Analysis and Control of Large Scale Aerospace Systems

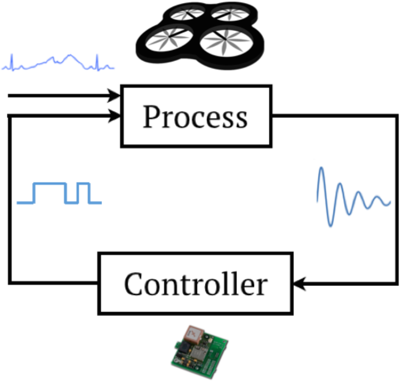
from Planetary Landing to Drone Traffic Management

Abhishek Halder

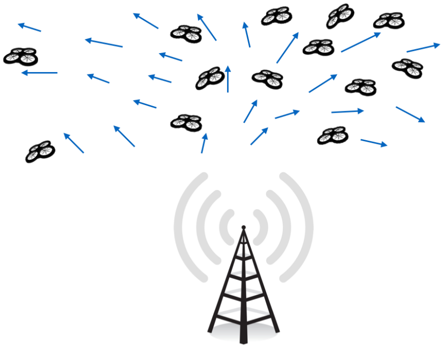
Department of Mechanical and Aerospace Engineering
University of California, Irvine
Irvine, CA 92697-3975

Motivation: Drone Traffic Management

Controlling A Drone

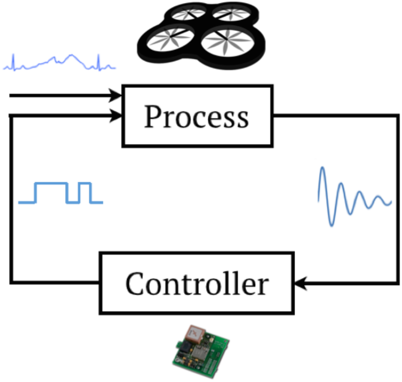


Controlling Swarm of Drones

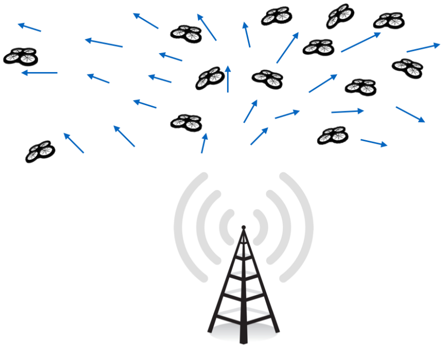


Motivation: Drone Traffic Management

Controlling A Drone

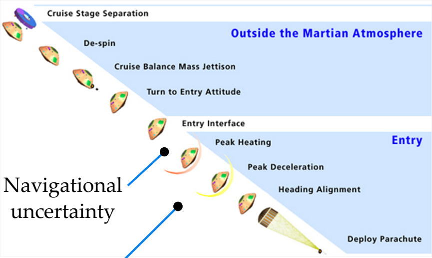


Controlling Swarm of Drones



Large number of agents $\rightsquigarrow$ Population density

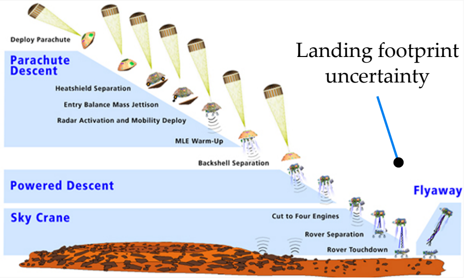
Motivation: Mars Entry-Descent-Landing



Navigational uncertainty

Heating uncertainty

Chute deployment uncertainty



Landing footprint uncertainty

Image credit: NASA JPL

Motivation: Mars Entry-Descent-Landing

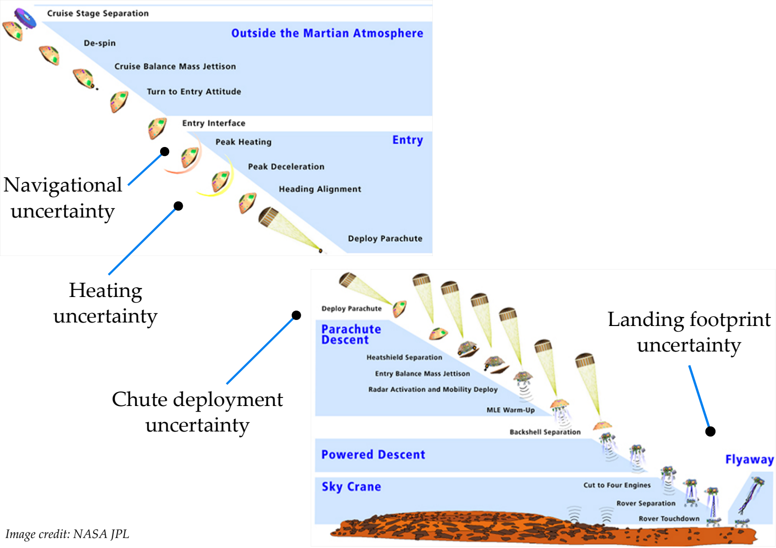
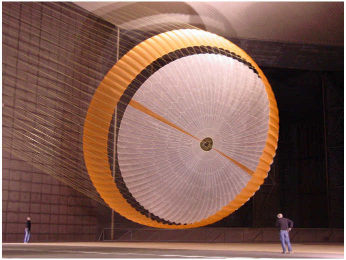


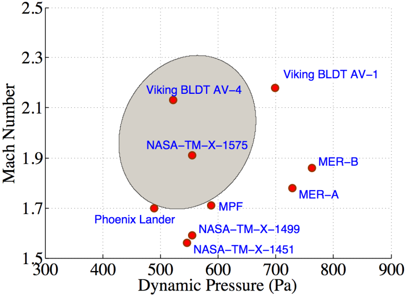
Image credit: NASA JPL

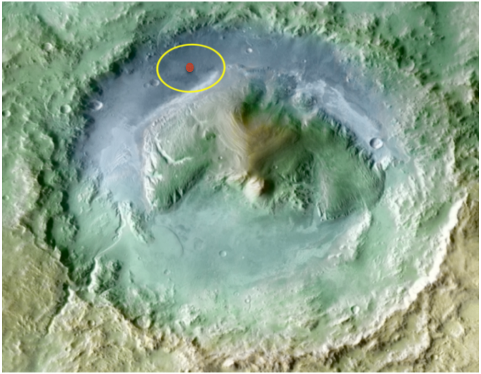
Large number of uncertain scenarios $\rightsquigarrow$ Probability density

Motivation: Mars Entry-Descent-Landing



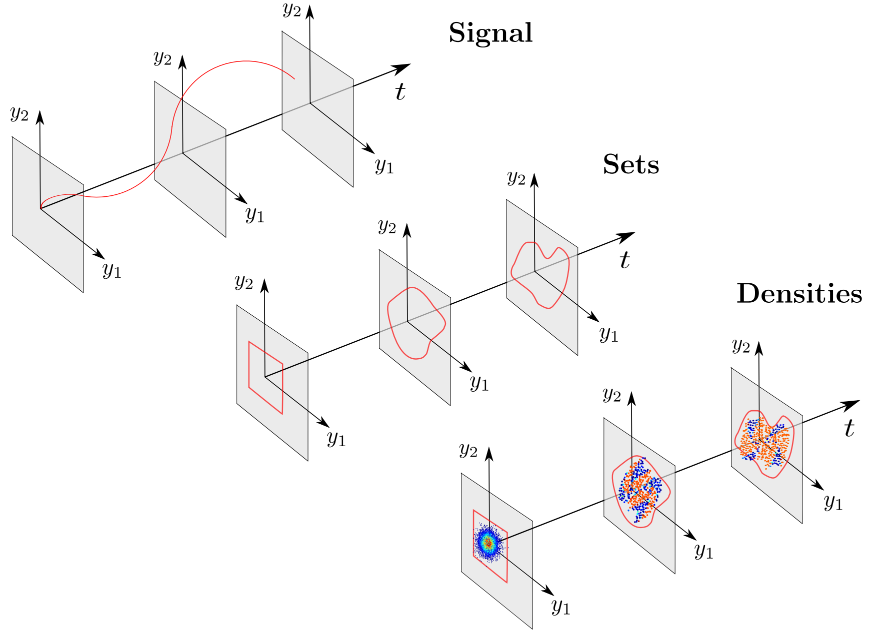
Supersonic parachute





Gale Crater (4.49S, 137.42E)

What to Analyze and Control

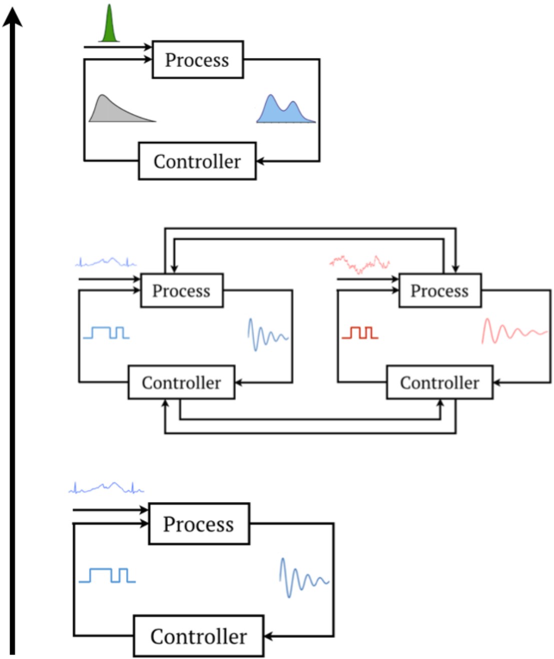


Outlook

Continuum of systems

Finitely many systems

One system



Outline of Today's Talk

Part I: An Application

Propagating Density in Planetary EDL

Part II: A Theory

Controlling Density

Part III: Ongoing and Future Research

Unmanned Aerial Systems Traffic Management

Part I. An Application

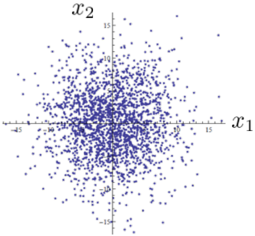
Propagating Density in Planetary EDL

Forecasting, Estimation, Validation, Verification

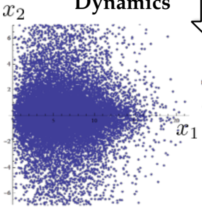
Joint work with R. Bhattacharya (Texas A&M), J. Balaram (JPL)

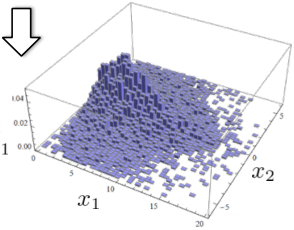
State-of-the-art

Nonlinear Dynamics with Monte Carlo on Samples

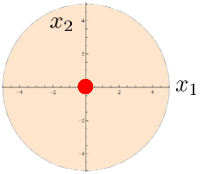


Dynamics

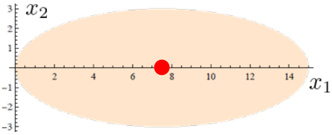




Linear Dynamics with Gaussian Uncertainty



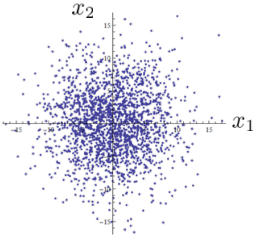
Dynamics

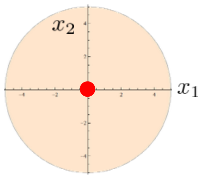


State-of-the-art

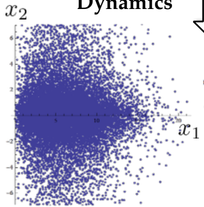
Nonlinear Dynamics with Monte Carlo on Samples

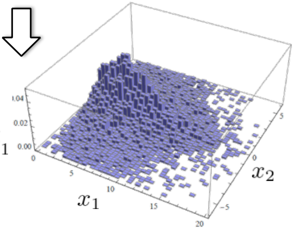
Linear Dynamics with Gaussian Uncertainty



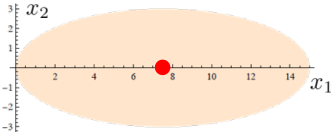


Dynamics





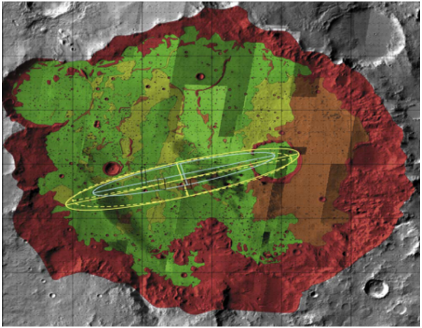
Dynamics



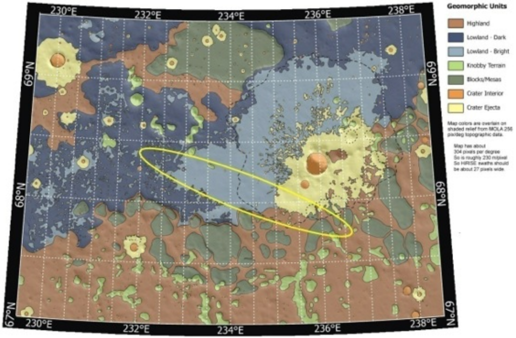
too expensive for EDL simulation

too ideal for EDL simulation

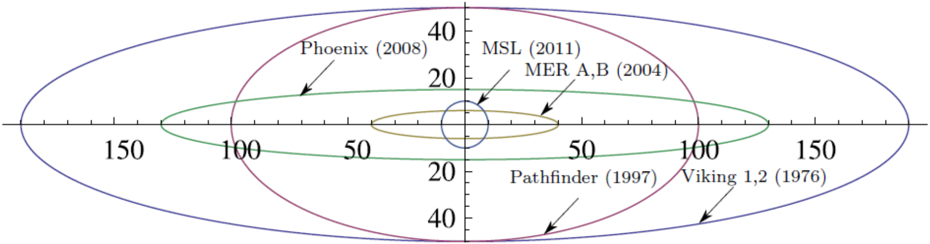
How Bad is Gaussian Fit



Source: Golombek et. al., J. Geophys. Research. 2003



Credit: NASA JPL, Univ. Washington, St. Louis, JHU APL, Univ. Arizona.



Propagating Joint Density Function

Trajectory dynamics

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{p}\right), \quad \mathbf{x} \in \mathbb{R}^{n_s}, \quad \mathbf{p} \in \mathbb{R}^{n_p}; \quad \mathbf{x}(0), \mathbf{p} \text{ random}$$

$$\dot{\mathbf{x}}_e(t) = \mathbf{f}_e\left(\mathbf{x}_e(t)\right), \quad \mathbf{x}_e := \begin{bmatrix} \mathbf{x} \\ \mathbf{p} \end{bmatrix} \in \mathbb{R}^{n_s+n_p}, \quad \mathbf{x}_e(0) \sim \rho_0\left(\mathbf{x}_e\right)$$

Density dynamics

Liouville PDE for joint density $\rho\left(\mathbf{x}_e(t), t\right)$

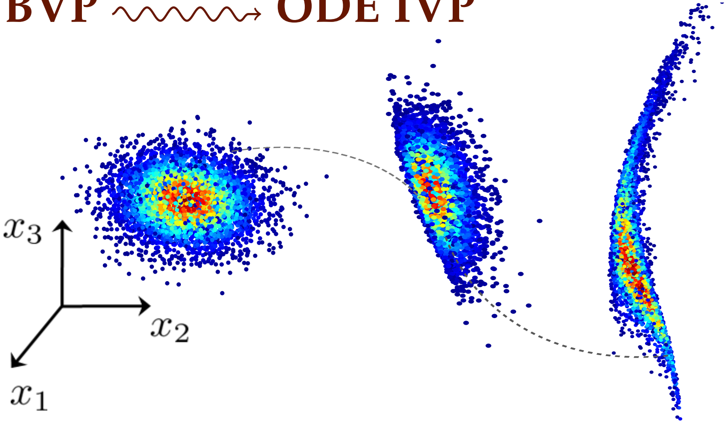
$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \mathbf{f}\right) = 0$$

Method of characteristics (MOC)

$$\dot{\mathbf{x}}_e(t) = \mathbf{f}_e\left(\mathbf{x}_e(t)\right), \quad \dot{\rho}(t) = -\rho \nabla \cdot \mathbf{f}, \quad \begin{bmatrix} \mathbf{x}_e(0) \\ \rho(0) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_e(0) \\ \rho_0(\mathbf{x}_e(0)) \end{bmatrix}$$

MOC

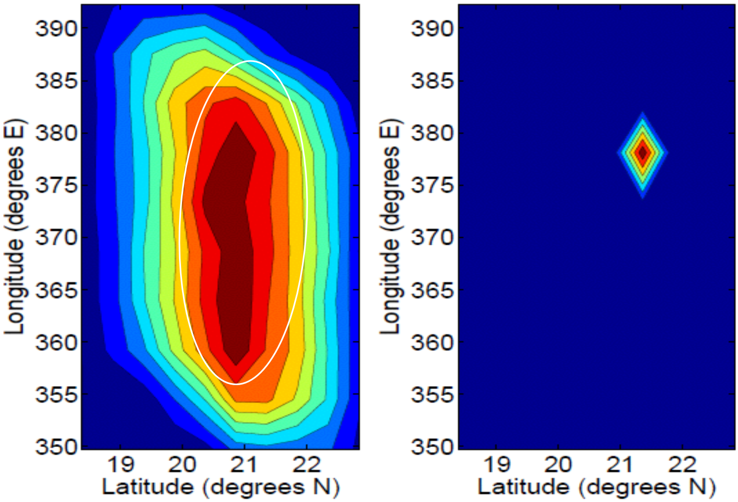
PDE BVP $\rightsquigarrow$ ODE IVP



MC simulation	**Liouville MOC**
Offline post-processing	Online
Histogram approximation	Exact arithmetic
Grid based	Meshless
n_s ODEs per sample	$n_s + 1$ ODEs per sample

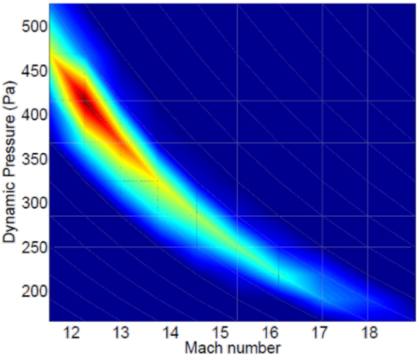
Application to Mars EDL

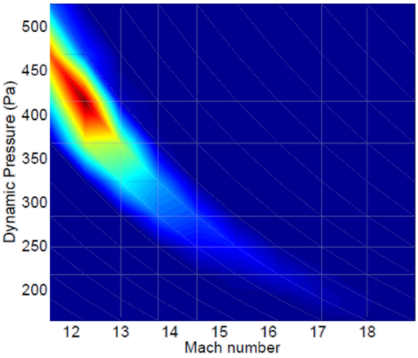
Landing Footprint Uncertainty



Application to Mars EDL

Chute Deployment Uncertainty

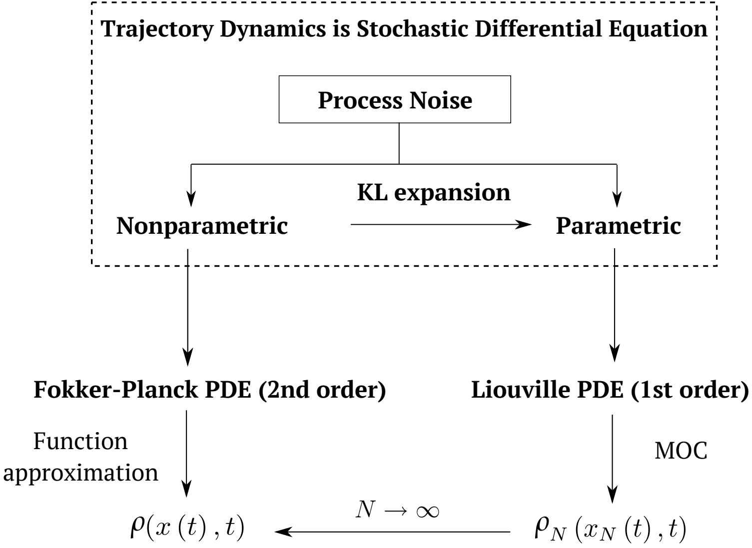




A.H., R. Bhattacharya, Dispersion Analysis in Hypersonic Flight During Planetary Entry Using Stochastic Liouville Equation, *Journal of Guidance, Control, and Dynamics*, 2011.

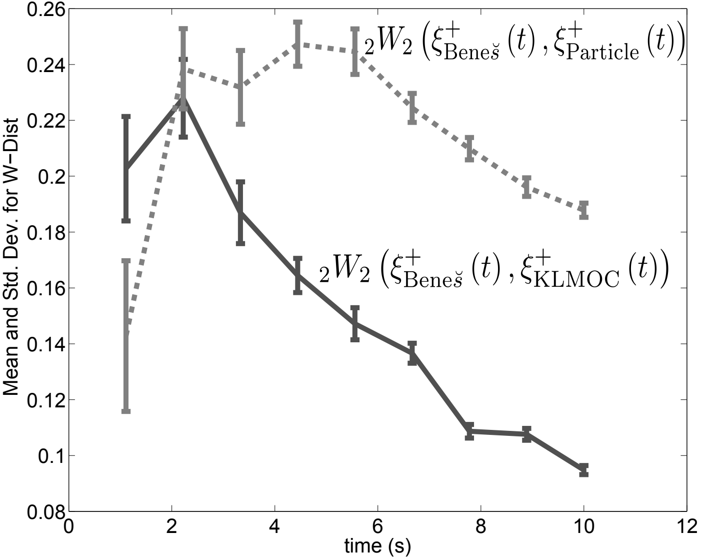
A.H., R. Bhattacharya, Beyond Monte Carlo: A Computational Framework for Uncertainty Propagation in Planetary Entry, Descent and Landing, *AIAA GNC*, 2010.

Extension for Process Noise



P. Dutta, **A.H.**, R. Bhattacharya, Uncertainty Quantification for Stochastic Nonlinear Systems using Perron-Frobenius Operator and Karhunen-Loève Expansion , *MSC*, 2012.

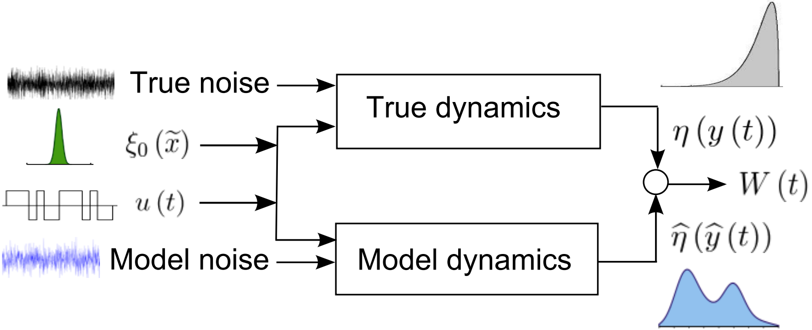
Application to Nonlinear Filtering



P. Dutta, **A.H.**, R. Bhattacharya, Nonlinear Estimation with Perron-Frobenius Operator and Karhunen-Loève Expansion , *IEEE Transactions on Aerospace and Electronic Systems*, 2015.

P. Dutta, **A.H.**, R. Bhattacharya, Nonlinear Filtering with Transfer Operator, *ACC*, 2013.

Model and Controller V&V



K. Lee, **A.H.**, R. Bhattacharya, Performance and Robustness Analysis of Stochastic Jump Linear Systems using Wasserstein Metric, *Automatica*, 2015.

A.H., R. Bhattacharya, Probabilistic Model Validation for Uncertain Nonlinear Systems, *Automatica*, 2014.

A.H., L. Lee, R. Bhattacharya, A Dynamical System Pair with Identical First Two Moments But Different Probability Densities, *CDC*, 2014.

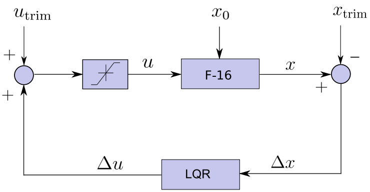
K. Lee, **A.H.**, R. Bhattacharya, Probabilistic Robustness Analysis of Stochastic Jump Linear Systems, *ACC*, 2014.

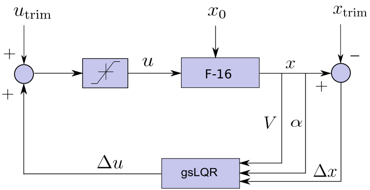
A.H., R. Bhattacharya, Frequency Domain Model Validation in Wasserstein Metric , *ACC*, 2013.

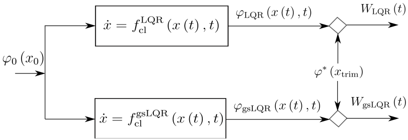
A.H., R. Bhattacharya, Further Results on Probabilistic Model Validation in Wasserstein Metric, *CDC*, 2012.

A.H., R. Bhattacharya, Model Validation: A Probabilistic Formulation, *CDC*, 2011.

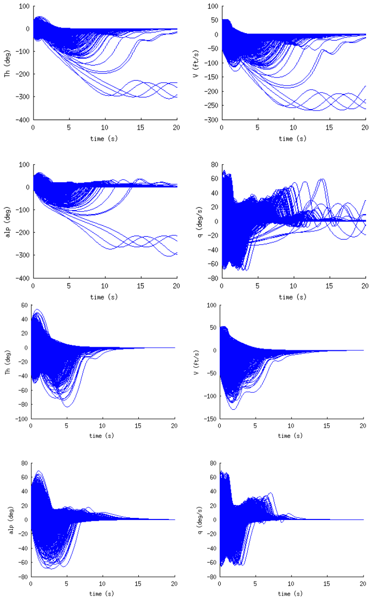
F-16 Flight Controller Verification



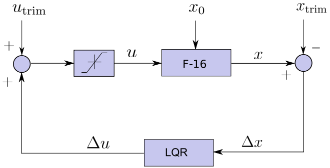


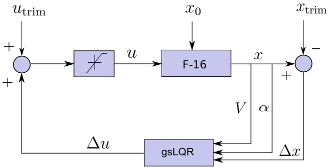


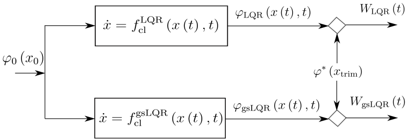
LQR vs. gsLQR Results: (MC)



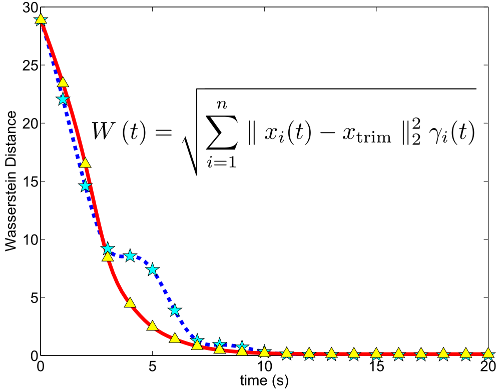
F-16 Flight Controller Verification







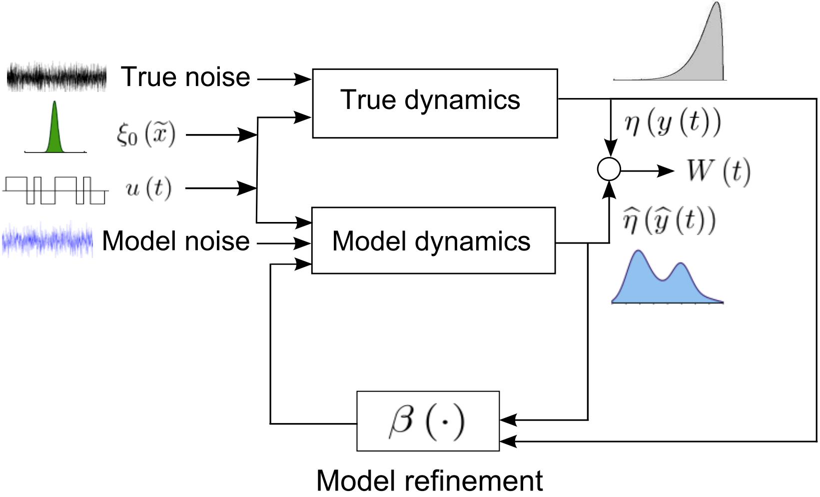
LQR vs. gsLQR Results: (MOC)



A.H., K. Lee, R. Bhattacharya, Optimal Transport Approach for Probabilistic Robustness Analysis of F-16 Controllers, *Journal of Guidance, Control, and Dynamics*, 2015.

A.H., K. Lee, R. Bhattacharya, Probabilistic Robustness Analysis of F-16 Controller Performance: An Optimal Transport Approach, *ACC*, 2013.

Model Refinement



A.H., R. Bhattacharya, Geodesic Density Tracking with Applications to Data Driven Modeling, *ACC*, 2014.

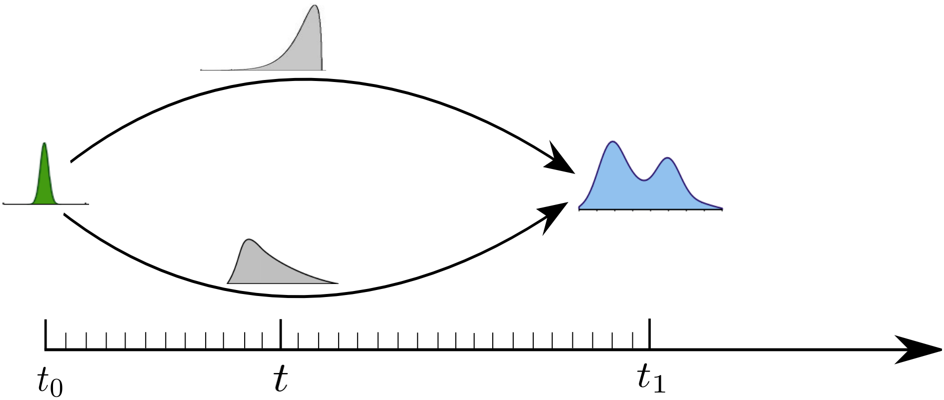
Part II. A Theory

Controlling Density

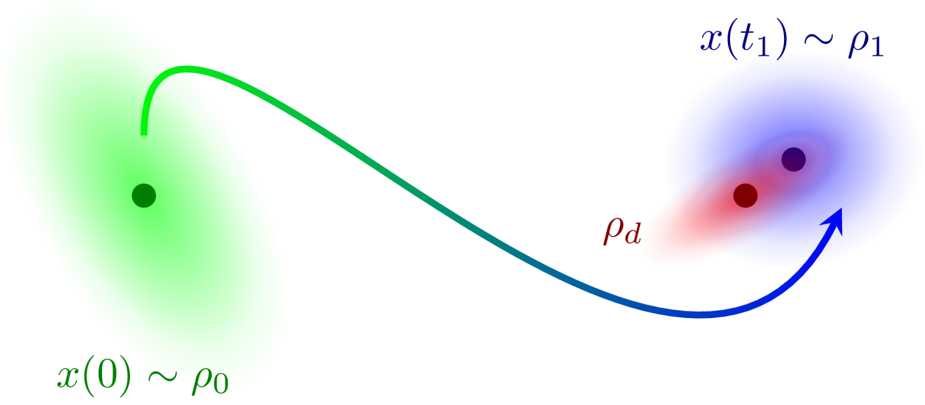
Finite Horizon LQG Density Regulator

Joint work with E.D.B. Wendel (Draper Laboratory)

How to Go from One Density to Another



or Close to Another



LQG State Regulator

$$\min_{u \in \mathcal{U}} \phi\left(x_1, x_d\right) \; + \; \mathbb{E}_x \left[\int_0^{t_1} (x^\top Q x \; + \; u^\top R u) \, \mathrm{d}t \right]$$

$$\mathrm{d}x(t) = Ax(t)\,\mathrm{d}t + Bu(t)\,\mathrm{d}t + F\,\mathrm{d}w(t),$$

$$x(0) = x_0 \text{ given}, \;\; x_d \text{ given}, \;\; t_1 \text{ fixed},$$

Typical terminal cost: MSE

$$\phi\left(x_1, x_d\right) = \mathbb{E}_{x_1}\left[(x_1 - x_d)^\top M (x_1 - x_d)\right]$$

LQG Density Regulator

$$\min_{u \in \mathcal{U}} \varphi\left(\rho_1, \rho_d\right) \;+\; \mathbb{E}_x \left[\int_0^{t_1} (x^\top Q x \;+\; u^\top R u) \, \mathrm{d}t \right]$$

$$\mathrm{d}x(t) = Ax(t)\,\mathrm{d}t + Bu(t)\,\mathrm{d}t + F\,\mathrm{d}w(t),$$

$$x(0) \sim \rho_0 \text{ given}, \quad x_d \sim \rho_d \text{ given}, \quad t_1 \text{ fixed},$$

Proposed terminal cost: MMSE

$$\varphi\left(x_1, x_d\right) = \inf_{y \sim \rho \in \mathcal{P}_2(\rho_1, \rho_d)} \mathbb{E}_y \left[(x_1 - x_d)^\top M (x_1 - x_d) \right],$$

where $y := \left(x_1, x_d\right)^\top$

Formulation: LQG Density Regulator

$$\min_{u\in\mathcal{U}} \overbrace{\inf_{y\sim\rho\in\mathcal{P}_2(\rho_1,\rho_d)} \mathbb{E}_y\left[(x_1-x_d)^\top M(x_1-x_d)\right]}^{\varphi(\rho_1,\rho_d)} + \mathbb{E}_x\left[\int_0^{t_1}(x^\top Qx + u^\top Ru)\,\mathrm{d}t\right]$$

$$\mathrm{d}x(t) = Ax(t)\,\mathrm{d}t + Bu(t)\,\mathrm{d}t + F\,\mathrm{d}w(t),$$

$$x(0)\sim\rho_0 = \mathcal{N}(\mu_0, S_0), \quad x_d\sim\rho_d=\mathcal{N}(\mu_d,S_d),$$

$$t_1 \text{ fixed}, \quad \mathcal{U} = \{u \,:\, u(x,t) = K(t)x + v(t)\}$$

∞ dim. TPBVP $\rightsquigarrow$ (n^2+3n) dim. TPBVP

$$\begin{pmatrix} \dot{\mu}(t) \\ \dot{z}(t) \end{pmatrix} = \begin{pmatrix} A & BR^{-1}B^\top \\ Q & -A^\top \end{pmatrix} \begin{pmatrix} \mu(t) \\ z(t) \end{pmatrix},$$

$$\dot{S}(t) = (A + BK^o)S(t) + S(t)(A + BK^o)^\top + FF^\top,$$

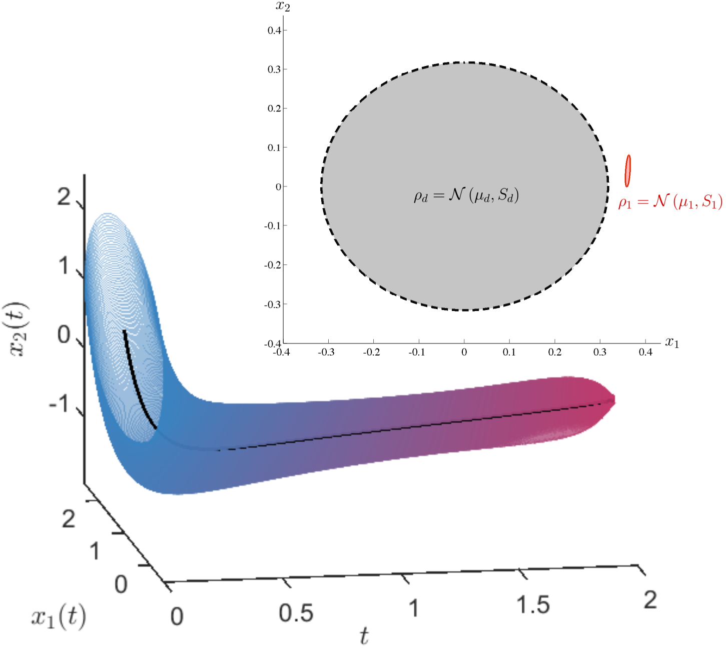
$$\dot{P}(t) = -A^\top P(t) - P(t)A - P(t)BR^{-1}B^\top P(t) + Q,$$

Boundary conditions:

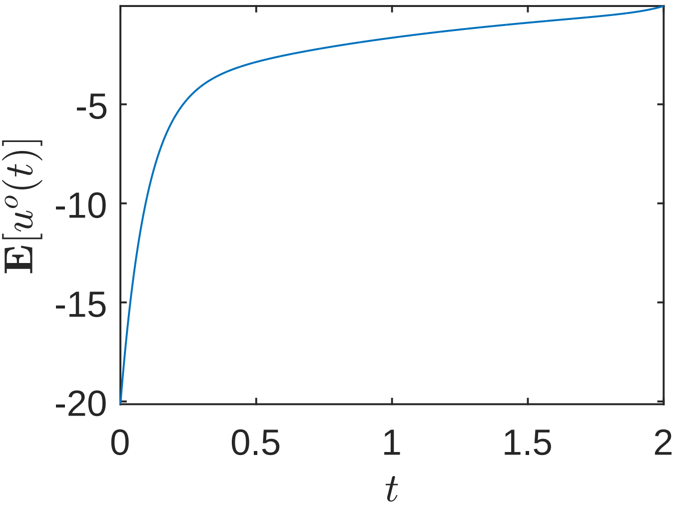
$$\mu(0) = \mu_0,\ z(t_1) = M(\mu_d - \mu_1),$$

$$S(0) = S_0,\ P(t_1) = \left(S_d^{\frac{1}{2}} \left(S_d^{-\frac{1}{2}} S_1^{-1} S_d^{-\frac{1}{2}}\right)^{\frac{1}{2}} S_d^{\frac{1}{2}} - I_n\right)M$$

Controlled State Covariance



Expected Optimal Control



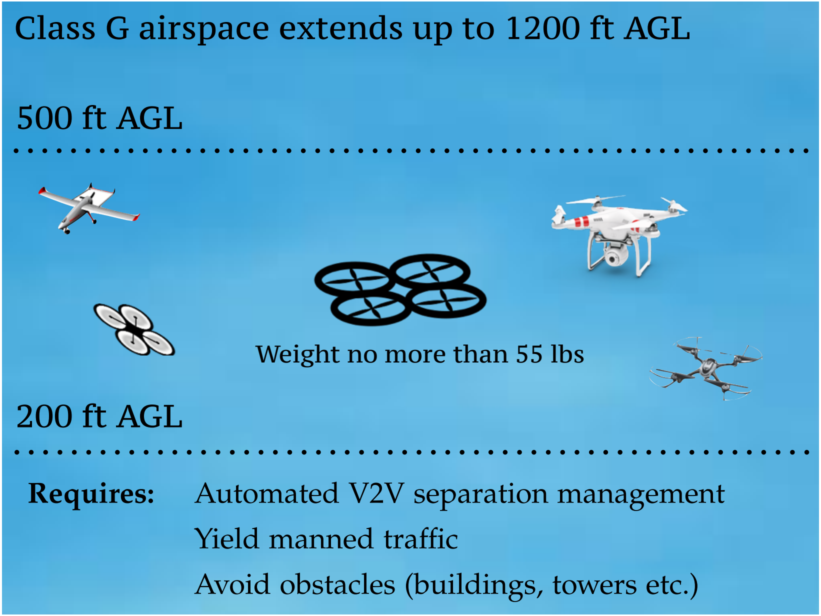
A.H., E.D.B. Wendel, Finite Horizon Linear Quadratic Gaussian Density Regulator with Wasserstein Terminal Cost, *ACC*, 2016.

Part III. Ongoing and Future Research

UTM

Unmanned Aerial Systems Traffic Management

Vision for UAS Traffic Management (UTM)



Technical Challenges

Dynamic Geofencing

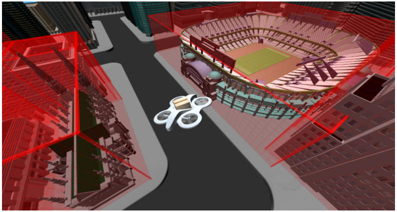
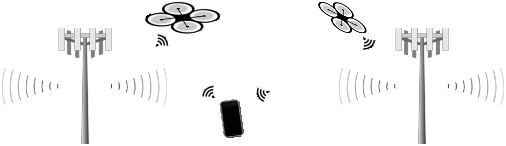
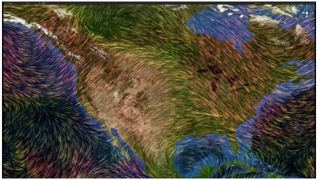


Image credit: NASA Ames Research Center

Control over LTE



Wind Uncertainty



Provable Safety

Protocols $\equiv$ Laws of the Sky

Offline Protocol

– How FAA approves a flight path request?

Motion Protocol

– What does an individual drone do in real time?

Communication Protocol

– What and how should a drone in flight talk?

Database Protocol

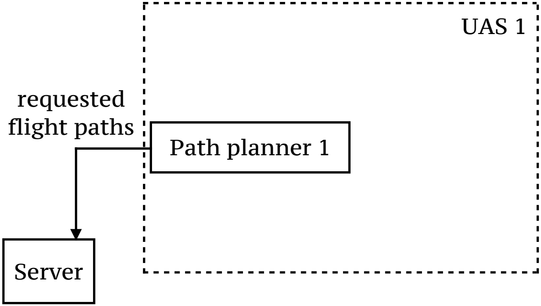
– Which other drones to talk with and when?

Offline Protocol

How FAA approves a flight path request?

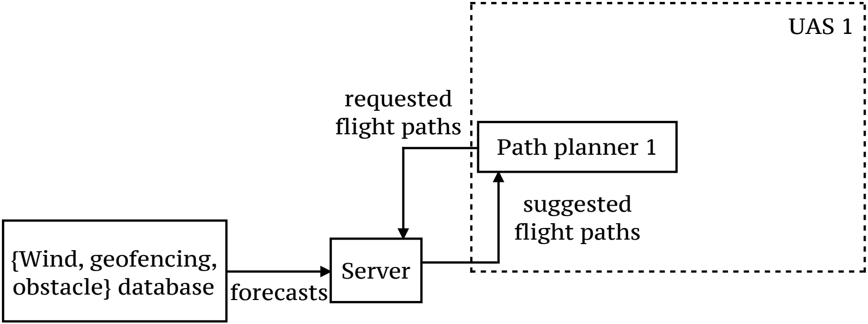
Offline Protocol

Path Planning and Deconfliction



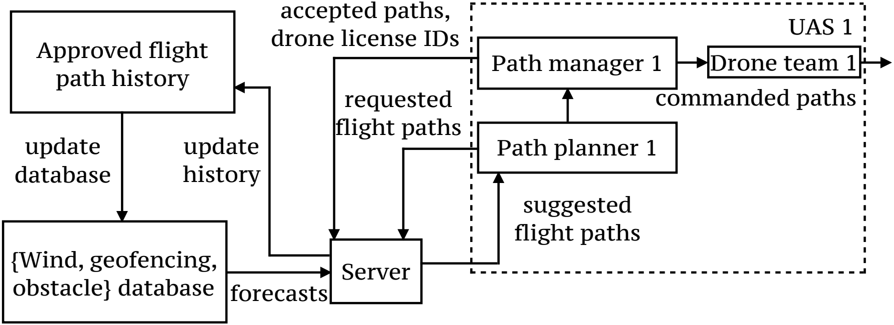
Offline Protocol

Path Planning and Deconfliction



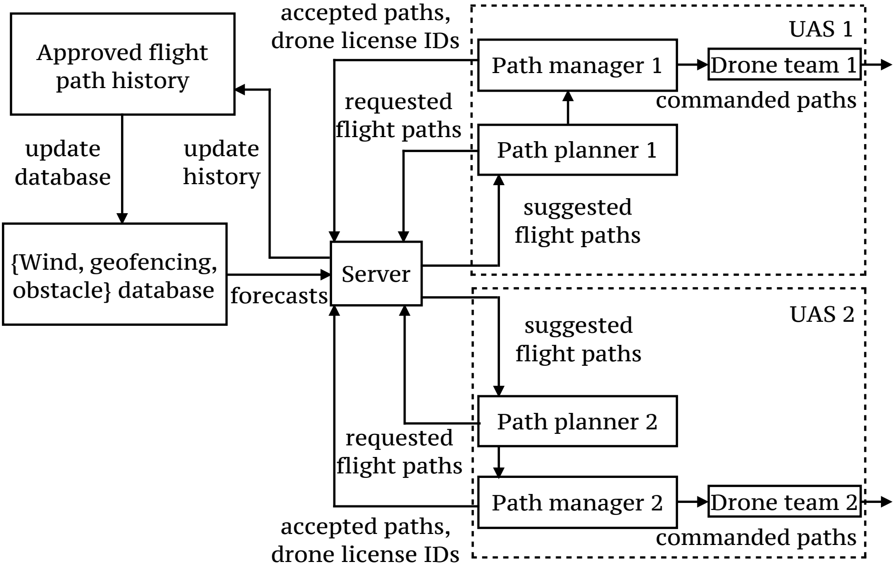
Offline Protocol

Path Planning and Deconfliction



Offline Protocol

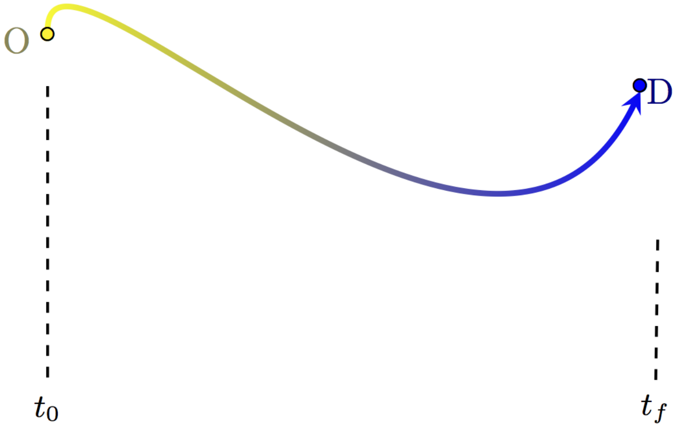
Path Planning and Deconfliction



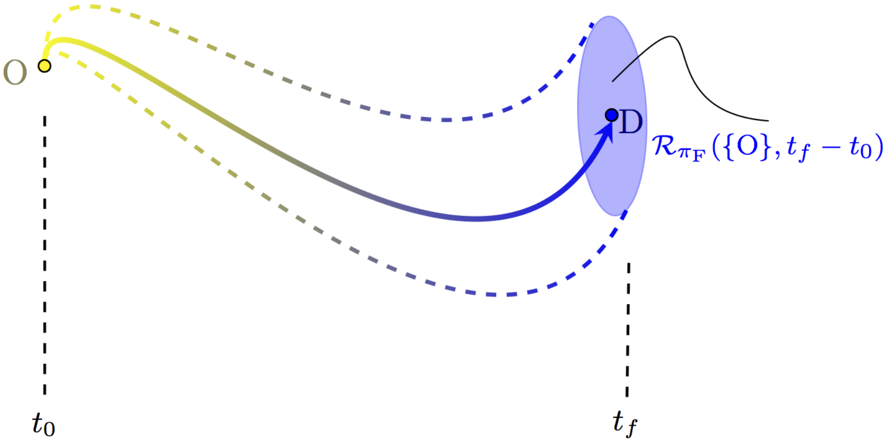
Motion Protocol

What does an individual drone do in real time?

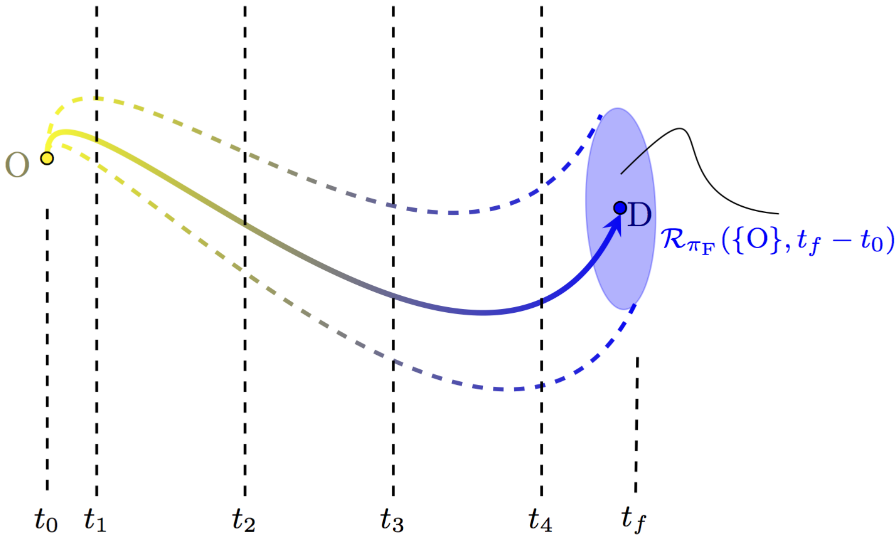
Input: Approved Flight Path



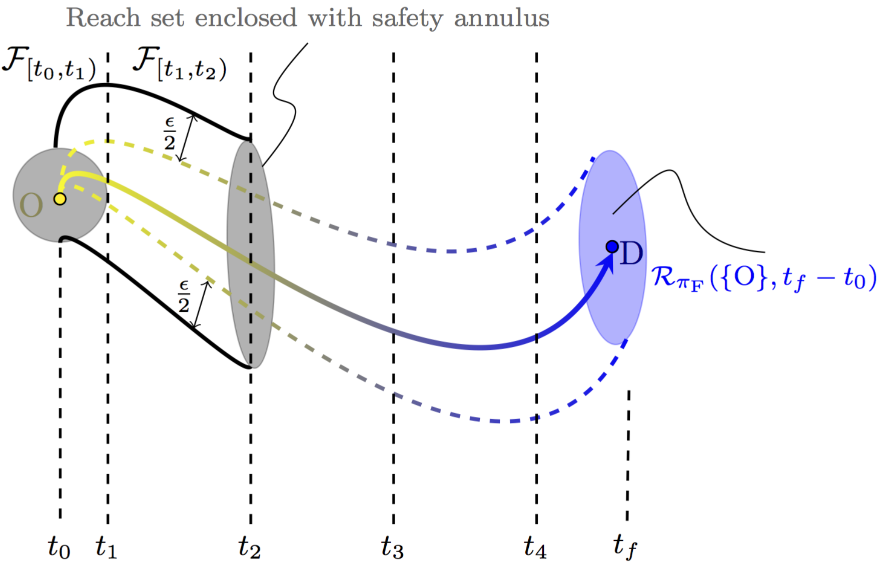
Reach Set Evolution due to Wind Uncertainty



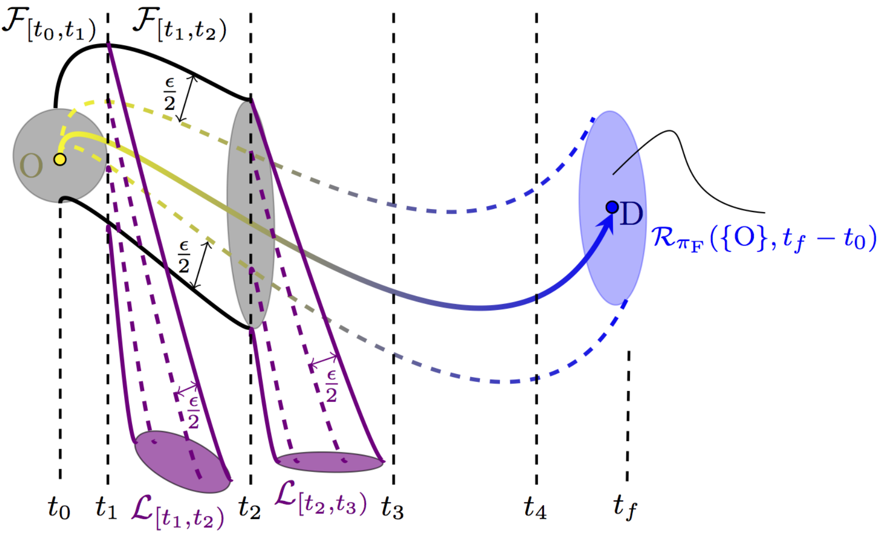
Discrete Decision Making Instances



4D Flight Tubes $\mathcal{F}_{[t_j,t_{j+1})}$

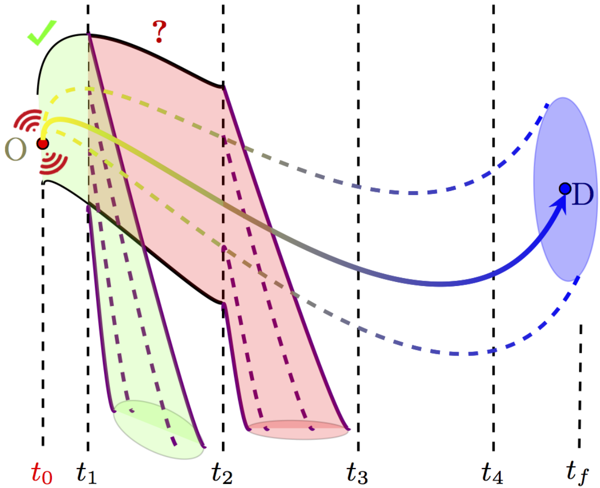


4D Flight + Landing Tubes $\{\mathcal{F}_{[t_j,t_{j+1})}, \mathcal{L}_{[t_{j+1},t_{j+2})}\}$



Motion Protocol: $t = t_0$

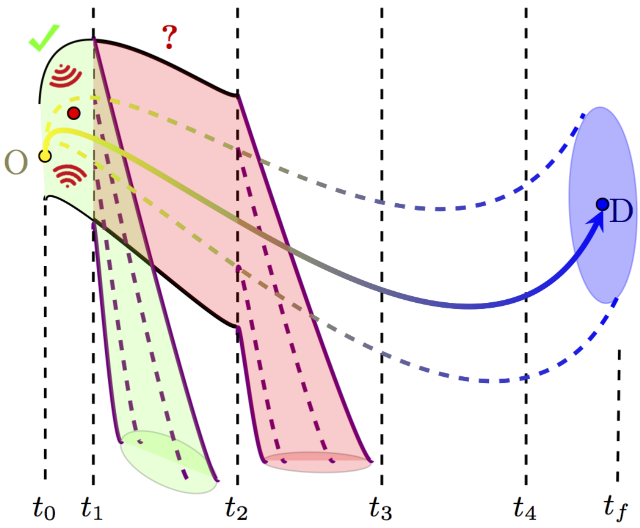
IF: Have all + ACKs for $\{\mathcal{F}_{[t_0,t_1)}, \mathcal{L}_{[t_1,t_2)}\}$ **AND** D $\in \mathcal{R}_{\pi_F}\left(\{O\}, t_f - t_0\right)$



THEN: Take-off **AND** broadcast req. for $\{\mathcal{F}_{[t_1,t_2)}, \mathcal{L}_{[t_2,t_3)}\}$

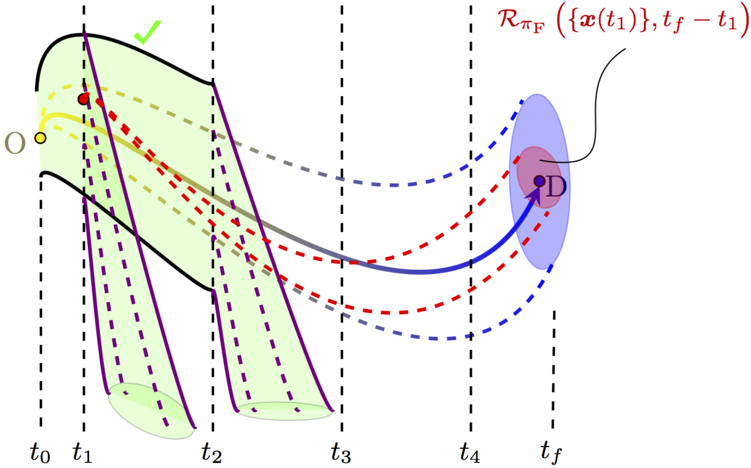
Motion Protocol: $t \in [t_0, t_1)$

Listening for $\pm$ ACKs, $\boldsymbol{x}(t) \in \mathcal{F}_{[t_0,t_1)}$



Motion Protocol: $t = t_1$

IF: All + ACKs **AND** D $\in \mathcal{R}_{\pi_F}\left(\{\boldsymbol{x}(t_1)\}, t_f - t_1\right)$

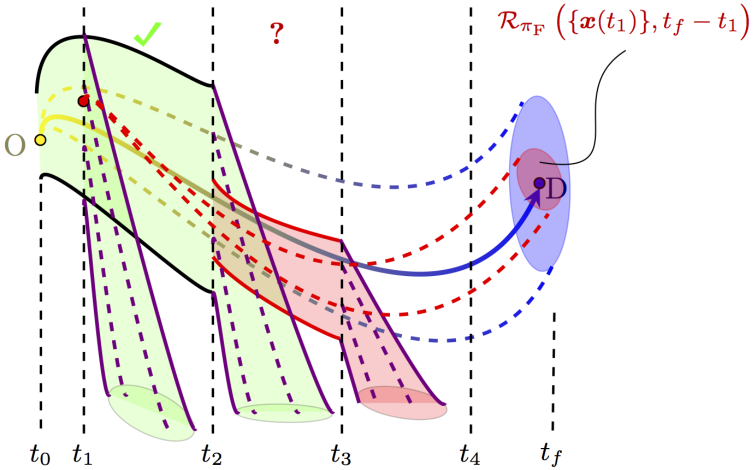


THEN: Continue in $\mathcal{F}_{[t_1,t_2)}$ **AND** broadcast req. for $\{\mathcal{F}_{[t_2,t_3)}, \mathcal{L}_{[t_3,t_4)}\}$

ELSE: Abort mission via $\mathcal{L}_{[t_1,t_2)}$

Motion Protocol: $t = t_1$

IF: All + ACKs **AND** D $\in \mathcal{R}_{\pi_{\mathrm{F}}}\left(\{\boldsymbol{x}(t_1)\}, t_f - t_1\right)$

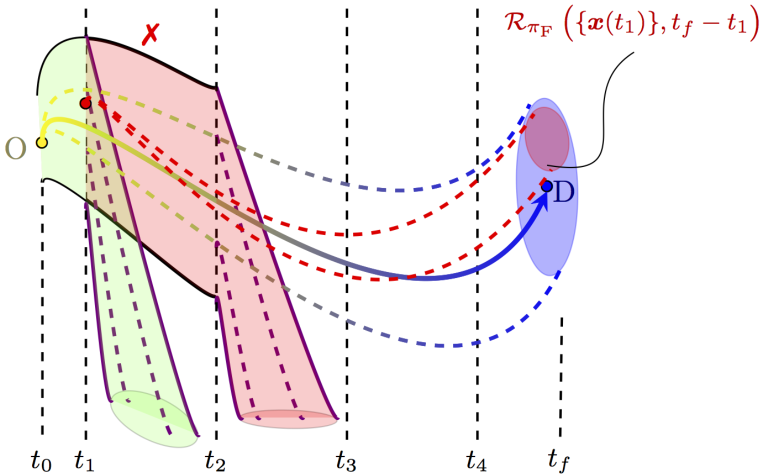


THEN: Continue in $\mathcal{F}_{[t_1,t_2)}$ **AND** broadcast req. for $\{\mathcal{F}_{[t_2,t_3)}, \mathcal{L}_{[t_3,t_4)}\}$

ELSE: Abort mission via $\mathcal{L}_{[t_1,t_2)}$

Motion Protocol: $t = t_1$

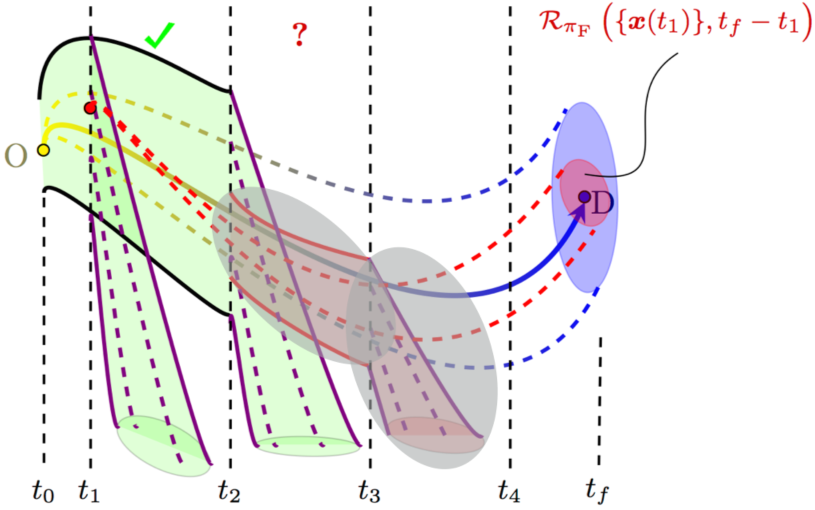
IF: All + ACKs **AND** D $\notin \mathcal{R}_{\pi_F}\left(\{\boldsymbol{x}(t_1)\}, t_f - t_1\right)$



THEN: Continue in $\mathcal{F}_{[t_1,t_2)}$ **AND** broadcast req. for $\{\mathcal{F}_{[t_2,t_3)}, \mathcal{L}_{[t_3,t_4)}\}$

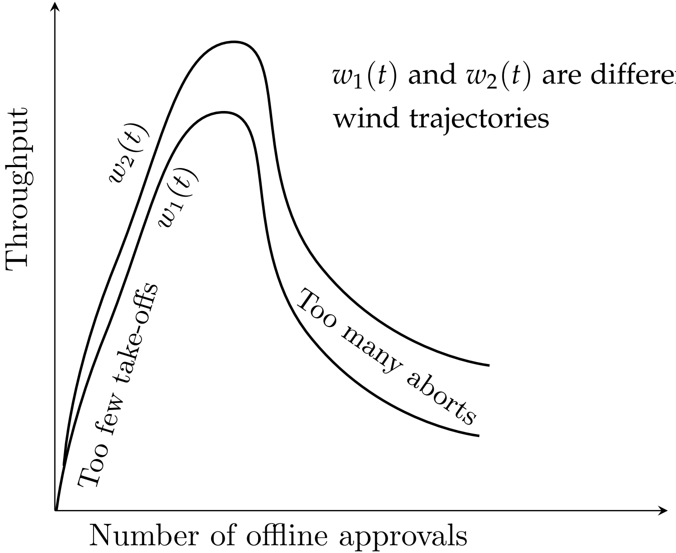
ELSE: Abort mission via $\mathcal{L}_{[t_1,t_2)}$

Algorithms for Motion Protocol

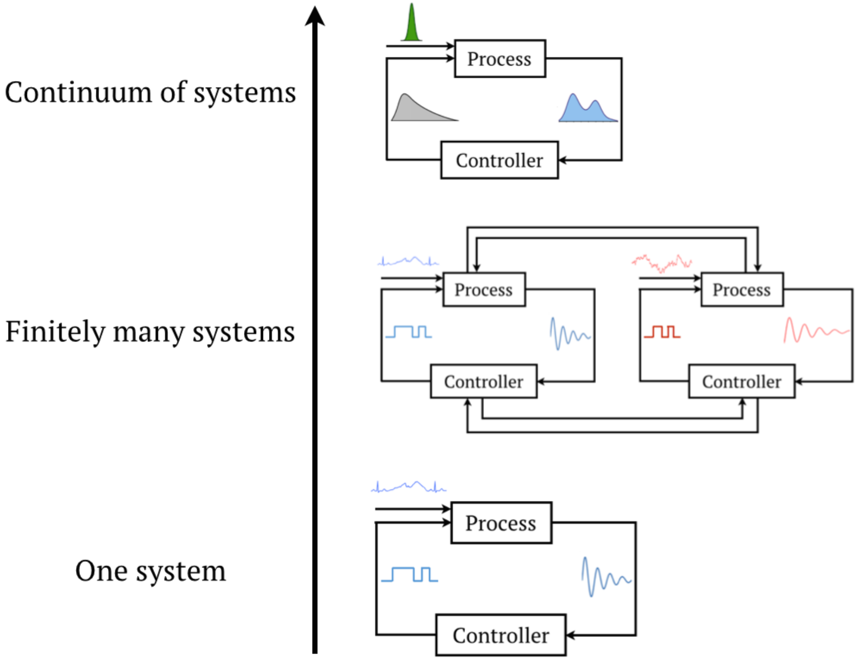


Compute minimum volume outer ellipsoids: SDP

Proposed Architecture: Performance



$w_1(t)$ and $w_2(t)$ are different wind trajectories


Continuum of systems
Process
Controller
Finitely many systems
Process
Controller
Process
Controller
One system
Process
Controller

Thank You

Backup Slides for Part I

Backup Slides for Part II

$\varphi\left(\mathcal{N}\left(\mu_1, S_1\right), \mathcal{N}\left(\mu_d, S_d\right)\right)$ **equals**

$$\left(\mu_1-\mu_d\right)^{\top} M\left(\mu_1-\mu_d\right)+$$

$$\min_{C \in \mathbb{R}^{n \times n}} \operatorname{tr}\left(\left(S_1+S_d-2C\right) M\right) \text{ s.t. } \begin{bmatrix} S_1 & C \\ C^{\top} & S_d \end{bmatrix} \succeq 0$$

$\varphi\left(\mathcal{N}\left(\mu_1, S_1\right), \mathcal{N}\left(\mu_d, S_d\right)\right)$ **equals**

$$\left(\mu_1 - \mu_d\right)^\top M \left(\mu_1 - \mu_d\right) +$$

$$\min_{C \in \mathbb{R}^{n \times n}} \operatorname{tr}\left(\left(S_1 + S_d - 2C\right)M\right) \text{ s.t. } \begin{bmatrix} S_1 & C \\ C^\top & S_d \end{bmatrix} \succeq 0$$

$$\Updownarrow$$

$$\max_{C \in \mathbb{R}^{n \times n}} \operatorname{tr}\left(CM\right) \quad \text{s.t.} \quad S_1 - C S_d^{-1} C^\top \succeq 0$$

$$\Updownarrow$$

$$C^* = S_1 S_d^{\frac{1}{2}} \left(S_d^{\frac{1}{2}} S_1 S_d^{\frac{1}{2}}\right)^{-\frac{1}{2}} S_d^{\frac{1}{2}}$$

This gives

$$\varphi\left(\mathcal{N}\left(\mu_1, S_1\right), \mathcal{N}\left(\mu_d, S_d\right)\right) = \left(\mu_1 - \mu_d\right)^\top M \left(\mu_1 - \mu_d\right)$$

$$+\mathrm{tr}\left(MS_1 + MS_d - 2\left[\left(\sqrt{S_d} M S_1 \sqrt{S_d}\right)\left(\sqrt{S_d} S_1 \sqrt{S_d}\right)^{-\frac{1}{2}}\right]\right)$$

Applying maximum principle:

$$K^o(t) = R^{-1} B^\top P(t),$$

$$v^o(t) = R^{-1} B^\top \left(z(t) - P(t)\mu(t)\right)$$

Matrix Geometric Mean

The minimal geodesic $\gamma^* : [0, 1] \mapsto \mathbf{S}_n^+$ connecting $\gamma(0) = S_d$ and $\gamma(1) = S_1^{-1}$, associated with the Riemannian metric $g_A(S_d, S_1^{-1}) = \operatorname{tr}\left(A^{-1}S_d A^{-1}S_1^{-1}\right)$, is

$$\gamma^*(t) = S_d \,\#_t\, S_1^{-1} = S_d^{\frac{1}{2}}\left(S_d^{-\frac{1}{2}}S_1^{-1}S_d^{-\frac{1}{2}}\right)^t S_d^{\frac{1}{2}}$$

$$= S_1^{-1} \,\#_{1-t}\, S_d = S_1^{-\frac{1}{2}}\left(S_1^{\frac{1}{2}}S_d S_1^{\frac{1}{2}}\right)^{1-t} S_1^{-\frac{1}{2}}$$

Geometric Mean:

$$\gamma^*\left(\frac{1}{2}\right) = S_d \,\#_{\frac{1}{2}}\, S_1^{-1} = S_1^{-1} \,\#_{\frac{1}{2}}\, S_d$$

Example

$$\begin{pmatrix} \mathrm{d}x_1 \\ \mathrm{d}x_2 \end{pmatrix} = \begin{bmatrix} 0 & 1 \\ 2 & -3 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \mathrm{d}t + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \, \mathrm{d}t + \begin{bmatrix} 0.01 \\ 0.01 \end{bmatrix} \mathrm{d}w$$

$$\rho_0 = \mathcal{N}\left((1,1)^\top, I_2\right), \quad \rho_d = \mathcal{N}\left((0,0)^\top, 0.1\, I_2\right),$$

$$Q = 100\, I_2, \quad R = 1, \quad M = I_2, \quad t_1 = 2$$

Backup Slides for Part III

Input-Output for Motion Protocol