Inria TAU Seminar

Learning-based Control on Dynamical Systems

R. Hosseinkhan Boucher[1], S. Douka[1], O. Semeraro[1], L. Mathelin[1]

Laboratoire Interdisciplinaire des Sciences du Numérique, Université Paris-Saclay, CNRS

Doctoral School: *Sciences et technologies de l'information et de la communication (STIC)*

Granted by the Agence Nationale de la Recherche (ANR) under projet ANR-21-CE46-0008 Reinforcement Learning as Optimal control for Shear Flows (REASON)

Dynamical Systems Control: Challenges

Challenges in Dynamical Systems Control

Optimal Control Problem

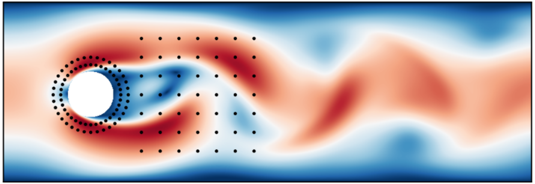
Dynamics: $\partial_t x\,(z,t) = P\,[x,u]\,(z,t)$

Objective: $\min_u J\,(u) = \int_0^T c(x(t),u(t))dt$

Example

P is the Navier-Stokes operator

Energy criterion: $c(x,u) = \|x\|^2 + \|u\|^2$



Cylinder flow drag reduction. Partial observation through sensors.

Challenges[1]

- Partial observability (PO) and delays
- Controllability
- Sampling complexity
- Robustness
- High dimensional hidden state space $\mathcal{X}$
- Extremely large degrees of freedom (sensor placement, actuators, amplitude, optimization problem). No benchmark

Rigorously

- Control problem with **continuous** time and **infinite** state space (Relaxed Stochastic Control)

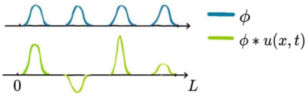
[1] J. Viquerat et al. "A review on deep reinforcement learning for fluid mechanics: An update", *AIP Publishing* (2022)

Controlled Kuramoto-Sivashinksy (KS)[1,2]

Controlled KS: $\partial_t x(z,t) + x(z,t)\,\partial_x x(z,t) = -\partial_x^2 x(z,t) - \partial_x^4 x(z,t) + \langle \phi, \boldsymbol{u} \rangle (z,t)$

$x(z+L,t) = x(z,t)$ and $(z,t) \in [0,L] \times [0,T]$

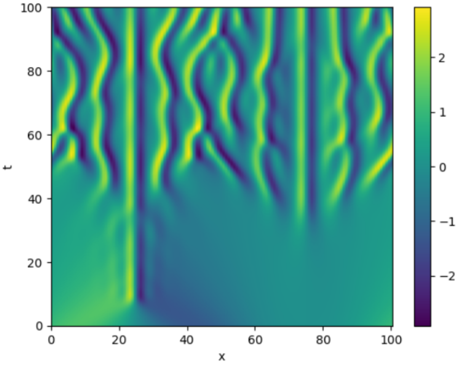
Control term: $\langle \phi, \boldsymbol{u} \rangle = \sum_{i=1}^{r} u_i f_{\mathcal{N}(\mu_i,\sigma^2)}$



ϕ **define** a given gaussian mixture, $\boldsymbol{u}$ is **unknown**

Properties

- Spatio-temporal chaos, 4th order non-linear
- Equilibria, relative equilibria, symmetries
- 4 equilibria $x_e^0(z) = 0$, $x_e^1(z)$, $x_e^2(z)$, $x_e^3(z)$



Evolution of the Kuramoto-Sivashinsky equation with $L = 100$

[1] Y. Kuramoto. "Diffusion-Induuced Chaos in Reaction Systems", *Progress of Theoretical Physics Supplement* (1978)

[2] G.I. Sivashinsky. "Nonlinear analysis of hydrodynamic instability in laminar flames—I. Derivation of basic equations", *Acta Astronautica* (1977)

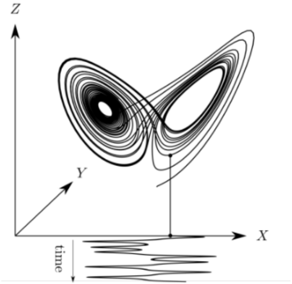
Controlled Lorenz[1]

Controlled Lorenz: $$\begin{cases} \partial_t x_1 = \sigma(x_2 - x_1) + \boldsymbol{u_1} \\ \partial_t x_2 = x_1(\rho - x_3) - x_2 + \boldsymbol{u_2} \\ \partial_t x_3 = x_1 x_2 - \beta x_3 + \boldsymbol{u_3} \end{cases}$$

Control Term: $\boldsymbol{u} = (\boldsymbol{u_1}, \boldsymbol{u_2}, \boldsymbol{u_3})$

Properties

- Chaos, instabilities, symmetries
- Equilibria x_e^0, x_e^1, x_e^2



Attracteur de Lorenz

[1] T. L. Vincent, J. Yu. "Control of a chaotic system", *Dynamics and Control* (1991)

Partially Observable Markov Decision Process (POMDP)

Dynamics

$\partial_t x(z,t) = P\left[x,u\right](z,t), \qquad x(\cdot,t) \in \mathbb{L}^2(\mathcal{X})$ and $u(\cdot,t) \in \mathbb{L}^2(\mathcal{U})$ for any $t \in [0,T]$

Spatial Discretisation

$\mathbb{L}^2(\mathcal{X}) \simeq \mathcal{X}^{d_\mathcal{X}} \qquad \mathbb{L}^2(\mathcal{U}) \simeq \mathcal{U}^{d_\mathcal{U}}$

Temporal Discretisation

$[0,T] \simeq (k\delta)_{0\leq k\leq n}$

Continuous operator $\longrightarrow$ Discrete[1] operator: $x_{k+1} = P(x_k, u_k)$, $x_k \in \mathcal{X}^{d_\mathcal{X}}$, $u_k \in \mathcal{U}^{d_\mathcal{U}}$

[1] **The same notations** (operator, time horizon etc.) as the **continuous time** framework will be used for the **discrete time** framework.

Partially Observable Markov Decision Process (POMDP)

Dynamics

$\partial_t x(z,t) = P\left[x,u\right](z,t), \qquad x(\cdot,t) \in \mathbb{L}^2(\mathcal{X})$ and $u(\cdot,t) \in \mathbb{L}^2(\mathcal{U})$ for any $t \in [0,T]$

Spatial Discretisation

$\mathbb{L}^2(\mathcal{X}) \simeq \mathcal{X}^{d_\mathcal{X}} \qquad \mathbb{L}^2(\mathcal{U}) \simeq \mathcal{U}^{d_\mathcal{U}}$

Temporal Discretisation

$[0,T] \simeq (k\delta)_{0\leq k \leq n}$

Continuous operator $\longrightarrow$ Discrete[1] operator: $x_{k+1} = P(x_k, u_k)$, $x_k \in \mathcal{X}^{d_\mathcal{X}}$, $u_k \in \mathcal{U}^{d_\mathcal{U}}$

Generalisation: Partially Observable Markov Decision Process (POMDP)

$$
\begin{aligned}
X_{k+1} &= P\left(X_k, U_k, \eta_k\right) \qquad && \eta_k \sim \mathcal{N}(0, \sigma_\eta^2 I_d) \\
Y_{k+1} &= Q\left(X_k\right) + \epsilon_k \qquad && \epsilon_k \sim \mathcal{N}(0, \sigma_\epsilon^2 I_d)
\end{aligned} \tag{1}
$$

with $X_0 \sim \mathcal{N}\left(x_e, \sigma_e^2 I_d\right)$.

[1] **The same notations** (operator, time horizon etc.) as the **continuous time** framework will be used for the **discrete time** framework.

Modeling as a Markov Decision Process (MDP)

State space $\mathcal{X}$, control space $\mathcal{U}$, observation space $\mathcal{Y}$

Random Dynamics

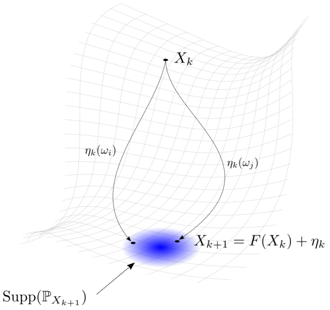
$\mathcal{P}\left(dx_{k+1} \mid (x_k, u_k)\right) \rightarrow$ probability on $\mathcal{X}$ given $(x_k, u_k) \in \mathcal{X} \times \mathcal{U}$

Random Observation

$\mathcal{Q}\left(dy_k \mid x_k\right) \rightarrow$ probability on $\mathcal{Y}$ given $x_k \in \mathcal{X}$

Random Control

$\pi(du_k \mid y_k) \rightarrow$ probability on $\mathcal{U}$ given $y_k \in \mathcal{Y}$



Transition Kernel $\mathcal{P}$

Modeling as a Markov Decision Process (MDP)

State space $\mathcal{X}$, control space $\mathcal{U}$, observation space $\mathcal{Y}$

Random Dynamics

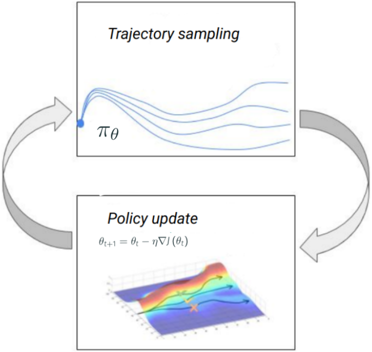
$\mathcal{P}\left(dx_{k+1} \mid (x_k, u_k)\right) \rightarrow$ probability on $\mathcal{X}$ given $(x_k, u_k) \in \mathcal{X} \times \mathcal{U}$

Random Observation

$\mathcal{Q}\left(dy_k \mid x_k\right) \rightarrow$ probability on $\mathcal{Y}$ given $x_k \in \mathcal{X}$

Random Control

$\pi(du_k \mid y_k) \rightarrow$ probability on $\mathcal{U}$ given $y_k \in \mathcal{Y}$



Policy gradient iterations to solve $\arg\min_\pi \mathbb{E}^\pi[\sum_{k=0}^{T} c(X_k, U_k)]$

Controlled Hidden Markov Chain

$$P^\pi\left(dx_0 du_0 dy_0 dx_1 du_1 \ldots dx_T\right) = P_{X_0}\left(dx_0\right) \mathcal{Q}\left(dy_0 \mid x_0\right) \pi\left(du_0 \mid y_0\right) \mathcal{P}\left(dx_1 \mid x_0, u_0\right)$$
$$\mathcal{Q}\left(dy_1 \mid x_1\right) \pi\left(du_1 \mid y_1\right) \cdots \pi\left(du_{T-1} \mid y_{T-1}\right) \mathcal{P}\left(dx_T \mid x_{T-1}, u_{T-1}\right)$$

Maximum Entropy: Noise Robustness

Robustness: Maximum Entropy and Flat Minima

Maximum Entropy in Reinforcement Learning

$$\arg\min_\pi \mathbb{E}^\pi \left[\sum_{k=0}^{T} \gamma^k \|X_k\|^2 - \alpha \mathcal{H}(\pi(du \mid X_k))\right], \quad \alpha > 0, \quad \mathcal{H} : \text{entropy}$$

Observations

- **Better exploration**
- **Robustness**
- **Flat minima and optimisation regularity** (recent work: Ahmed et al. ICLR (2019)[1])

[1] A. Ahmed et al. "Understanding Flat Minima in Neural Networks", *ICLR* (2019)

Robustness: Maximum Entropy and Flat Minima

Maximum Entropy in Reinforcement Learning

$$\arg\min_\pi \ \mathbb{E}^\pi \left[\sum_{k=0}^{T} \gamma^k \|X_k\|^2 - \alpha \mathcal{H}(\pi(du \mid X_k))\right], \quad \alpha > 0, \quad \mathcal{H} : \text{entropy}$$

Observations

- **Better exploration**
- **Robustness**
- **Flat minima and optimisation regularity** (recent work: Ahmed et al. ICLR (2019)[1])

Questions:

Why does entropy improve robustness? Why does entropy regularise the optimisation landscape?

Objective

Understanding robustness-entropy-regularity synergy

Hypothesis

Entropy $\longrightarrow$ Policy Complexity

[1] A. Ahmed et al. "Understanding Flat Minima in Neural Networks", *ICLR* (2019)

Excess Risk Under Noise

Partial Observability

$$\begin{aligned} X_{k+1} &= P\left(X_k, U_k, \eta_k\right) \\ Y_{k+1} &= Q\left(X_k\right) + \epsilon_k \qquad \epsilon_k \sim \mathcal{N}(0, \sigma_\epsilon^2 I_d) \end{aligned} \tag{2}$$

Notation

When $\epsilon \equiv 0 \longrightarrow P^{\pi}$

When $\epsilon \not\equiv 0 \longrightarrow P^{\pi,\epsilon}$

Excess Risk Under Noise

Partial Observability

$$
\begin{aligned}
X_{k+1} &= P\left(X_k, U_k, \eta_k\right) \\
Y_{k+1} &= Q\left(X_k\right) + \epsilon_k \qquad\qquad \epsilon_k \sim \mathcal{N}(0, \sigma_\epsilon^2 I_d)
\end{aligned} \tag{2}
$$

Notation

When $\epsilon \equiv 0 \longrightarrow P^\pi$

When $\epsilon \not\equiv 0 \longrightarrow P^{\pi,\epsilon}$

Rate of Excess Risk Under Noise

$$
\mathring{\mathcal{R}}^\pi = \frac{J^{\pi,\epsilon} - J^\pi}{J^\pi} \tag{3}
$$

with $J^{\pi,\epsilon} = \mathbb{E}^{\pi,\epsilon}\left[\sum_{k=0}^{T} \gamma^k \left\|X_k\right\|^2\right]$

Training with different temperature levels α

Objective

$$\pi_\alpha^* = \arg\min_\pi \mathbb{E}^\pi \left[\sum_{k=0}^T \gamma^k \|X_k\|^2 - \alpha \mathcal{H}(\pi(du \mid X_k))\right], \quad \alpha > 0$$

Initial condition $X_0 \sim \mathcal{N}(x_e^2, \sigma^2)$ and $\pi_\theta(\cdot|X_k) \sim \mathcal{N}_{d_\mathcal{U}}(\mu_\theta(X_k), \theta_{\sigma_\pi} I_{d_\mathcal{U}})$

Goal control $x_k \longrightarrow x_e^0 = 0$

Training with different temperature levels α

Objective

$$\pi^*_\alpha = \arg\min_\pi \mathbb{E}^\pi \left[\sum_{k=0}^{T} \gamma^k \|X_k\|^2 - \alpha \mathcal{H}(\pi(du \mid X_k))\right], \quad \alpha > 0$$

Initial condition $X_0 \sim \mathcal{N}(x_e^2, \sigma^2)$ and $\pi_\theta(\cdot|X_k) \sim \mathcal{N}_{d_\mathcal{U}}(\mu_\theta(X_k), \theta_{\sigma_\pi} I_{d_\mathcal{U}})$

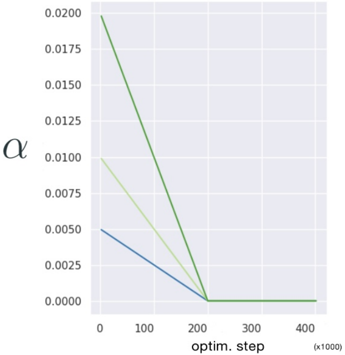
Goal control $x_k \longrightarrow x_e^0 = 0$

Hypothesis

With $\alpha > 0$ the policies π^*_α are more robust than $\pi^*_{\alpha=0}$

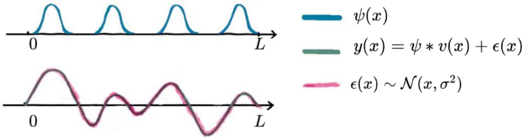
Experimental Plan

- Fix 5 **entropy levels** α
- 10 trainings for each α for 2m of iterations with the system
- α decreases linearly
- Study of the regularity of π^*_α and its robustness



α decreases linearly during optim

Evaluation of the policy with noisy observation



Noisy observable

Hypothesis

$\epsilon \nearrow \quad \longrightarrow \quad J^{\pi^*,\epsilon} \nearrow$ (noise impacts perf)

$\alpha > 0 \quad \longrightarrow \quad \mathring{\mathcal{R}}^{\pi,\alpha} \searrow$ (robustness)

Experimental Plan

- *Test* π^*_α with *different noise levels* ϵ on Y
- Compare $J^{\pi^*,\epsilon}$ according to J^{π^*} *i.e.* $\mathring{\mathcal{R}}^{\pi} = \frac{J^{\pi^*,\epsilon} - J^{\pi^*}}{J^{\pi^*}}$

with $J^{\pi^*} = \mathbb{E}^{\pi^*}\left[\sum_{k=0}^{T} \gamma^k \|X_k\|^2\right]$

and $J^{\pi^*,\epsilon}$ same quantity evaluated with noisy observables

:

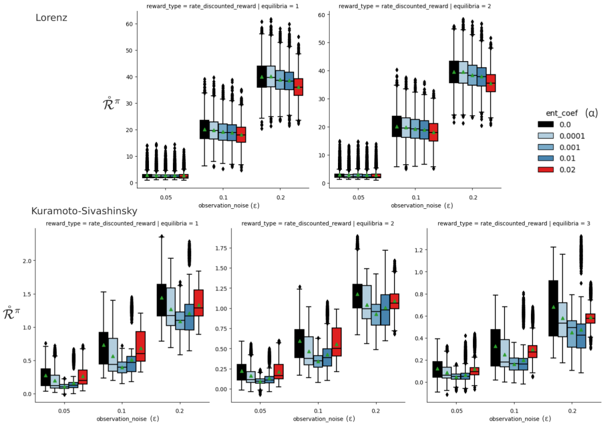
Observation noise robustness by Maximum Entropy

Experiment

- Evaluate 10 models θ^*_α for each value of α
- **Total :** 50 models $\theta^*_{\alpha_i}$
- $\forall \theta^*_{\alpha_i}$ evaluate 200 trajectories until T

Results

- Noise ϵ **increases** globally the cost J^{π^*}
- **KS and Lorenz**: $\alpha = 0$ noise sensitive
- **KS**: $\alpha_{\max}$ noise sensitive



Variation $\frac{J^{\pi^*_\alpha,\epsilon} - J^{\pi^*}}{J^{\pi^*}}$. Each **bar block** : noise intensity ϵ.
Colors: $\alpha = 0$ (black), $\alpha > 0$ (blue), $\alpha_{\max}$ (red)

Complexity measures[1]

Complexity Measure

$\mathcal{M}: \pi \in \Pi \rightarrow \mathbb{R}_+$

$\mathcal{M}(\pi)$ measures the **complexity** of the model π

Robustness Measure

$\mathring{\mathcal{R}}^\pi \leq f(\mathcal{M}(\pi))$

where f is an increasing function

Objective

Identify proper complexity measures for robustness

[1] B. Neyshabur et al. "Exploring Generalization in Deep Learning" *NIPS* (2017)

Complexity Measure: Lipschitz Upper Bound

Lipshitz Bound

$\pi_\theta(\cdot|X_k) \sim \mathcal{N}_{d_\mathcal{U}}\left(\mu_\theta(X_k), \theta_{\sigma_\pi} I_{d_\mathcal{U}}\right)$

If $\mu_\theta\left(x\right) = \left(\sigma_l \circ \sigma_{l-1} \circ \ldots \circ \sigma_1\right)\left(x\right)$,

$Lips(\mu_\theta) \leq \prod_{i=1}^{l} Lips\left(\sigma_i\right) = \prod_{i=1}^{l} \|\theta_i\|$,

where θ_i weight matrix i.

Complexity Measure: Lipschitz Upper Bound

Lipshitz Bound

$\pi_\theta(\cdot|X_k) \sim \mathcal{N}_{d_\mathcal{U}}\left(\mu_\theta(X_k),\, \theta_{\sigma_\pi} I_{d_\mathcal{U}}\right)$

If $\mu_\theta\,(x) = (\sigma_l \circ \sigma_{l-1} \circ \ldots \circ \sigma_1)\,(x)$,

$Lips(\mu_\theta) \leq \prod_{i=1}^{l} Lips\,(\sigma_i) = \prod_{i=1}^{l} \|\theta_i\|$,

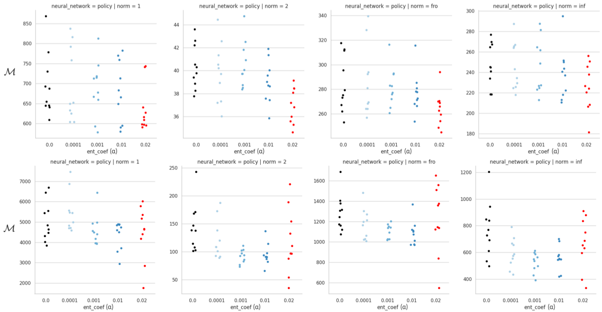
where θ_i weight matrix i.

Lipshitz Complexity Measure

- $\mathcal{M}\,(\pi_\theta) = \prod_{i=1}^{l} \|\theta_i\|$

Result

Low $\mathcal{M}(\pi_\theta^\alpha)$ corresponds to low $\mathring{\mathcal{R}}^\pi$



$\mathcal{M}\,(\pi_\theta^\alpha) = \prod_{i=1}^{l} \|\theta_i^\alpha\|$

Colors: $\alpha = 0$, $\alpha > 0$, $\alpha_{\max}$

Top: Lorenz, Bottom: KS

Complexity Measure: Conditional Fisher Information Trace

Hessian and Fisher Information

$\alpha > 0 \longrightarrow$ Flat Minima (already observed)[1]

$$\nabla_\theta^2 J^{\pi_\theta} = \mathbb{E}^{\pi_\theta}\left[\sum_{h,i,j=0}^{T} c\left(X_h, U_h\right)\left(\nabla_\theta \log \pi_\theta\left(U_i \mid X_j\right) \nabla_\theta \log \pi_\theta\left(U_j \mid X_j\right)^T + \nabla_\theta^2\left[\log \pi_\theta\left(U_i \mid X_j\right)\right]\right)\right]$$

Fisher Information: $\mathcal{I}(\theta) = -\mathbb{E}^{X\sim\rho, U\sim\pi_\theta(\cdot|X)}\left[\nabla_\theta^2 \log \pi_\theta\left(U|X\right)\right]$

Complexity Measure: Conditional Fisher Information Trace

Hessian and Fisher Information

$\alpha > 0 \longrightarrow$ Flat Minima (already observed)[1]

$$\nabla_\theta^2 J^{\pi_\theta} = \mathbb{E}^{\pi_\theta}\left[\sum_{h,i,j=0}^{T} c\left(X_h, U_h\right)\left(\nabla_\theta \log \pi_\theta\left(U_i \mid X_j\right) \nabla_\theta \log \pi_\theta\left(U_j \mid X_j\right)^T + \nabla_\theta^2\left[\log \pi_\theta\left(U_i \mid X_j\right)\right]\right)\right]$$

Fisher Information: $\mathcal{I}(\theta) = -\mathbb{E}^{X \sim \rho, U \sim \pi_\theta(\cdot|X)}\left[\nabla_\theta^2 \log \pi_\theta\left(U|X\right)\right]$

Fisher Information Complexity Measure

- $\mathcal{M}\left(\pi_\theta\right) = -Tr\left(\mathbb{E}^{X \sim \rho^{\pi_\theta}, U \sim \pi_\theta(\cdot|X)}\left[\nabla_{\theta_\mu}^2 \log \pi_\theta\left(U|X\right)\right]\right)$

Complexity Measure: Conditional Fisher Information Trace

Hessian and Fisher Information

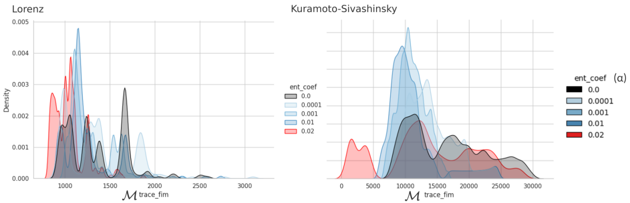
$\alpha > 0 \longrightarrow$ Flat Minima (already observed)[1]

$$\nabla_\theta^2 J^{\pi_\theta} = \mathbb{E}^{\pi_\theta}\left[\sum_{h,i,j=0}^{T} c\left(X_h, U_h\right)\left(\nabla_\theta \log \pi_\theta\left(U_i \mid X_j\right) \nabla_\theta \log \pi_\theta\left(U_j \mid X_j\right)^T + \nabla_\theta^2\left[\log \pi_\theta\left(U_i \mid X_j\right)\right]\right)\right]$$

Fisher Information: $\mathcal{I}(\theta) = -\mathbb{E}^{X\sim\rho, U\sim\pi_\theta(\cdot|X)}\left[\nabla_\theta^2 \log \pi_\theta\left(U|X\right)\right]$

Fisher Information Complexity Measure

- $\mathcal{M}\left(\pi_\theta\right) = -Tr\left(\mathbb{E}^{X\sim\rho^{\pi_\theta}, U\sim\pi_\theta(\cdot|X)}\left[\nabla_{\theta_\mu}^2 \log \pi_\theta\left(U|X\right)\right]\right)$



$\mathcal{M}\left(\pi_\theta^\alpha\right) = -Tr(\mathbb{E}^{X\sim\rho^{\pi_\theta}, U\sim\pi_\theta(\cdot|X)}\left[\nabla_{\theta_\mu}^2 \log \pi_\theta\left(U|X\right)\right])$

Colors: $\alpha = 0$, $\alpha > 0$, $\alpha_{\max}$

Result

Low $\mathcal{M}\left(\pi_\theta^\alpha\right)$ corresponds to low $\mathring{\mathcal{R}}^\pi$

Limits and perspectives

Hypothesis

Entropy $\longrightarrow$ **Flat Minimum** Already observed in (Ahmed et al. 2019)

Flat Minimum $\longleftrightarrow$ **Robustness** $\longleftrightarrow$ **Fisher Information of** θ_π ✓

Robustness of the results

- For α_{max} we lose robustness because we no longer solve the same objective
- Lorenz (fully observable) does not discriminate policies (because deterministic solution?)

Perspectives

Optimization scheme based on flat minima or Fisher Information

Learning Based Control: Sampling Strategies

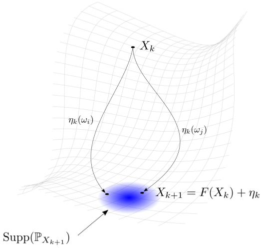
Gaussian Process Modeling

Controlled Hidden Markov Chain

$$P^{\pi}\left(dx_0 du_0 dx_1 du_1 \ldots dx_T\right) = P_{X_0}\left(dx_0\right)\mathcal{Q}\left(dy_0 \mid x_0\right)\pi\left(x_0, du_0\right)\mathcal{P}\left(dx_1 \mid x_0, u_0\right)$$
$$\mathcal{Q}\left(dy_1 \mid x_1\right)\,\pi\left(x_1, du_1\right)\cdots\pi\left(x_{T-1}, du_{T-1}\right)\mathcal{P}\left(dx_T \mid x_{T-1}, u_{T-1}\right)$$

Learning Dynamics with Gaussian Process

$$\hat{\mathcal{P}}_{\mathcal{D}}(\;\cdot\;, (x,u)) \sim \mathcal{N}\left(\mu_{(x,u)},\; k_{(x,u),\,(x,u)} \mid \mathcal{D}\right) \tag{4}$$



Transition Kernel $\mathcal{P}$

[1] C. E. Rasmussen et al. "Gaussian Processes in Reinforcement Learning" *NIPS* (2003)

Dynamics Approximation with Model Predictive Control

Model Predictive Control

$$\pi^{\mathrm{MPC}}(x) = u_0^* \tag{5}$$

$$s.t. \quad \left(u_0^*, \ldots, u_{T^{\mathrm{MPC}}}^*\right) = \underset{\left(u_0, \ldots, u_{T^{\mathrm{MPC}}}\right)}{\arg\min} \; \mathbb{E}^{\left(u_0, \ldots, u_{T^{\mathrm{MPC}}}\right)}\left[\sum_{k=0}^{T^{\mathrm{MPC}}} c\left(X_k, u_k\right) \mid X_0 = x\right] \tag{6}$$

New Problem

- Sampling budget $\longrightarrow n$
- Collect $\mathcal{D}_n$ such that $\hat{\mathcal{P}}_{\mathcal{D}} \simeq \mathcal{P}$
- Data $\mathcal{D}_n = \{(x_0, u_0), \ldots, (x_n, u_n,)\}$ is collected online along an observed dynamics
- How to sample next data point?

[1] R. Y. Rubinstein et al. The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning *Springer* (2004)

Entropy Map

How to quantify the uncertainty on $P_{X_{k+1}}$?

Infinitesimal volume element of $\mathcal{X} \longrightarrow dx$

Uncertainty on dx

$I(dx) = \log(\frac{1}{P_{X_{k+1}}(dx)})$

Entropy Map

How to quantify the uncertainty on $P_{X_{k+1}}$?

Infinitesimal volume element of $\mathcal{X} \longrightarrow dx$

Uncertainty on dx

$I(dx) = \log(\frac{1}{P_{X_{k+1}}(dx)})$

Entropy (average uncertainty)

$\mathcal{H}(P_{X_{k+1}}) = \int_{\mathbb{R}} \log \frac{1}{P_{X_{k+1}}(dx)} P_{X_{k+1}}(dx)$

Entropy Map

How to quantify the uncertainty on $P_{X_{k+1}}$?
Infinitesimal volume element of $\mathcal{X} \longrightarrow dx$

Uncertainty on dx

$$I(dx) = \log\left(\frac{1}{P_{X_{k+1}}(dx)}\right)$$

Entropy (average uncertainty)

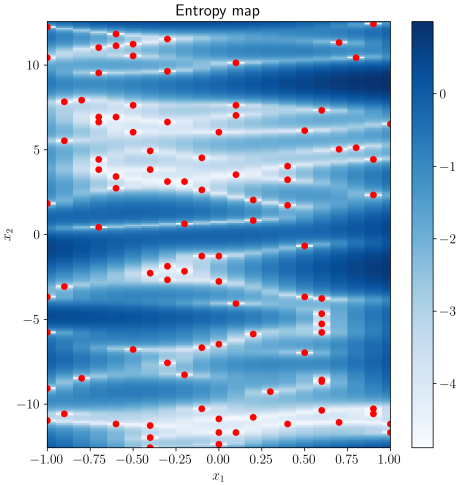
$$\mathcal{H}(P_{X_{k+1}}) = \int_{\mathbb{R}} \log \frac{1}{P_{X_{k+1}}(dx)} P_{X_{k+1}}(dx)$$

Gaussian Entropy

$$P_{X_{k+1}}(dx) = f_{\mathcal{N}(\mu_{(x,\,u)}, k_{(x,\,u),\,(x,\,u)})}(x)dx$$

$$\mathcal{H}(P_{X_{k+1}}) = \frac{1}{2}\log\left(2\pi e\, k_{(x,\,u),\,(x,\,u)}\right)$$

In the Gaussian case, the variance k characterise the entropy



Expected Information Gain

Process *trajectory* $\longrightarrow H_T = (X_0,\ U_0, \ldots, U_T, X_T)$
Optimal trajectory under $\hat{\mathcal{P}}_{\mathcal{D}} \longrightarrow H_T^*$

Expected Information Gain

$$\mathrm{EIG}_n(x, u) = \mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n\right] - \mathbb{E}_{P_{X_{n+1} \mid \mathcal{D}_n,\, X_n = x,\, U_n = u}}\left[\mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n, X_n = x, U_n = u, X_{n+1}\right]\right] \tag{7}$$

EIG $\longrightarrow$ conditional mutual information between $\hat{H}_T^*$ and X_{n+1} given $\mathcal{D}_n$

[1] V. Mehta et al. "An Experimental Design Perspective on Model-Based Reinforcement Learning" *ICLR* (2022)

Expected Information Gain

Process *trajectory* $\longrightarrow$ $H_T = (X_0,\ U_0, \ldots, U_T, X_T)$
Optimal trajectory under $\hat{\mathcal{P}}_{\mathcal{D}} \longrightarrow H_T^*$

Expected Information Gain

$$\mathrm{EIG}_n(x,u) = \mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n\right] - \mathbb{E}_{P_{X_{n+1} \mid \mathcal{D}_n,\, X_n = x,\, U_n = u}}\left[\mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n, X_n = x, U_n = u, X_{n+1}\right]\right] \tag{7}$$

EIG $\longrightarrow$ conditional mutual information between $\hat{H}_T^*$ and X_{n+1} given $\mathcal{D}_n$

By symmetry of cond. MI

$$\mathrm{EIG}_n(x,u) = \mathcal{H}\left[X_{n+1} \mid \mathcal{D}_n,\ X_n = x,\ U_n = u\right] - \mathbb{E}_{P_{\hat{H}_T^* \mid \mathcal{D}_n}}\left[\mathcal{H}\left[X_{n+1} \mid \mathcal{D}_n, X_n = x,\ U_n = u,\ \hat{H}_T^*\right]\right] \tag{8}$$

[1] V. Mehta et al. "An Experimental Design Perspective on Model-Based Reinforcement Learning" *ICLR* (2022)

Randomized Decision Epochs: Temporal Abstraction with options[1]

New Problem

- Data $\mathcal{D}_n = \{(x_0, u_0), \ldots, (x_n, u_n,)\}$ is collected online along an observed dynamics
- When to sample next data point?

[1] R. S. Sutton et al. "Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning", *NIPS* (1999)

Randomized Decision Epochs: Temporal Abstraction with options[1]

New Problem

- Data $\mathcal{D}_n = \{(x_0, u_0), \dots, (x_n, u_n,)\}$ is collected online along an observed dynamics
- When to sample next data point?

Hypothesis

Wait for auto-decorrelation of (X_{n+1}, U_{n+1}) from $\mathcal{D}_n$

[1] R. S. Sutton et al. "Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning", *NIPS* (1999)

Randomized Decision Epochs: Temporal Abstraction with options[1]

New Problem

- Data $\mathcal{D}_n = \{(x_0, u_0), \dots, (x_n, u_n,)\}$ is collected online along an observed dynamics
- When to sample next data point?

Hypothesis

Wait for auto-decorrelation of (X_{n+1}, U_{n+1}) from $\mathcal{D}_n$

Random decision epochs $\longrightarrow (\kappa_j)_{j\in\mathbb{N}}$

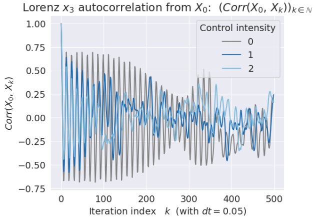
Semi-Markov Decision Process $\longrightarrow (X_{\kappa_j})_{j\in\mathbb{N}}$

$\mathcal{P}^{\mathrm{SMDP}}(dx' \mid (x, (u, t))) = P(X_{k+t} \mid X_k = x,\ U_{k:k+t-1} = u)$

Time-delay $\longrightarrow t \in \mathcal{T}$

Constant control between κ_j and κ_{j+1}

New action space $\longrightarrow \mathcal{U} \times \mathcal{T}$



$(Cov(X_0, X_k))_{k\in\mathbb{N}}$ for the controlled Lorenz system x_3 component under multiple control intensities.

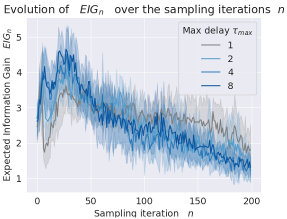
[1] R. S. Sutton et al. "Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning", *NIPS* (1999)

Semi-Markovian Expected Information Gain

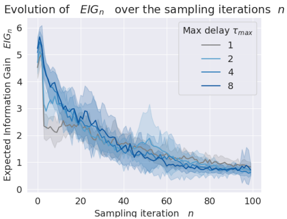
New Information Gain

$$\mathrm{EIG}_n^{\text{SM-TIP}}(x,(u,t)) =$$

$$\mathcal{H}\left[X_{\kappa_n+t+1} \mid \mathcal{D}_n,\, X_{\kappa_n+t}=x,\, U_{\kappa_n+t}=u,\, \kappa_n\right] - \mathbb{E}_{P_{\hat{H}_T^*\mid\mathcal{D}_n}}\left[\mathcal{H}\left[X_{\kappa_n+t+1} \mid \mathcal{D}_n, X_{\kappa_n+t}=x,\, U_{\kappa_n+t}=u,\, \hat{H}_T^*,\, \kappa_n\right]\right] \tag{9}$$



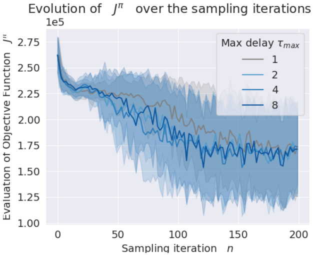
(a) Lorenz

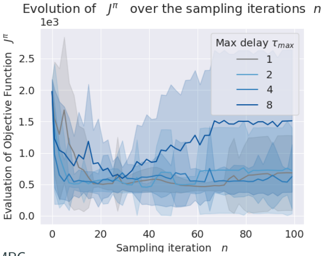


(b) Pendulum

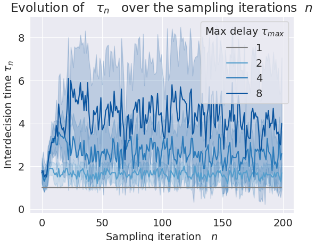
Expected Information Gain $\mathrm{EIG}^{\text{SM-TIP}}$ over n.

Objective Function and Inter-Decision Time

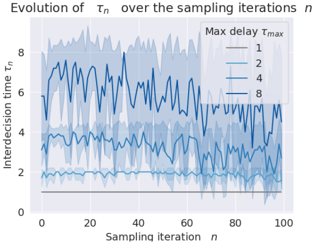




Objective function $J^{\widehat{\pi}^{\mathrm{MPC}}}$ during training.



(a) Lorenz



(b) Pendulum

Limits and perspectives

Hypothesis

Temporal Abstraction $\longrightarrow$ Information (fixed sampling budget n) ✓

Robustness of the results

- Evaluation fairness when one model can explore the dynamics further than the other
- How to characterise the δ_t treshold where temporal abstraction is beneficial?

Perspectives

Correct bootstrapping error in the SMDP