On Optimal Control of Complex Dynamical Systems

Riccardo Bonalli

Autonomous Systems Laboratory, Department of Aeronautics and Astronautics, Stanford University

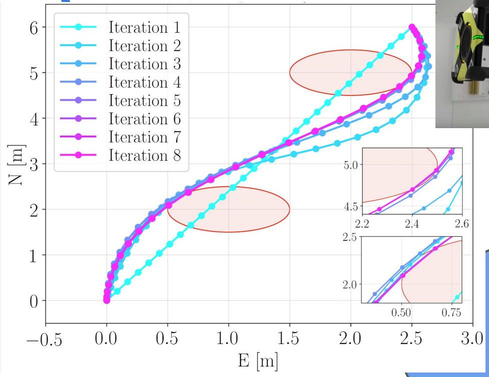
A quick overview of my research

Goal

Devise algorithms for autonomous systems that select the best available strategy and make robust decisions under uncertainty

A quick overview of my research

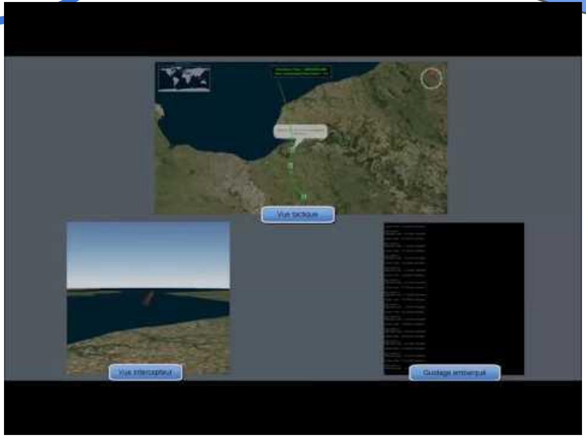
Stochastic optimal control



Goal

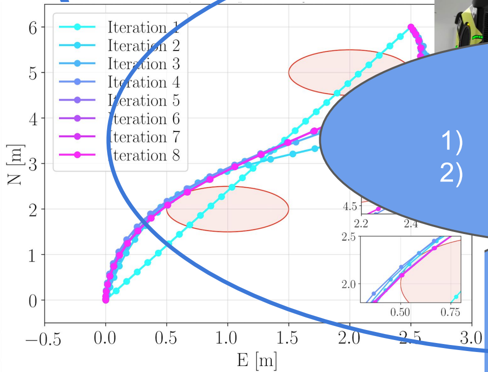
Devise algorithms for autonomous systems that select the best available strategy and make robust decisions under uncertainty

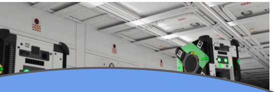
Deterministic optimal control



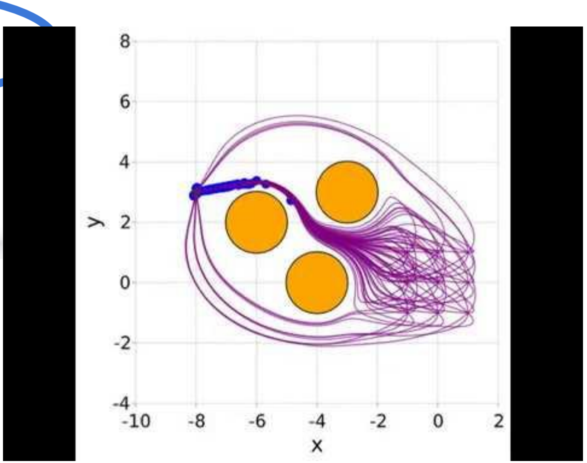
A quick overview of my research

Stochastic optimal control





Differential geometric policy generation



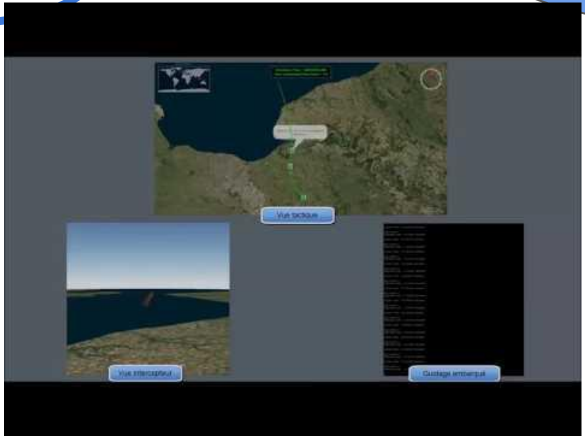
Today's topic!

1) SCP
2) Pullback Bundle Dynamical Systems

Goal

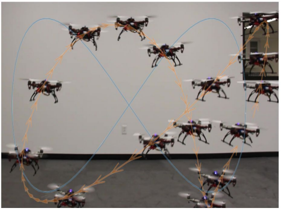
Devise algorithms for autonomous systems that select the best available strategy and make robust decisions under uncertainty

Deterministic optimal control

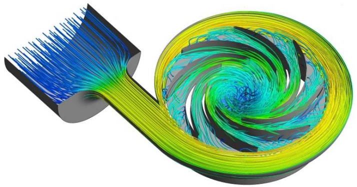


Some ongoing/future projects

Provable reinforcement learning

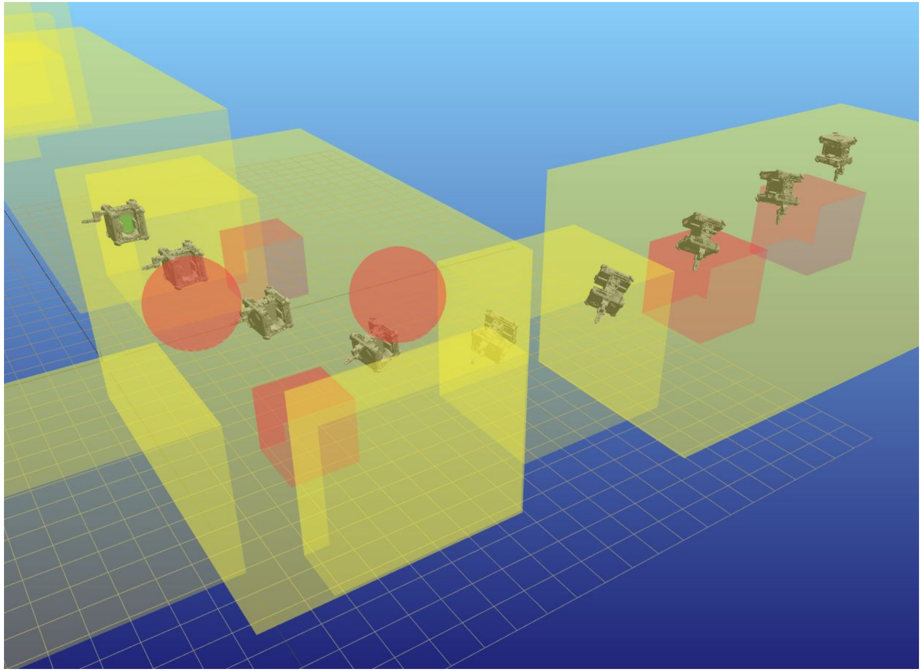


Large scale control algorithms



First topic

Stochastic non-linear optimal control through Sequential Convex Programming (SCP)



Optimal trajectories for Astrobee computed via SCP

Optimal control of finite-dimensional dynamical systems

$$\min_{u \in \mathcal{U}} \; \mathbb{E}\left[\int_0^{t_f} f^0(s, u(s), x(s)) \, \mathrm{d}s \right]$$

$$\mathrm{d}x(s) = f(s, u(s), x(s)) \, \mathrm{d}s + \sigma(s, x(s)) \, \mathrm{d}B_s$$

$$x(0) = x^0, \quad \mathbb{E}[\, g(x(t_f)) \,] = 0$$

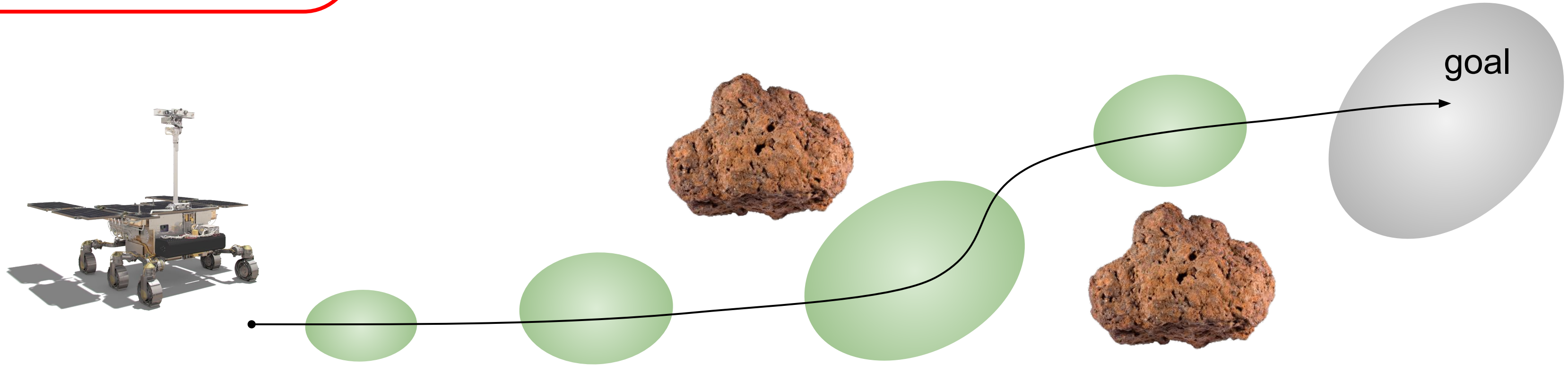
$$\mathbb{E}[\, h(x(s)) \,] \leq 0, \quad s \in [0, t_f]$$

← Cost

← Dynamics

← Initial / Final Conditions

← State Constraints



Several approaches have already been proposed

Original LQR papers

Linear systems:

- W. M. Wonham, *On a matrix Riccati equation of stochastic control*, SIAM J. Control, 6(4): 681-697, 1968.
- U. G. Haussmann, *Optimal stationary control with state and control dependent noise*, SIAM J. Control, 9(2): 184-198, 1971.
- **J.M. Bismut, *Linear quad* *ndom coefficients*, SIAM J. Control, 14(3): 419-444**
- S. Chen, X. Li, and X. Z *ndefinite control weight costs*, SIAM J. Control, 36(5):
- T.E. Duncan, B. Pasik-Du *n state dependent fractional brownian noise and stochastic* 60(2): 199-202, 2017.

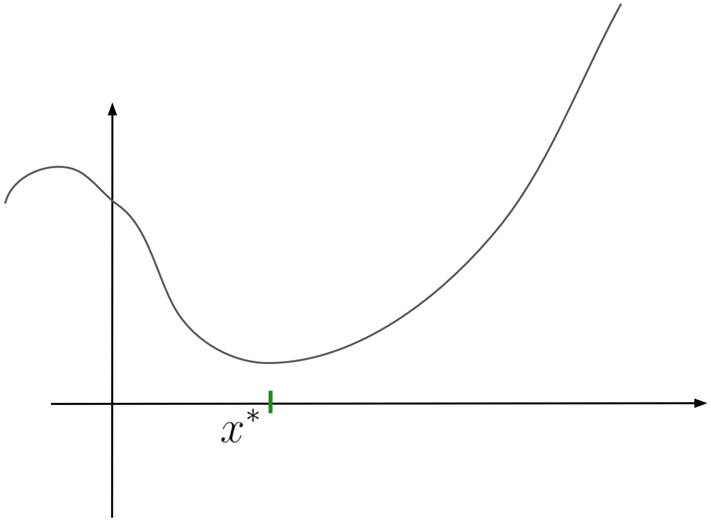
Remark

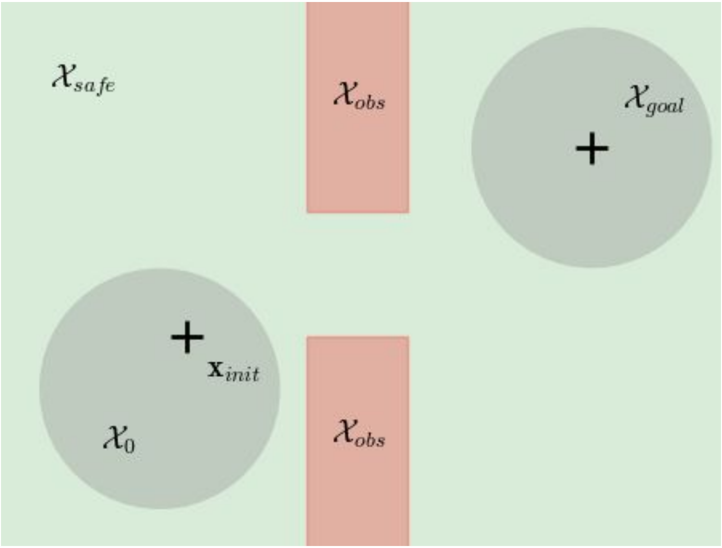
We might somehow be able to leverage the existing works on linear systems: **Sequential Convex Programming (SCP)**

Nonlinear systems:

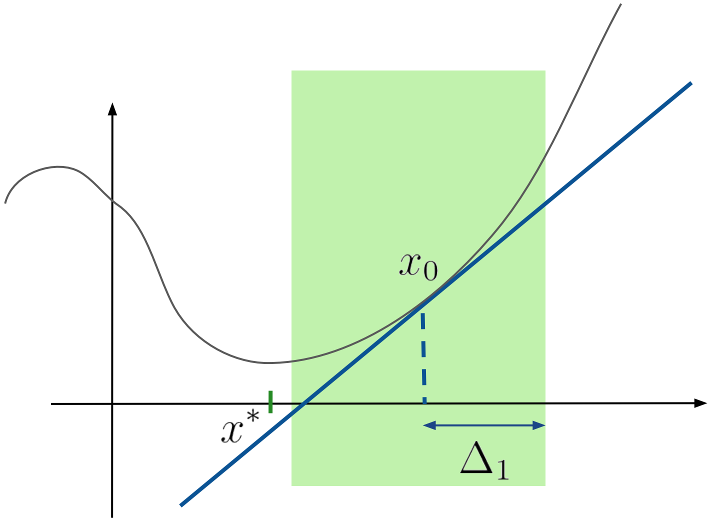
- A. Meshab, S. Streif, R. Findeisen, R.D. Braatz, *Stochastic nonlinear model predictive control with probabilistic constraints*, American Control Conference, 2014, Portland (Oregon).
- S. Satoh, H.J. Kappen, M. Saeki, *An iterative method for nonlinear stochastic optimal control based on path integrals*, IEEE Transactions on Automatic Control, 62(1): 262-276, 2017.

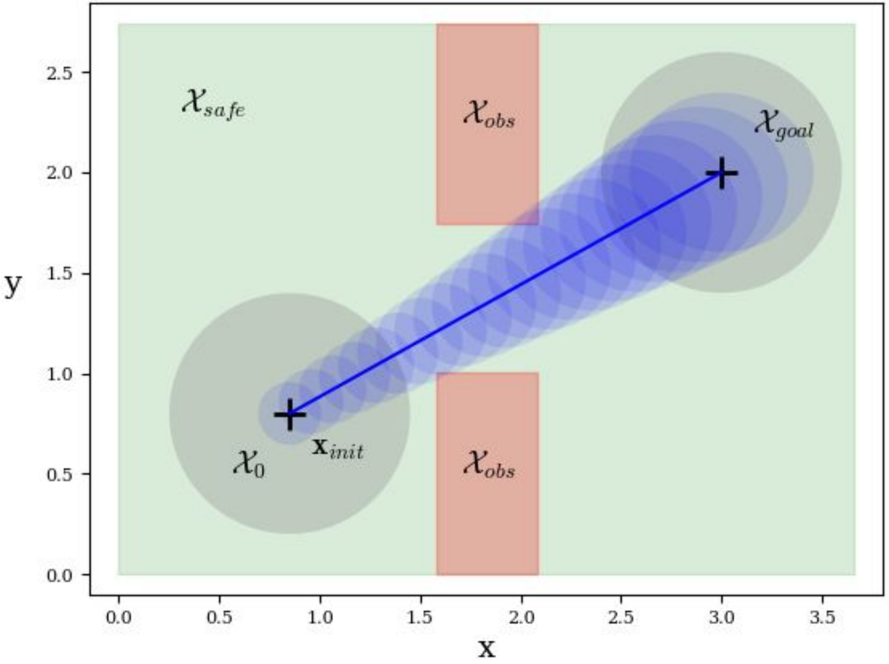
Intuitive introduction to SCP





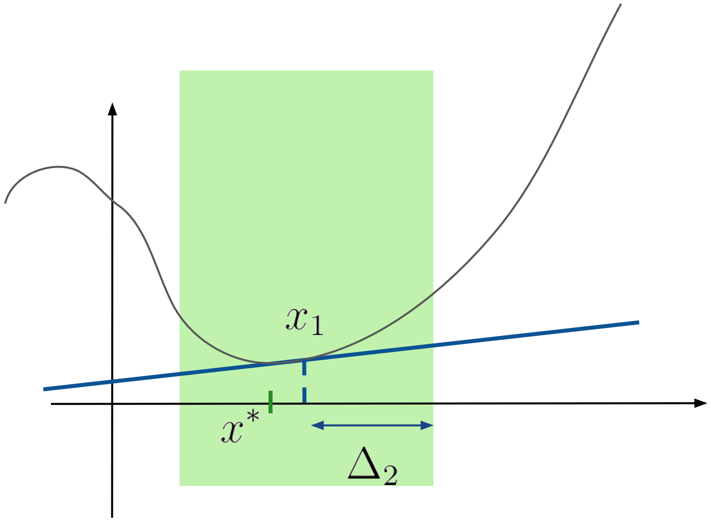
Intuitive introduction to SCP

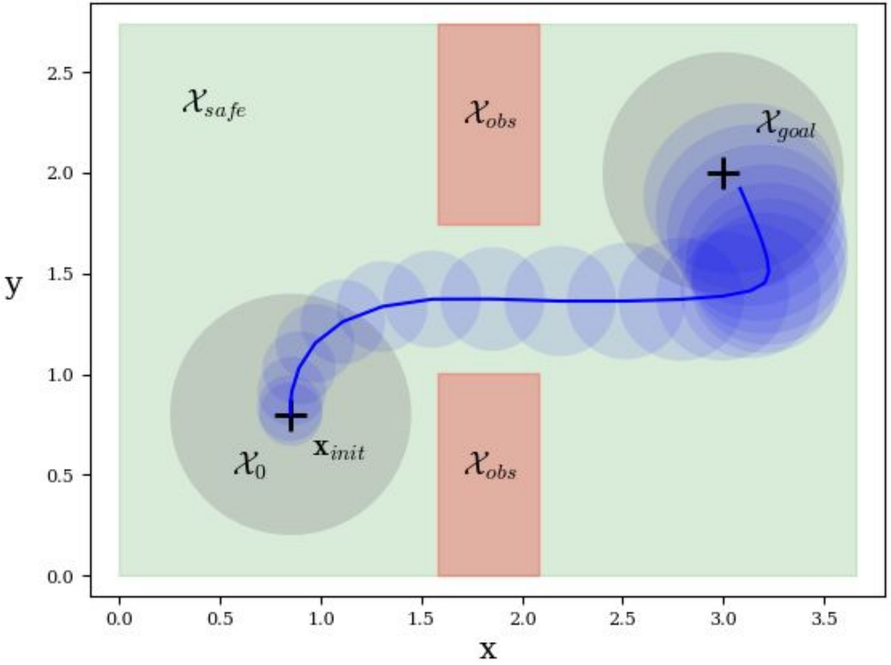




Intuitive introduction to SCP

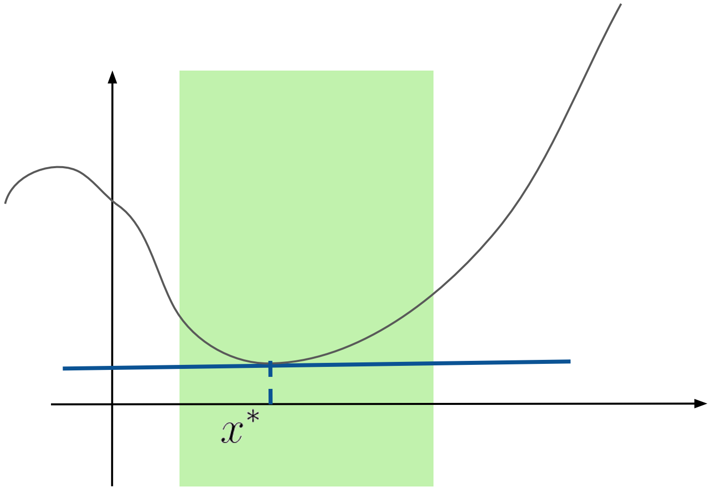
A few iterations later...



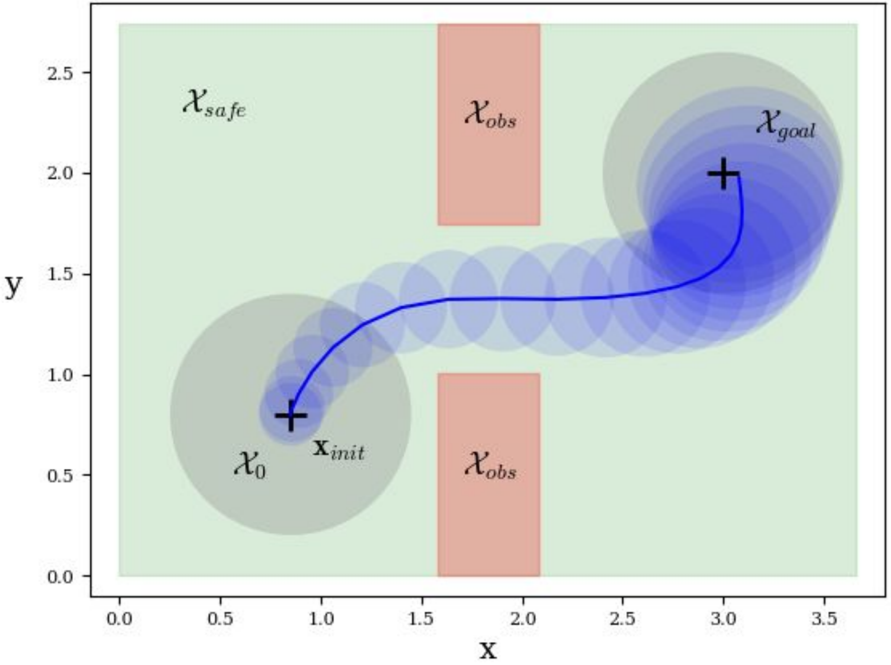


Intuitive introduction to SCP

A few iterations later...



Convergence!



Stochastic SCP formulation

$$
(\mathbf{OCP}) \quad
\begin{aligned}
&\min_{u \in \mathcal{U}} \quad \mathbb{E}\left[\int_0^{t_f} u(s)^2 + h(x(s)) \, \mathrm{d}s\right] \\
&\mathrm{d}x(s) = b(x(s), u(s)) \, \mathrm{d}s + \sigma(x(s)) \, \mathrm{d}B_s \\
&\qquad\quad \triangleq \big(f_0(x(s)) + u(s) f_1(x(s))\big) \, \mathrm{d}s + \sigma(x(s)) \, \mathrm{d}B_s \\
&x(0) = x^0, \quad \mathbb{E}\left[g(x(t_f))\right] = 0
\end{aligned}
$$

$$\Downarrow$$

$$
(\mathbf{COCP})_{k+1} \quad
\begin{aligned}
&\min_{u \in \mathcal{U}} \quad \mathbb{E}\left[\int_0^{t_f} u(s)^2 + h(x_k(s)) + \frac{\partial h}{\partial x}(x_k(s))(x(s) - x_k(s)) \, \mathrm{d}s\right] \\
&\mathrm{d}x(s) = \left(b(x_k(s), u(s)) + \frac{\partial b}{\partial x}(x_k(s), u_k(s))(x(s) - x_k(s))\right) \, \mathrm{d}s \\
&\qquad + \left(\sigma(x_k(s)) + \frac{\partial \sigma}{\partial x}(x_k(s))(x(s) - x_k(s))\right) \, \mathrm{d}B_s \\
&x(0) = x^0, \quad \mathbb{E}\left[g(x_k(t_f)) + \frac{\partial g}{\partial x}(x_k(t_f))(x(t_f) - x_k(t_f))\right] = 0
\end{aligned}
$$

Be careful: the linearization makes sense only locally. Add trust-region constraints:

$$\mathbb{E}\left[\int_0^{t_f} \|x(s) - x_k(s)\|^2 \, \mathrm{d}s\right] \leq \Delta_{k+1}$$

$$\Delta_{k+1} \in \mathbb{R}_+, \quad \Delta_{k+1} \longrightarrow 0$$

Are we really solving the original problem?

This begs the question: “Are we doing something meaningful? I.e., when convergence is achieved, what is the quantity we come up with?”

Our answer

Under mild assumptions, SCP finds a local optimum for (OCP), in the sense of the Pontryagin Maximum Principle (PMP)*

*The PMP are necessary conditions for local optimality

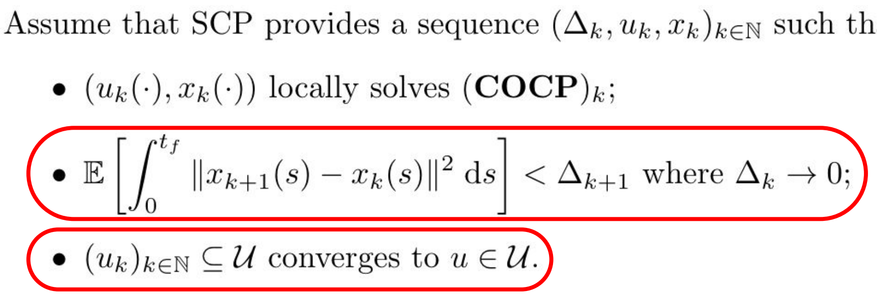
The proof leverage the **continuity** properties of stochastic Itô variational inequalities with respect to **convexification**

The stochastic Pontryagin Maximum Principle

Let $\mathcal{U} = L^2([0, t_f]; U)$ or $\mathcal{U} = L^2([0, t_f] \times \Omega; U)$, where $U \subseteq \mathbb{R}$. For $(\mathbf{OCP})$, we define the Hamiltonian

$$H(x, p, p^0, q, u) = p^\top \big(f_0(x) + u f_1(x)\big) + p^0 \big(u^2 + h(x)\big) \boxed{+\, q^\top \sigma(x).}$$

Theoretical guarantees for stochastic SCP [1]



Assume that SCP provides a sequence $(\Delta_k, u_k, x_k)_{k\in\mathbb{N}}$ such that:

- $(u_k(\cdot), x_k(\cdot))$ locally solves $(\mathbf{COCP})_k$;
- $\mathbb{E}\left[\int_0^{t_f} \|x_{k+1}(s) - x_k(s)\|^2 \, \mathrm{d}s\right] < \Delta_{k+1}$ where $\Delta_k \to 0$;
- $(u_k)_{k\in\mathbb{N}} \subseteq \mathcal{U}$ converges to $u \in \mathcal{U}$.

It may be relaxed by using final constraints

We may adopt weak convergences for deterministic controls

Main result of convergence

Note: this may be leveraged to accelerate SCP!

[1] R. Bonalli, T. Lew and M. Pavone, *Sequential Convex Programming for Non-Linear Stochastic Optimal Control*. Submitted to SIAM Journal on Control and Optimization.

Numerical results

Optimal control of a drifting vehicle

$$\min_{(u_1,u_2)\in L^2([0,t_f];\mathbb{R}^2)} \int_0^{t_f} u_1(s)^2 + u_2(s)^2 \; \mathrm{d}s$$

$$+\,\mathbb{E}\left[\int_0^{t_f} h(x(s),y(s)) \; \mathrm{d}s\right]$$

Collision avoidance constraints penalized within the cost

$$\mathrm{d}x = v\cos\theta \; \mathrm{d}s + 0.1v^2\omega^2 \; \mathrm{d}B^1$$

$$\mathrm{d}y = v\sin\theta \; \mathrm{d}s + 0.1v^2\omega^2 \; \mathrm{d}B^2$$

$$\mathrm{d}\theta = \omega \; \mathrm{d}s + 0.01v^2\omega^2 \; \mathrm{d}B^3$$

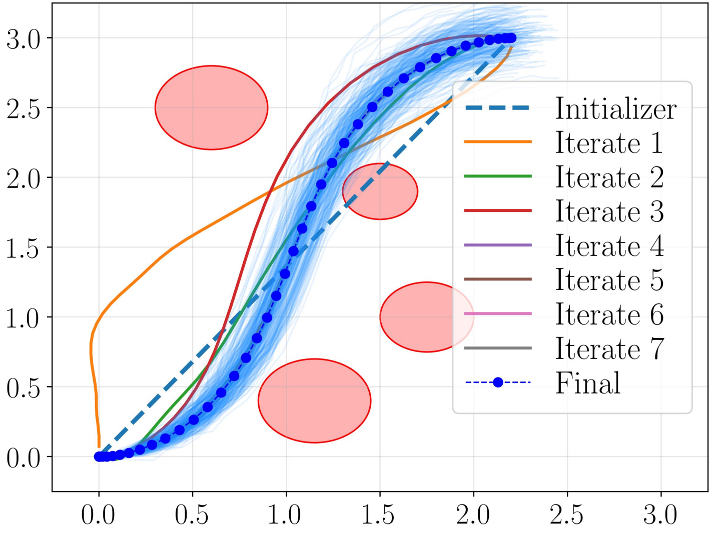
$$\mathrm{d}v = u_1 \; \mathrm{d}s$$

$$\mathrm{d}\omega = u_2 \; \mathrm{d}s$$

Simulating vehicle drifting

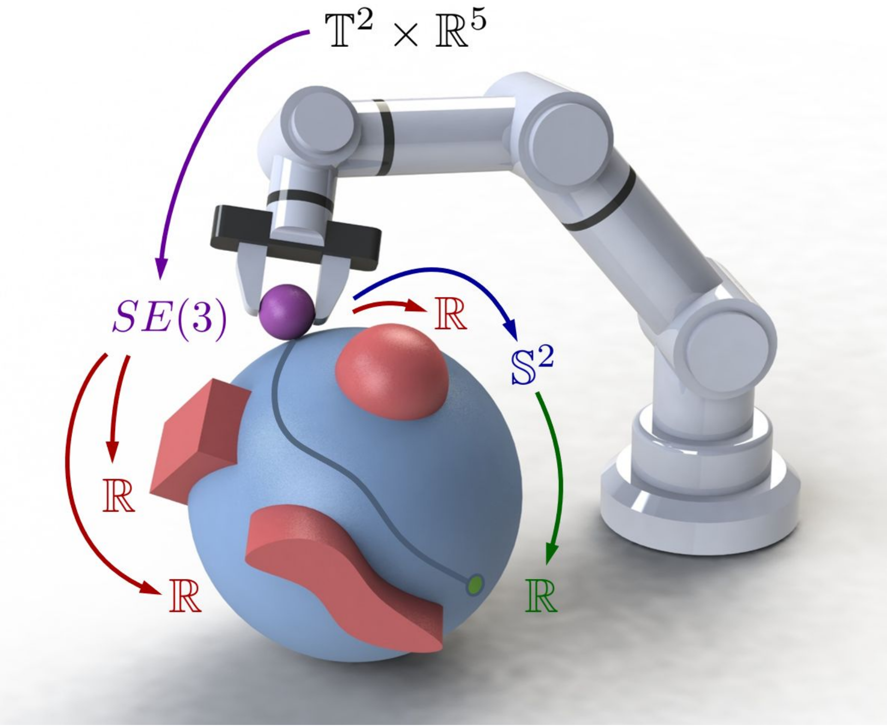
$$(x,y,\theta,v,\omega)(0) = \boldsymbol{x}^0,$$

$$\mathbb{E}[(x,y,\theta,v,\omega)(t_f) - \boldsymbol{x}^f] = 0$$



Second topic

Real-time motion policies for complex dynamical systems through Pullback Bundle Dynamical Systems (PBDSs)



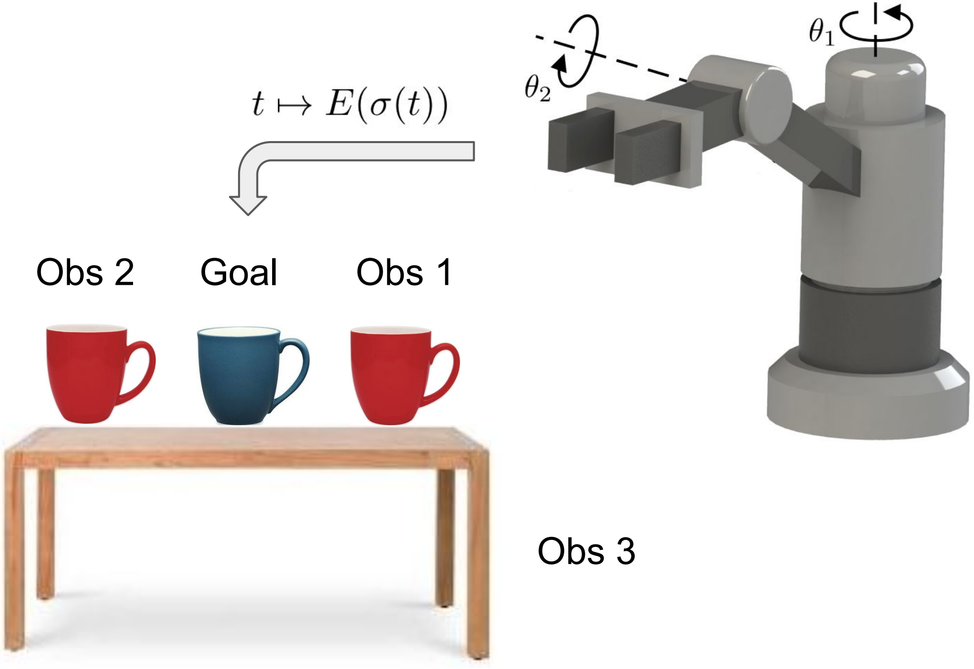
Example PBDS task map tree

Policy synthesis is generally difficult

$$(\theta_1, \theta_2) \in M \triangleq \mathbb{S}^1 \times \mathbb{S}^1$$

Objective: find $\sigma(t) = (\theta_1(t), \theta_2(t))$ such that

$$E(\sigma(t)) \notin \text{Obs } 1\ , \quad E(\sigma(t)) \notin \text{Obs } 2,$$
$$E(\sigma(t)) \notin \text{Obs } 3\ , \quad E(\sigma(t_f)) \in \text{Goal}$$



Proposed solution: plan on task spaces

$$(\theta_1, \theta_2) \in M \triangleq \mathbb{S}^1 \times \mathbb{S}^1$$

$$f_i : M \longrightarrow N_i \triangleq [0, +\infty)$$

$$d_i \triangleq f_i(\theta_1, \theta_2) \in N_i$$

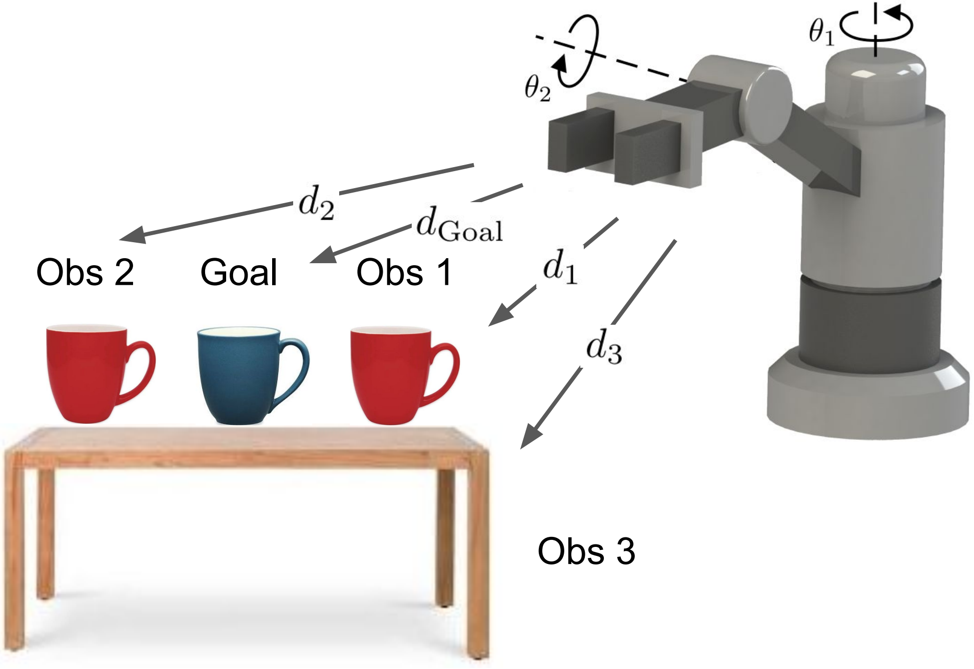
Objective:

1) design $d_i : [0, t_f] \longrightarrow N_i$ such that

$$d_1(t) > 0,\ d_2(t) > 0,\ d_3(t) > 0$$

$$d_4(t_f) = d_{\text{Goal}}(t_f) = 0$$

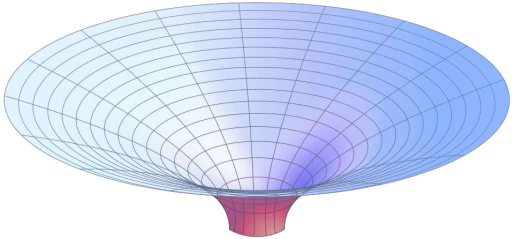
2) find $\sigma(t) = (\theta_1(t), \theta_2(t))$ such that $(f_1, f_2, f_3, f_4)(\sigma(t))$ "resembles" $(d_1, d_2, d_3, d_4)(t)$



Some works along this line

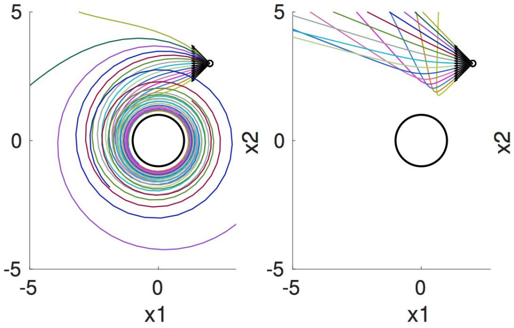
1) Artificial Potential Functions [1,2]

   Computationally efficient, but might get trapped in local minima



2) RMPflow [3]
   (Geometric Dynamical Systems weighted by Riemannian Motion Policies (RMPs))

   Adopting appropriate Riemannian metrics to avoid potentials, but geometrically inconsistent and difficult to tune



[1] O. Khatib, *Real-time obstacle avoidance for manipulators and mobile robots*. In IEEE International Conference on Robotics and Automation, 1985.
[2] H. Lukas, A. Billard, and J.-J. Slotine. *Avoidance of convex and concave obstacles with convergence ensured through contraction*. IEEE Robotics and Automation Letters 4.2 (2019): 1462-1469.
[3] C. Ching-An, M. Mukadam, J. Issac, S. Birchfield, D. Fox, B. Boots, and N.Ratliff. *RMPflow: A computational graph for automatic motion policy generation.* In International Workshop on the Algorithmic Foundations of Robotics, 2018.

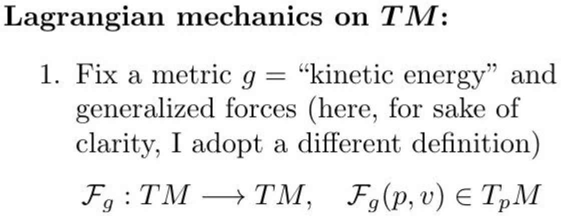
Our objective

Leverage Lagrangian mechanics and bundle theory to devise a local-minima free, computationally efficient, geometrically consistent, and easy-to-use method

How do we do it?

1) Design efficient trajectories on task spaces as solutions to a new class of mechanical systems, i.e., Pullback Bundle Dynamical Systems (PBDSs)
2) Recover a trajectory in the configuration manifold that achieves all the tasks by projecting the accelerations related to each PBDS over appropriate subspaces

How does Lagrangian mechanics work?



Lagrangian mechanics on TM:

1. Fix a metric $g =$ "kinetic energy" and generalized forces (here, for sake of clarity, I adopt a different definition)

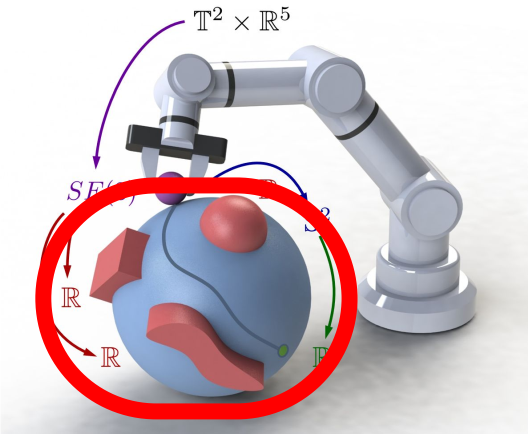
$$\mathcal{F}_g : TM \longrightarrow TM, \quad \mathcal{F}_g(p, v) \in T_pM$$

Building a curve that satisfies all the tasks

1. For the ith task, design a task mapping $f_i : M \longrightarrow N_i$ and equip the task manifold N_i with a Riemannian metric $g_i(z) : T_zN_i \times T_zN_i \longrightarrow \mathbb{R}$, generalized forces $\mathcal{F}_i : TN_i \longrightarrow T^*N_i$, and a cost Riemannian metric $\omega_i(z,w) : T_{(z,w)}TN_i \times T_{(z,w)}TN_i \longrightarrow \mathbb{R}$.

Practical example

Planning on the unitary sphere: reaching a point while avoiding any collision



1) Configuration manifold

$$M = \mathbb{S}^2 \triangleq \{p \in \mathbb{R}^3 : \ \|p\| = 1\}$$

2) Task 1: goal attraction

$$f_1 : M \longrightarrow N_1 = \mathbb{R} : p \mapsto q_1 = \|p - p_{\text{goal}}\|^2, \quad \Phi_1(q_1) = q_1^2$$

3) Task 2: collision avoidance (one for each obstacle)

$$f_2 : M \longrightarrow N_2 = \mathbb{R} : p \mapsto q_2 = \min_{x \in \text{obs}} \|p - x\|^2$$

$$g_2(q_2) = \exp\left(\frac{\alpha}{q_2^2}\right), \quad \omega_2(q_2, v_2) = \begin{cases} 1, & q_2 < d_{\text{unsafe}}, \ v_2 < 0 \\ 0, & \text{otherwise} \end{cases}$$

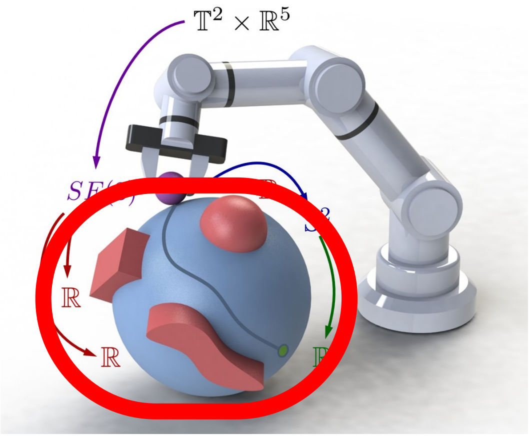
4) Task 3: energy dissipation

$$f_3 : M \longrightarrow M : p \mapsto p, \quad \mathcal{F}_3^D(p) \cdot v = -v$$

$$g_3 = \text{“round metric”}, \quad \omega_3 = \text{“induced metric”}$$

Practical example

Planning on the unitary sphere: reaching a point while avoiding any collision



1) Configuration manifold

$$M = \mathbb{S}^2 \triangleq \{p \in \mathbb{R}^3 : \ \|p\| = 1\}$$

2) Task 1: goal attraction

$$f_1 : M \longrightarrow N_1 = \mathbb{R} : p \mapsto q_1 = \|p - p_{\text{goal}}\|^2, \quad \Phi_1(q_1) = q_1^2$$

3) Task 2: collision avoidance (one for each obstacle)

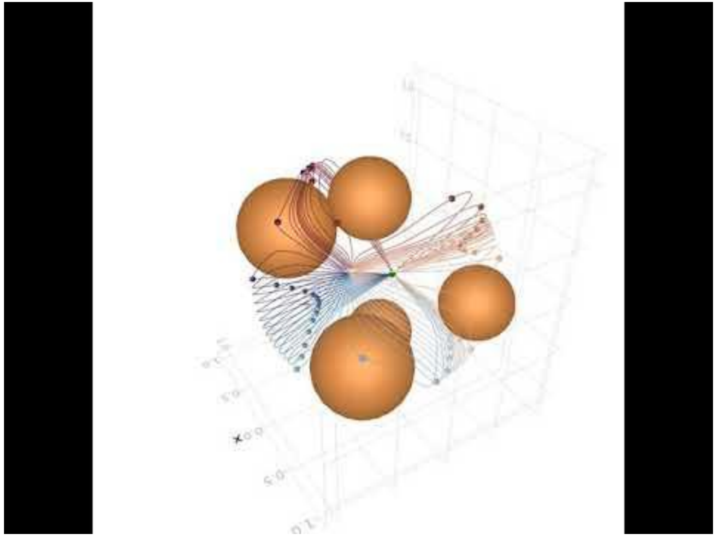
$$f_2 : M \longrightarrow N_2 = \mathbb{R} : p \mapsto q_2 = \min_{x \in \text{obs}} \|p - x\|^2$$

$$g_2(q_2) = \exp\left(\frac{\alpha}{q_2^2}\right), \quad \omega_2(q_2, v_2) = \begin{cases} 1, & q_2 < d_{\text{unsafe}}, \ v_2 < 0 \\ 0, & \text{otherwise} \end{cases}$$

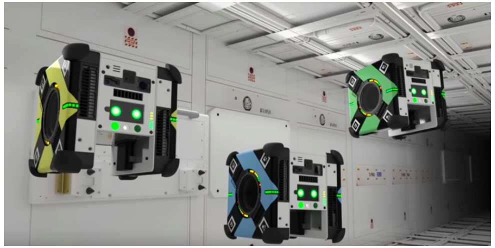
4) Task 3: energy dissipation

$$f_3 : M \longrightarrow M : p \mapsto p, \quad \mathcal{F}_3^D(p) \cdot v = -v$$

$$g_3 = \text{“round metric”}, \quad \omega_3 = \text{“induced metric”}$$



Future directions



SCP

- Extend to more general frameworks, e.g., considering risk measures and scenario optimization
- Analysis of the convergence for the training of deep neural networks
- Leverage this framework to deal wit linear partial differential equations

PBDS

- Extend the for
- Extend the fo e represented

Thank you for your attention!
Any question?

Collaborators

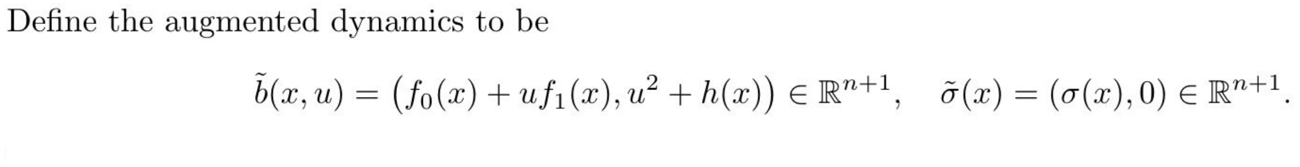
M. Pavone

T. Lew

A. Bylard

Detailed approaches

Convergence of SCP - sketch of proof



Define the augmented dynamics to be

$$\tilde{b}(x,u) = \big(f_0(x) + u f_1(x), u^2 + h(x)\big) \in \mathbb{R}^{n+1}, \quad \tilde{\sigma}(x) = (\sigma(x), 0) \in \mathbb{R}^{n+1}.$$

Lebesgue points for stochastic controls are correctly introduced via Bochner integration

How do we make it work in practice?

Some numerical strategies

- For every fixed realization of the Brownian motion, solve a deterministic convex problem (expensive)
- For specific costs, solve a sequence of LQR problems with stochastic coefficients (hard problem)
- We propose another procedure that leverages the structure and the results entailed by SCP

Some simplifying assumptions

- Controls are deterministic
- The cost can be written as function of the mean of stochastic trajectories
- The method is such that, at each iteration, the optimal trajectory has “small variance” at each time
- The dynamic takes the following form:

$$\begin{cases} \mathrm{d}x(s) = b(x(s), u(s))\ \mathrm{d}s + \boxed{y(s)\mathrm{d}B_s} \\ \mathrm{d}y(s) = c(y(s), u(s))\ \mathrm{d}s, \quad c \text{ is affine in } u \end{cases}$$

Also “functions” of the variable y may be considered

Deterministic reformulation 1

$$\begin{cases} \mathrm{d}x(s) = \left(b(x_k(s), u(s)) + \dfrac{\partial b}{\partial x}(x_k(s), u_k(s))(x(s) - x_k(s)) \right) \mathrm{d}s + y(s)\mathrm{d}B \\ \mathrm{d}y(s) = \left(c(y_k(s), u(s)) + \dfrac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s)) \right) \mathrm{d}s \end{cases}$$

The process is Gaussian and may be characterized by its mean and covariance!

$$\begin{cases} \mathrm{d}x(s) \approx \Big(M_k(s)x(s) + d_k(s, u(s)) \Big) \mathrm{d}s + y(s)\mathrm{d}B_s \\ \qquad \triangleq \boxed{\left(\dfrac{\partial b}{\partial x}(\mathbb{E}[x_k(s)], u_k(s))x(s) + \left(b(\mathbb{E}[x_k(s)], u(s)) - \dfrac{\partial b}{\partial x}(\mathbb{E}[x_k(s)], u_k(s))\mathbb{E}[x_k(s)] \right) \right)} \mathrm{d}s + y(s)\mathrm{d}B_s \\ \mathrm{d}y(s) = \left(c(y_k(s), u(s)) + \dfrac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s)) \right) \mathrm{d}s \end{cases}$$

$$\begin{cases} \dot{m}(t) = M_k(s)m(s) + d_k(s, u(s)) \\ \boxed{\dot{\Sigma}(t) = M_k(s)\Sigma(s) + \Sigma(s)M_k(s)^\top + y(s)y(s)^\top} \\ \dot{y}(s) = c(y_k(s), u(s)) + \dfrac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s)) \end{cases}$$

System of ODEs. The variance may be controlled thanks to the variable y, and forced to be small!

Deterministic reformulation 2

$$\begin{cases} \dot{m}(t) = M_k(s)m(s) + d_k(s, u(s)) \\ \dot{\Sigma}(t) = M_k(s)\Sigma(s) + \Sigma(s)M_k(s)^\top \boxed{+\, y(s)y(s)^\top} \\ \dot{y}(s) = c(y_k(s), u(s)) + \dfrac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s)) \end{cases}$$

$$\Downarrow$$

$$\begin{cases} \dot{m}(t) = M_k(s)m(s) + d_k(s, u(s)) \\ \dot{\Sigma}(t) = M_k(s)\Sigma(s) + \Sigma(s)M_k(s)^\top \boxed{+\, y(s)y_k(s)^\top} \\ \dot{y}(s) = c(y_k(s), u(s)) + \dfrac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s)) \end{cases}$$

Presence of a term which is quadratic in y. This prevents from using convex optimization to solve each subproblem

Idea!

Use the convergences necessarily entailed by SCP

Final convex subproblems

$$\min_{u \in \mathcal{U}} \int_0^{t_f} u(s)^2 + h(m_k(s)) + \frac{\partial h}{\partial x}(m_k(s))(m(s) - m_k(s)) + \mathrm{tr}(\Sigma_x(s)) \, \mathrm{d}s$$

The variance is penalized to make the previous approximation well-posed

$$\begin{aligned}
\dot{m}(t) &= M_k(s)m(s) + d_k(s, u(s)) \\
\dot{\Sigma}_x(t) &= M_k(s)\Sigma_x(s) + \Sigma_x(s)M_k(s)^\top + y(s)y_k(s)^\top \\
\dot{y}(s) &= c(y_k(s), u(s)) + \frac{\partial c}{\partial x}(y_k(s), u_k(s))(y(s) - y_k(s))
\end{aligned}$$

Linear dynamics of the mean and the covariance

$$(m, y)(0) = (m^0, y^0), \quad g_k(t_f, m(t_f), y(t_f)) = 0$$

Initial and final conditions as functions of the mean

$$\left(\mathbb{E}\left[\int_0^{t_f} \|x(s) - x_k(s)\|^2 \, \mathrm{d}s \right] \leq \right) 2 \int_0^{t_f} \mathrm{tr}(\Sigma_{x - x_k}(s)) + \|m(s) - m_k(s)\|^2 \, \mathrm{d}s \leq \Delta_{k+1}$$

We bound with the covariance of the “error” trajectory. Its dynamics must be included in the formulation (this is done exactly as before)

Standard tools in Lagrangian mechanics

$M = \text{configuration manifold},\ N_i = \text{i-th task manifold},\ f_i : M \longrightarrow N_i = \text{i-th task mapping}$

We adapt those tools to introduce a new kind of mechanics

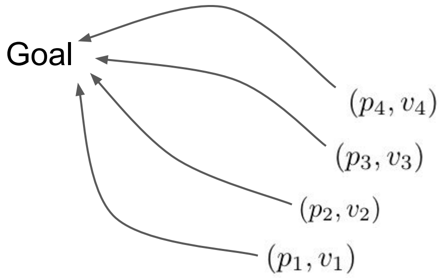
$M = \text{configuration manifold},\ N_i = \text{i-th task manifold},\ f_i : M \longrightarrow N_i = \text{i-th task mapping}$

From Lagrangian mechanical systems to PBDSs

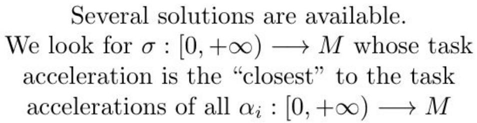
Lagrangian mechanics on TM:

1. Fix a metric $g =$ "kinetic energy" and generalized forces (here, for sake of clarity, I adopt a different definition)

   $$\mathcal{F}_g : TM \longrightarrow TM, \quad \mathcal{F}_g(p, v) \in T_pM$$



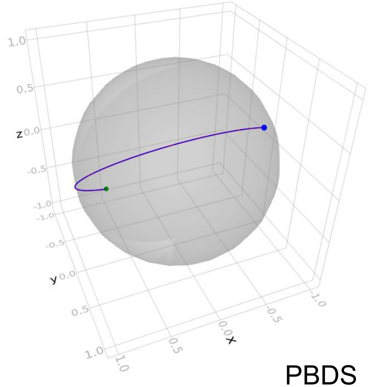
Building a curve that satisfies all the tasks



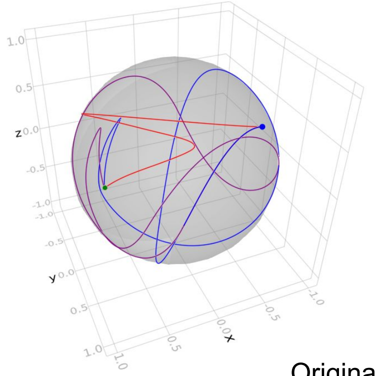
Several solutions are available.
We look for $\sigma : [0, +\infty) \longrightarrow M$ whose task acceleration is the "closest" to the task accelerations of all $\alpha_i : [0, +\infty) \longrightarrow M$

This looks pretty complex, what are the benefits?

1) Geometric well-posedness



PBDS



Original RMPflow

2) Global, geometrically consistent stability

When $\mathcal{F}_{g_i} = \mathcal{F}^D_{g_i} - \operatorname{grad}_{g_i} \Phi_i$ with $\mathcal{F}^D_{g_i}$ = "dissipative forces" and Φ_i = "potential", under appropriate assumptions we can invoke LaSalle principle to prove global stability through the following Lyapunov function (here, $f_i^* g_i$ = "pullback metric" and $f_i^* \Phi_i$ = "pullback potential"):

$$V : TM \longrightarrow \mathbb{R} : (p, v) \mapsto \frac{1}{2} \sum_{i=1}^{n_{\text{task}}} f_i^* g_i(p) \cdot (v, v) + \sum_{i=1}^{n_{\text{task}}} f_i^* \Phi_i(p)$$

3) Computational efficiency and ease-of-use

Let's see how to set things up on a practical example!