsymbol{\Sigma}_0 = \mathrm{diag}(0.4^2, 0.3^2)$, $x_\mathrm{d} \sim N(\mu_\mathrm{d}, \boldsymbol{\Sigma}_\mathrm{d})$, $\mu_d = 0$ and $\boldsymbol{\Sigma}_\mathrm{d} = \mathrm{diag}(0.8^2, 0.6^2)$
- Parameters for numerical simulations: $\zeta = \gamma = \alpha = 0.05$, $\delta = -1$, $\beta = 2$, sampling period: $\tau = 0.1$, system parameters: $N = 100$

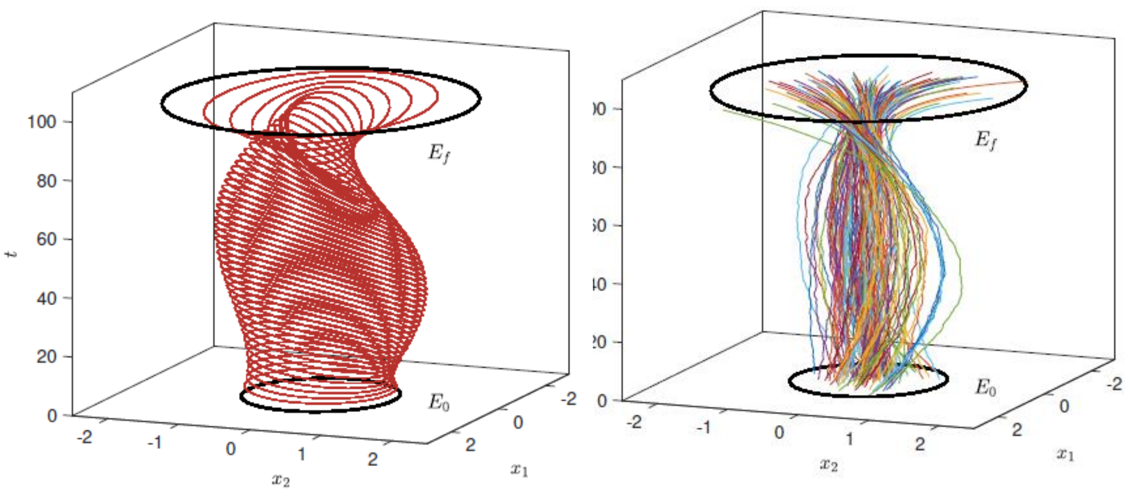


Figure 1. Three-dimensional illustration of the time-evolution of sample trajectories and state covariance of a nonlinear stochastic system

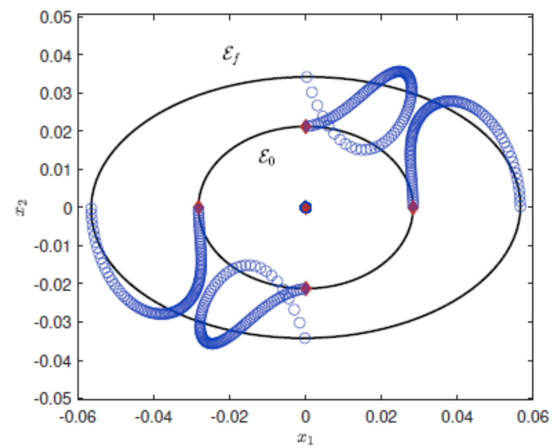


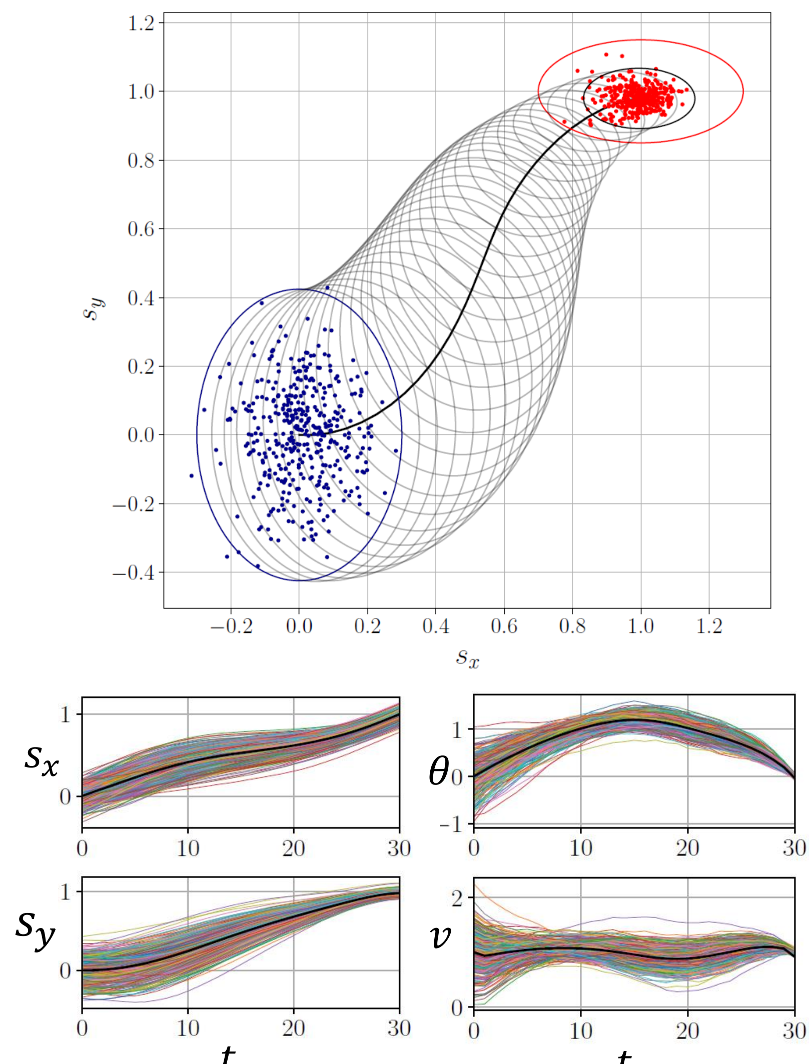
Figure 2. Time-evolution of the sigma points of the closed-loop system

Numerical Simulations (DCP Formulation)

- Consider the discrete-time nonlinear stochastic model (unicycle car):

$$
\begin{aligned}
s_x(t+1) &= s_x(t) + v(t)\tau\cos\theta(t) + \epsilon_x(t)\\
s_y(t+1) &= s_y(t) + v(t)\tau\sin\theta(t) + \epsilon_y(t)\\
\theta(t+1) &= \theta(t) + u_\theta(t)v(t)\tau + \epsilon_\theta(t)\\
v(t+1) &= v(t) + u_v(t)\tau + \epsilon_v(t)
\end{aligned}
$$

- (s_x, s_y): position vector, θ: heading angle, v: speed
- **Goal:** Shrink the uncertainty in the coordinate s_y, the heading angle θ, and the speed v, while retaining the uncertainty in s_x.
- Terminal covariance (red) is close to the desired one even though we only considered soft terminal constraints (terminal cost $\mathcal{W}^2(\rho_{\mathrm{d}}, \rho_N)$)



Model-free Covariance Steering Based on Variational Gaussian Process Regression

Covariance Steering Based on Data-Driven Predictive Models

- The previous tools for covariance steering assumed knowledge of a state space model of the uncertain system
- Such models may not be available a priori in many applications (e.g., the system dynamics may change during system operation)
- Proposed approach: Compute data-driven prediction models by learning the system dynamics from available data (experiment, simulations) and use these models for control design purposes
- In particular, use sparse variational Gaussian Process regression tools to capture the effects of model uncertainty and process noise (small computational / inference cost)

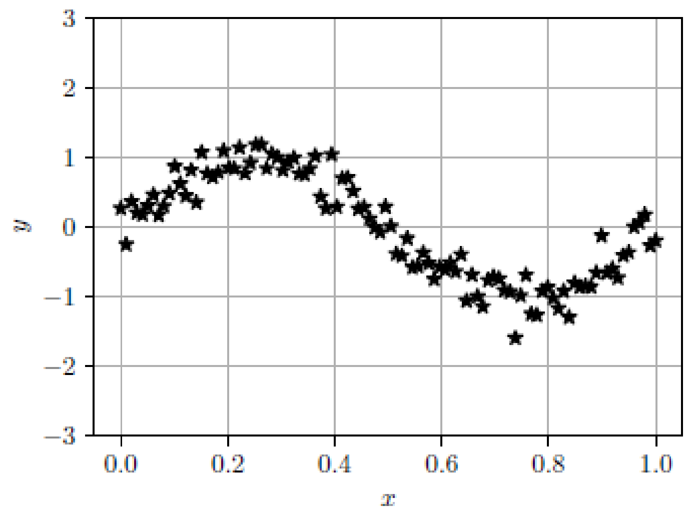
Brief Introduction to Gaussian Processes for (non-parametric) Modeling

- Unknown function $f\colon \mathbb{R}^n \to \mathbb{R}$ with noisy observations at known locations x_i:

$$y_i = f(x) + \epsilon_i$$

- Observation likelihood:

$$p(y_i | f(x_i)) = \mathcal{N}(y_i | f(x_i), \sigma_\epsilon^2)$$



- Data: N observations at N locations: $y = \in \mathbb{R}^N, \boldsymbol{X} = [x_1, \dots, x_N]^{\mathrm{T}} \in \mathbb{R}^{N \times n} [y_1, \dots, y_N]^{\mathrm{T}}$

Why Gaussian Processes?
- Flexibility (non-parametric approach: distributions over functions)
- Provide uncertainty estimates
- Degrade gracefully (they know what they don't know)

Basic Concepts and Steps of GP regression

- Assume that f belongs to a family of functions with a Gaussian Prior:

$$f(x) \sim N(f(x) | m(x), k(x,x))$$

- *Prior* over the vector $\mathbf{f} = [f(x_1), \dots, f(x_N)]^{\mathrm{T}}$:

$$p(\mathbf{f}; \boldsymbol{X}) = \mathcal{N}(\mathbf{f} | m(x), k(x,x))$$

- *Joint density* of $\boldsymbol{y}$ and $\mathbf{f}$: $p(\boldsymbol{y}, \mathbf{f}; \boldsymbol{X}) = p(\boldsymbol{y} | \mathbf{f}, \boldsymbol{X}) p(\mathbf{f}; \boldsymbol{X})$

- Marginalize out $\mathbf{f}$ to obtain *marginal likelihood*:

$$p(\boldsymbol{y}; \boldsymbol{X}) = \int p(\boldsymbol{y} | \mathbf{f}; \boldsymbol{X}) p(\mathbf{f}; X) \mathrm{d}\mathbf{f} = \mathcal{N}(\boldsymbol{y} | m(\boldsymbol{X}), k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_\epsilon^2 I))$$

- Optimize *hyperparameters* $\Theta_\star = \{\theta_m, \theta_k, \sigma_\epsilon\}$:

$$\Theta_\star = \operatorname*{argmin}_{\Theta} (-\log(p(\boldsymbol{y}; \boldsymbol{X})))$$

where θ_m, θ_k are the mean and kernel hyperparameters

Basic Concepts and Steps of GP regression

- *Prediction* (predict y_* on a test location x_*):

$$p(y_*; x_*, \boldsymbol{y}, \boldsymbol{X}) = \int p(\boldsymbol{y}_*, \boldsymbol{y}; x_*, \boldsymbol{X})\mathrm{d}\boldsymbol{y} = N(\boldsymbol{y}_* | \mu_*, \sigma_*)$$

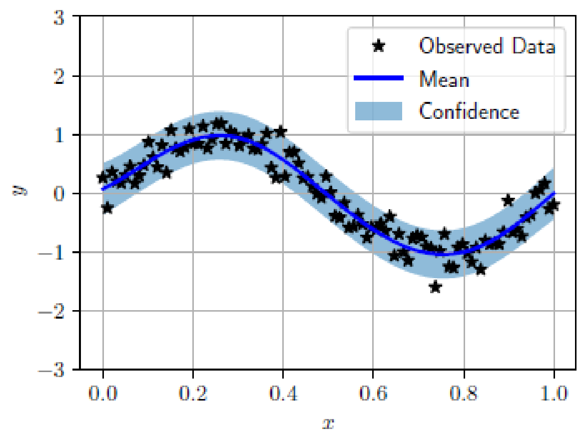
  - $\mu_* = m(x_*) + k(x_*, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_\epsilon^2 I]^{-1}(\boldsymbol{y} - m(\boldsymbol{X}))$
  - $\sigma_* = k(x_*, x_*) - k(x_*, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_\epsilon^2 I]^{-1} k(\boldsymbol{X}, x_*)$

- *Inference*: invert $N \times N$ matrix (scales with cube of data size N). Does not scale to more than a few thousand data points

- Example:

$m(x) = c,$

$$k(x, x') = \sigma_f^2 \exp(-\frac{1}{2}(x - x')^T \boldsymbol{L}^{-1}(x - x'))$$



Scaling GP regression to Big Data: Sparse Variational GP Regression

- Sparse approximation of GPs (goal: reduce cost of inference)
- Introduce M *inducing locations/values* ($M < N$):

$$\boldsymbol{Z} = [z_1 \dots z_M]^{\mathrm{T}}, \qquad \boldsymbol{u} = [f(z_1), \dots, f(z_M)]^{\mathrm{T}}$$

- *Joint density*: $p(\boldsymbol{y}, \mathbf{f}, \boldsymbol{u}) = p(\boldsymbol{y} \mid \mathbf{f}; \boldsymbol{X})\, p(\mathbf{f} \mid \boldsymbol{u}; \boldsymbol{X}, \boldsymbol{Z})\, p(\boldsymbol{u}; \boldsymbol{Z})$
    - $p(\mathbf{f} \mid \boldsymbol{u}; \boldsymbol{X}, \boldsymbol{Z}) = \mathcal{N}(\mathbf{f} \mid \tilde{\mu}, \widetilde{\boldsymbol{\Sigma}})$, where
        - $[\tilde{\mu}]_i = m(x_i) + k(x_i, \boldsymbol{Z}) k(\boldsymbol{Z}, \boldsymbol{Z})^{-1} (u - m(\boldsymbol{Z}))$
        - $\left[\widetilde{\boldsymbol{\Sigma}}\right]_{ij} = k\left(x_i, x_j\right) - k(x_i, \boldsymbol{Z}) k(\boldsymbol{Z}, \boldsymbol{Z})^{-1}\, k\left(\boldsymbol{Z}, x_j\right)$
    - Gaussian prior on $\boldsymbol{u}$: $p(\boldsymbol{u}; \boldsymbol{Z}) = \mathcal{N}(\boldsymbol{u} \mid m(\boldsymbol{Z}), k(\boldsymbol{Z}, \boldsymbol{Z}))$
- *Variational posterior*: $q(\mathbf{f}, \boldsymbol{u}) = p(\mathbf{f} \mid \boldsymbol{u}; \boldsymbol{X}, \boldsymbol{Z}) q(\boldsymbol{u})$
  where $q(\boldsymbol{u}) = \mathcal{N}(\boldsymbol{u} \mid m_u, \boldsymbol{S}_{\boldsymbol{u}})$ ($m_u, \boldsymbol{S}_{\boldsymbol{u}}$: are parameters)

Scaling GP regression to Big Data: Sparse Variational GP Regression

- Marginalize out $\boldsymbol{u}$:
  $q(\mathbf{f}|m_u, \boldsymbol{S_u}; \boldsymbol{X}, \boldsymbol{Z}) = \int p(\mathbf{f}|\boldsymbol{u}; \boldsymbol{X}, \boldsymbol{Z})q(\boldsymbol{u})\mathrm{d}\boldsymbol{u} = \mathcal{N}(\mathbf{f}|\mu(\boldsymbol{X}), \boldsymbol{\Sigma}(\boldsymbol{X}, \boldsymbol{X}))$
  where $[\mu(\boldsymbol{X})]_i = \mu_f(x_i)$ and $[\boldsymbol{\Sigma}(\boldsymbol{X}, \boldsymbol{X})]_{i,j} = \boldsymbol{\Sigma}_f(x_i, x_j)$, with μ_f *and* $\boldsymbol{\Sigma}_f$ defined as before* (based on $\mathbf{u}, \boldsymbol{Z}$)

- Find optimal *variational parameters* $\boldsymbol{Z}, m_u$ and $\boldsymbol{S_u}$ and *hyperparameters* that maximize the following lower bound:

$$\log p(\boldsymbol{y}\,|\boldsymbol{X}) \geq \mathbb{E}_{q(\mathbf{f},\boldsymbol{u})}[\log(p(\boldsymbol{y}, \mathbf{f}, \boldsymbol{u})/q(\mathbf{f}, \boldsymbol{u}))] = \mathcal{L}$$

where $\mathcal{L} = \sum \quad \mathbb{E}_{q(f_i|m,\boldsymbol{S},x_i,\boldsymbol{Z})}[\log p(y_i|f_i) - \mathrm{KL}(q(\boldsymbol{u})||p(\boldsymbol{u}))]$

- Predict y_* on a test location x_*:

$$p(y_*; x_*, m, \boldsymbol{S}, \boldsymbol{Z})) = \mathcal{N}(y_*|\mu_f(x_*), \boldsymbol{\Sigma}_f(x_*, x_*) + \sigma_\epsilon^2)$$

*Tsolovikos, Alexandros, and Efstathios Bakolas. "Nonlinear Covariance Steering using Variational Gaussian Process Predictive Models." *arXiv preprint arXiv:2010.00778* (2020).

System Identification using SVGPs

- **Given:**

  1. Given a black box simulator corresponding to the originally unknown discrete-time nonlinear stochastic system:

  $$z(t+1) = g\big(z(t), u(t)\big) + \epsilon(t)$$

  2. Observation data at known locations $x_i = [z(t_i); u(t_i)]$:

  $$y_i = g\big(z(t_i), u(t_i)\big) + \epsilon(t_i),$$

- **Objective:**

  Use data $D = \{y_i, z_i, u_i\}_{i=1}^{N}$ to learn a SVGP-based prediction model for the system dynamics:

  $$z(t+1) = G\big(z(t), u(t)\big) + w(t)$$

  where $G\big(z(t), u(t)\big) \coloneqq \mu_f([z(t); u(t)])$ and

  $$w(t) \sim N(w_t | 0, \boldsymbol{\Sigma}_f([z(t); u(t)], [z(t), u(t)]) + \sigma_\epsilon^2)$$

Tsolovikos, Alexandros, and Efstathios Bakolas. "Nonlinear Covariance Steering using Variational Gaussian Process Predictive Models." *arXiv preprint arXiv:2010.00778* (2020).

Modifications to Model-Based Greedy Nonlinear Covariance Steering Algorithm

- Successive linearization of the SVGP-based predictive model:

$$z(t+1) = A_{GP} z(t) + B_{GP} u(t) + d_{GP}$$

$$A_{GP} = \frac{\partial}{\partial z} G\big(\mu_z(t), \mu_u(t)\big), \quad B_{GP} = \frac{\partial}{\partial u} G(\mu_z(t), \mu_u(t))$$
$$d_{GP} = -A_{GP}\mu_z(t) - B_{GP}\mu_u(t) + G(\mu_z(t), \mu_u(t))$$

- The unscented transform will also be adjusted*. In particular,
  - GP-based predictive model will be used for sigma points propagation:

$$\hat{\sigma}_{k+1}^{i} = G_{cl}^{k}(\sigma_k^i),\ G_{cl}^{k}(t,z) := G\big(z, \phi_k(t,z)\big)$$

  - The noise covariance used in the computation of the next stage covariance will also be adjusted appropriately

$$\widehat{\boldsymbol{\Sigma}}_{k+1} = \sum_{i=0}^{2n} \delta_k^i \left(\hat{\sigma}_{k+1}^{i} - \hat{\mu}_{k+1}\right)\left(\hat{\sigma}_{k+1}^{i} - \hat{\mu}_{k+1}\right)' + W_k$$

where $W_k = \boldsymbol{\Sigma}_f([z(t); u(t)], [z(t); u(t)]) + \sigma_\epsilon^2 I$

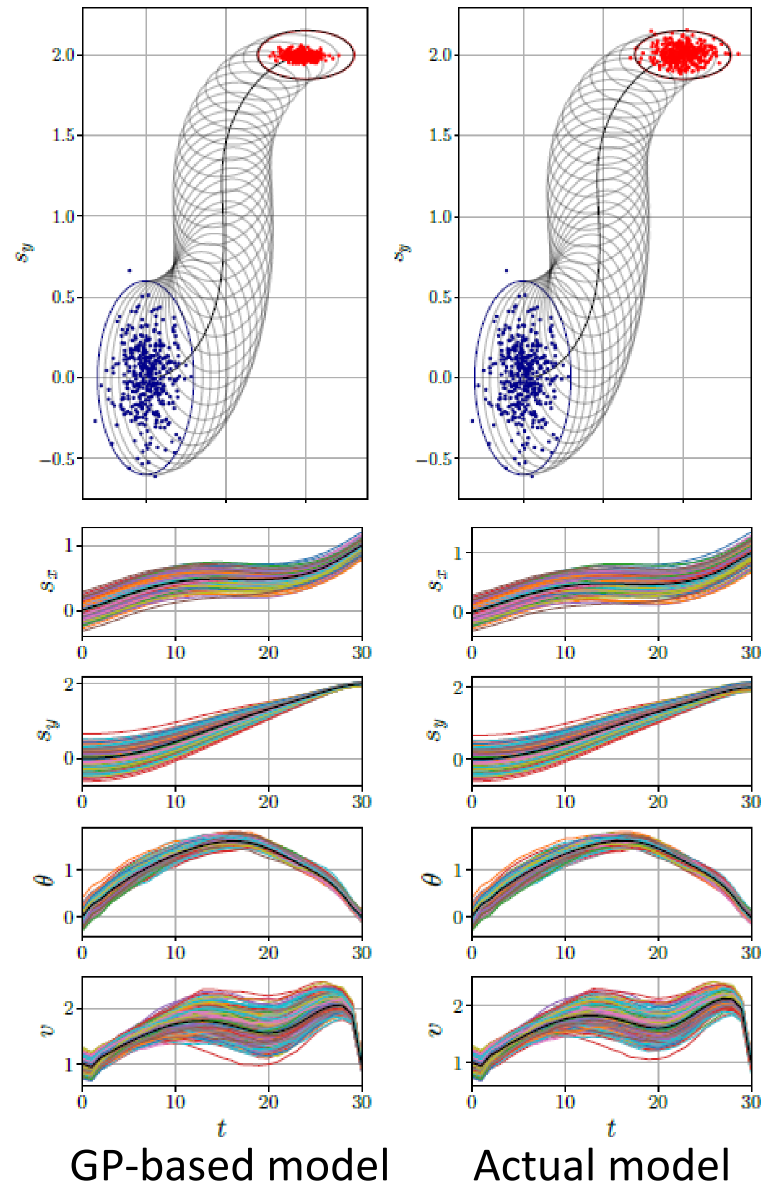
* Ko, Jonathan, et al. "GP-UKF: Unscented Kalman filters with Gaussian process prediction and observation models." *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.

Simulation Results

- Consider again the system:

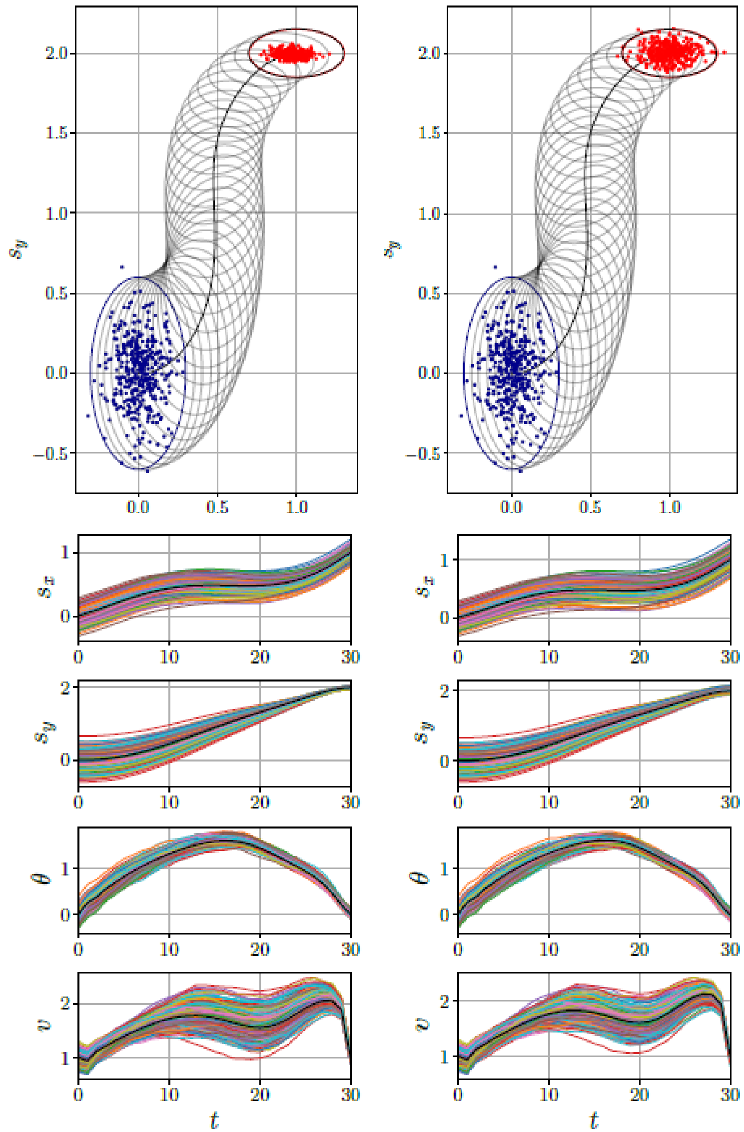
$$s_x(t+1) = s_x(t) + v(t)\tau\cos\theta(t) + \epsilon_x(t)$$
$$s_y(t+1) = s_y(t) + v(t)\tau\sin\theta(t) + \epsilon_y(t)$$
$$\theta(t+1) = \theta(t) + u_\theta(t)v(t)\tau + \epsilon_\theta(t)$$
$$v(t+1) = v(t) + u_v(t)\tau + \epsilon_v(t)$$

- (s_x, s_y): position vector, θ: heading angle, v: speed
- Data from a black box simulator are used for training a GP-based prediction model (use squared exponential kernel)
- **Goal:** Shrink the uncertainty in the coordinate s_y, the heading angle θ, and the speed v, while retaining the uncertainty in s_x.



Remarks on numerical simulations

- Left figures: results based on the SVGP-prediction models; Right figures: results based on model-based covariance steering (SDP formulation)
- The uncertainty predicted by the SVGP model is very close to the uncertainty predicted by the model-based approach.
- The actual terminal distribution obtained using SVGP model (visualization based on red particles from 400 Monte Carlo realizations) is more concentrated near the mean than in the model-based approach (right figure).
- The covariance steering based on the SVGP model is more *cautious* than model-based covariance steering.



GP-based model   Actual model

Summary & Concluding Remarks

- We discussed ways to address covariance / distribution steering problems for discrete-time nonlinear stochastic systems using model-based and model-free (data-driven) approaches
  - successive linearization of system dynamics (based on either a given state-space model or a SVGP-based predictive model) along the ensuing mean (state and input) trajectories
  - the solution of a sequence of linearized covariance steering problems which are either associated with SDP (convex) programs or difference of convex functions programs
  - the unscented transform for the computation of one-stage predictions of the state mean and state covariance

Opportunities for Future Research

- Study the partial state information case for both the model-based and the model-free cases
- Improve performance of linearized covariance steering algorithms (consider different control policy parametrizations)
- Explore different linearization methods in order to reduce the frequency of linearization
- Explore better ways to improve scalability and computational efficiency, and make connections with nonlinear MPC methods (infinite-horizon case)
- Study the distribution steering problems in the class of multimodal distributions

- PhD students who contributed to this research

Alex
Tsolovikos

Isin
Balci

- This research was supported in part by NSF through grants ECCS-1924790, CCMI-1753687 and CMMI-1937957