Measure-valued Proximal Recursions for Learning and Control

Iman Nodozi

inodozi@ucsc.edu

Department of Electrical and Computer Engineering
University of California, Santa Cruz

Committee: Profs. Dejan Milutinovic (Chair), Abhishek Halder (Advisor), Yu Zhang

Dissertation Defense, March 15, 2024

Measure-valued Optimization Problems

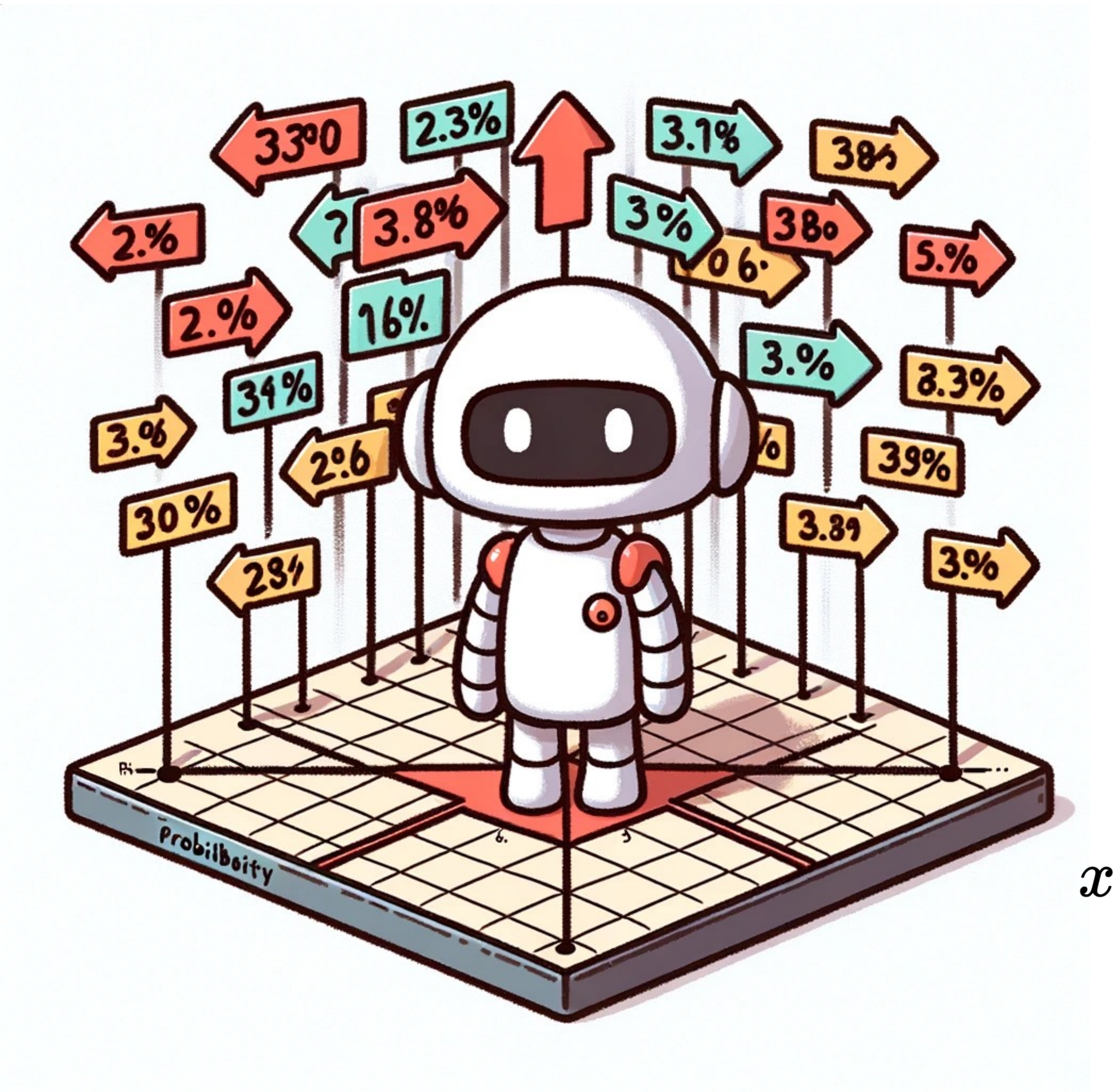
$$\mu^{\text{opt}} = \arg\inf_{\mu} F(\mu)$$

Measure-valued Optimization Problems

$$\mu^{\text{opt}} = \arg\inf_{\mu \in \mathscr{P}_2(\mathbb{R}^d)} F(\mu)$$

Manifold of probability measures supported on $\mathbb{R}^d$ with finite second moments

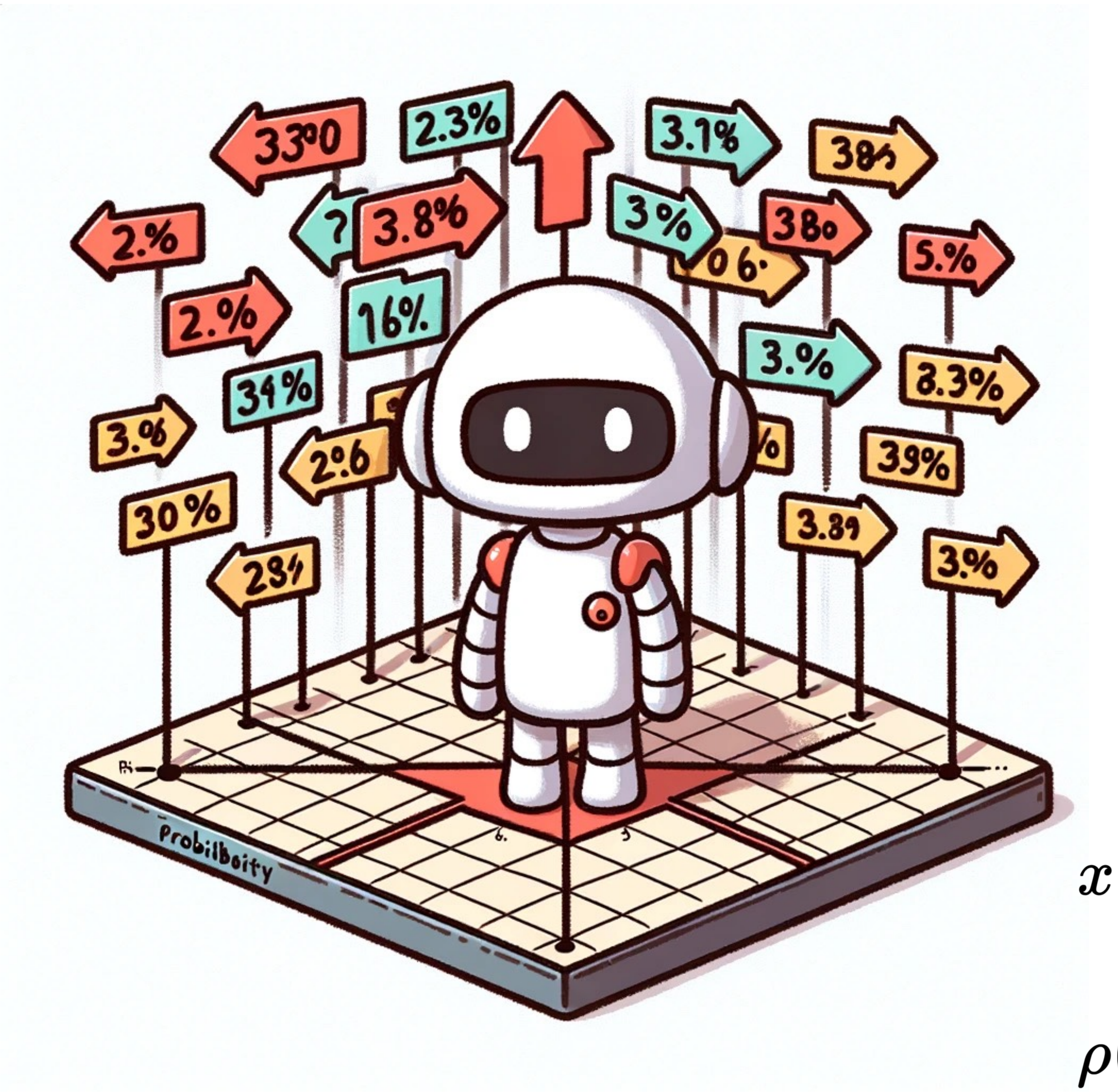
Probability Distribution



$$x(t) = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \in \mathcal{X} \equiv \mathbb{R}^2 \times \mathbb{S}^1$$

Images generated by ChatGPT

Probability Distribution

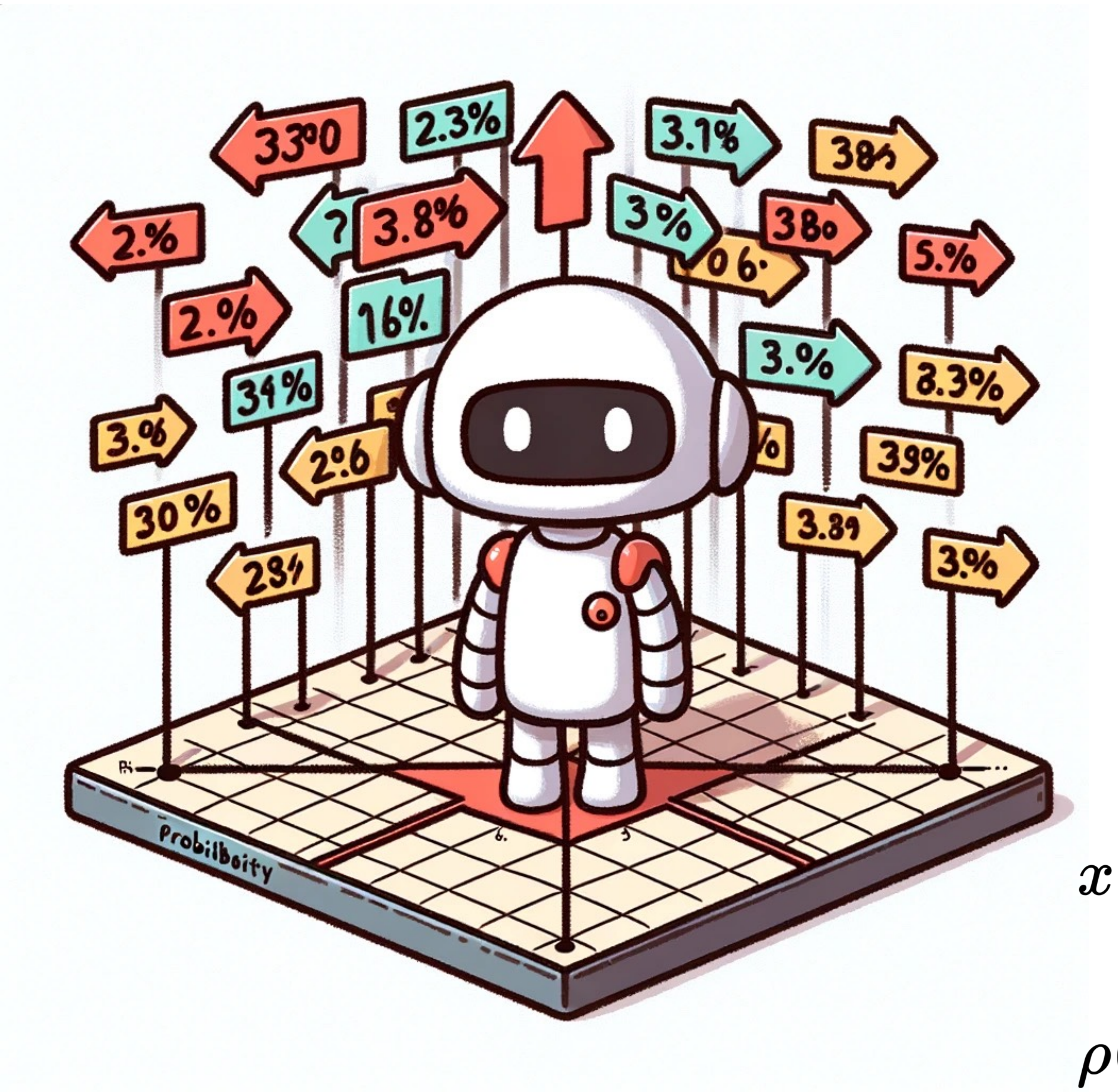


$$x(t) = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \in \mathcal{X} \equiv \mathbb{R}^2 \times \mathbb{S}^1$$

$$\rho(x,t) : \mathcal{X} \times [0, \infty) \to \mathbb{R}_{\geq 0}$$

Images generated by ChatGPT

Probability Distribution



$$x(t) = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \in \mathcal{X} \equiv \mathbb{R}^2 \times \mathbb{S}^1$$

$$\rho(x,t) : \mathcal{X} \times [0,\infty) \to \mathbb{R}_{\geq 0}$$

measure

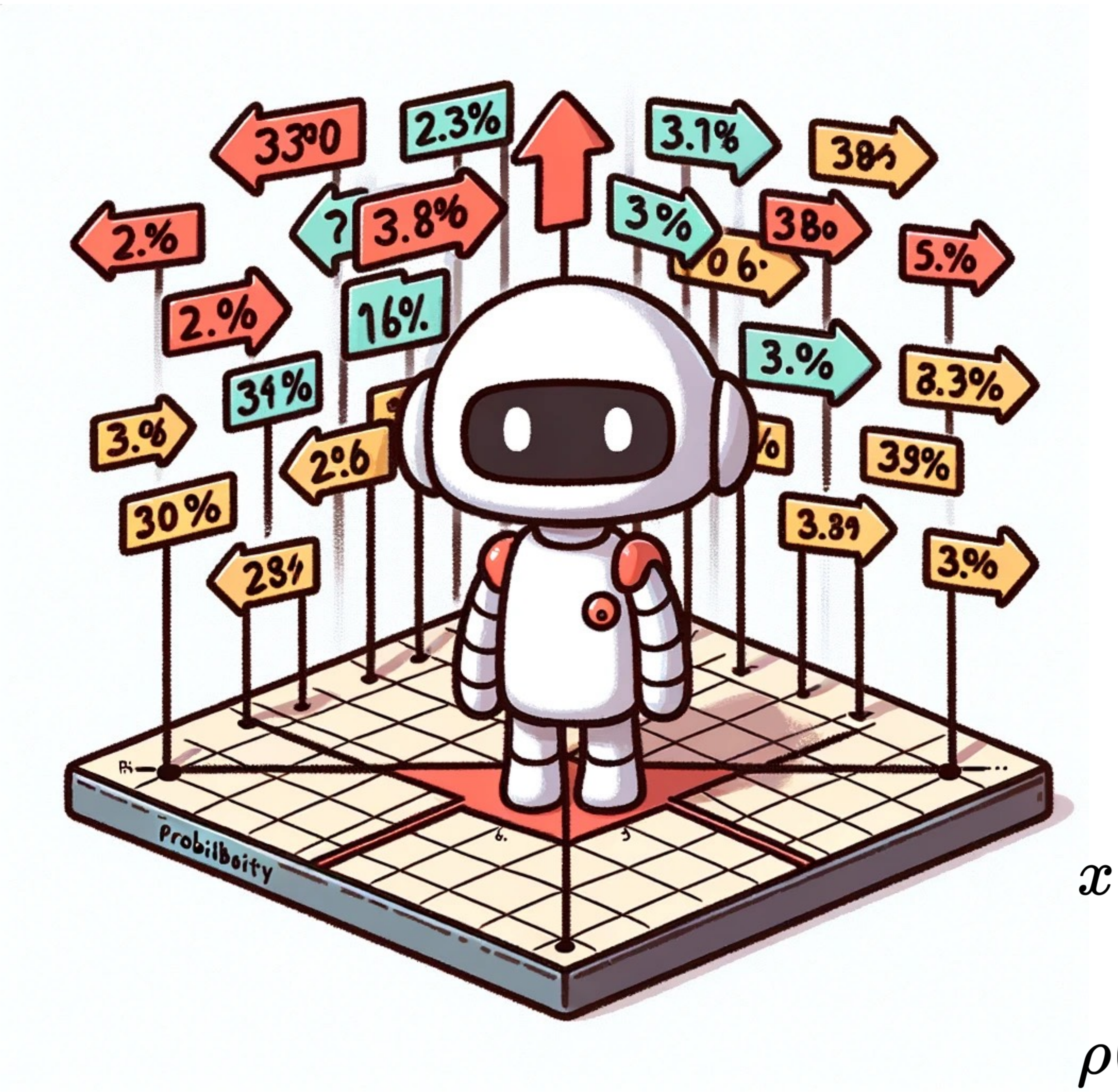
Density function

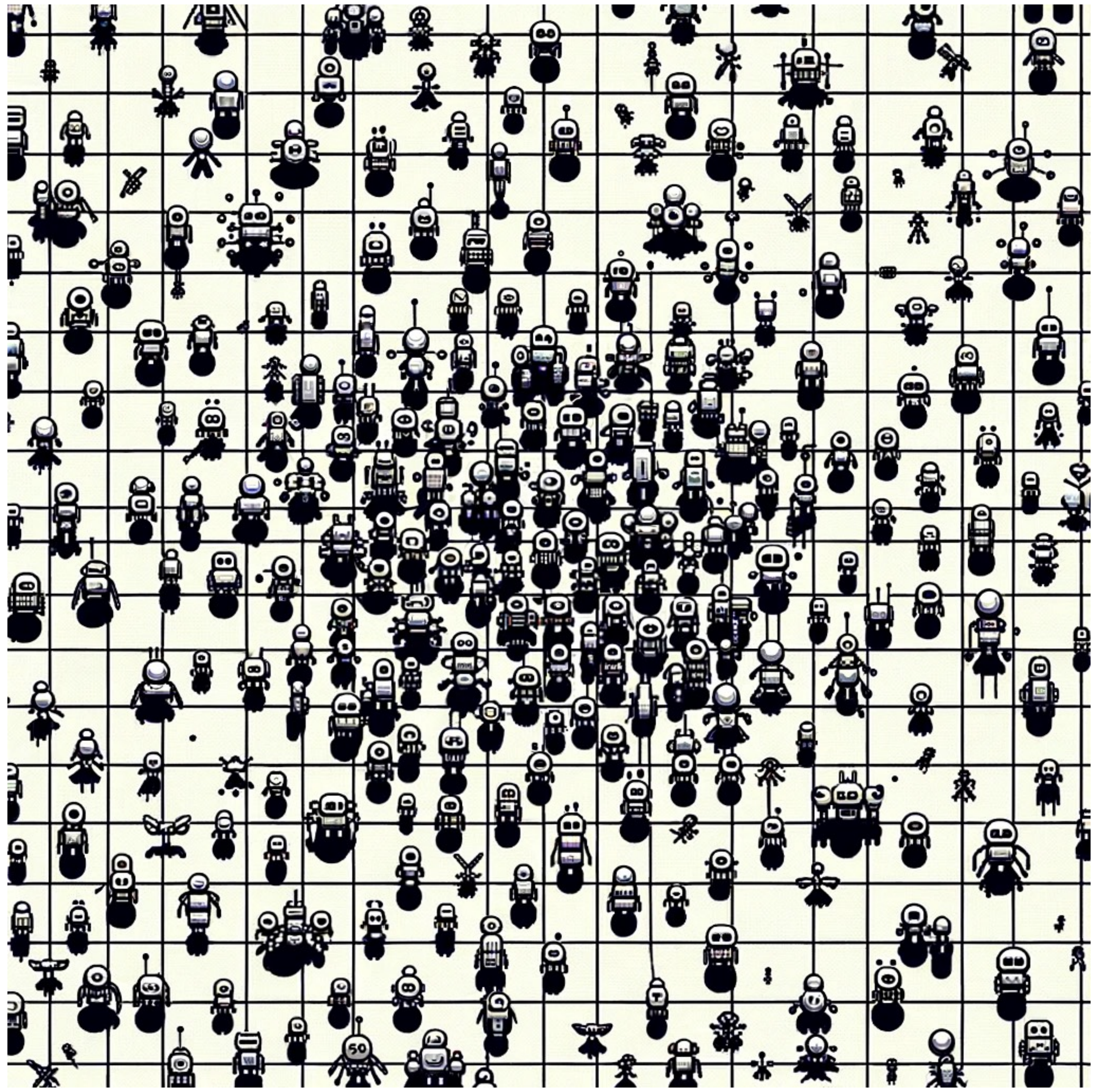
$$\int_{\mathcal{X}} \mathrm{d}\mu = \int_{\mathcal{X}} \rho \mathrm{d}x = 1 \quad \text{for all } t \in [0,\infty)$$

Images generated by ChatGPT

Probability Distribution

Population Distribution





$$x(t) = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \in \mathcal{X} \equiv \mathbb{R}^2 \times \mathbb{S}^1$$

$$\rho(x,t) : \mathcal{X} \times [0,\infty) \to \mathbb{R}_{\geq 0}$$

measure

Density function

$$\int_{\mathcal{X}} \mathrm{d}\mu = \int_{\mathcal{X}} \rho \mathrm{d}x = 1 \quad \text{for all } t \in [0,\infty)$$

Images generated by ChatGPT

Geometry on the Space of Prob. Measures

2-Wasserstein distance metric

Ground cost, e.g., $\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{y}\|_2^2$

$$W_2(\mu_0, \mu_1) := \left(\inf_{m} \int_{\mathcal{X}\times\mathcal{Y}} c(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}m(\boldsymbol{x}, \boldsymbol{y}) \right)^{1/2}$$

$$\text{subject to} \quad \int_{\mathcal{Y}} \mathrm{d}m = \mu_0(\,\mathrm{d}\boldsymbol{x}), \quad \int_{\mathcal{X}} \mathrm{d}m = \mu_1(\,\mathrm{d}\boldsymbol{y})$$

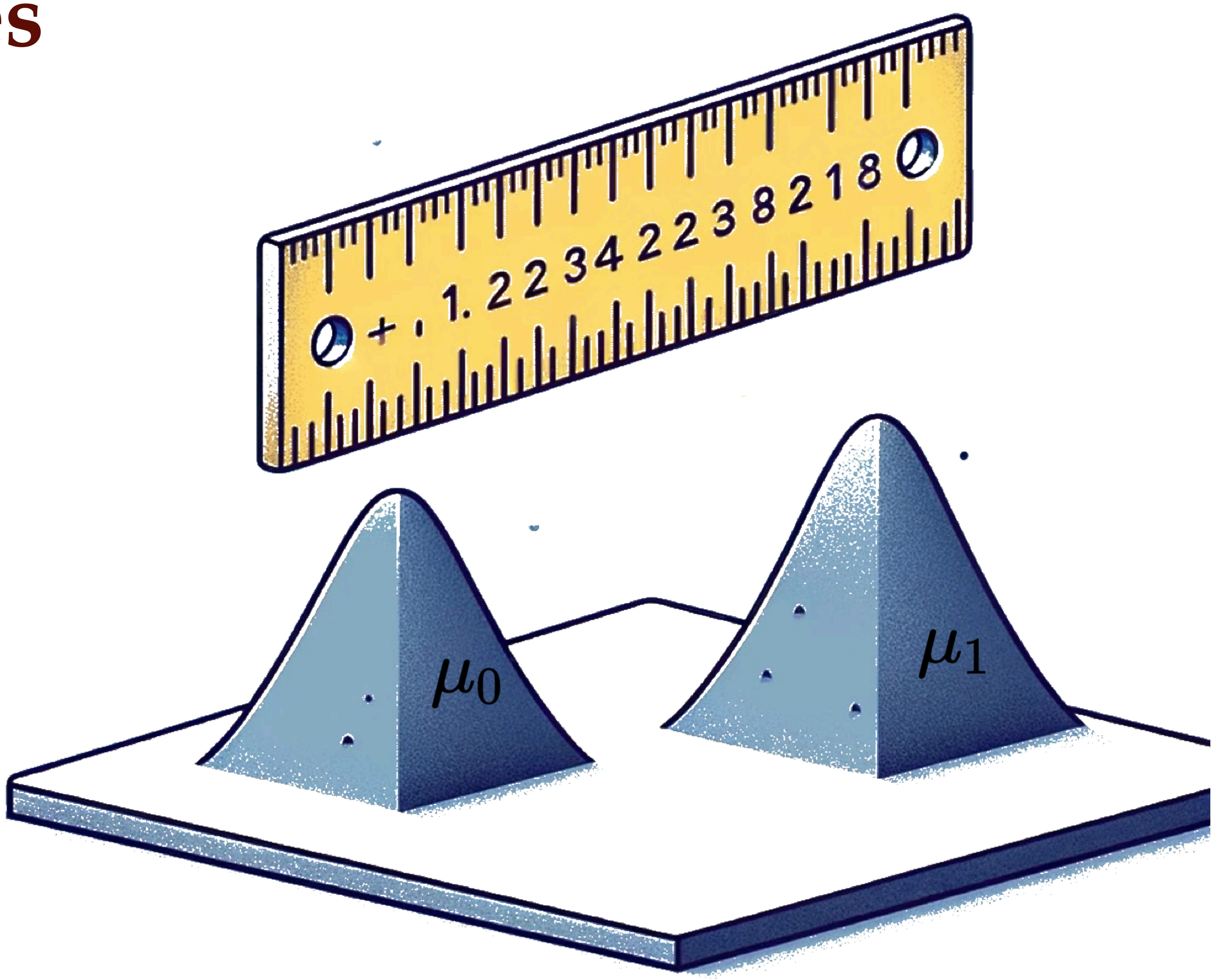


Image generated by ChatGPT

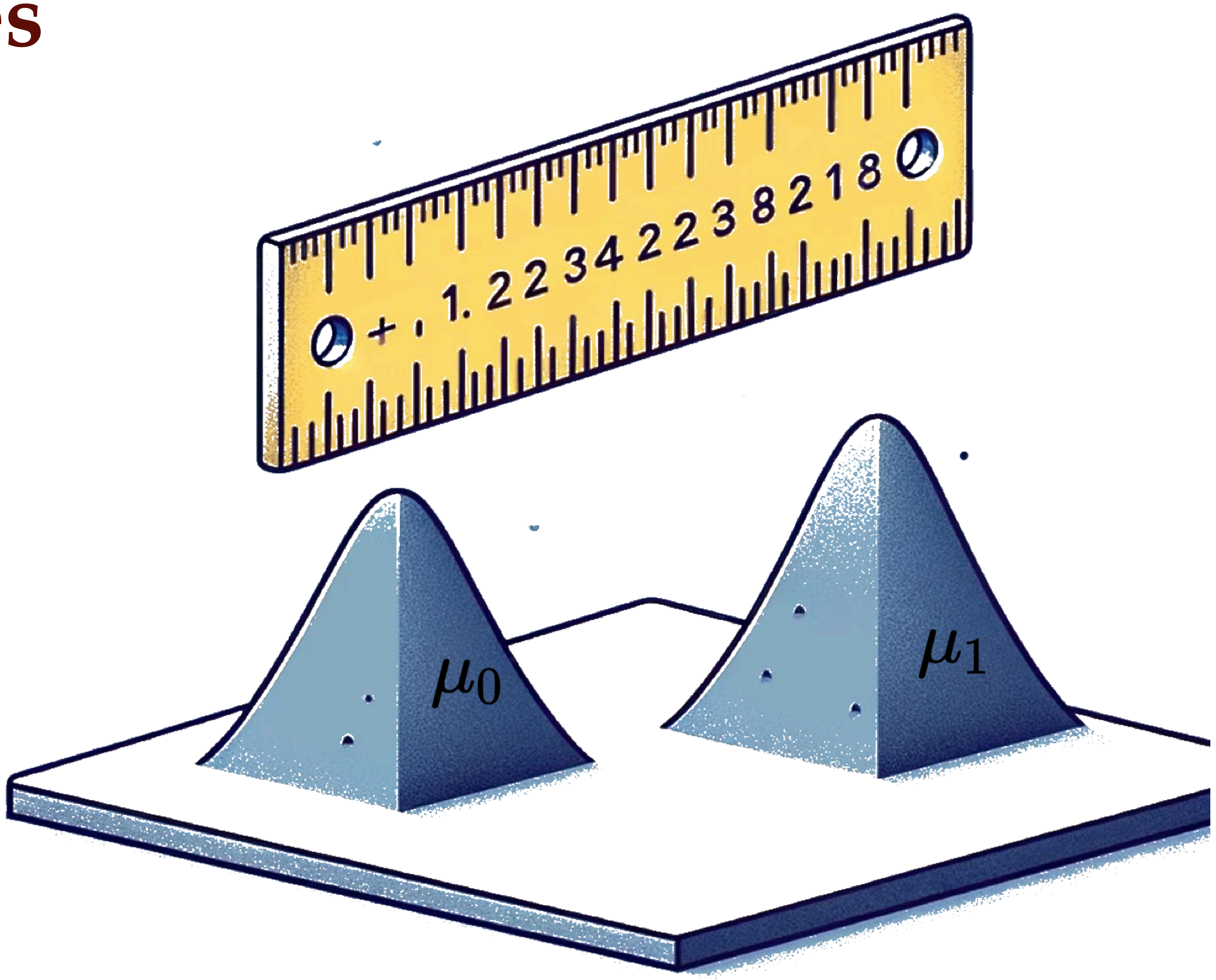
Geometry on the Space of Prob. Measures

2-Wasserstein distance metric

Ground cost, e.g., $\frac{1}{2}\|x - y\|_2^2$

$$W_2(\mu_0, \mu_1) := \left(\inf_m \int_{\mathcal{X}\times\mathcal{Y}} c(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}m(\boldsymbol{x}, \boldsymbol{y})\right)^{1/2}$$

$$\text{subject to} \quad \int_{\mathcal{Y}} \mathrm{d}m = \mu_0(\,\mathrm{d}\boldsymbol{x}), \quad \int_{\mathcal{X}} \mathrm{d}m = \mu_1(\,\mathrm{d}\boldsymbol{y})$$



Sinkhorn divergence:

$$W_\varepsilon(\mu_0, \mu_1) := \left(\inf_m \int_{\mathcal{X}\times\mathcal{Y}} \{c(\boldsymbol{x}, \boldsymbol{y}) + \varepsilon \log m\} \mathrm{d}m(\boldsymbol{x}, \boldsymbol{y})\right)^{1/2}, \quad \varepsilon > 0$$

$$\text{subject to} \quad \int_{\mathcal{Y}} \mathrm{d}m = \mu_0(\,\mathrm{d}\boldsymbol{x}), \quad \int_{\mathcal{X}} \mathrm{d}m = \mu_1(\,\mathrm{d}\boldsymbol{y})$$

Image generated by ChatGPT

Connection with Wasserstein Gradient Flows

$$\frac{\partial \mu}{\partial t} = -\nabla^{W_2} F(\mu) := \nabla \cdot \left(\mu \nabla \frac{\delta F}{\delta \mu} \right) \qquad (\star)$$

Wasserstein gradient

Minimizer of $\underset{\mu \in \mathcal{P}_2(\mathbb{R}^d)}{\arg\inf} F(\mu)$ ⟵ Stationary solution of $(\star)$

Transient solution of $(\star)$ ⟶ Discrete time-stepping realizing

grad. descent of $\underset{\mu \in \mathcal{P}_2(\mathbb{R}^d)}{\arg\inf} F(\mu)$

Wasserstein proximal recursion à la Jordan-Kinderlehrer-Otto (JKO) scheme

Gradient Flows

Gradient Flow in $\mathcal{X}$

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = -\nabla f(\boldsymbol{x}), \quad \boldsymbol{x}(0) = x_0$$

Recursion:

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{x}_{k-1} - h\nabla f(\boldsymbol{x}_k) \\ &= \arg\min_{x\in\mathcal{X}} \left\{ \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}_{k-1}\|_2^2 + hf(\boldsymbol{x}) \right\} \\ &=: \mathrm{prox}_{hf}^{\|\cdot\|_2}(\boldsymbol{x}_{k-1}) \end{aligned}$$

Convergence:

$$x_k \to \boldsymbol{x}(t = kh) \quad \text{as} \quad h \downarrow 0$$

Gradient Flow in $\mathcal{P}_2(\mathcal{X})$

$$\frac{\partial\mu}{\partial t} = -\nabla^W F(\mu), \quad \mu(\boldsymbol{x}, 0) = \mu_0$$

Recursion:

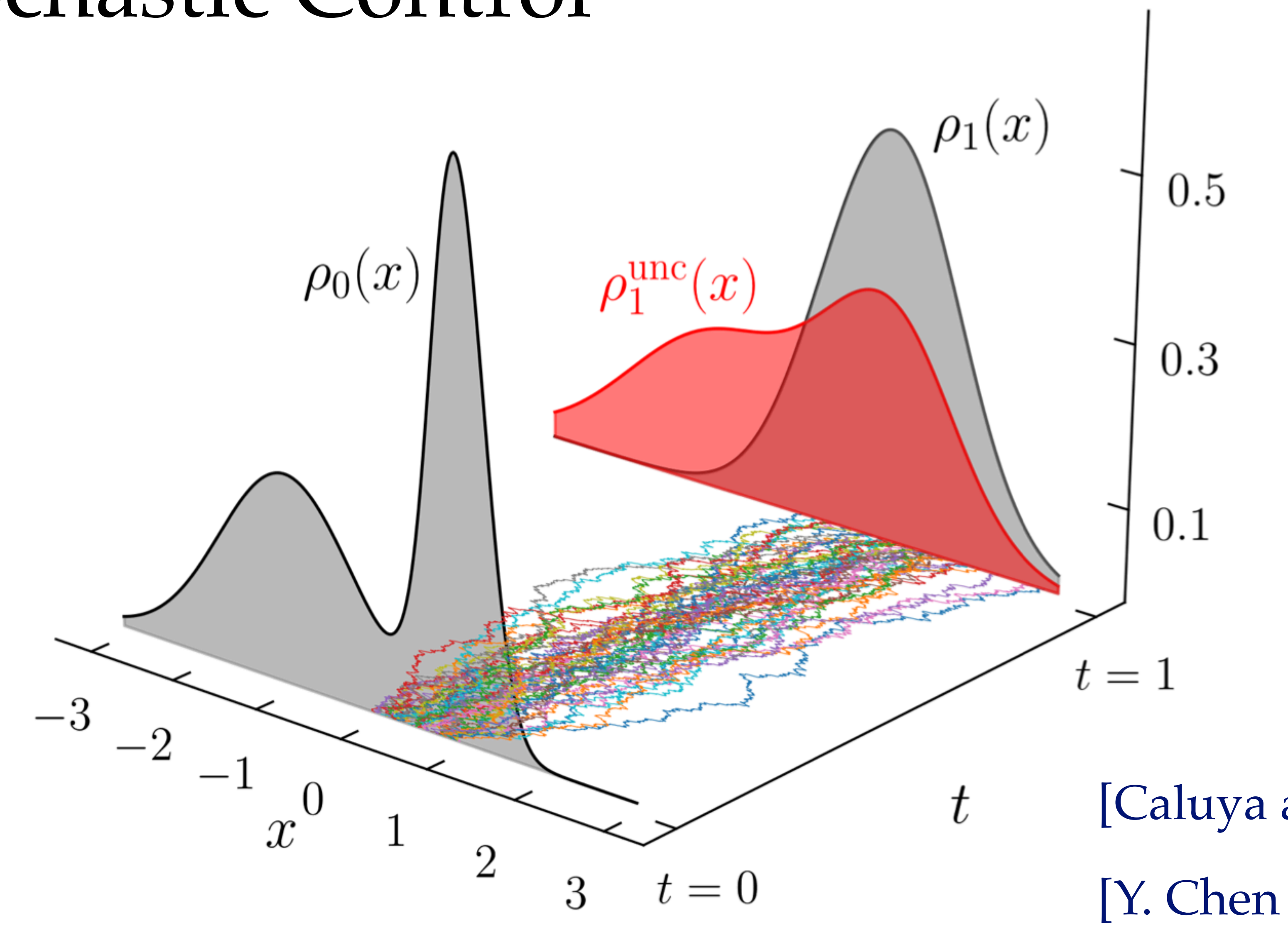
$$\begin{aligned} \mu_k &= \mu(\cdot, t = kh) \\ &= \arg\min_{\mu\in\mathcal{P}_2(\mathcal{X})} \left\{ \frac{1}{2}W^2(\mu, \mu_{k-1}) + hF(\mu) \right\} \\ &=: \mathrm{prox}_{hF}^{W}(\mu_{k-1}) \end{aligned}$$

Convergence:

$$\mu_k \to \mu(\cdot, t = kh) \quad \text{as} \quad h \downarrow 0$$

Motivating Applications

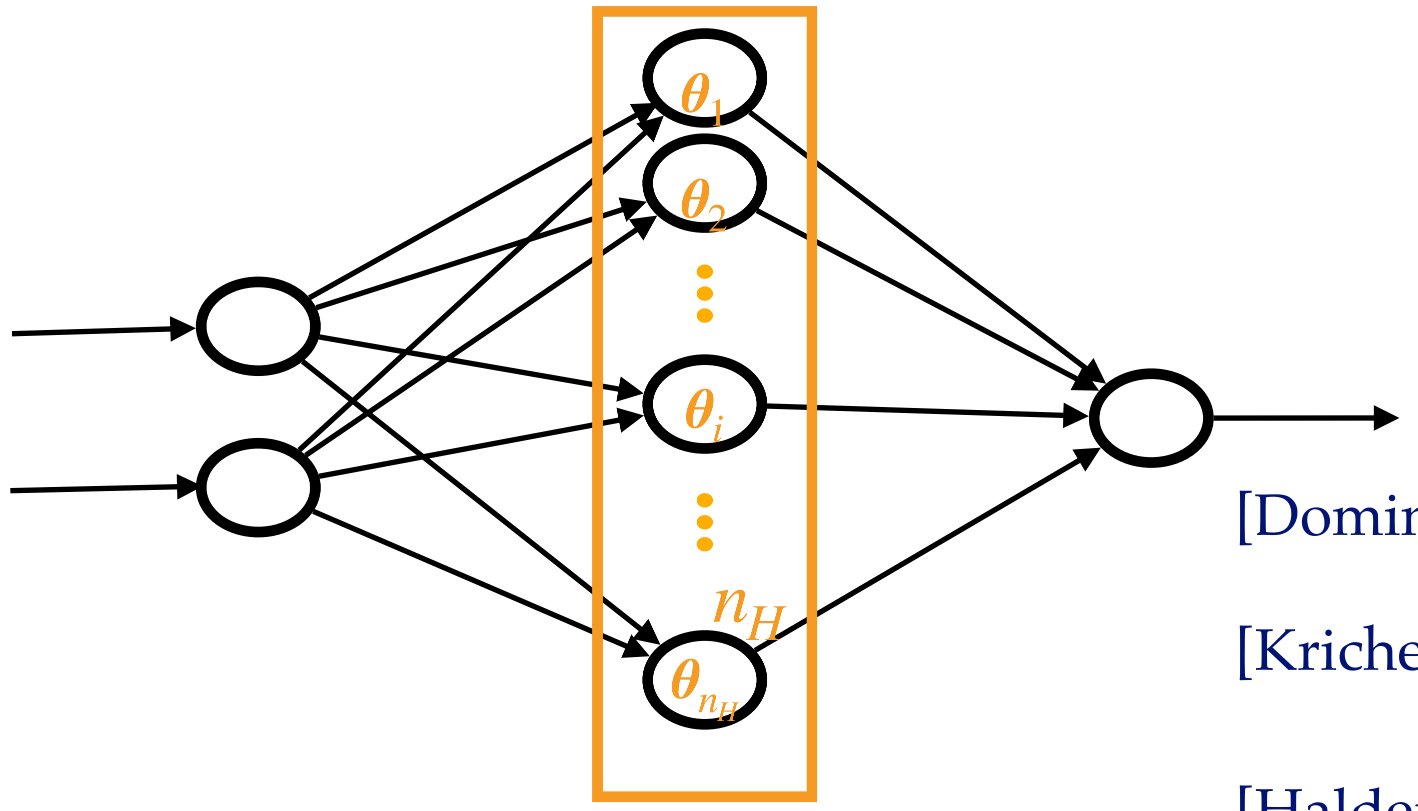
Stochastic Control



[Caluya and Halder,., 2021]

[Y. Chen et al., 2021]

Stochastic learning



[Domingo-Enrich et al., 2020]

[Krichene, et al., 2020]

[Halder et al., 2020]

Stochastic Modeling

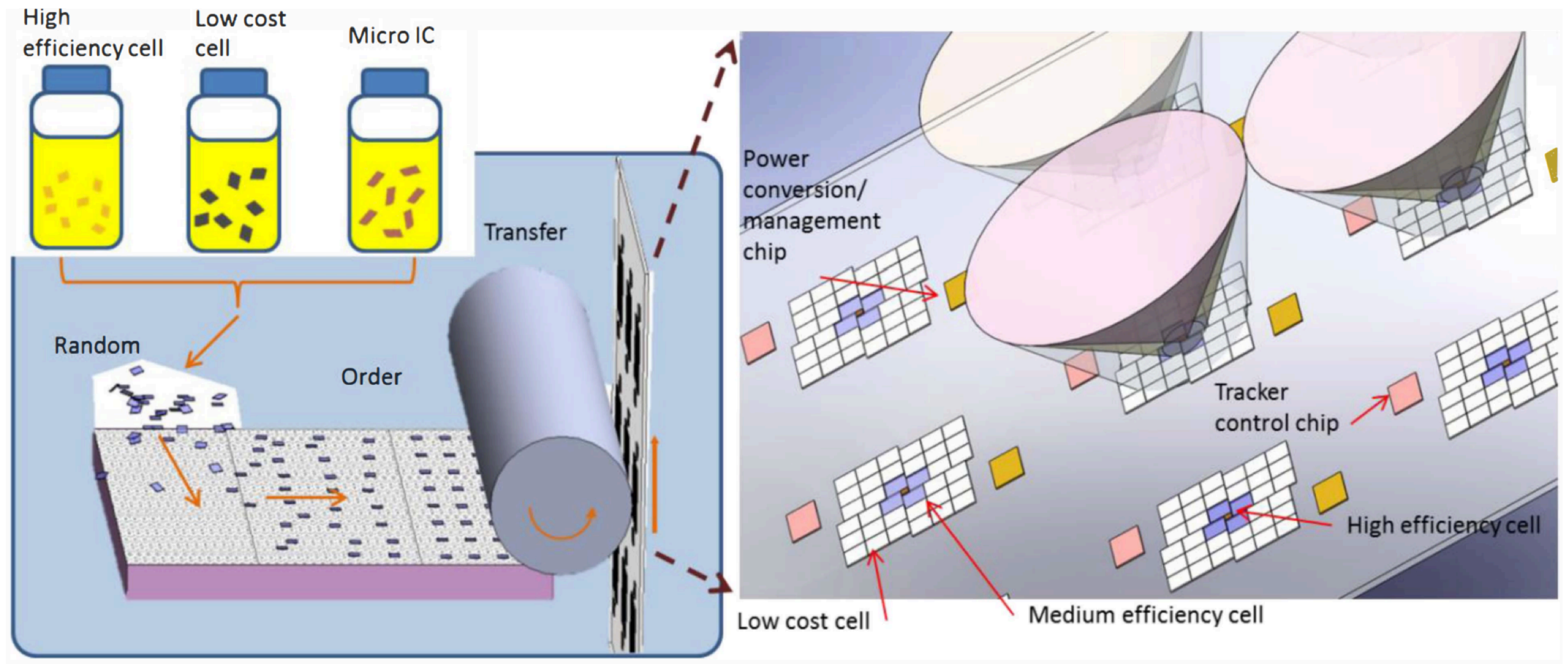


Image credit: PARC

Generative AI

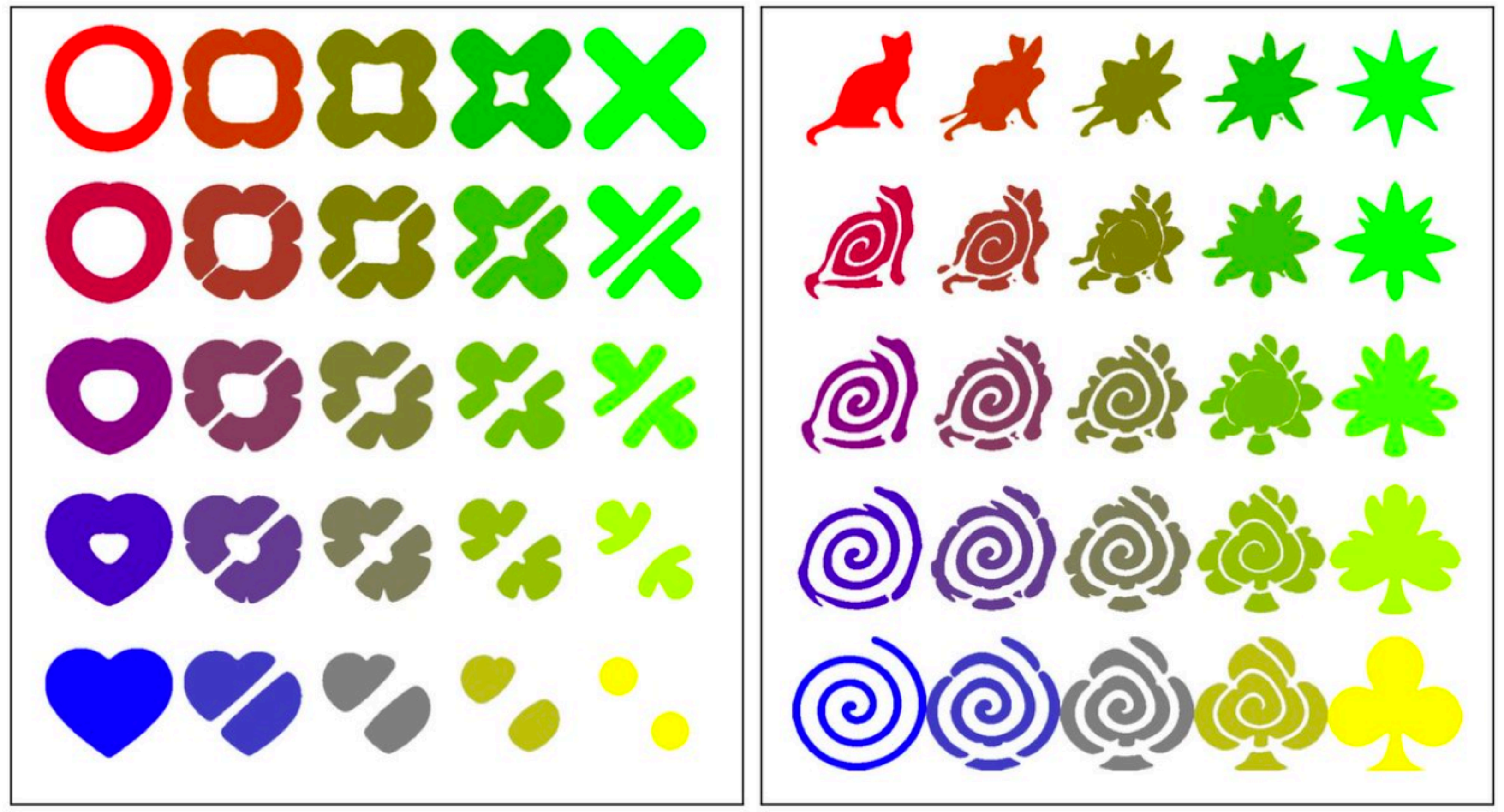


Image credit: G. Pyre

Contributions

Part I: Optimal Stochastic Control of Generalized Schrödinger Bridge in

The Control-affine Case Knowing the Model Structure [**I. Nodozi**, A. Halder., CDC 22]

The Control Non-affine Case Knowing the Model Structure [**I. Nodozi**, et. al., ACC 23]

The Control Non-affine Case not Knowing the Model Structure [**I. Nodozi**, et. al., IEEE TCST 23]

Part II: Stochastic Modeling and Solving of Chiplet Population Dynamics

A Controlled Mean Field Model for Chiplet Population Dynamics [**I. Nodozi**, et. al., IEEE LCSS 23]

Part III: Stochastic Learning

Centralized Computing: Mean Field Learning [A. Teter, **I. Nodozi**, A. Halder, TMLR 23]

Distributed Computing: Wasserstein Consensus ADMM [**I. Nodozi**, A. Halder. arXiv]

Part I: Optimal Stochastic Control of Generalized Schrödinger Bridge

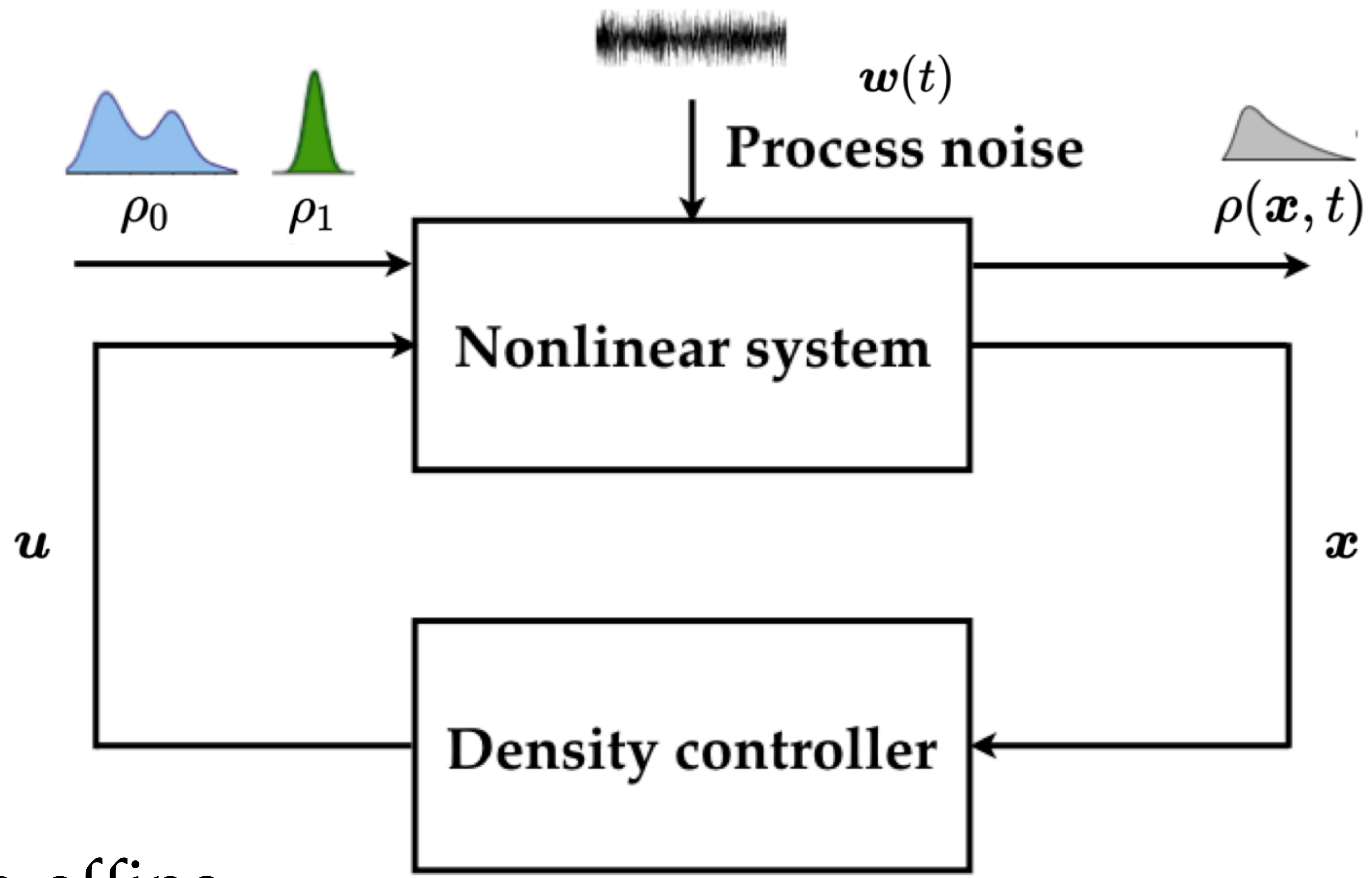
Stochastic Control

$$\inf_{\boldsymbol{u}\in\mathscr{U}} \quad \mathbb{E}_{\mu^u}\left\{\int_0^T \frac{1}{2}\|\boldsymbol{u}(\boldsymbol{x},t)\|_2^2 \,\mathrm{d}t\right\}$$

subject to

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(t,\boldsymbol{x},\boldsymbol{u})\mathrm{d}t + \sqrt{2\beta^{-1}}\boldsymbol{g}(t,\boldsymbol{x},\boldsymbol{u})\mathrm{d}\boldsymbol{w}$$

$$\boldsymbol{x}(t=0) \sim \mu_0(\boldsymbol{x}), \quad \boldsymbol{x}(t=T) \sim \mu_T(\boldsymbol{x})$$



Control affine

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x},t)\mathrm{d}t + \boldsymbol{B}(t)\boldsymbol{u}(\boldsymbol{x},t)\mathrm{d}t + \sqrt{2\epsilon}\boldsymbol{B}(t)\mathrm{d}\boldsymbol{w}$$

Case study: Nonuniform Noisy Kuramoto Oscillators

Control non-affine

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(t,\boldsymbol{x},\boldsymbol{u})\mathrm{d}t + \sqrt{2\beta^{-1}}\boldsymbol{g}(t,\boldsymbol{x},\boldsymbol{u})\mathrm{d}\boldsymbol{w}$$

Case study: Controlled Self-assembly
- Model-based
- Model-free

Stochastic Control/ Control-affine

Conditions for Optimality

$$\frac{\partial}{\partial t}\rho^{\text{opt}} + \nabla \cdot \left(\rho^{\text{opt}}\left(\boldsymbol{f} + \boldsymbol{B}(t)^{\top}\nabla\psi\right)\right) = \epsilon\left\langle \boldsymbol{D}(t), \text{Hess}\left(\rho^{\text{opt}}\right)\right\rangle$$

Controlled FPK PDE

$$\frac{\partial\psi}{\partial t} + \frac{1}{2}\left\|\boldsymbol{B}(t)^{\top}\nabla\psi\right\|_2^2 + \langle\nabla\psi, \boldsymbol{f}\rangle = -\epsilon\langle\boldsymbol{D}(t),\ \text{Hess}\,(\psi)\rangle$$

HJB PDE

$$\boldsymbol{u}^{\text{opt}}(\boldsymbol{x}, t) = \boldsymbol{B}(t)^{\top}\nabla\psi(\boldsymbol{x}, t)$$

Optimal policy

$$\rho^{\text{opt}}(\boldsymbol{x},0) = \rho_0(\boldsymbol{x}), \quad \rho^{\text{opt}}(\boldsymbol{x}, T) = \rho_T(\boldsymbol{x})$$

Boundary conditions

Stochastic Control/ Control-affine

Hopf-Cole a.k.a. Fleming's logarithmic transform: $\left(\rho^{\text{opt}}, \psi\right) \mapsto \underbrace{(\widehat{\varphi}, \varphi)}_{\text{Schrödinger factors}}$

$$\varphi(\boldsymbol{x}, t) = \exp\left(\frac{\psi(\boldsymbol{x}, t)}{2\epsilon}\right)$$

$$\hat{\varphi}(\boldsymbol{x}, t) = \rho^{\text{opt}}(\boldsymbol{x}, t)\exp\left(-\frac{\psi(\boldsymbol{x}, t)}{2\epsilon}\right)$$

1931
31.3.

This is still a challenge to mathematicians, to solve a certain boundary value problem. 14.III.52. E.S.

Überreicht vom Verfasser

S

ÜBER DIE UMKEHRUNG DER NATURGESETZE

VON

E. SCHRÖDINGER

SONDERAUSGABE AUS DEN SITZUNGSBERICHTEN
DER PREUSSISCHEN AKADEMIE DER WISSENSCHAFTEN
PHYS.-MATH. KLASSE. 1931. IX

Sur la théorie relativiste de l'électron et l'interprétation de la mécanique quantique

PAR

E. SCHRÖDINGER

I. — **Introduction**

J'ai l'intention d'exposer dans ces conférences diverses idées concernant la mécanique quantique et l'interprétation qu'on en donne généralement à l'heure actuelle ; je parlerai principalement de la théorie quantique relativiste du mouvement de l'électron. Autant que nous pouvons nous en rendre compte aujourd'hui, il semble à peu près sûr que la mécanique quantique de l'électron, sous sa forme idéale, *que nous ne possédons pas encore*, doit former un jour la base de toute la physique. A cet intérêt tout à fait général, s'ajoute, ici à Paris, un intérêt particulier : vous savez tous que les bases de la théorie moderne de l'électron ont été posées à Paris par votre célèbre compatriote Louis de BROGLIE.

Stochastic Control/ Control-affine

2 coupled nonlinear PDEs → boundary-coupled linear PDEs!!

$$\frac{\partial \varphi}{\partial t} = -\langle \nabla \varphi, \boldsymbol{f} \rangle - \epsilon \langle \boldsymbol{D}(t), \text{Hess}(\varphi) \rangle$$

Forward Fokker-Planck PDE

$$\frac{\partial \hat{\varphi}}{\partial t} = -\nabla \cdot (\hat{\varphi} \boldsymbol{f}) + \epsilon \langle \boldsymbol{D}(t), \text{Hess}(\hat{\varphi}) \rangle$$

Backward Fokker-Planck PDE

Initial and Terminal conditions

$$\varphi(\boldsymbol{x}, 0)\hat{\varphi}(\boldsymbol{x}, 0) = \rho_0(\boldsymbol{x})$$

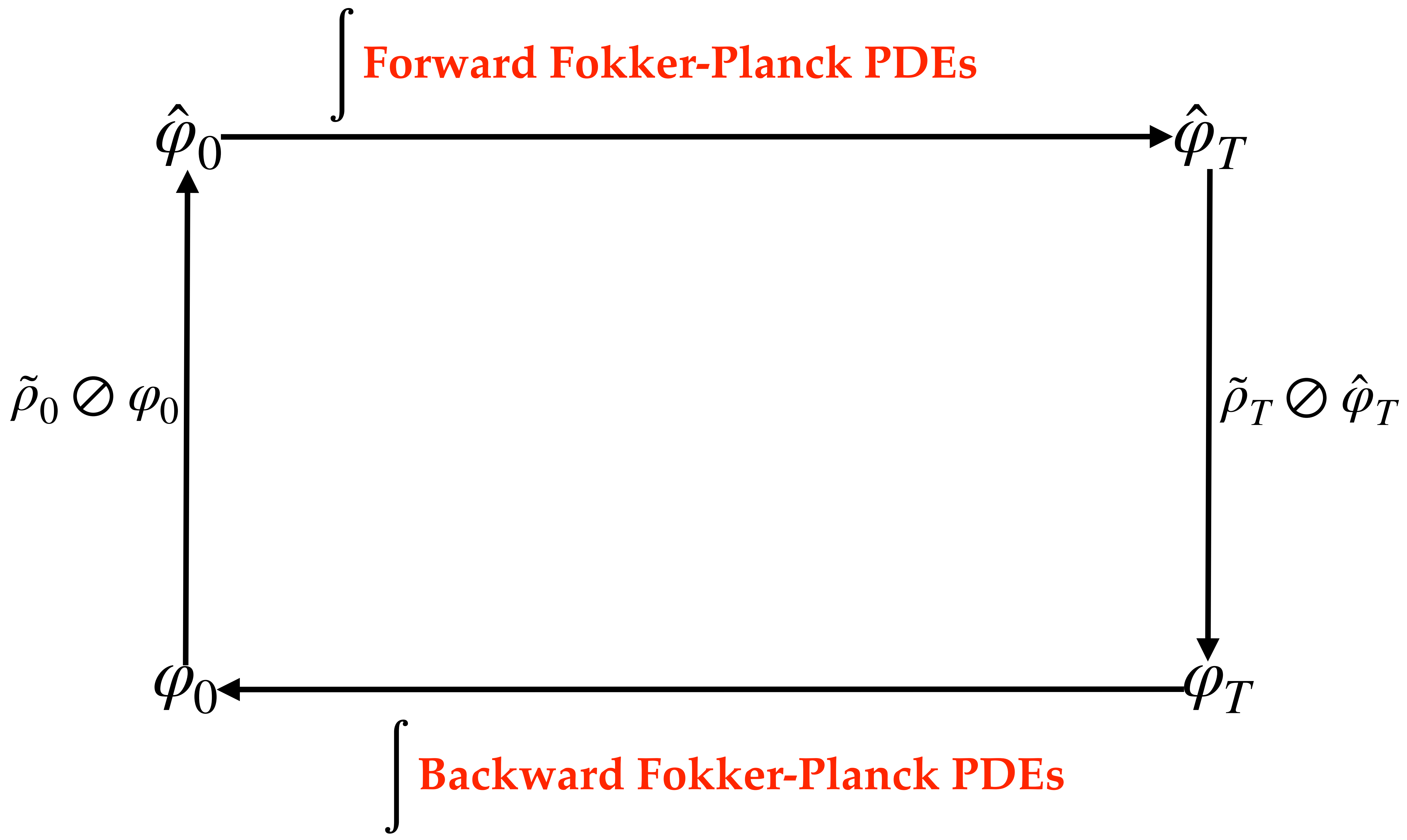
$$\varphi(\boldsymbol{x}, T)\hat{\varphi}(\boldsymbol{x}, T) = \rho_T(\boldsymbol{x})$$

Optimal controlled joint state PDF: $\rho^{\text{opt}}(\boldsymbol{x}, t) = \varphi(\boldsymbol{x}, t)\hat{\varphi}(\boldsymbol{x}, t)$

Optimal control: $\boldsymbol{u}^{\text{opt}}(\boldsymbol{x}, t) = 2\epsilon \boldsymbol{B}(t)^\top \nabla \log \varphi$

Stochastic Control/ Control-affine

Fixed Point Recursion Over Pair $(\varphi, \hat{\varphi})$

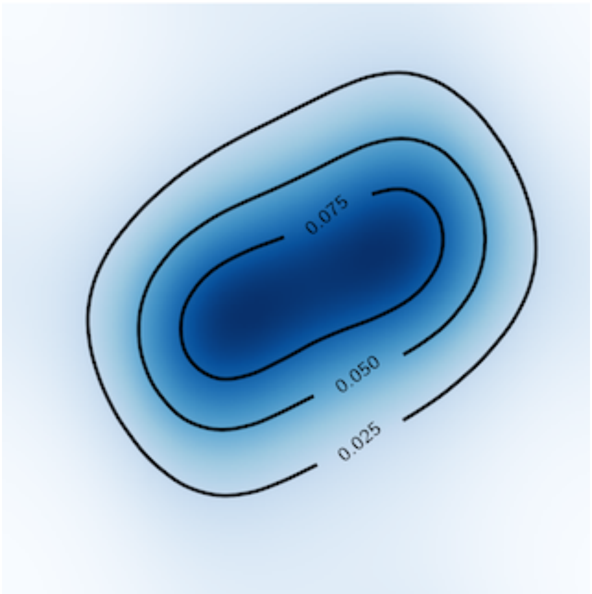


Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

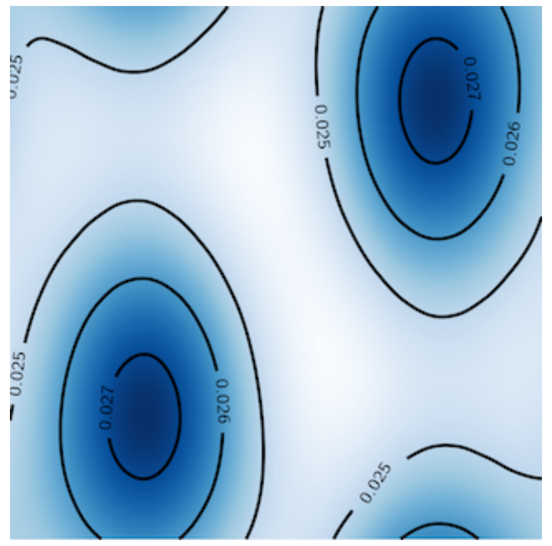
First order Case Study

$$\inf_{u \in \mathcal{U}} \quad \mathbb{E}_{\mu^u}\left[\int_0^T \frac{1}{2} u^2 \, \mathrm{d}t\right],$$

$$\mathrm{d}\boldsymbol{\theta} = \left(-\nabla_{\boldsymbol{\theta}} V(\boldsymbol{\theta}) + \boldsymbol{S}\boldsymbol{u}\right) \mathrm{d}t + \sqrt{2}\boldsymbol{S}\mathrm{d}\boldsymbol{w}$$



$\boldsymbol{\theta}(t=0) \sim \mu_0$ (Desynchronized)



$\boldsymbol{\theta}(t=T) \sim \tilde{\mu}_T$ (Synchronized)

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

Uncontrolled forward-backward Kolmogorov PDEs:

$$\frac{\partial \hat{\varphi}}{\partial t} = \nabla_{\xi} \cdot \left(\hat{\varphi} \Upsilon \nabla_{\xi} \tilde{V}\right) + \Delta_{\xi} \hat{\varphi}$$

Forward Fokker-Planck PDE

$$\frac{\partial \varphi}{\partial t} = \left\langle \nabla_{\xi} \varphi, \Upsilon \nabla_{\xi} \tilde{V} \right\rangle - \Delta_{\xi} \varphi$$

Backward Fokker-Planck PDE

Initial and Terminal conditions

$$\hat{\varphi}_0(\boldsymbol{\xi})\varphi_0(\boldsymbol{\xi}) = \rho_0(\boldsymbol{S}\boldsymbol{\xi})\left(\prod_{i=1}^{n} \sigma_i\right)$$

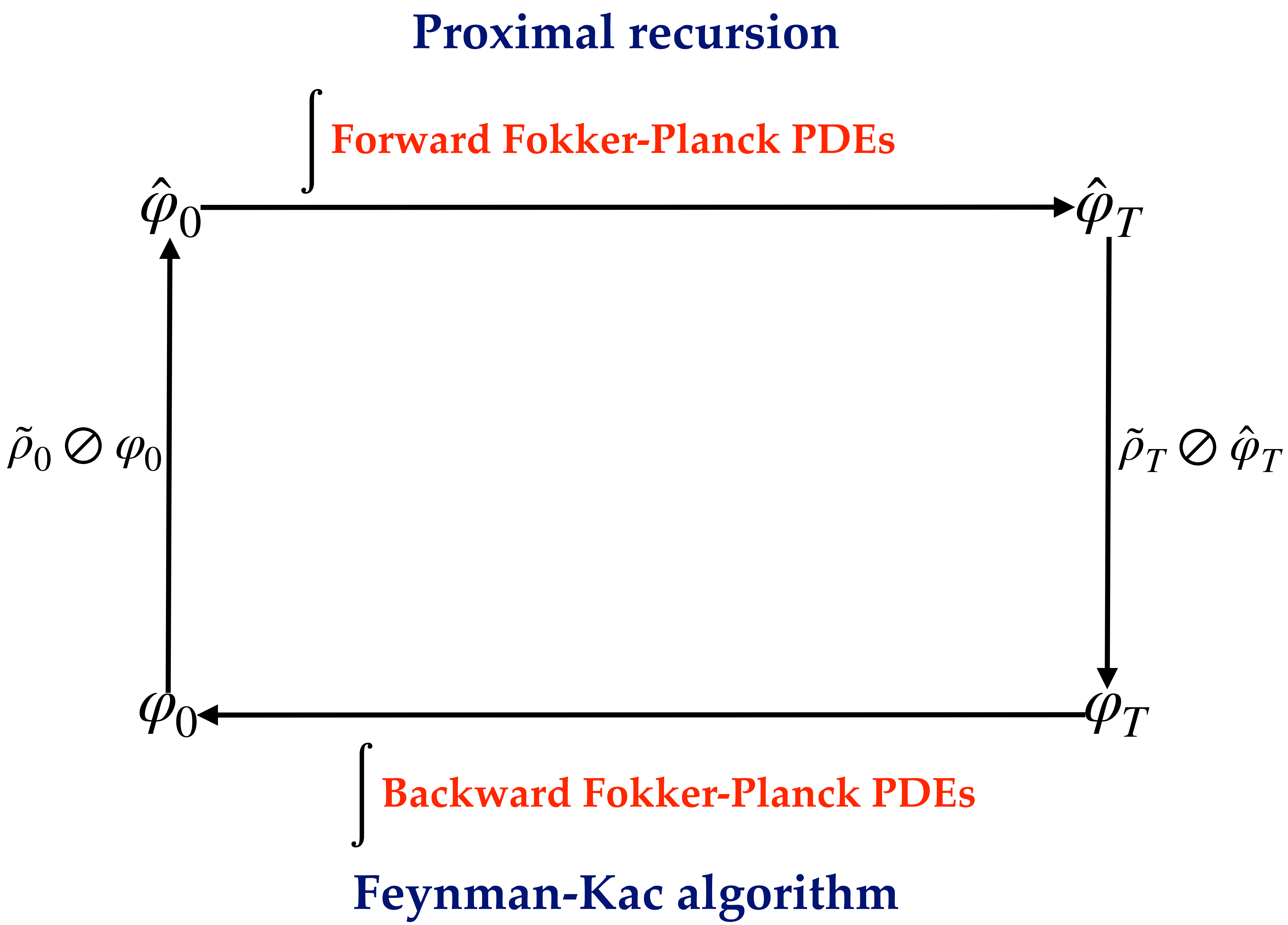
$$\hat{\varphi}_T(\boldsymbol{\xi})\varphi_T(\boldsymbol{\xi}) = \rho_T(\boldsymbol{S}\boldsymbol{\xi})\left(\prod_{i=1}^{n} \sigma_i\right)$$

Optimal controlled joint state PDF: $$\rho^{\text{opt}}(\boldsymbol{\theta}, t) = \hat{\varphi}(\boldsymbol{S}^{-1}\boldsymbol{\theta}, t)\varphi(\boldsymbol{S}^{-1}\boldsymbol{\theta}, t) / \left(\prod_{i=1}^{n} \sigma_i\right)$$

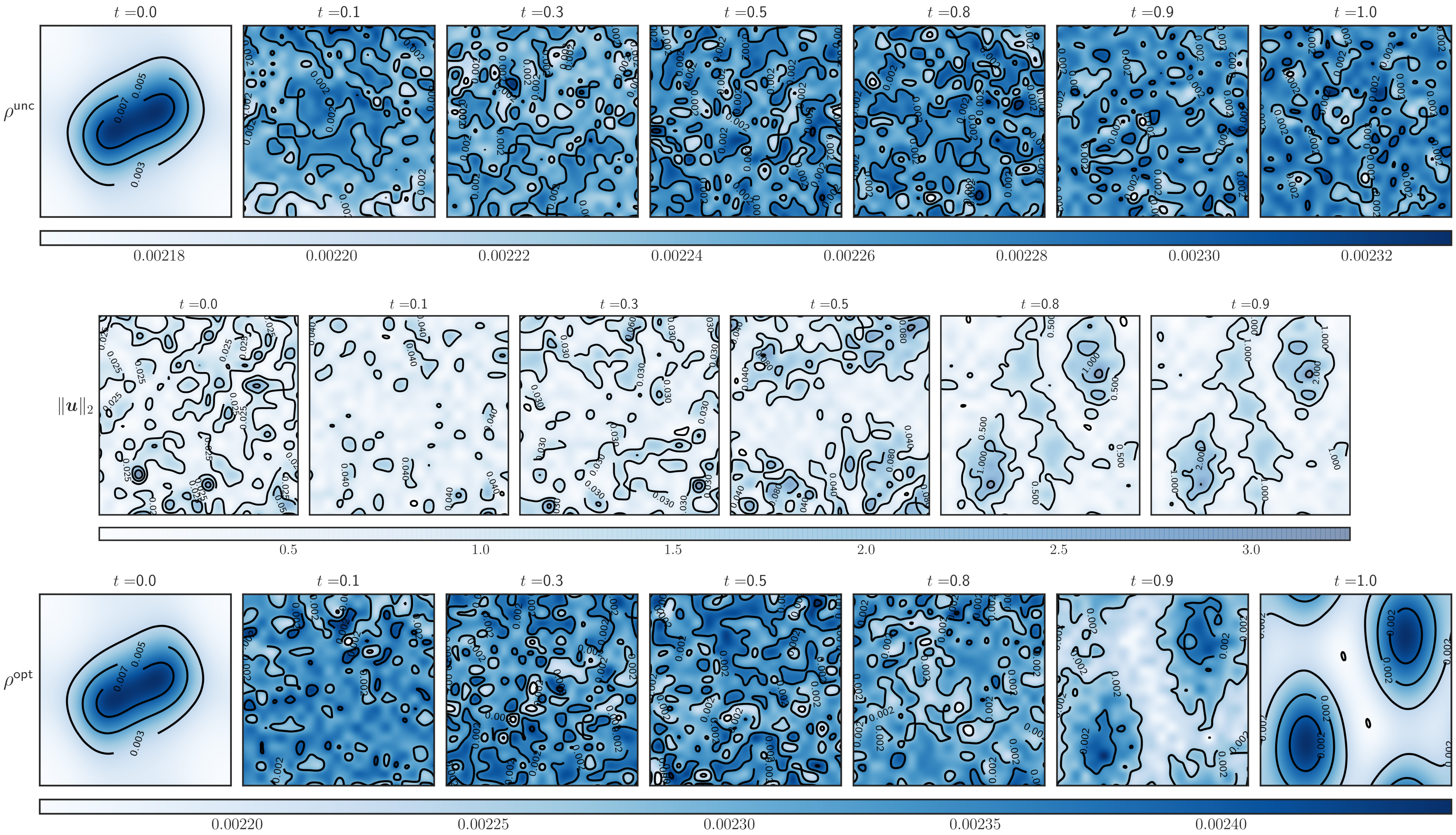
Optimal control: $$\boldsymbol{u}^{\text{opt}}(\boldsymbol{\theta}, t) = \boldsymbol{S}\, \nabla_{\boldsymbol{\theta}} \log \varphi\left(\boldsymbol{S}^{-1}\boldsymbol{\theta}, t\right)$$

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

Fixed Point Recursion Over Pair $(\varphi, \hat{\varphi})$

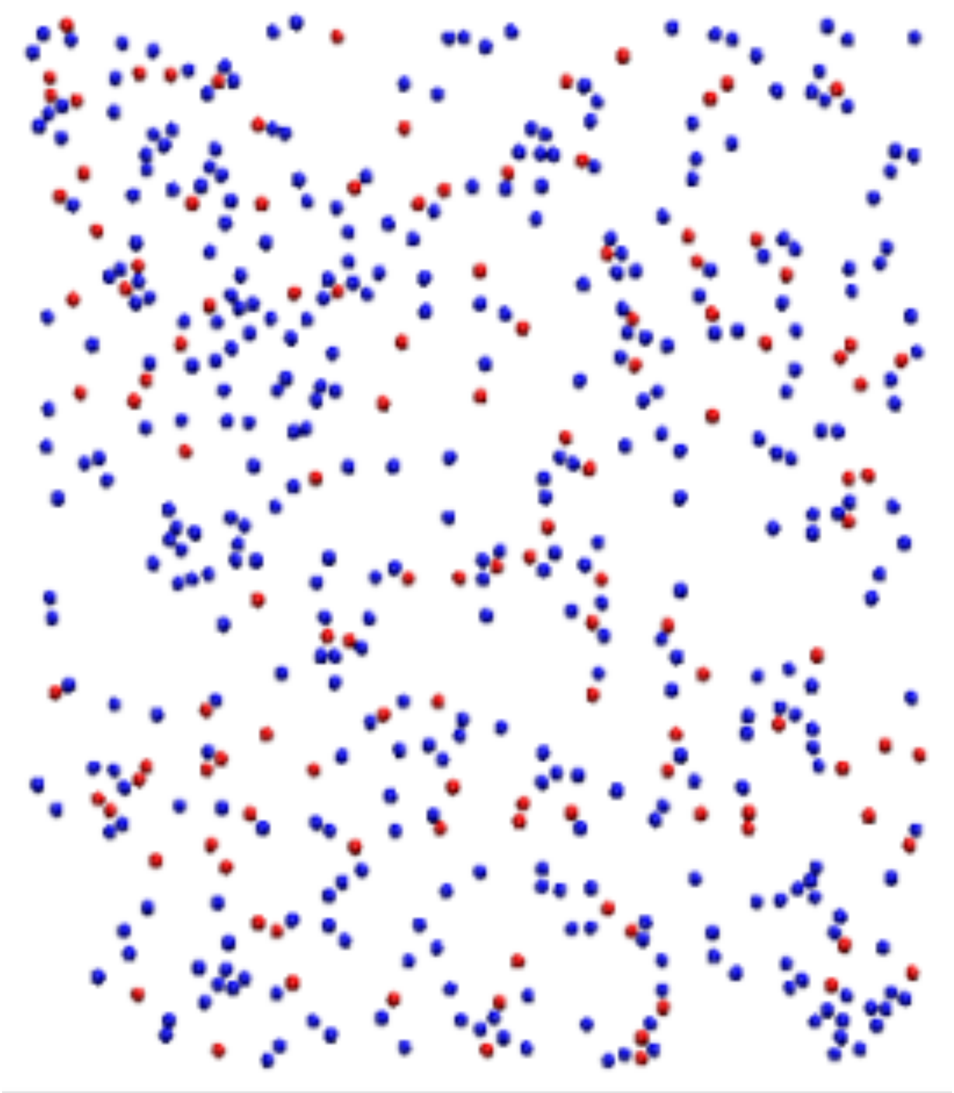


Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

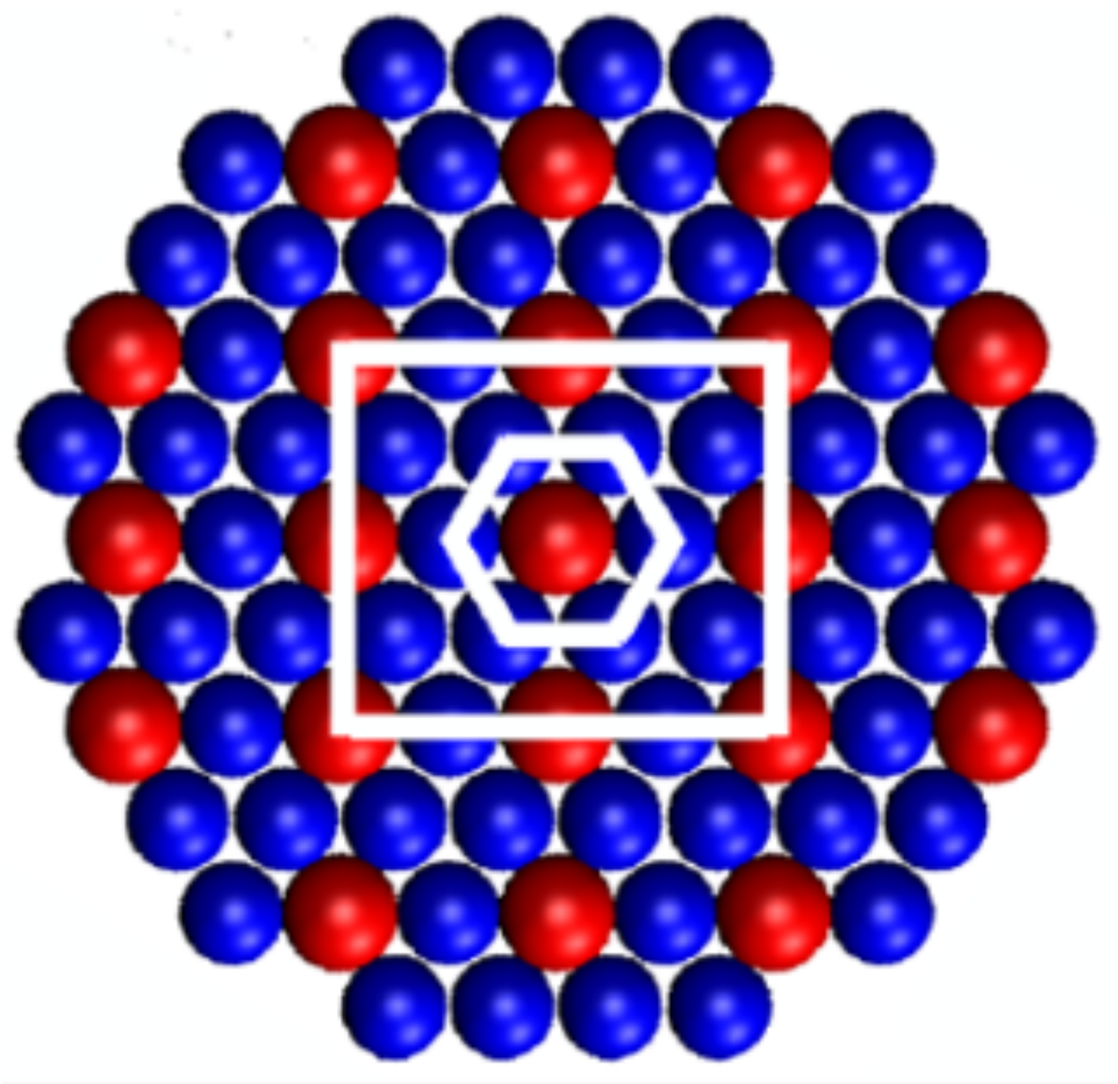


Stochastic Control / Control Non-affine

Controlled Self-assembly



Dispersed particles

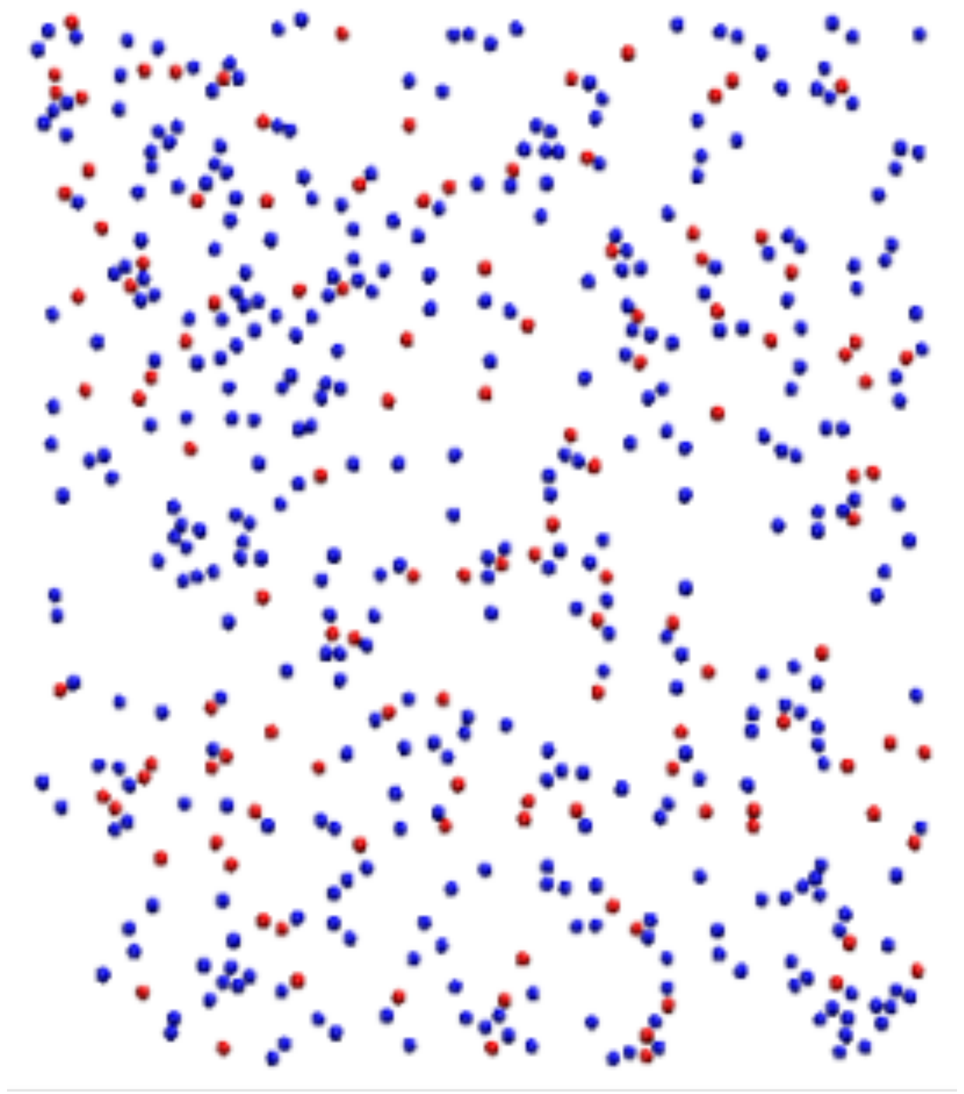


Ordered structure

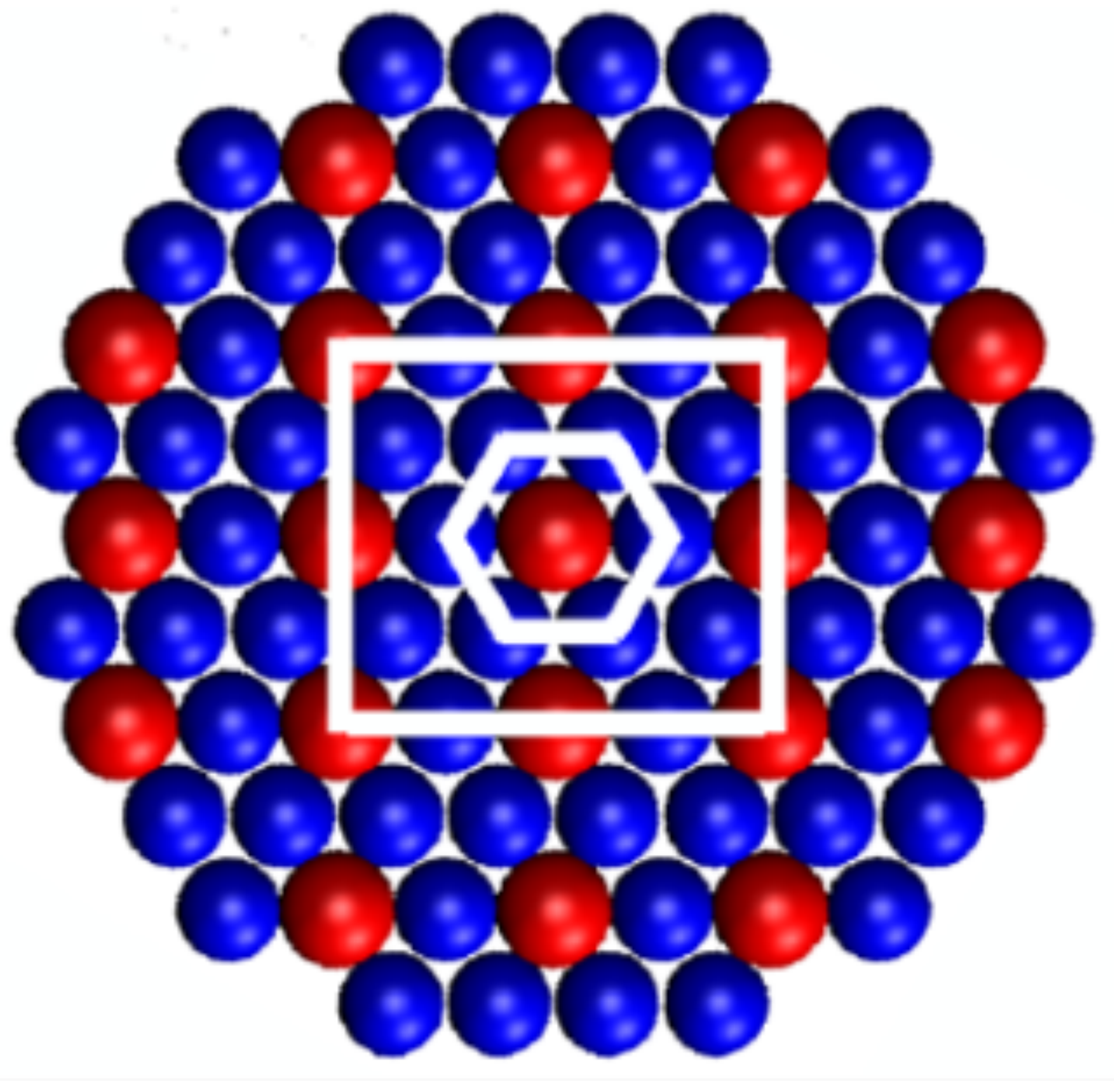
Applications: Precision (e.g., sub nm scale) manufacturing of materials with advanced electrical, magnetic or optical properties

Two controlled colloidal SA case studies: (1) model-based, (2) data-driven

Controlled Self-assembly Case Study 1: Model Based



Dispersed particles



Ordered structure

Typical state variable: $\langle C_6 \rangle \in (0,6)$

Average number of hexagonally close packed neighboring particles in 2D assembly ⇝ measure of crystallinity order

Typical control variable: u

Electric field voltage

Technical challenge:

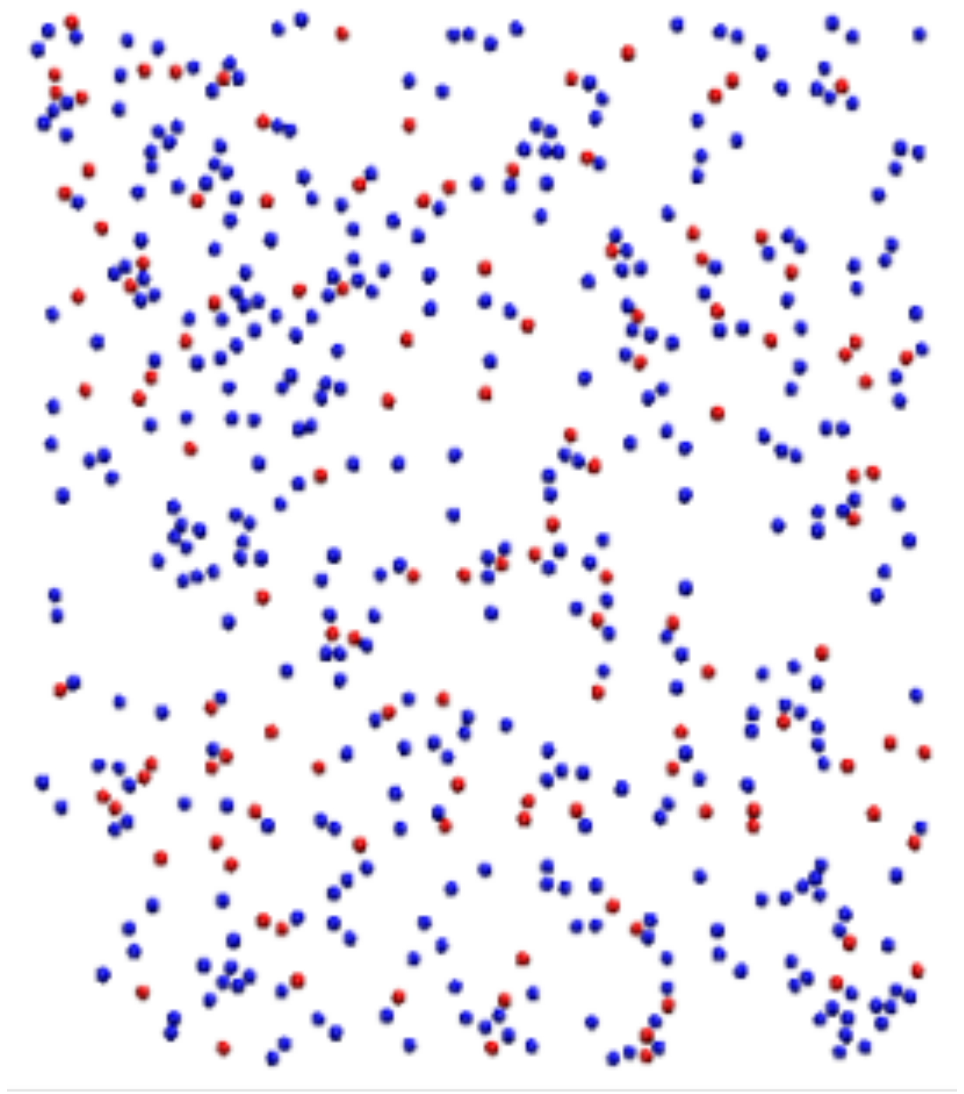
Nonlinear+ noisy molecular dynamics ⇝ $\langle C_6 \rangle$ is a controlled stochastic process

Controlled Self-assembly Case Study 2: Data Driven

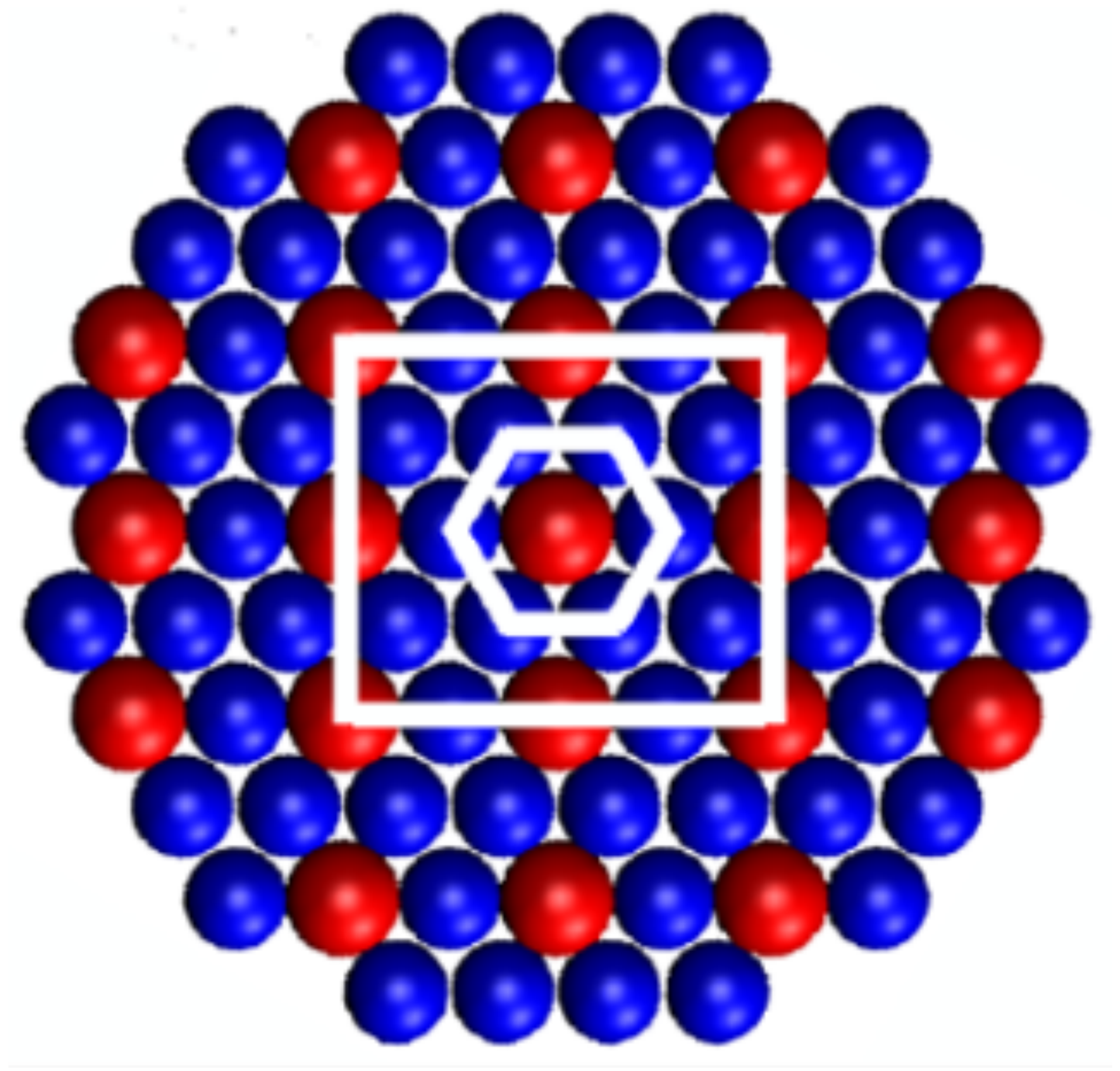
Typical state variable: $(\langle C_{10} \rangle, \langle C_{12} \rangle) \in [0,1]^2$

Steinhart bond order parameters

useful for distinguishing
between BCC and FCC structures



Dispersed particles



Ordered structure

Typical control variable: u

$(u_1, u_2) =$ (temperature, pressure)

Technical challenge:

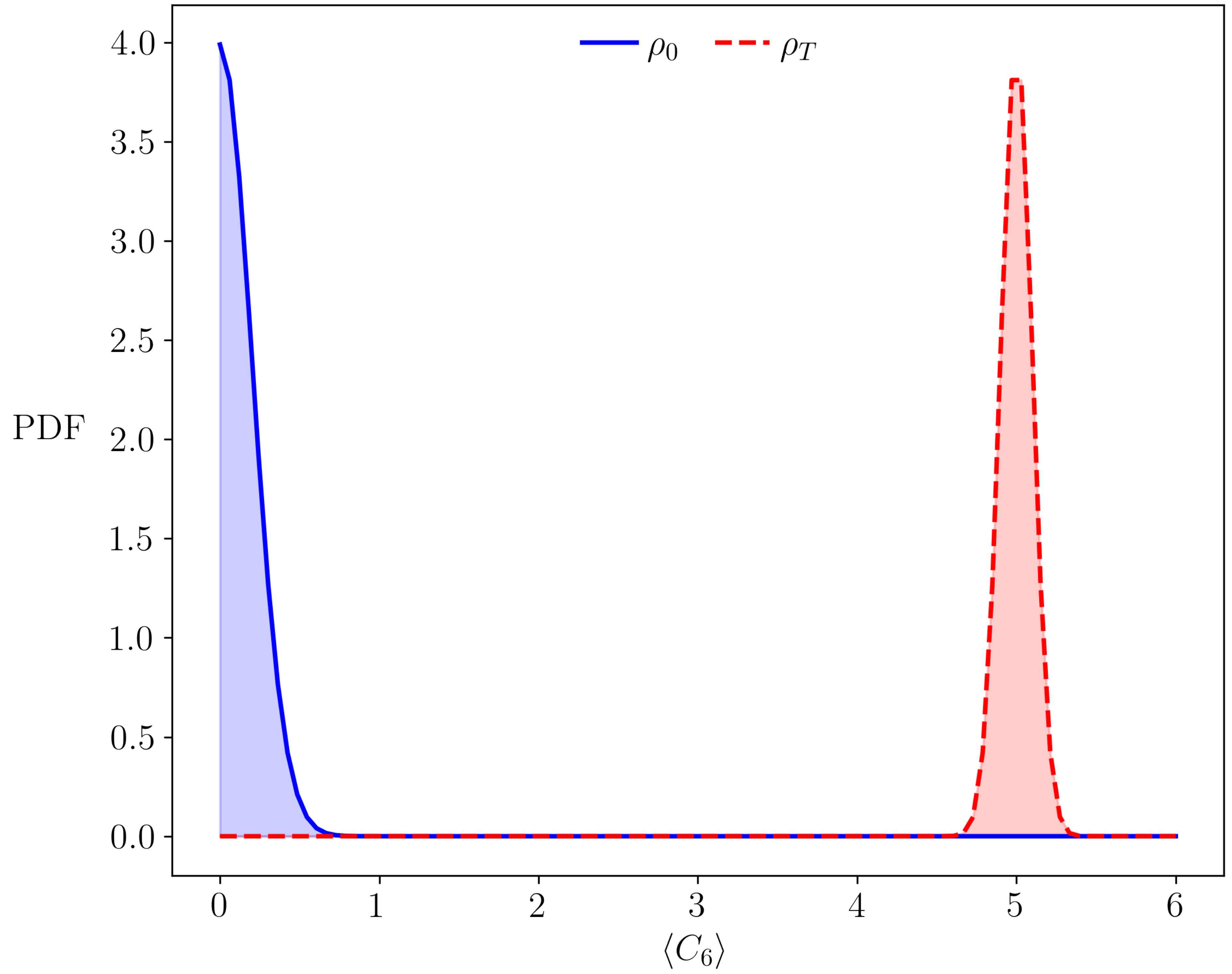
Difficult to deduce first principle physics-based controlled dynamics over $(\langle C_{10} \rangle, \langle C_{12} \rangle)$

Intuition for Case Study 1:

$\langle C_6 \rangle \approx 0 \;\Leftrightarrow$ Crystalline disorder

$\langle C_6 \rangle \approx 5 \;\Leftrightarrow$ Crystalline order

Steer the PDF of the stochastic state $\langle C_6 \rangle$ from disordered at $t = t_0 \equiv 0$ to ordered at $t = T \equiv 200\,\mathrm{s}$



Typical prescribed finite horizon for controlled self-assembly

Endpoint PDF constraints: $\langle C_6 \rangle(t = t_0) \sim \rho_0$ (given)

$\langle C_6 \rangle(t = T) \sim \rho_T$ (given)

Control policy to accomplish the PDF steering: $u = \pi\left(\langle C_6 \rangle, t\right)$

Underdetermined

Case 1: Minimum Effort Self-assembly

Proposed formulation:

$$\inf_{u \in \mathscr{U}} \quad \mathbb{E}_{\mu^u}\left[\int_0^T \frac{1}{2}u^2 \, \mathrm{d}t\right],$$

drift landscape — diffusion landscape — free energy landscape

$$D_1(x^u, u) := \frac{\partial}{\partial x} D_2(x^u, u) - \frac{\partial}{\partial x} F(x^u, u) \frac{D_2(x^u, u)}{k_B \theta}$$

$D_2(x^u, u)$: either from model or learnt from MD simulation data

subject to $\mathrm{d}x^u = D_1(x^u, u)\,\mathrm{d}t + \sqrt{2D_2(x^u, u)}\,\mathrm{d}w,$

x^u: $\langle C_6 \rangle$

w: standard Wiener process

$$x^u(t=0) \sim \mathrm{d}\mu_0 = \rho_0\,\mathrm{d}x^u, \quad x^u(t=T) \sim \mathrm{d}\mu_T = \rho_T\,\mathrm{d}x^u$$

Stochastic Control / Control Non-affine

Case 1: Minimum Effort Self-assembly

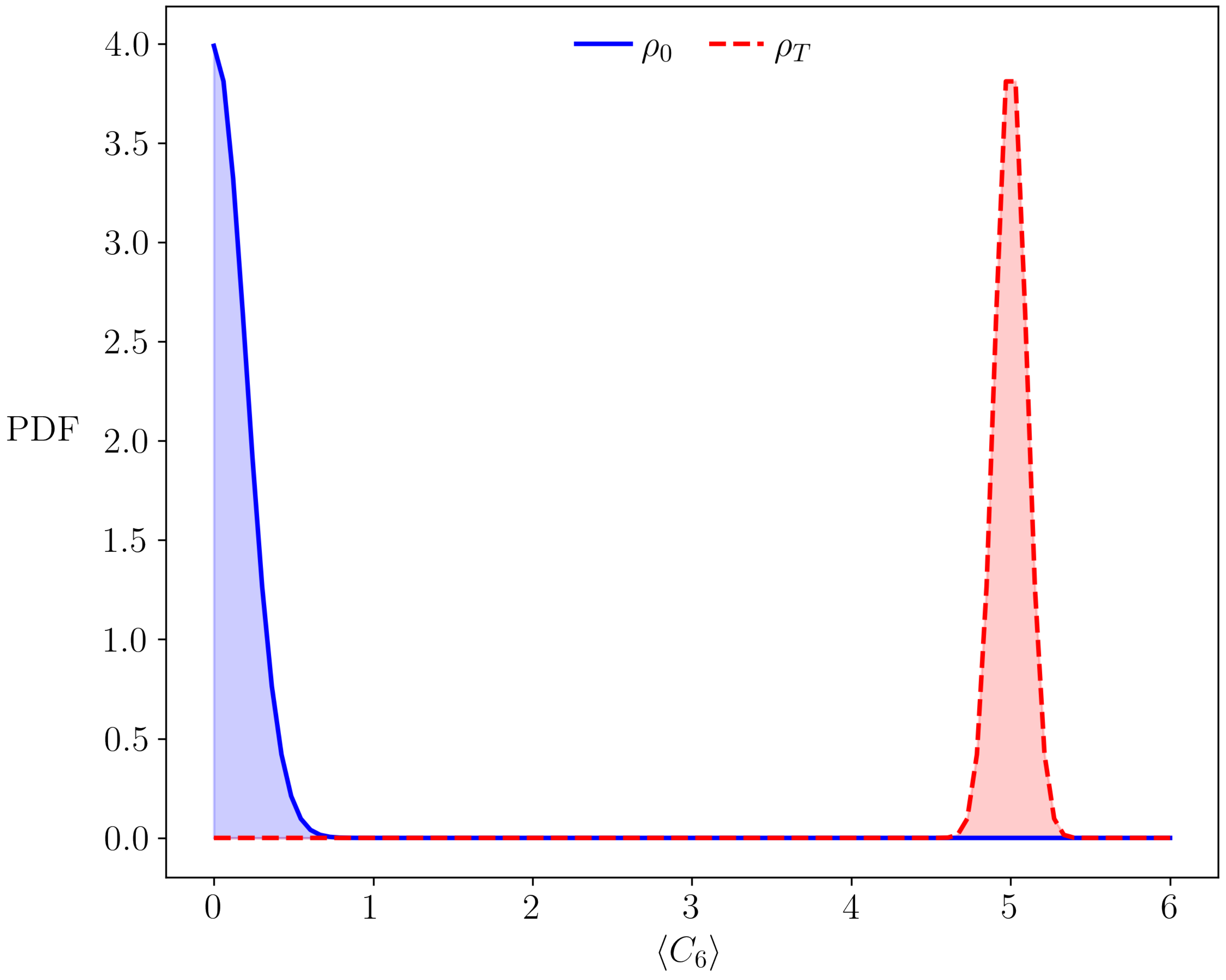
Equivalent formulation:

$$\inf_{(\rho^u, u)} \int_0^T \int_{\mathbb{R}} \frac{1}{2} u^2(x^u, t)\, \rho^u(x^u, t)\, \mathrm{d}x^u\, \mathrm{d}t$$

subject to $$\frac{\partial \rho^u}{\partial t} = -\frac{\partial}{\partial x^u}\left(D_1 \rho^u\right) + \frac{\partial^2}{\partial x^{u2}}\left(D_2 \rho^u\right)$$

$$\rho^u(x^u, t=0) = \rho_0, \quad \rho^u(x^u, t=T) = \rho_T$$

Guaranteed existence-uniqueness for compactly supported ρ_0, ρ_T



Case 1: Conditions for Optimality

$$\frac{\partial \psi}{\partial t} = \frac{1}{2}\left(\pi^{\text{opt}}\right)^2 - \frac{\partial \psi}{\partial x} D_1 - \frac{\partial^2 \psi}{\partial x^{u2}} D_2$$ **HJB PDE**

$$\frac{\partial \rho^u}{\partial t} = -\frac{\partial}{\partial x^u}\left(D_1 \rho^u\right) + \frac{\partial^2}{\partial x^{u2}}\left(D_2 \rho^u\right)$$ **Controlled FPK PDE**

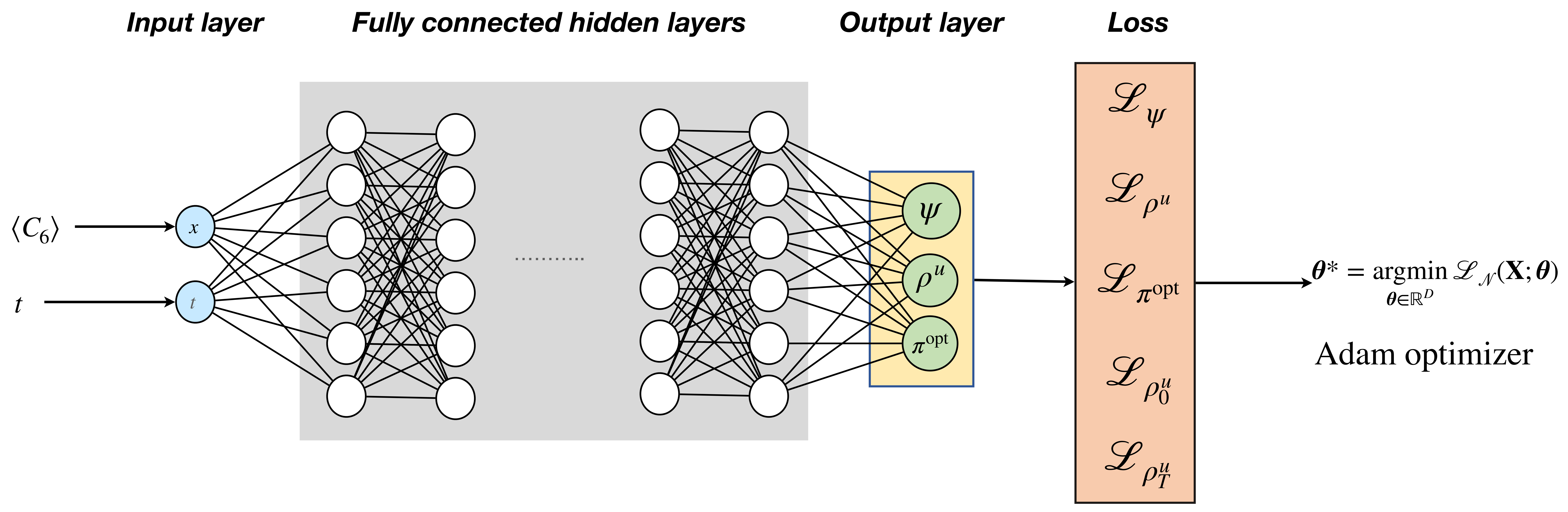
$$\pi^{\text{opt}}(x^u, t) = \frac{\partial \psi}{\partial x^u}\frac{\partial D_1}{\partial u} + \frac{\partial^2 \psi}{\partial x^{u2}}\frac{\partial D_2}{\partial u}$$ **Optimal policy**

$$\rho^u(x^u, t=0) = \rho_0, \quad \rho^u(x^u, t=T) = \rho_T$$ **Boundary conditions**

value function	optimally controlled PDF	optimal policy

To be solved for the triple: $\psi\left(x^u, t\right), \rho^u\left(x^u, t\right), \pi^{\text{opt}}\left(x^u, t\right)$

Case 1: Train Physics Informed Neural Network (PINN) to Learn the Solution of the GSBP



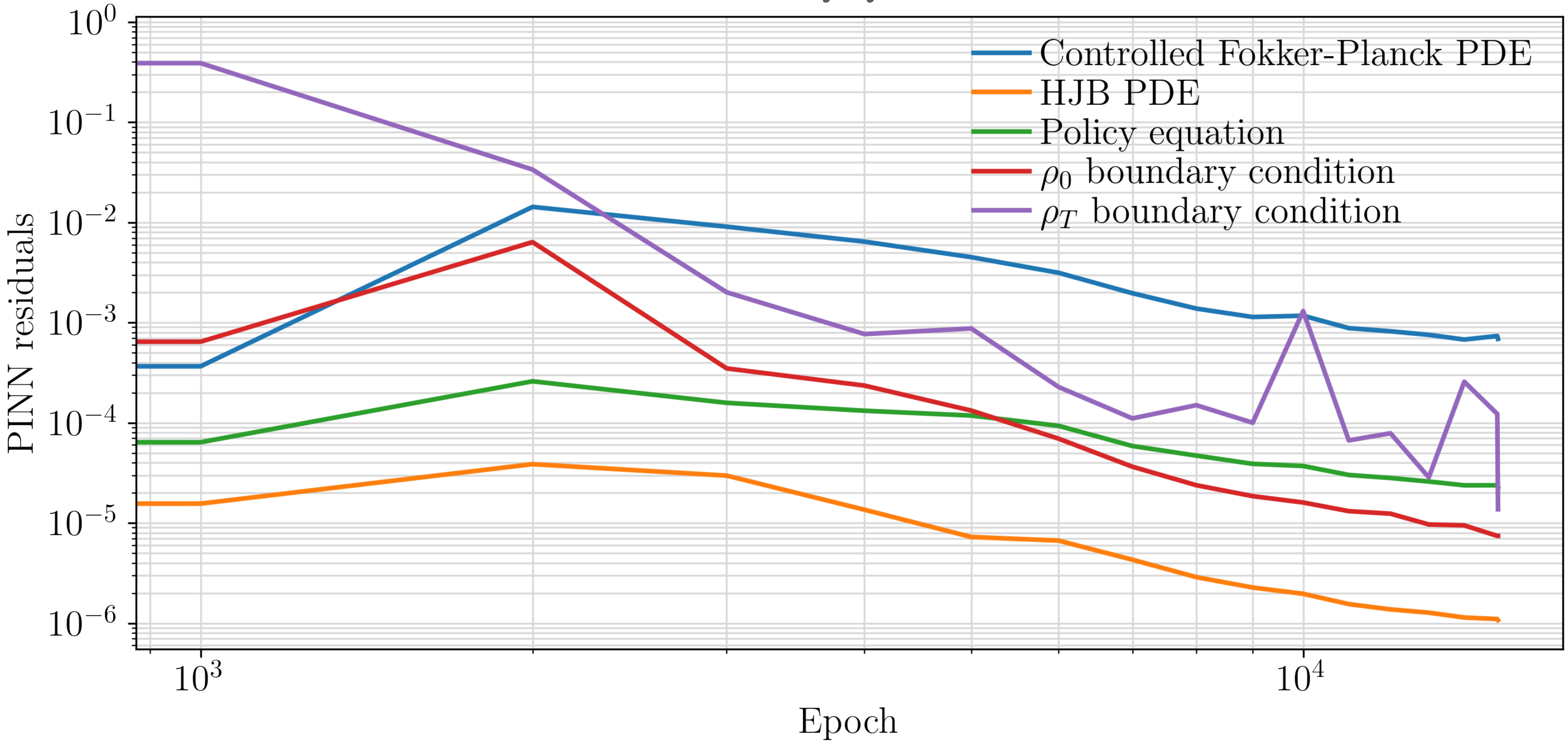
$$\mathscr{L}_{\mathcal{N}} = \mathscr{L}_\psi + \mathscr{L}_{\rho^u} + \mathscr{L}_{\pi^{\text{opt}}} + \mathscr{L}_{\rho_0^u} + \mathscr{L}_{\rho_T^u}$$

[Lu Lu.et al, 2021] [Niaki.et al, 2021]

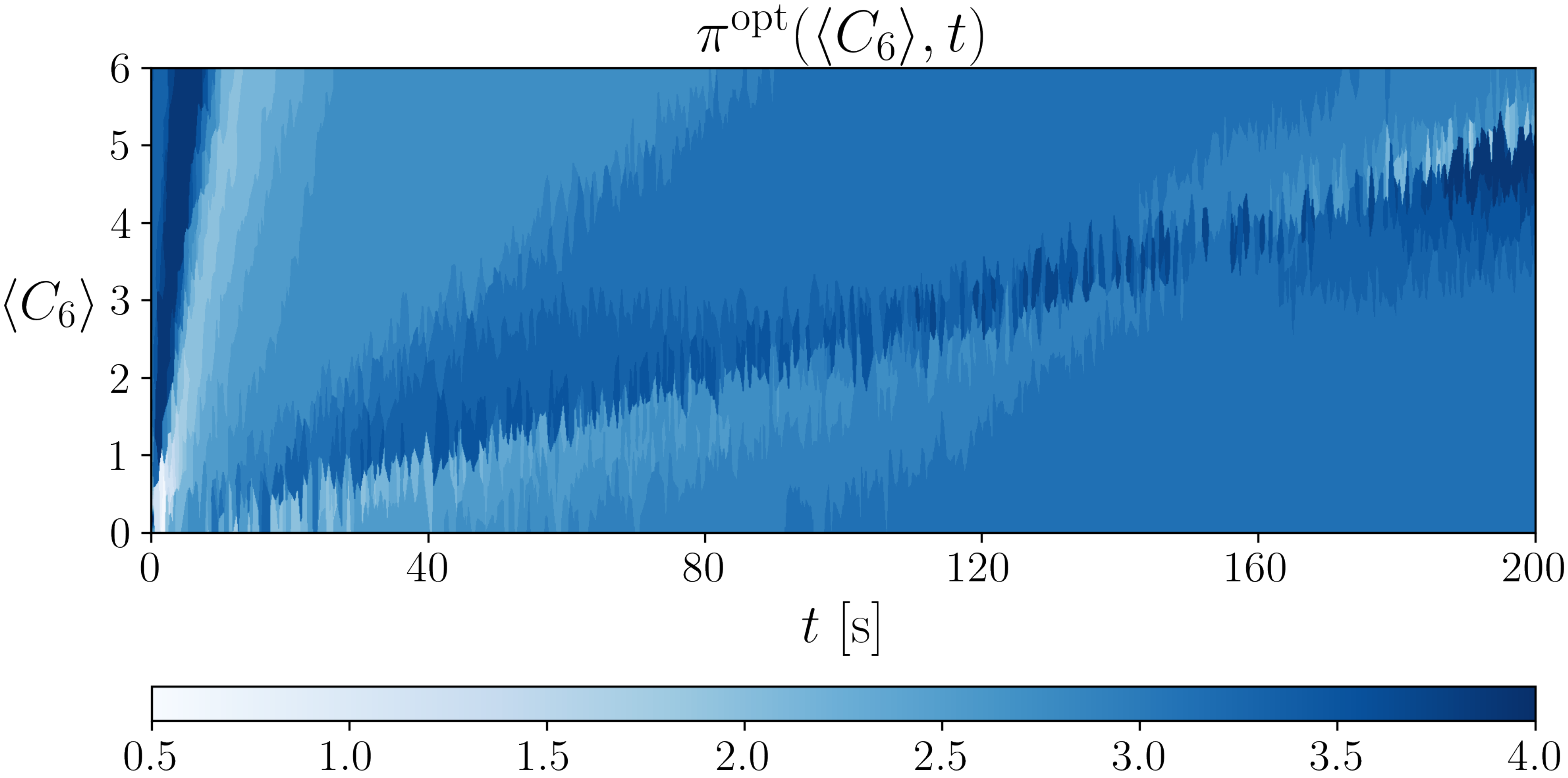
Stochastic Control / Control Non-affine

Case 1: Residual for PINN Training

Benchmark controlled self-assembly system: [Y Xue, et al, *IEEE Trans. Control Sys. Technology*, 2014]

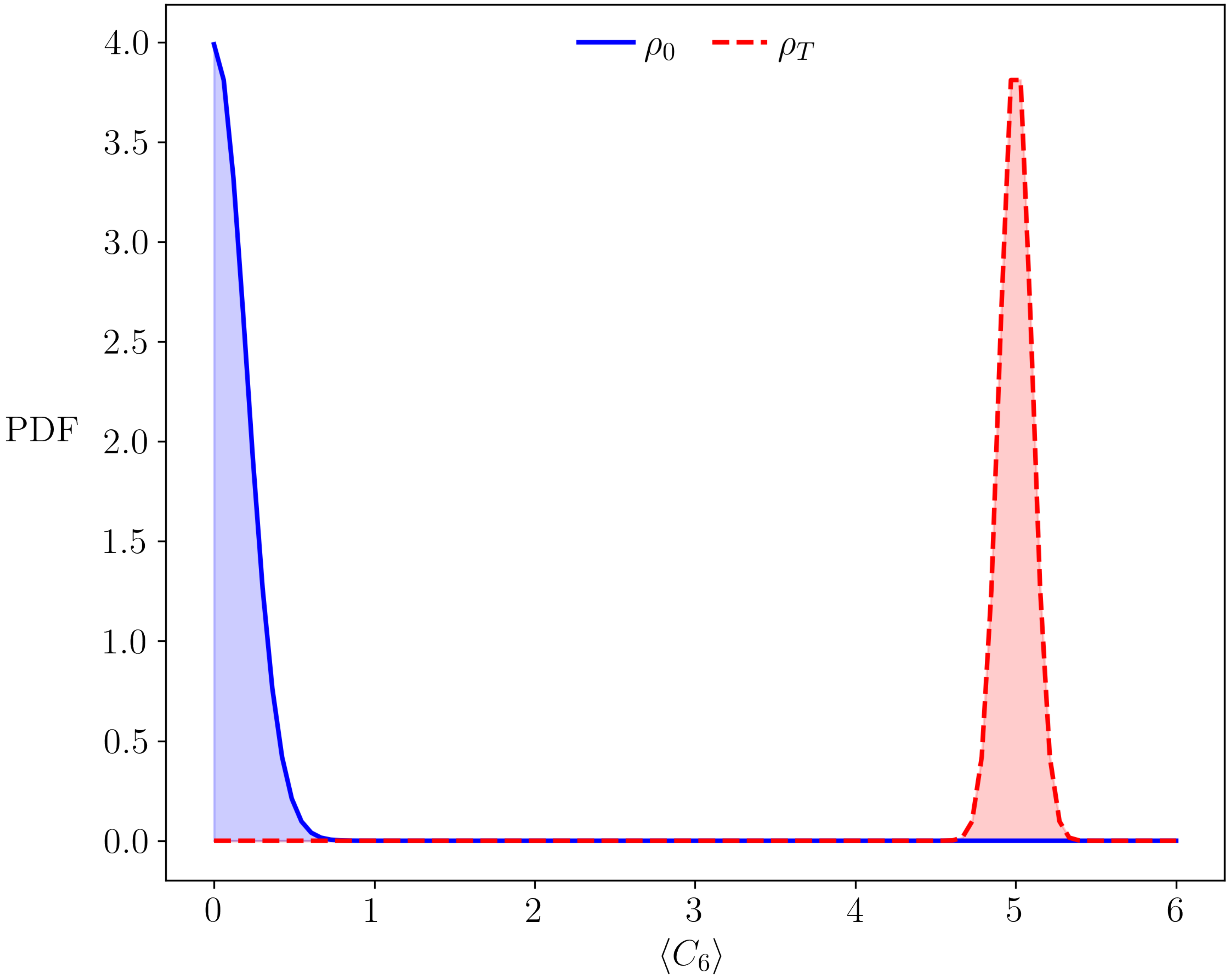


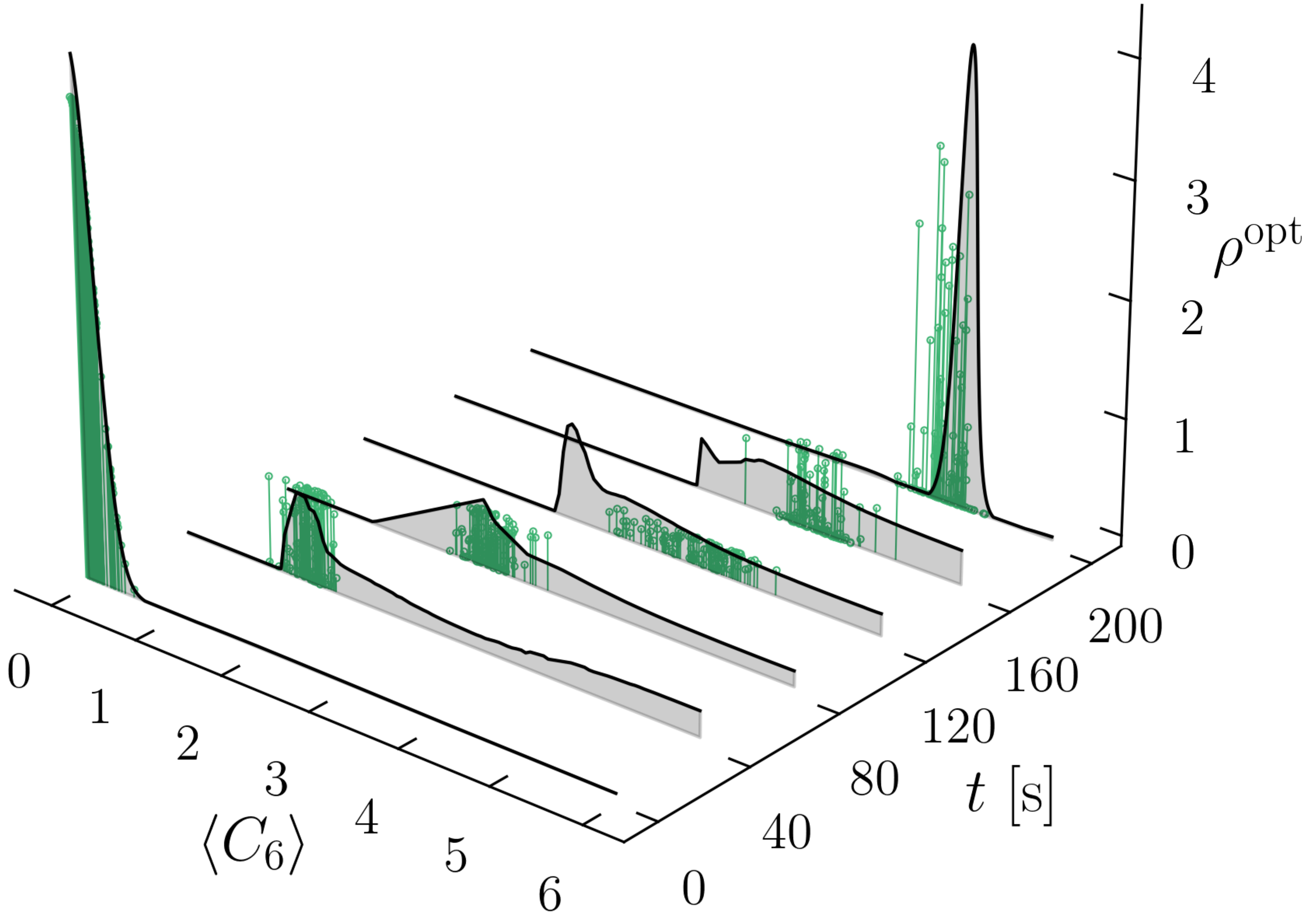
Case 1: Optimal Policy



Stochastic Control / Control Non-affine

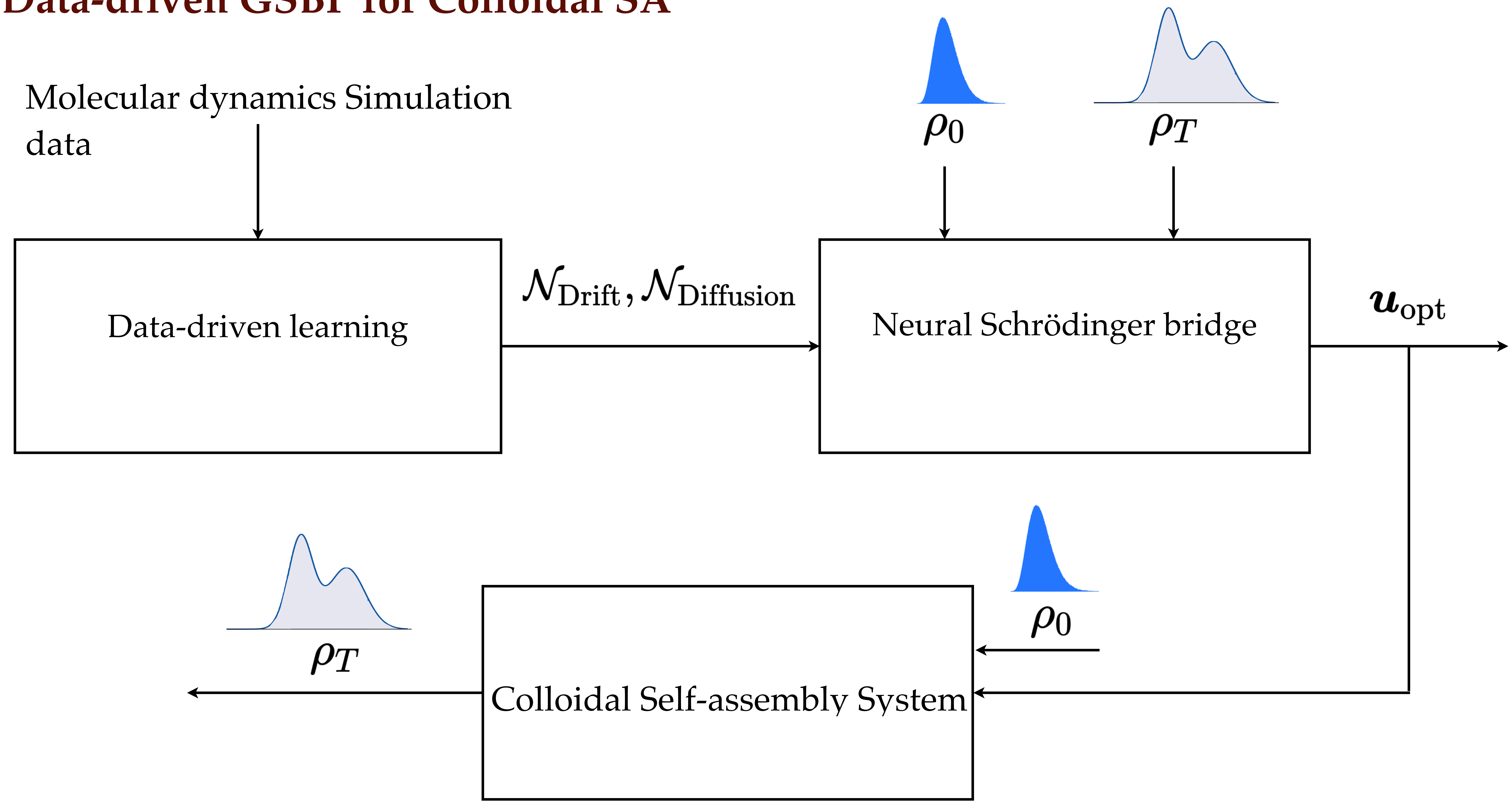
Case 1: Optimally Controlled State PDFs



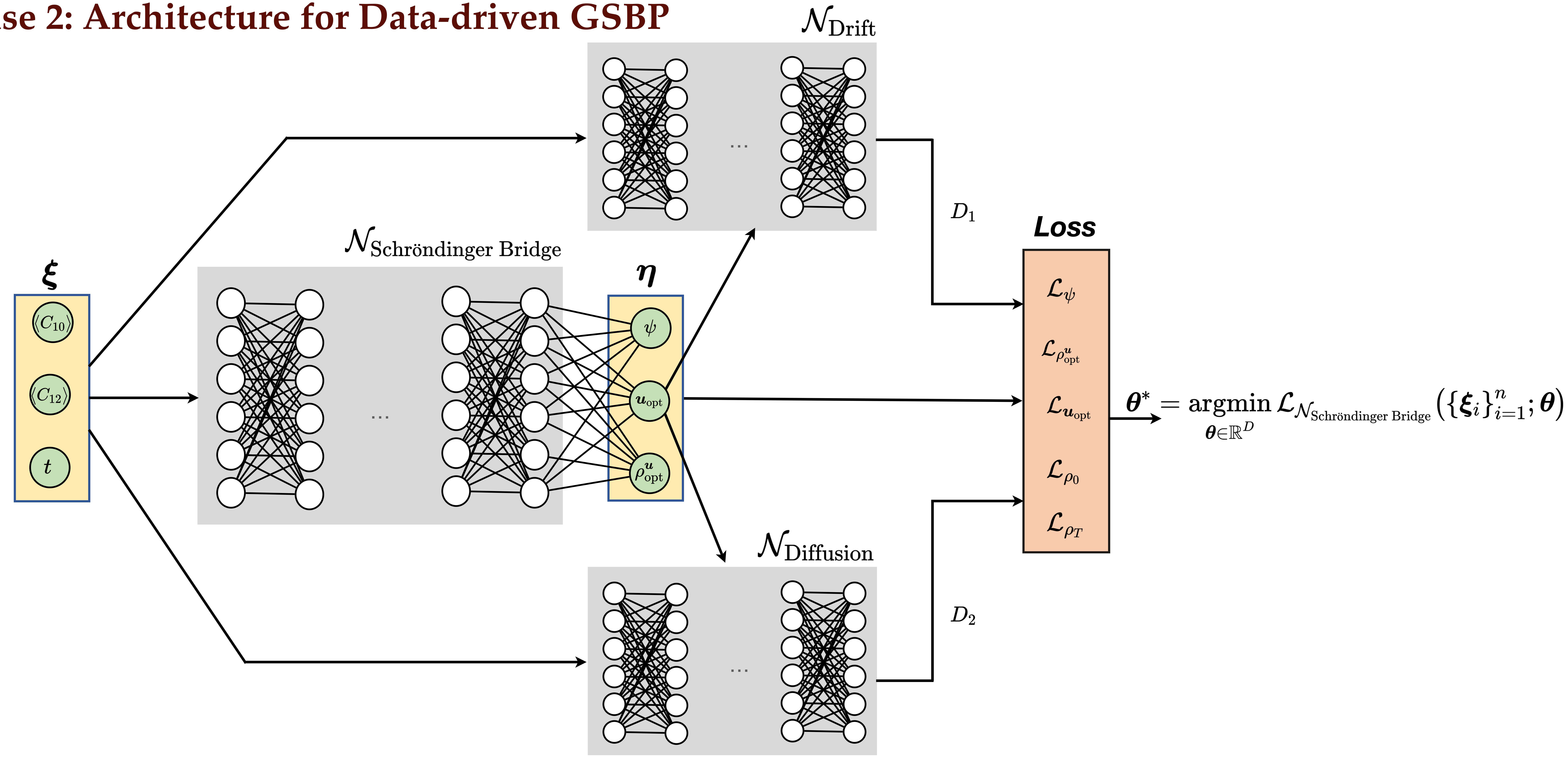


... the MSE losses are not appropriate for enforcing the endpoint PDF constraints

Case 2: Data-driven GSBP for Colloidal SA



Case 2: Architecture for Data-driven GSBP



Case 2: Sinkhorn Losses for Boundary Conditions

$$W_{\varepsilon}^{2}(\mu_0,\mu_1) := \inf_{\pi\in\Pi_2(\mu_0,\mu_T)} \int_{\mathbb{R}^n\times\mathbb{R}^n} \left\{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2 + \varepsilon\log\pi(\boldsymbol{x},\boldsymbol{y})\right\}\mathrm{d}\pi(\boldsymbol{x},\boldsymbol{y})$$

For boundary conditions, use Sinkhorn losses: $\mathcal{L}_{\rho_i} := W_{\varepsilon}^{2}\left(\rho_i, \rho_i^{\text{epoch index}}(\boldsymbol{\theta})\right)$

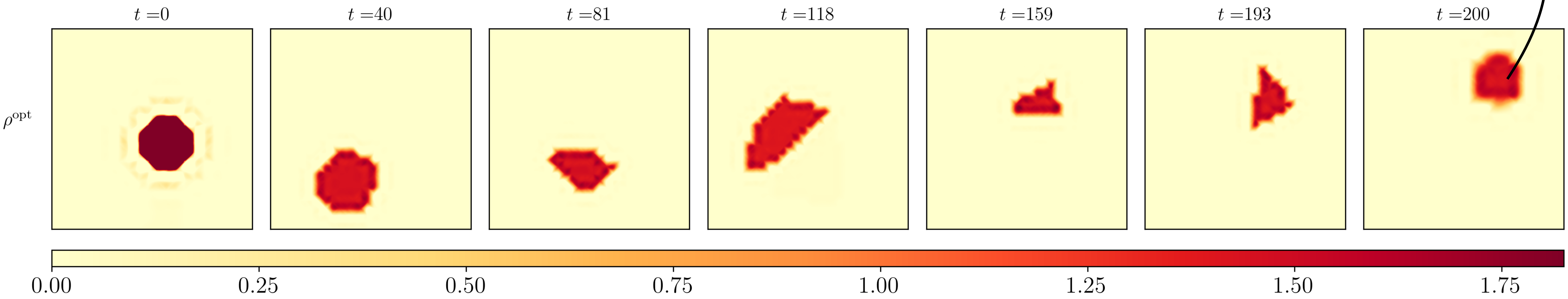
Implementation friendly for PINN training:

$$\text{Autodiff}_{\boldsymbol{\theta}}\, W_{\varepsilon}^{2}\left(\rho_i, \rho_i^{\text{epoch index}}(\boldsymbol{\theta})\right) \quad \forall i\in\{0,T\}$$

Stochastic Control / Control Non-affine

Case 2: Synthesize BCC Crystalline Structure by PDF Steering in $(\langle C_{10}\rangle, \langle C_{12}\rangle)$ Space

Body-centered cube (BCC) crystal



Data-driven:

Uses PINN with Sinkhorn losses + the drift-diffusion are themselves NNs

Part II: Stochastic Modeling and Solving of Chiplet Population Dynamics

Stochastic Modeling

Model dynamics of "chiplet population": large ensemble of micro/nano sized particles immersed in dielectric fluid

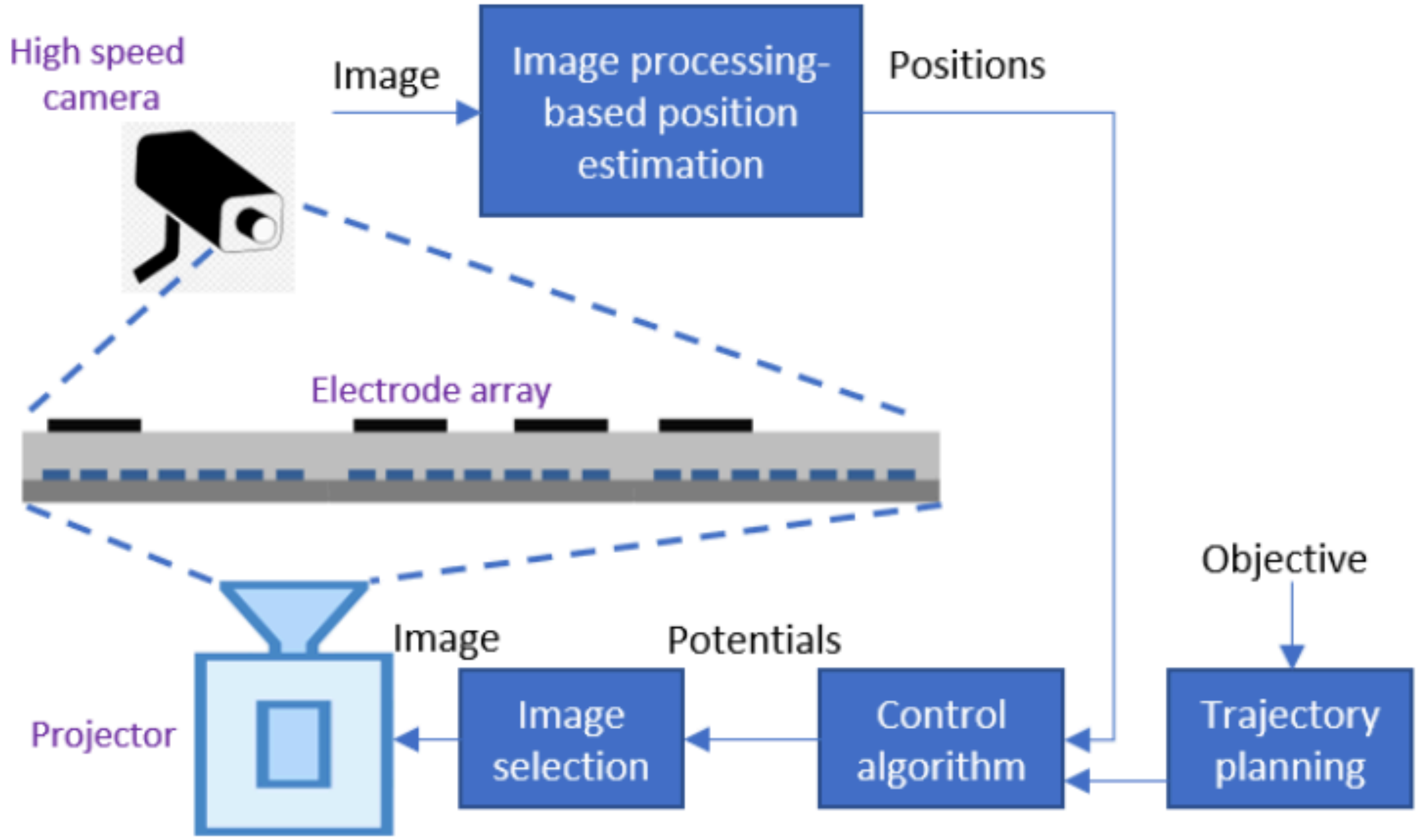


Image credit: PARC

Motivating applications

Xerographic micro-assembly for printer systems

Manufacturing of photovoltaic solar cells

Actuation and control

Electric potential generated by very large array of small electrodes

⇣

Spatio-temporally non-uniform dielectrophoretic forces on the chiplets

Stochastic Modeling

Main Idea: derive controlled dynamics in the limit both

electrodes and # chiplets $\to \infty$

Derived model

2D position of an individual chiplet: $\boldsymbol{x}(t) \in \mathbb{R}^2$

Electric voltage — Typically [-400, 400] Volt

Causal deterministic control policy: $u : \mathbb{R}^2 \times [0, \infty) \mapsto [u_{\min}, u_{\max}] \subset \mathbb{R}$

At low Reynold's number in dielectric fluid (ignoring small mass of chiplet):

$$\underbrace{\mu \dot{\boldsymbol{x}}}_{\text{viscons drag force}} = \underbrace{\boldsymbol{f}^u}_{\text{controlled interaction force}} + \text{noise}$$

At time t, normalized chiplet population density function (PDF): $\rho(\boldsymbol{x}, t) \in \mathcal{P}_2(\mathbb{R}^2)$

The vector field: $\boldsymbol{f}^u : \mathbb{R}^2 \times [0, \infty) \times \mathcal{U} \times \mathcal{P}_2(\mathbb{R}^2) \mapsto \mathbb{R}^2$

Stochastic Modeling

Derived model: nonlocal Itô SDE

W.l.o.g. viscous coefficient $\mu = 1$ (else re-scale vector field)

Itô SDE for the i th chiplet:

$$\mathrm{d}\boldsymbol{x}_i = \boldsymbol{f}^u(\boldsymbol{x}_i, t, u, \rho^n)\mathrm{d}t + \sqrt{2\beta^{-1}}\,\mathrm{d}\boldsymbol{w}_i(t) \quad \text{with i.i.d. } \boldsymbol{x}_{0i} \sim \rho_0 \in \mathcal{P}_2\left(\mathbb{R}^2\right) \quad \forall i \in [\![n]\!],$$

$$\rho^n := \frac{1}{n}\sum_{i=1}^{n} \delta_{\boldsymbol{x}_i}$$

Standard Wiener process

Non-local vector field:

$$\boldsymbol{f}^u(\boldsymbol{x}, t, u, \rho) = -\nabla\left(\int_{\mathbb{R}^2} \phi^u(\boldsymbol{x}, \boldsymbol{y}, t)\rho(\boldsymbol{y}, t)\mathrm{d}\boldsymbol{y}\right) = -\nabla(\rho * \phi^u)$$

Controlled interaction potential

Comma ... not minus

Generalized convolution

Stochastic Modeling

Derived model: controlled interaction potential ϕ^u

Non-local vector field: $$\boldsymbol{f}^u(\boldsymbol{x}, t, u, \rho) = -\nabla\left(\int_{\mathbb{R}^2} \phi^u(\boldsymbol{x}, \boldsymbol{y}, t)\rho(\boldsymbol{y}, t)\mathrm{d}\boldsymbol{y}\right)$$

Controlled interaction potential $= \phi^u_{\text{cc}}(\boldsymbol{x}, \boldsymbol{y}, t) + \phi^u_{\text{ce}}(\boldsymbol{x}, \boldsymbol{y}, t)$

$$\phi^u_{\text{cc}}(\boldsymbol{x}, \boldsymbol{y}, t) := C_{\text{cc}}(\|\boldsymbol{x} - \boldsymbol{y}\|_2)(\bar{u}(\boldsymbol{y}, t) - \bar{u}(\boldsymbol{x}, t))^2/2$$
$$\phi^u_{\text{ce}}(\boldsymbol{x}, \boldsymbol{y}, t) := C_{\text{ce}}(\|\boldsymbol{x} - \boldsymbol{y}\|_2)(u(\boldsymbol{y}, t) - \bar{u}(\boldsymbol{x}, t))^2/2$$

Capacitances (in practice, from COMSOL electrostatic simulation)

$$\bar{u}(\boldsymbol{x}, t) := \frac{\int_{\mathbb{R}^2} C_{\text{ce}}(\|\boldsymbol{x} - \boldsymbol{y}\|_2)u(\boldsymbol{y}, t)\rho(\boldsymbol{y}, t)\mathrm{d}\boldsymbol{y}}{\int_{\mathbb{R}^2} C_{\text{ce}}(\|\boldsymbol{x} - \boldsymbol{y}\|_2)\rho(\boldsymbol{y}, t)\mathrm{d}\boldsymbol{y}}$$

Consistency guarantee for the mean field limit

Theorem. The random empirical measure $\rho^n \rightharpoonup \rho$ a.s. in the limit $n \uparrow \infty$

where ρ solves the nonlinear McKean-Vlasov-Fokker-Planck-Kolmogorov IVP

$$\begin{aligned}\frac{\partial \rho}{\partial t} &= -\nabla \cdot (\rho \boldsymbol{f}^u) + \beta^{-1} \Delta \rho \\ &= \nabla \cdot \left(\rho \nabla \left(\rho * \phi^u + \beta^{-1}(1 + \log \rho)\right)\right)\end{aligned}$$

$$\rho(\cdot, t = 0) = \rho_0 \in \mathcal{P}\left(\mathbb{R}^2\right) \text{ (given)}.$$

Stochastic Modeling

Chiplet mean field dynamics as Wass. grad flow

Theorem. Define "energy functional" $\Phi(\rho) := \mathbb{E}_\rho\left[\rho * \phi^u + \beta^{-1}\log\rho\right]$

Then

(i) $$\frac{\partial \rho}{\partial t} = -\nabla^W \Phi(\rho)$$

(ii) $\Phi(\cdot)$ is a Lyapunov functional for the mean field dynamics.

Stochastic Modeling

Wasserstein proximal recursion

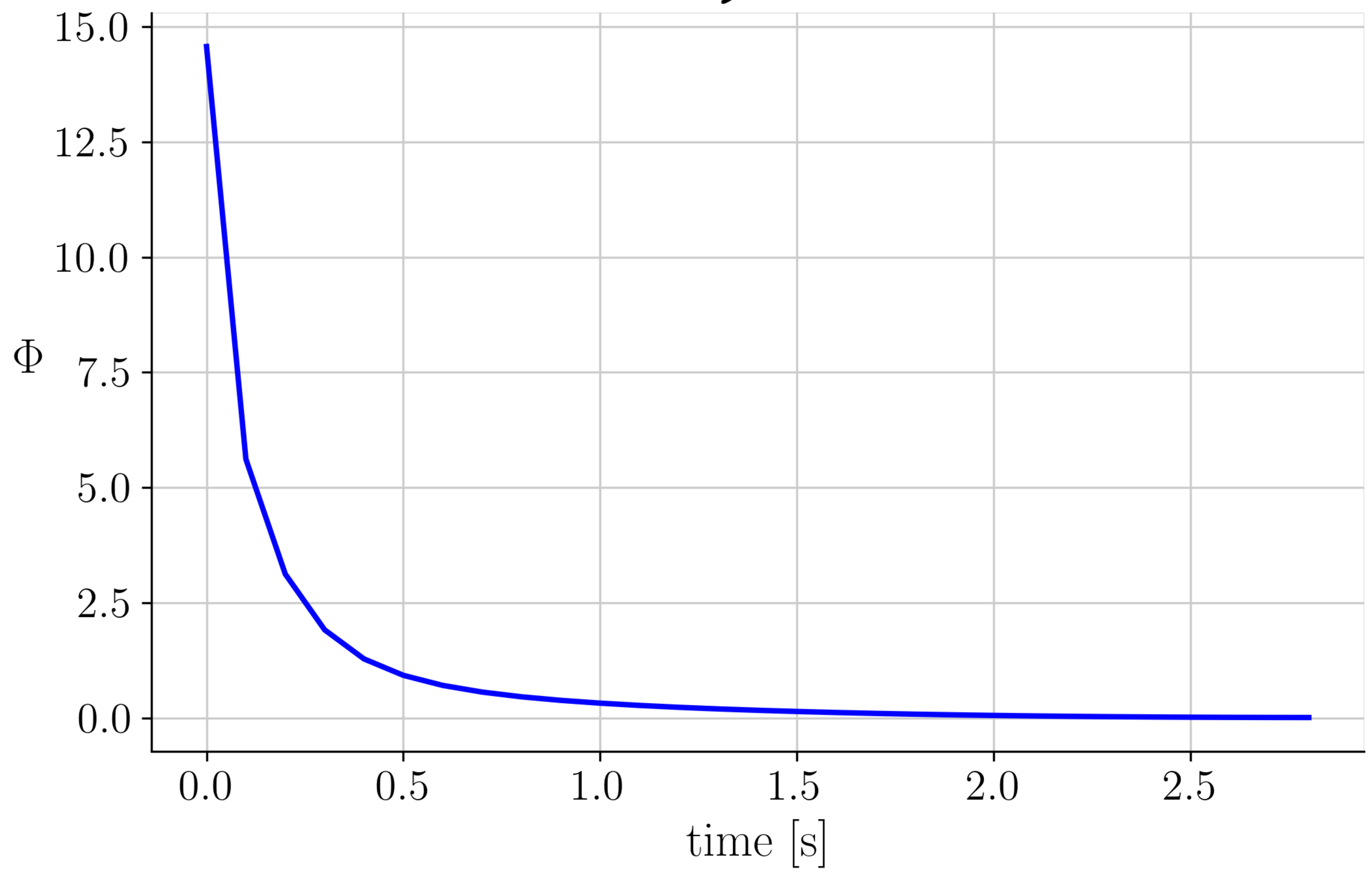
Theorem. Let $\widehat{\Phi}(\varrho, \varrho_{k-1}) := \mathbb{E}_{\varrho}\left[\varrho_{k-1} * \phi^{u} + \beta^{-1} \log \varrho\right], \varrho, \varrho_{k-1} \in \mathcal{P}_2\left(\mathbb{R}^2\right) \forall k \in \mathbb{N}$

Then the proximal recursion

$$\begin{aligned} \varrho_k &= \operatorname{prox}^{W}_{\tau\widehat{\Phi}}(\varrho_{k-1}) \\ &:= \underset{\varrho\in\mathcal{P}_2\left(\mathbb{R}^2\right)}{\arg\inf}\left\{\frac{1}{2}W^2(\varrho, \varrho_{k-1}) + \tau\widehat{\Phi}(\varrho, \varrho_{k-1})\right\} \end{aligned}$$

approximates the transient solutions of the mean

field nonlinear PDE IVP



Part III: Stochastic Learning

Centralized Computing Can Become Intensive: Mean Field SGD Dynamics in NN Classification

Free energy functional $F(\rho) := R(\hat{f}(\boldsymbol{x}, \rho))$

For quadratic loss:

$$F(\mu) = F_0 + \int_{\mathbb{R}^p} V(\boldsymbol{\theta})\mathrm{d}\mu(\boldsymbol{\theta}) + \int_{\mathbb{R}^{2p}} U(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}})\mathrm{d}\mu(\boldsymbol{\theta})\mathrm{d}\mu(\tilde{\boldsymbol{\theta}})$$

depend on activation functions of the NN

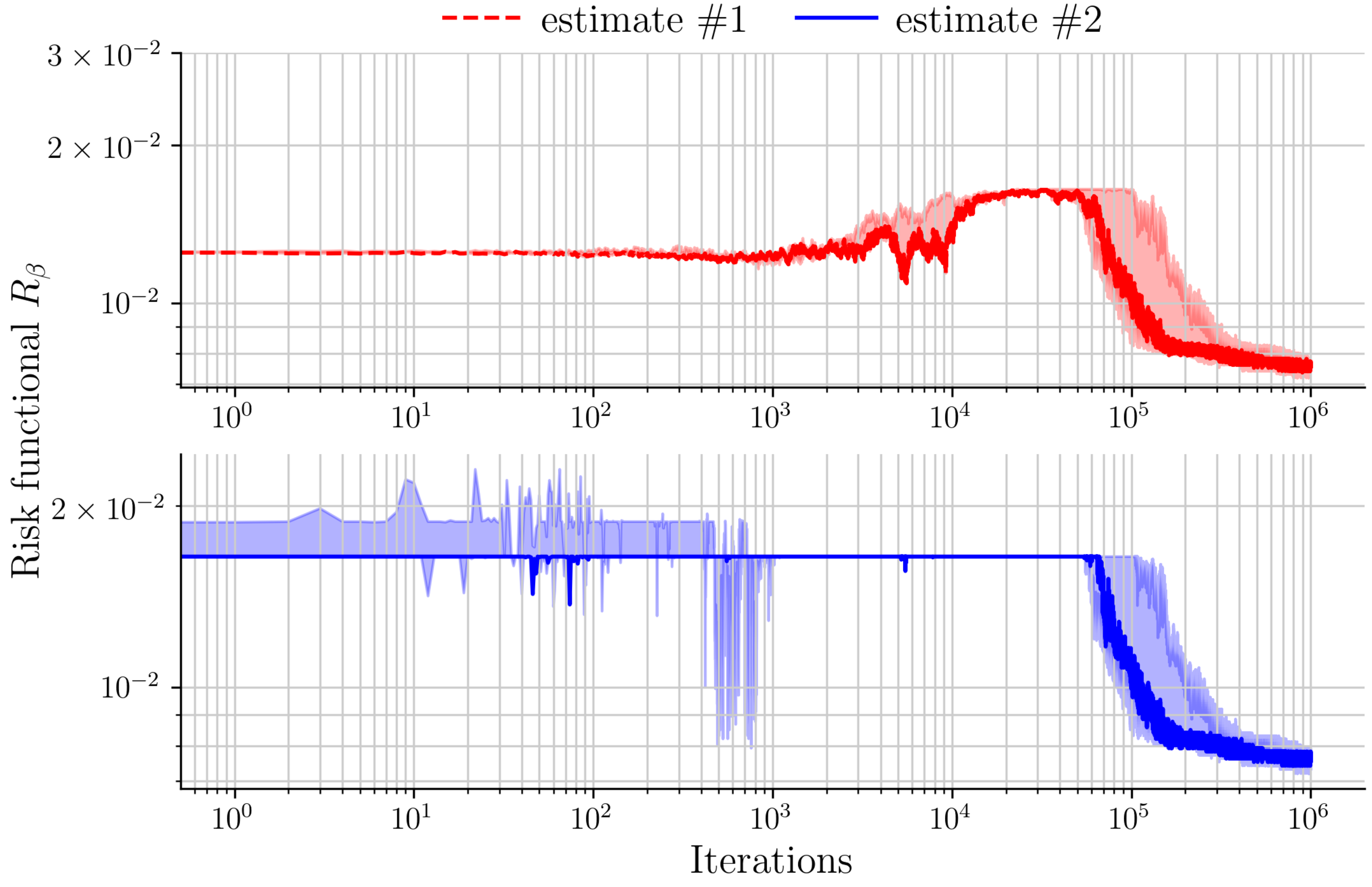
Neuronal population measure dynamics: $$\frac{\partial \mu}{\partial t} = \nabla \cdot \left(\mu \nabla \frac{\delta F}{\delta \mu}\right) =: -\nabla^{W_2} F(\mu)$$

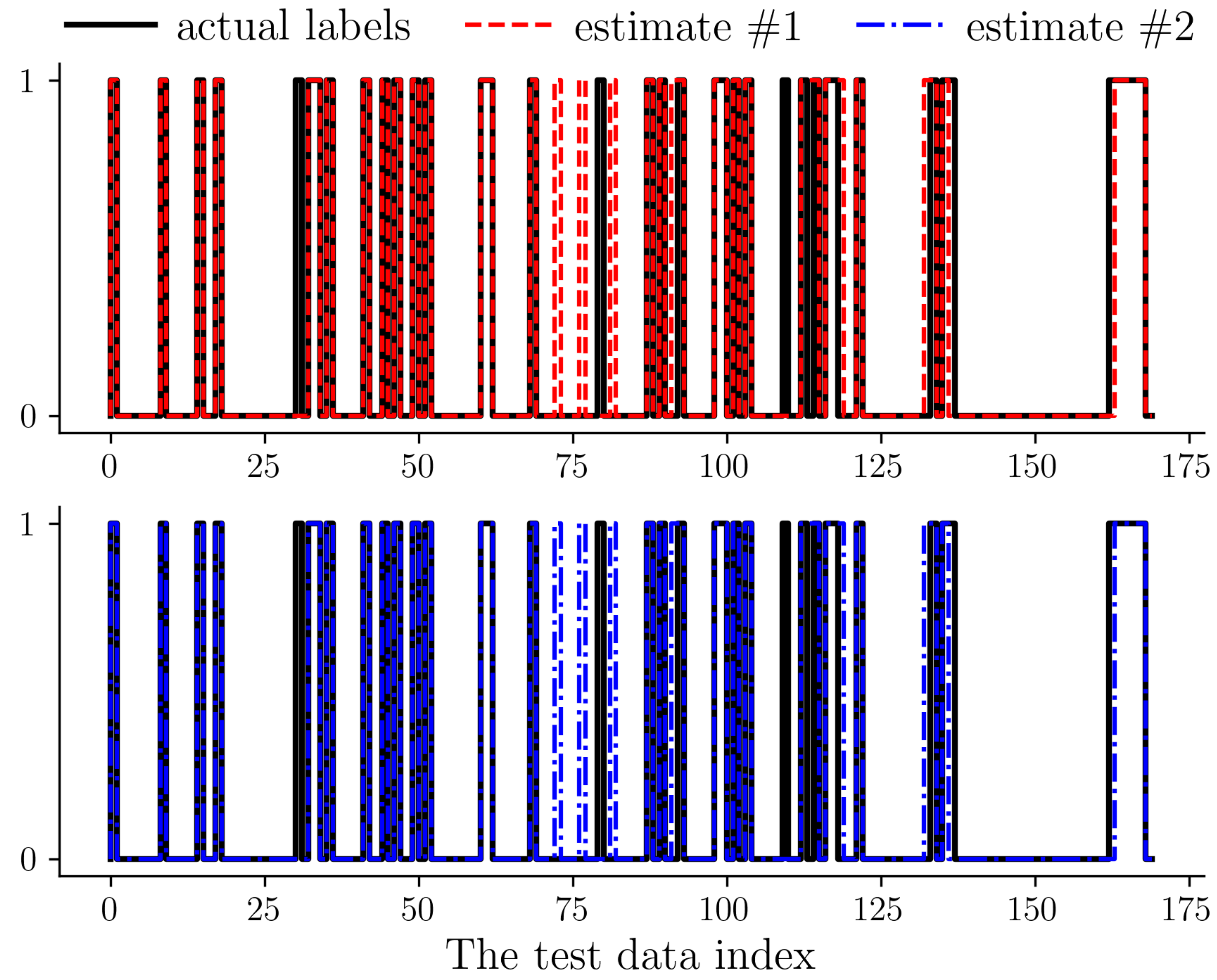
Wasserstein proximal recursion: $$\mu_{k+1} = \mathrm{prox}_{hF}^{W}(\mu_k)$$

Stochastic Learning / Centralized Computing

Centralized Computing Can Become Intensive: Mean Field SGD Dynamics in NN Classification

Case study: Wisconsin Breast Cancer (Diagnostic) Data Set





Classification accuracy for the WBDC dataset		
β	Estimate #1	Estimate #2
0.03	91.17%	92.35%
0.05	92.94%	92.94%
0.07	78.23%	92.94%

CPU: 3.4 GHz 6 core intel i5 8GB RAM (≈ 33 hrs runtime)

GPU: Jetson TX2 NVIDIA Pascal GPU 256 CUDA cores, 64 bit NVIDIA Denver + ARM Cortex A57 CPUs (≈ 2 hrs runtime)

Our Present Work: Distributed Algorithm

$$\arg\inf_{\mu \in \mathcal{P}_2(\mathbb{R}^d)} F_1(\mu) + F_2(\mu) + \ldots + F_n(\mu)$$

Our Present Work: Distributed Algorithm

Main idea:

$$\operatorname*{arg\,inf}_{\mu \in \mathcal{P}_2(\mathbb{R}^d)} F_1(\mu) + F_2(\mu) + \ldots + F_n(\mu)$$

re-write

$$\operatorname*{arg\,inf}_{(\mu_1,\ldots,\mu_n,\zeta) \in \mathcal{P}_2^{n+1}(\mathbb{R}^d)} \quad F_1(\mu_1) + F_2(\mu_2) + \ldots + F_n(\mu_n)$$

$$\text{subject to} \quad \mu_i = \zeta \text{ for all } i \in [n]$$

Our Present Work: Distributed Algorithm

Main idea:

$$\arg\inf_{\mu\in\mathcal{P}_2(\mathbb{R}^d)} F_1(\mu) + F_2(\mu) + \ldots + F_n(\mu)$$

re-write

$$\arg\inf_{(\mu_1,\ldots,\mu_n,\zeta)\in\mathcal{P}_2^{n+1}(\mathbb{R}^d)} F_1(\mu_1) + F_2(\mu_2) + \ldots + F_n(\mu_n)$$

$$\text{subject to} \quad \mu_i = \zeta \text{ for all } i \in [n]$$

Define Wasserstein augmented Lagrangian:

$$L_\alpha(\mu_1,\ldots,\mu_n,\zeta,\nu_1,\ldots,\nu_n) := \sum_{i=1}^{n}\left\{F_i(\mu_i) + \frac{\alpha}{2}W^2(\mu_i,\zeta) + \int_{\mathbb{R}^d}\nu_i(\boldsymbol{\theta})(\mathrm{d}\mu_i - \mathrm{d}\zeta)\right\}$$

regularization > 0

Lagrange multipliers

Stochastic Learning/ Distributed Computing

Proposed Consensus ADMM

$$
\begin{aligned}
\mu_i^{k+1} &= \underset{\mu_i \in \mathcal{P}_2(\mathbb{R}^d)}{\arg\inf}\, L_\alpha\left(\mu_1, \ldots, \mu_n, \zeta^k, \nu_1^k, \ldots, \nu_n^k\right)\\
\zeta^{k+1} &= \underset{\zeta \in \mathcal{P}_2(\mathbb{R}^d)}{\arg\inf}\, L_\alpha\left(\mu_1^{k+1}, \ldots, \mu_n^{k+1}, \zeta, \nu_1^k, \ldots, \nu_n^k\right)\\
\nu_i^{k+1} &= \nu_i^k + \alpha\left(\mu_i^{k+1} - \zeta^{k+1}\right)
\end{aligned}
$$

where $i \in [n], k \in \mathbb{N}_0$

Define

$$\nu_{\text{sum}}^k(\boldsymbol{\theta}) := \sum_{i=1}^n \nu_i^k(\boldsymbol{\theta}), \quad k \in \mathbb{N}_0$$

and simplify the recursions to

$$
\begin{aligned}
\mu_i^{k+1} &= \text{prox}^W_{\frac{1}{\alpha}\left(F_i(\cdot) + \int \nu_i^k \,\mathrm{d}(\cdot)\right)}\left(\zeta^k\right)\\
\zeta^{k+1} &= \underset{\zeta \in \mathcal{P}_2(\mathbb{R}^d)}{\arg\inf}\left\{\left(\sum_{i=1}^n W^2\left(\mu_i^{k+1}, \zeta\right)\right) - \frac{2}{\alpha}\int_{\mathbb{R}^d} \nu_{\text{sum}}^k(\boldsymbol{\theta})\mathrm{d}\zeta\right\}\\
\nu_i^{k+1} &= \nu_i^k + \alpha\left(\mu_i^{k+1} - \zeta^{k+1}\right)
\end{aligned}
$$

Discrete Version of the Proposed ADMM

Euclidean distance matrix

Outer layer ADMM

$$
\begin{aligned}
\boldsymbol{\mu}_i^{k+1} &= \operatorname{prox}^{W}_{\frac{1}{\alpha}\left(F_i(\boldsymbol{\mu}_i)+\langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i\rangle\right)}\left(\boldsymbol{\zeta}^k\right) \\
&= \underset{\boldsymbol{\mu}_i \in \Delta^{N-1}}{\arg\inf} \left\{ \min_{\boldsymbol{M} \in \Pi_N(\boldsymbol{\mu}_i, \boldsymbol{\zeta}^k)} \frac{1}{2}\langle \boldsymbol{C}, \boldsymbol{M}\rangle + \frac{1}{\alpha}\left(F_i(\boldsymbol{\mu}_i) + \langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i\rangle\right) \right\} \\
\boldsymbol{\zeta}^{k+1} &= \underset{\boldsymbol{\zeta} \in \Delta^{N-1}}{\arg\inf} \left\{ \left(\sum_{i=1}^{n} \min_{\boldsymbol{M}_i \in \Pi_N(\boldsymbol{\mu}_i^{k+1}, \boldsymbol{\zeta})} \frac{1}{2}\langle \boldsymbol{C}, \boldsymbol{M}_i\rangle \right) - \frac{2}{\alpha}\langle \boldsymbol{\nu}_{\text{sum}}^k, \boldsymbol{\zeta}\rangle \right\} \\
\boldsymbol{\nu}_i^{k+1} &= \boldsymbol{\nu}_i^k + \alpha\left(\boldsymbol{\mu}_i^{k+1} - \boldsymbol{\zeta}^{k+1}\right)
\end{aligned}
$$

Inner layer ADMM

where N is the number of samples

Stochastic Learning/ Distributed Computing

$$\mu_i^{k+1} = \mathrm{prox}^W_{\frac{1}{\alpha}\left(F_i(\cdot)+\int \nu_i^k \,\mathrm{d}(\cdot)\right)}\left(\zeta^k\right)$$

Split free energy functionals: $\Phi_i(\mu_i) := F_i(\mu_i) + \int_{\mathbb{R}^d} \nu_i^k \,\mathrm{d}\mu_i$

∴ Distributed Wasserstein prox ≈ time updates of $\dfrac{\partial \tilde{\mu}_i}{\partial t} = -\nabla^W \Phi_i(\tilde{\mu}_i)$

Examples:

$\Phi_i(\cdot) = F_i(\cdot) + \int \nu_i^k \mathrm{d}(\cdot)$	PDE	Name
$\int_{\mathbb{R}^d} \left(V(\boldsymbol{\theta}) + \nu_i^k(\boldsymbol{\theta})\right) \mathrm{d}\mu_i(\boldsymbol{\theta})$	$\dfrac{\partial \widetilde{\mu}_i}{\partial t} = \nabla \cdot \left(\widetilde{\mu}_i \left(\nabla V + \nabla \nu_i^k\right)\right)$	Liouville equation
$\int_{\mathbb{R}^d} \left(\nu_i^k(\boldsymbol{\theta}) + \beta^{-1} \log \mu_i(\boldsymbol{\theta})\right) \mathrm{d}\mu_i(\boldsymbol{\theta})$	$\dfrac{\partial \widetilde{\mu}_i}{\partial t} = \nabla \cdot \left(\widetilde{\mu}_i \nabla \nu_i^k\right) + \beta^{-1} \Delta \widetilde{\mu}_i$	Fokker-Planck equation
$\int_{\mathbb{R}^d} \nu_i^k(\boldsymbol{\theta}) \mathrm{d}\mu_i(\boldsymbol{\theta}) + \int_{\mathbb{R}^{2d}} U(\boldsymbol{\theta}, \boldsymbol{\sigma}) \mathrm{d}\mu_i(\boldsymbol{\theta}) \mathrm{d}\mu_i(\boldsymbol{\sigma})$	$\dfrac{\partial \widetilde{\mu}_i}{\partial t} = \nabla \cdot \left(\widetilde{\mu}_i \left(\nabla \nu_i^k + \nabla \left(U \circledast \widetilde{\mu}_i\right)\right)\right)$	Propagation of chaos equation
$\int_{\mathbb{R}^d} \left(\nu_i^k(\boldsymbol{\theta}) + \frac{\beta^{-1}}{m-1} \mathbf{1}^\top \mu_i^m\right) \mathrm{d}\mu_i(\boldsymbol{\theta}), m > 1$	$\dfrac{\partial \widetilde{\mu}_i}{\partial t} = \nabla \cdot \left(\widetilde{\mu}_i \nabla \nu_i^k\right) + \beta^{-1} \Delta \widetilde{\mu}_i^m$	Porous medium equation

$\boldsymbol{\mu}_i$ update ⇝ Outer Consensus (Sinkhorn) ADMM

Example: $\Phi(\boldsymbol{\mu}) := \langle \boldsymbol{a}, \boldsymbol{\mu} \rangle, \boldsymbol{a} \in \mathbb{R}^N \setminus \{\mathbf{0}\}, \boldsymbol{\mu}, \boldsymbol{\zeta} \in \Delta^{N-1}, \boldsymbol{\Gamma} := \exp(-\boldsymbol{C}/2\varepsilon), \varepsilon > 0$

$$\operatorname{prox}_{\frac{1}{\alpha}\Phi}^{W_\varepsilon}(\boldsymbol{\zeta}) = \exp\left(-\frac{1}{\alpha\varepsilon}\boldsymbol{a}\right) \odot \left(\boldsymbol{\Gamma}^\top \left(\zeta \oslash \left(\boldsymbol{\Gamma} \exp\left(-\frac{1}{\alpha\varepsilon}\boldsymbol{a}\right)\right)\right)\right)$$

ζ update $\rightsquigarrow$ Inner (Euclidean) ADMM

Theorem. Consider the convex problem

$$\left(\boldsymbol{u}_1^{\text{opt}}, \ldots, \boldsymbol{u}_n^{\text{opt}}\right) = \underset{(\boldsymbol{u}_1,\ldots,\boldsymbol{u}_n)\in\mathbb{R}^{nN}}{\arg\min} \sum_{i=1}^{n} \langle \boldsymbol{\mu}_i^{k+1}, \log(\boldsymbol{\Gamma}\exp(\boldsymbol{u}_i/\varepsilon))\rangle$$

$$\text{subject to} \sum_{i=1}^{n} \boldsymbol{u}_i = \frac{2}{\alpha}\boldsymbol{\nu}_{\text{sum}}^{k}.$$

Then

$$\boldsymbol{\zeta}^{k+1} = \exp\left(\boldsymbol{u}_i^{\text{opt}}/\varepsilon\right) \odot \left(\boldsymbol{\Gamma}\left(\boldsymbol{\mu}_i^{k+1} \oslash \left(\boldsymbol{\Gamma}\exp\left(\boldsymbol{u}_i^{\text{opt}}/\varepsilon\right)\right)\right)\right) \in \boldsymbol{\Delta}^{N-1} \forall \boldsymbol{i} \in [n]$$

ζ update $\rightsquigarrow$ Inner (Euclidean) ADMM

Theorem. Let $f_i(\boldsymbol{u}_i) := \left\langle \boldsymbol{\mu}_i^{k+1}, \log(\boldsymbol{\Gamma} \exp(\boldsymbol{u}_i/\varepsilon))\right\rangle, \quad \boldsymbol{u}_i \in \mathbb{R}^N, \quad$ for all $i \in [n]$,

Then the following Euclidean ADMM solves

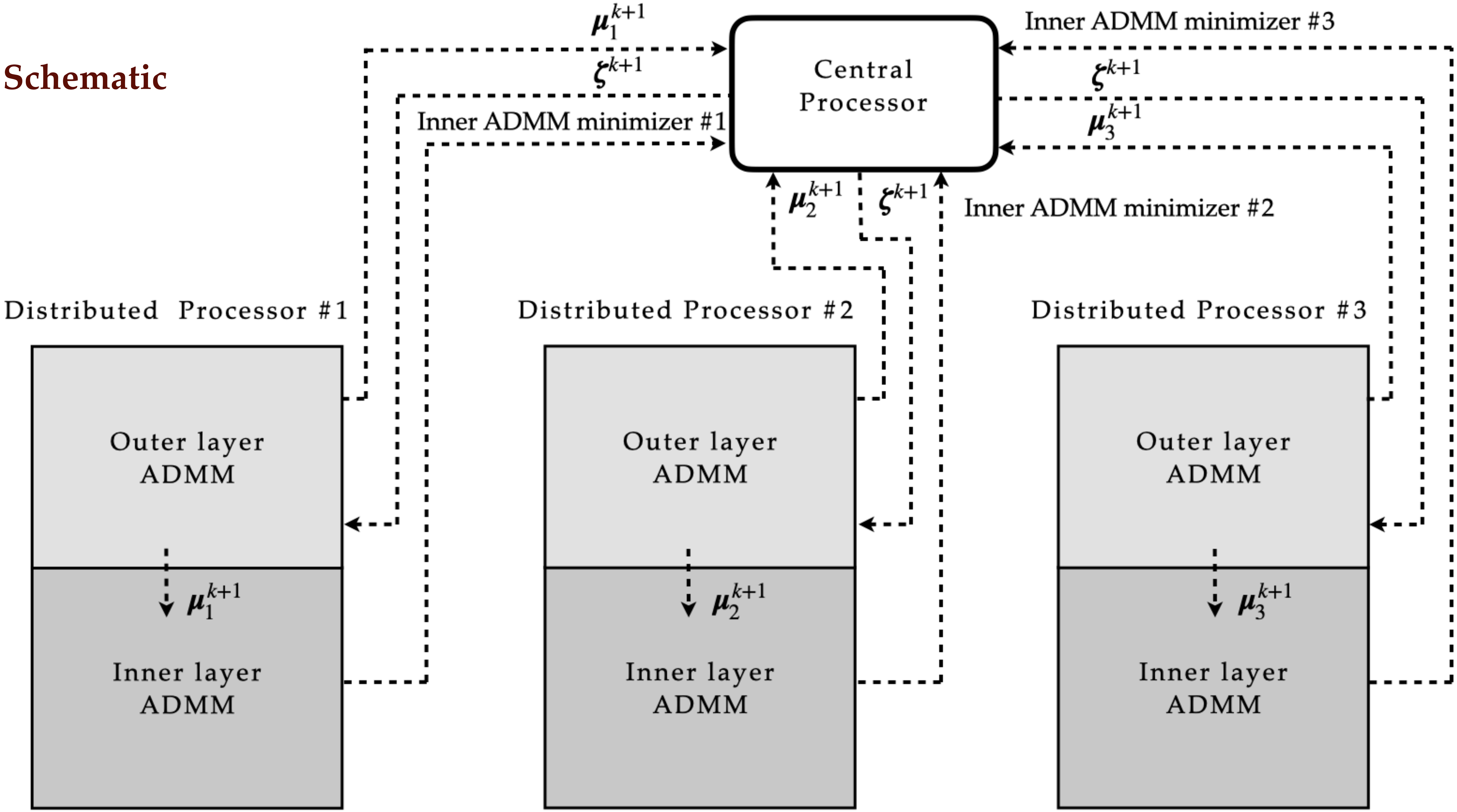
$$
\begin{aligned}
\boldsymbol{u}_i^{\ell+1} &= \operatorname{prox}_{\frac{1}{\tau} f_i}^{\|\cdot\|_2}\left(\boldsymbol{z}_i^{\ell} - \tilde{\boldsymbol{\nu}}_i^{\ell}\right) \\
\boldsymbol{z}_i^{\ell+1} &= \left(\boldsymbol{u}_i^{\ell+1} - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{u}_i^{\ell+1}\right) + \left(\tilde{\boldsymbol{\nu}}_i^{\ell} - \frac{1}{n}\sum_{i=1}^{n} \tilde{\boldsymbol{\nu}}_i^{\ell}\right) + \frac{2}{n\alpha}\boldsymbol{\nu}_{\text{sum}}^{k} \\
\tilde{\boldsymbol{\nu}}_i^{\ell+1} &= \tilde{\boldsymbol{\nu}}_i^{\ell} + \left(\boldsymbol{u}_i^{\ell+1} - \boldsymbol{z}_i^{\ell+1}\right)
\end{aligned}
$$

Theorem.

Guaranteed convergence for inner layer ADMM under some constraints on hyper-parameters

Stochastic Learning/ Distributed Computing

Overall Schematic



Stochastic Learning/ Distributed Computing

Experiment #1 Aggregation-drift-diffusion nonlinear PDE

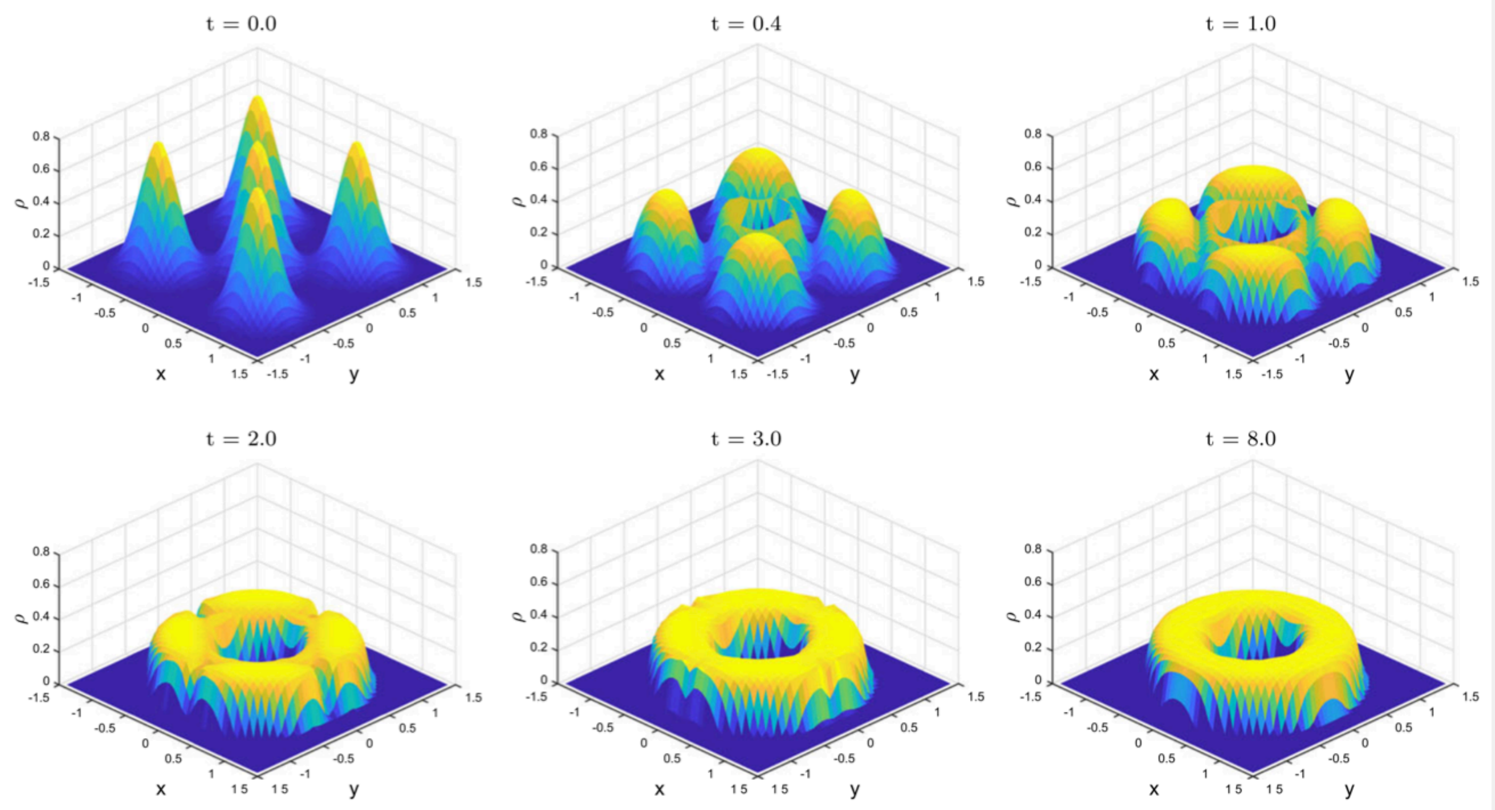
$$\frac{\partial \mu}{\partial t} = \underbrace{\nabla \cdot (\mu \nabla (U * \mu))}_{i=1} + \underbrace{\nabla \cdot (\mu \nabla V) + \beta^{-1} \Delta \mu^2}_{i=2}$$

$$U(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|_2^2 - \ln \|\boldsymbol{x}\|_2$$

$$V(\boldsymbol{x}) = -\frac{1}{4}\ln \|\boldsymbol{x}\|_2$$

Centralized computation:

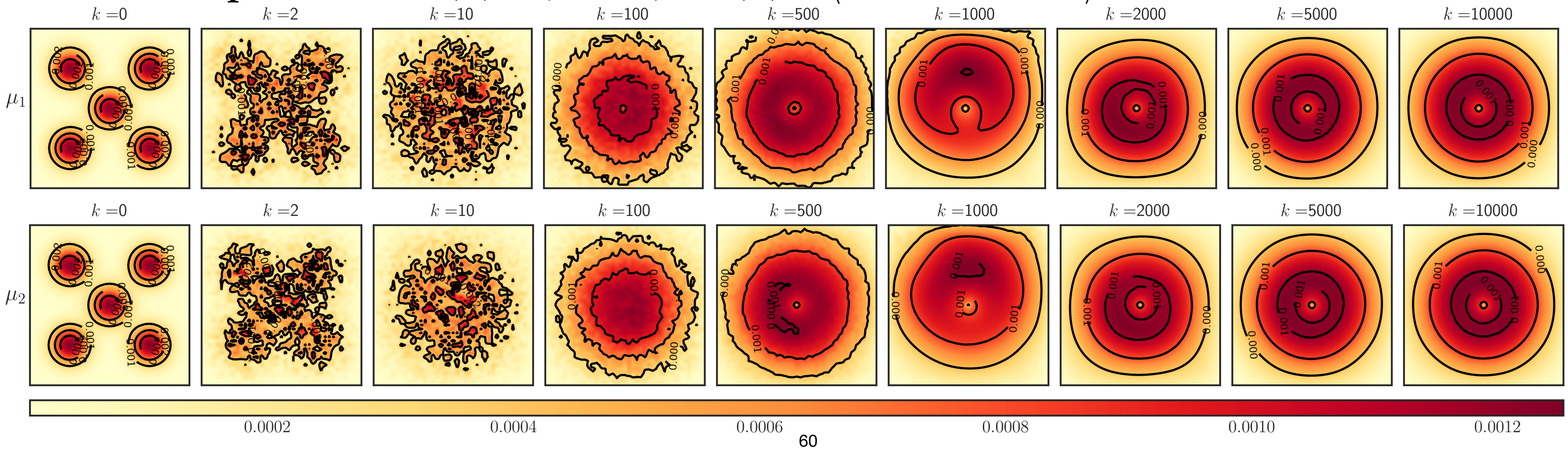
Carrillo, Craig, Wang and Wei, *FOCM*, 2021



$$\lim_{\beta^{-1}\downarrow 0} \mu_\infty = \mathrm{Unif}(\mathscr{A})$$

Annulus with inner radius $1/2$ and outer radius $\sqrt{5}/2$

Distributed computation: $F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}, \boldsymbol{\mu} \rangle \quad F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{V}_k + \beta^{-1} \log \boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$



Stochastic Learning/ Distributed Computing

Experiment #1 Aggregation-drift-diffusion nonlinear PDE

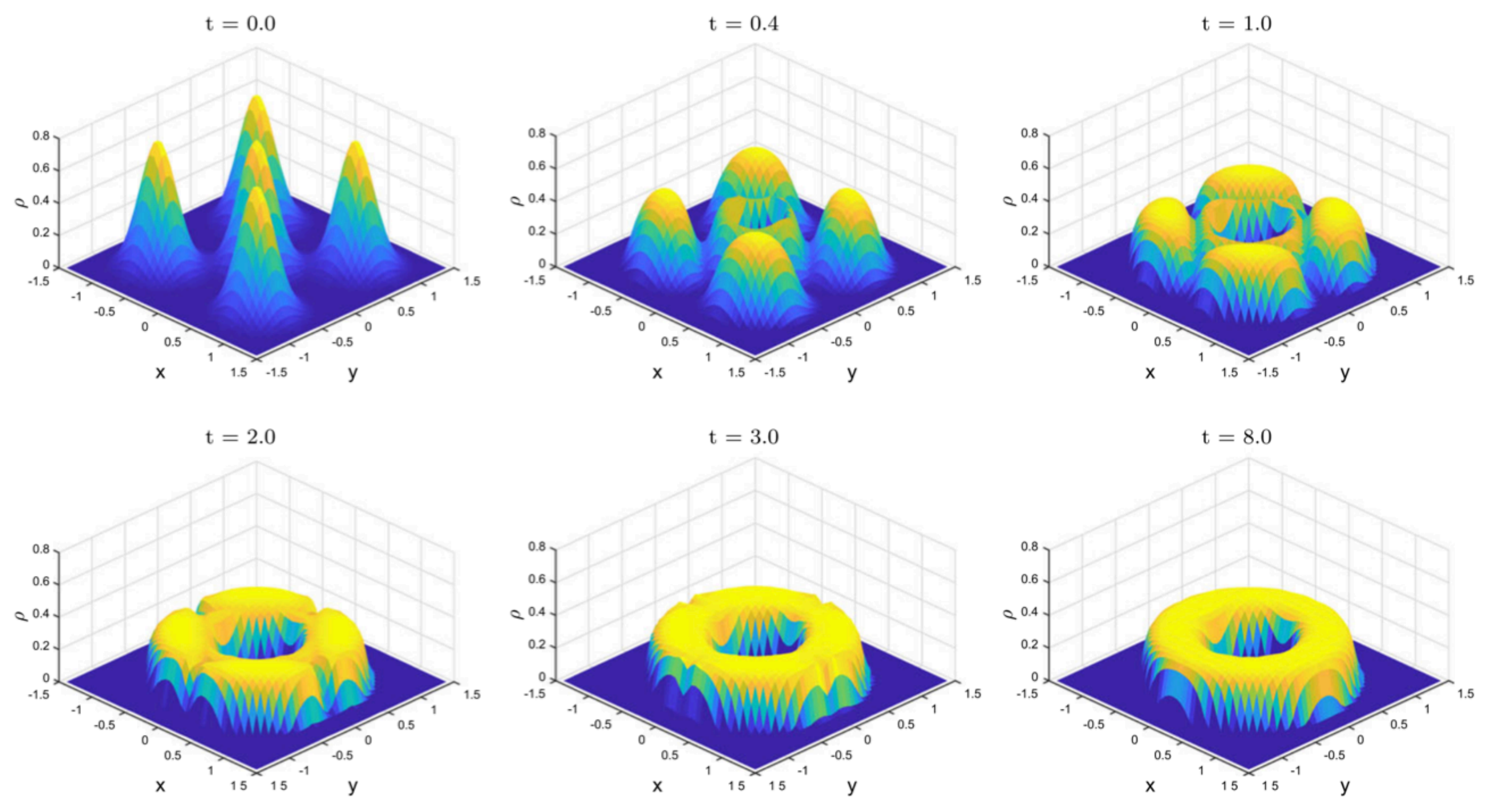
$$\frac{\partial \mu}{\partial t} = \underbrace{\nabla \cdot (\mu \nabla (U * \mu))}_{i=1} + \underbrace{\nabla \cdot (\mu \nabla V) + \beta^{-1} \Delta \mu^2}_{i=2}$$

$$U(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|_2^2 - \ln \|\boldsymbol{x}\|_2$$

$$V(\boldsymbol{x}) = -\frac{1}{4} \ln \|\boldsymbol{x}\|_2$$

Centralized computation:

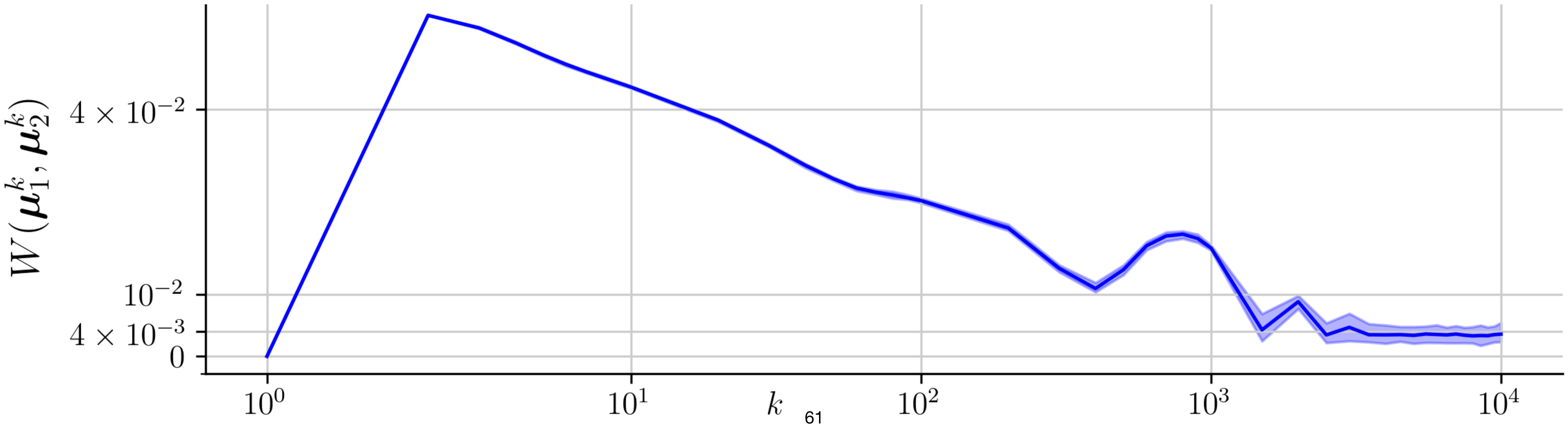
Carrillo, Craig, Wang and Wei, *FOCM*, 2021



$$\lim_{\beta^{-1} \downarrow 0} \mu_\infty = \mathrm{Unif}(\mathscr{A})$$

Annulus with inner radius 1/2 and outer radius $\sqrt{5}/2$

Distributed computation: $F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}, \boldsymbol{\mu} \rangle \quad F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{V}_k + \beta^{-1} \log \boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$



Stochastic Learning/ Distributed Computing

Experiment #2 Wasserstein barycenter

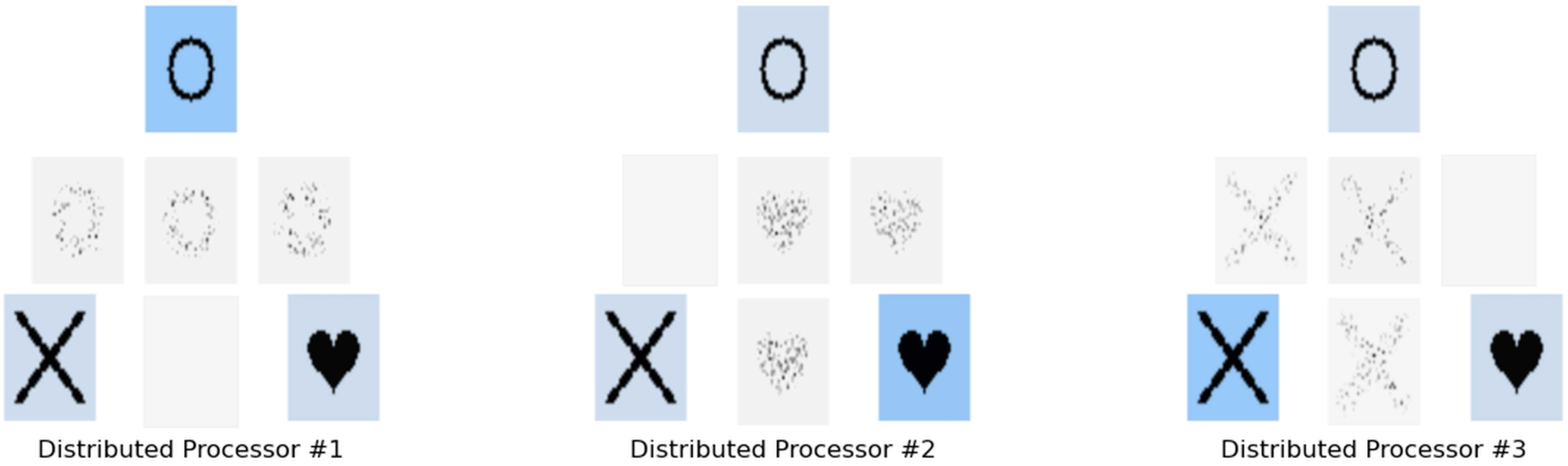
$$\arg\inf_{\boldsymbol{\mu}\in\mathcal{P}_2(\mathcal{X})} \sum_{i=1}^{n} w_i W^2(\boldsymbol{\mu}, \boldsymbol{\xi}_i)$$

Stochastic Learning/ Distributed Computing

Experiment #2 Wasserstein barycenter

$$\arg\inf_{\boldsymbol{\mu}\in\mathcal{P}_2(\mathcal{X})} \sum_{i=1}^{n} w_i W^2(\boldsymbol{\mu}, \boldsymbol{\xi}_i)$$

Experiment #2 Wasserstein barycenter



Thank You

Acknowledgment:

1923278, 2112754, 2112755

Regent's Fellowship

BSOE Dissertation Year Fellowship

Backup Slides

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

First order Case Study

$$\inf_{u\in\mathcal{U}} \quad \mathbb{E}_{\mu^u}\left[\int_0^T \frac{1}{2}u^2 \,\mathrm{d}t\right],$$

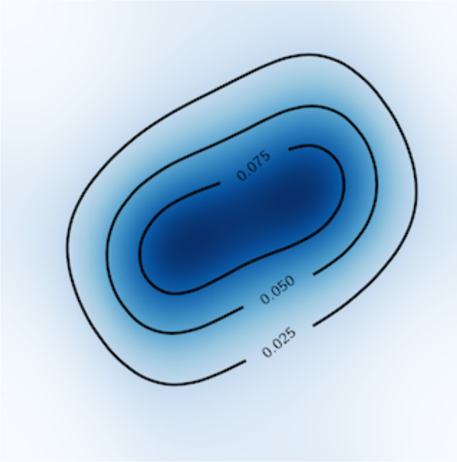
$$\inf_{u\in\mathcal{U}} \quad \mathbb{E}_{\mu^u}\left[\int_0^T \frac{1}{2}u^2 \,\mathrm{d}t\right],$$

$$\mathrm{d}\boldsymbol{\theta} = \left(-\nabla_{\boldsymbol{\theta}} V(\boldsymbol{\theta}) + \boldsymbol{S}\boldsymbol{u}\right)\mathrm{d}t + \sqrt{2}\boldsymbol{S}\mathrm{d}\boldsymbol{w}$$

Change of variables

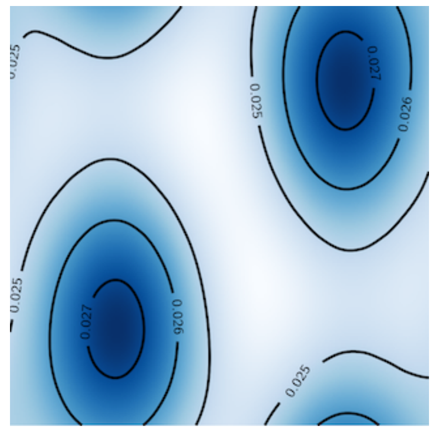
$$\boldsymbol{\theta} \mapsto \boldsymbol{\xi} := \boldsymbol{S}^{-1}\boldsymbol{\theta}$$

$$\mathrm{d}\boldsymbol{\xi} = \left(\boldsymbol{u} - \boldsymbol{\Upsilon}\,\nabla_{\boldsymbol{\xi}}\tilde{V}(\boldsymbol{\xi})\right)\mathrm{d}t + \sqrt{2}\,\mathrm{d}\boldsymbol{w}$$



$\boldsymbol{\theta}(t=0) \sim \mu_0$ (Desynchronized)

$\boldsymbol{\xi}(t=0) \sim \tilde{\mu}_0$ (Desynchronized)



$\boldsymbol{\theta}(t=T) \sim \tilde{\mu}_T$ (Synchronized)

$\boldsymbol{\xi}(t=T) \sim \tilde{\mu}_T$ (Synchronized)

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

First order

$$\mathrm{d}\boldsymbol{\theta} = \left(-\nabla_{\boldsymbol{\theta}} V(\boldsymbol{\theta}) + \boldsymbol{S}\boldsymbol{u}\right)\mathrm{d}t + \sqrt{2}\boldsymbol{S}\mathrm{d}\boldsymbol{w}$$

Second order

$$\begin{pmatrix} \mathrm{d}\boldsymbol{\theta} \\ \mathrm{d}\boldsymbol{\omega} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\omega} \\ -\boldsymbol{M}^{-1}\nabla_{\boldsymbol{\theta}} V(\boldsymbol{\theta}) - \boldsymbol{M}^{-1}\boldsymbol{\Gamma}\boldsymbol{\omega} + \boldsymbol{M}^{-1}\boldsymbol{S}\boldsymbol{u} \end{pmatrix}\mathrm{d}t + \begin{pmatrix} \boldsymbol{0}_{n\times 1} \\ \sqrt{2}\boldsymbol{M}^{-1}\boldsymbol{S}\mathrm{d}\boldsymbol{w} \end{pmatrix}$$

Potential function

$$V(\boldsymbol{\theta}) := \sum_{i<j} k_{ij}(1 - \cos(\theta_i - \theta_j - \varphi_{ij})) - \sum_{i=1}^{n} P_i \theta_i$$

Coupling > 0

Phase difference $\in [0, \pi/2)$

Linear coeff. > 0

Positive diagonal matrices $\boldsymbol{M}, \boldsymbol{\Gamma}, \boldsymbol{S}$

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

$$\inf_{(\rho,u)} \int_0^T \int_{\mathcal{X}} \|\boldsymbol{u}(\boldsymbol{x},t)\|_2^2 \; \rho(\boldsymbol{x},t) \; \mathrm{d}\boldsymbol{x}\mathrm{d}t$$

First order, $\mathcal{X} \equiv \mathbb{T}^n$

$\boldsymbol{S}\boldsymbol{S}^\top$

$$\text{s.t} \quad \frac{\partial \rho}{\partial t} = -\nabla_{\boldsymbol{\theta}} \cdot \left(\rho\left(\boldsymbol{S}\boldsymbol{u} - \nabla_{\boldsymbol{\theta}} V\right)\right) + \left\langle \boldsymbol{D}, \textbf{Hess}\left(\rho\right)\right\rangle$$

Second order, $\mathcal{X} \equiv \mathbb{T}^n \times \mathbb{R}^n$

$$\text{s.t} \quad \frac{\partial \rho}{\partial t} = \nabla_{\boldsymbol{\omega}} \cdot \left(\rho\left(\boldsymbol{M}^{-1}\nabla_{\boldsymbol{\theta}} V(\boldsymbol{\theta}) + \boldsymbol{M}^{-1}\boldsymbol{\Gamma}\boldsymbol{\omega} - \boldsymbol{M}^{-1}\boldsymbol{S}\boldsymbol{u} + \boldsymbol{M}^{-1}\boldsymbol{D}\boldsymbol{M}^{-1}\nabla_{\boldsymbol{\omega}}\log\rho\right) - \left\langle \boldsymbol{\omega}, \nabla_{\boldsymbol{\theta}}\rho\right\rangle\right.$$

Initial and Terminal conditions $\quad \rho(\boldsymbol{x}, t=0) = \rho_0, \qquad \rho(\boldsymbol{x}, t=T) = \rho_T$

Stochastic Control/ Control-affine: Nonuniform Noisy Kuramoto Oscillators

The Second Order Case

Uncontrolled forward-backward Kolmogrov PDEs

$$\frac{\partial \hat{\varphi}}{\partial t} = -\left\langle \boldsymbol{\eta}, \nabla_{\xi}\hat{\varphi} \right\rangle + \nabla_{\eta} \cdot \left(\hat{\varphi} \left(\widetilde{\boldsymbol{\Upsilon}}\, \nabla_{\xi} \boldsymbol{U}(\boldsymbol{\xi}) + \nabla_{\eta} \boldsymbol{F}(\boldsymbol{\eta}) \right) \right) + \Delta_{\eta} \hat{\varphi}$$

$$\frac{\partial \varphi}{\partial t} = -\left\langle \boldsymbol{\eta}, \nabla_{\xi}\varphi \right\rangle + \left\langle \widetilde{\boldsymbol{\Upsilon}}\, \nabla_{\xi} \boldsymbol{U}(\boldsymbol{\xi}) + \nabla_{\eta} \boldsymbol{F}(\boldsymbol{\eta}), \nabla_{\eta} \varphi \right\rangle - \Delta_{\eta} \varphi$$

Initial and Terminal conditions

$$\hat{\varphi}_0(\boldsymbol{\xi})\varphi_0(\boldsymbol{\xi}) = \rho_0 \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{S}\boldsymbol{M}^{-1} \right) \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} \right) \left(\prod_{i=1}^{n} \frac{\sigma_i^2}{m_i^2} \right)$$

$$\hat{\varphi}_T(\boldsymbol{\xi})\varphi_T(\boldsymbol{\xi}) = \rho_T \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{S}\boldsymbol{M}^{-1} \right) \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} \right) \left(\prod_{i=1}^{n} \frac{\sigma_i^2}{m_i^2} \right)$$

Optimal controlled joint state PDF: $\rho^{\mathrm{opt}}(\boldsymbol{\theta}, \boldsymbol{\omega}, t) = \hat{\varphi} \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{M}\boldsymbol{S}^{-1} \right) \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{\omega} \end{pmatrix}, t \right) \varphi \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{M}\boldsymbol{S}^{-1} \right) \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{\omega} \end{pmatrix}, t \right) \left(\prod_{i=1}^{n} \frac{m_i^2}{\sigma_i^2} \right)$

Optimal control: $\boldsymbol{u}^{\mathrm{opt}} \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{M}\boldsymbol{S}^{-1} \right) \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{\omega} \end{pmatrix}, t \right) = \left(\boldsymbol{I}_2 \otimes \boldsymbol{S}\boldsymbol{M}^{-1} \right) \nabla_{\boldsymbol{\theta}} \log \varphi \left(\left(\boldsymbol{I}_2 \otimes \boldsymbol{M}\boldsymbol{S}^{-1} \right) \begin{pmatrix} \boldsymbol{\theta} \\ \boldsymbol{\omega} \end{pmatrix}, t \right)$

Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators

Proximal recursion

$$\hat{\phi}_k = \text{prox}^{d}_{h\Psi}\left(\hat{\phi}_{k-1}\right) := \underset{\hat{\phi}}{\arg\inf}\, \frac{1}{2}\left(d\left(\hat{\phi}, \hat{\phi}_{k-1}\right)\right)^2 + h\Psi(\hat{\phi})$$

Distance

Step size

Energy-like functional

First order: $d \equiv W_{\Upsilon}$ $\quad \Psi(\hat{\phi}) \equiv \int_{\prod_{i=1}^{n}[0,2\pi/\sigma_i)} (\tilde{V} + \log \hat{\phi})\hat{\phi}\mathrm{d}\boldsymbol{\xi}$

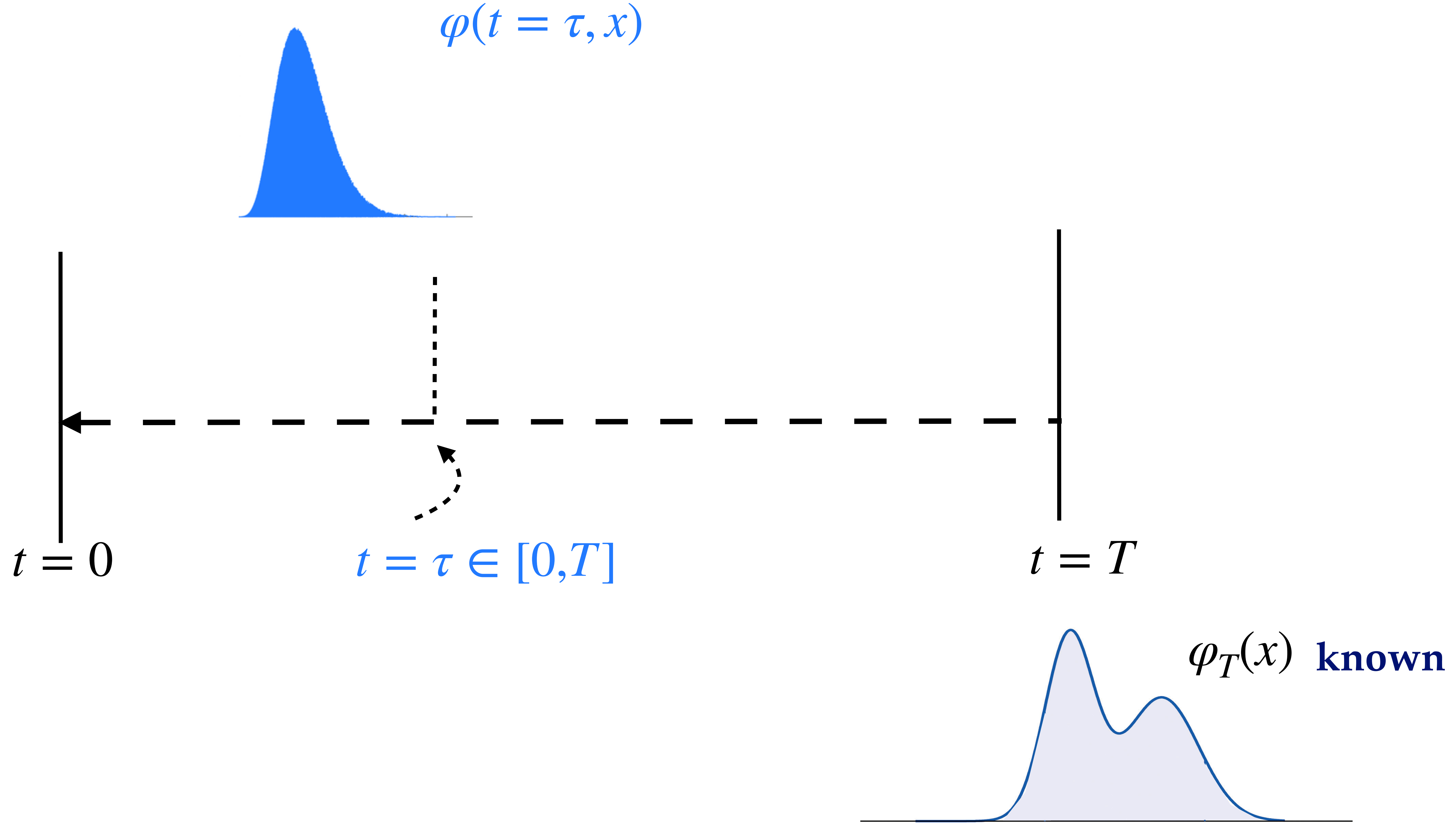
Second order: $d \equiv W_{h,\tilde{\Upsilon}}$

$$\Psi(\hat{\phi}) \equiv \int \left(\prod_{i=1}^{n}[0, 2\pi m_i/\sigma_i)\right) \times \mathbb{R}^n (F + \log \hat{\phi})\hat{\phi}\mathrm{d}\boldsymbol{\xi}\mathrm{d}\boldsymbol{\eta}$$

Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators

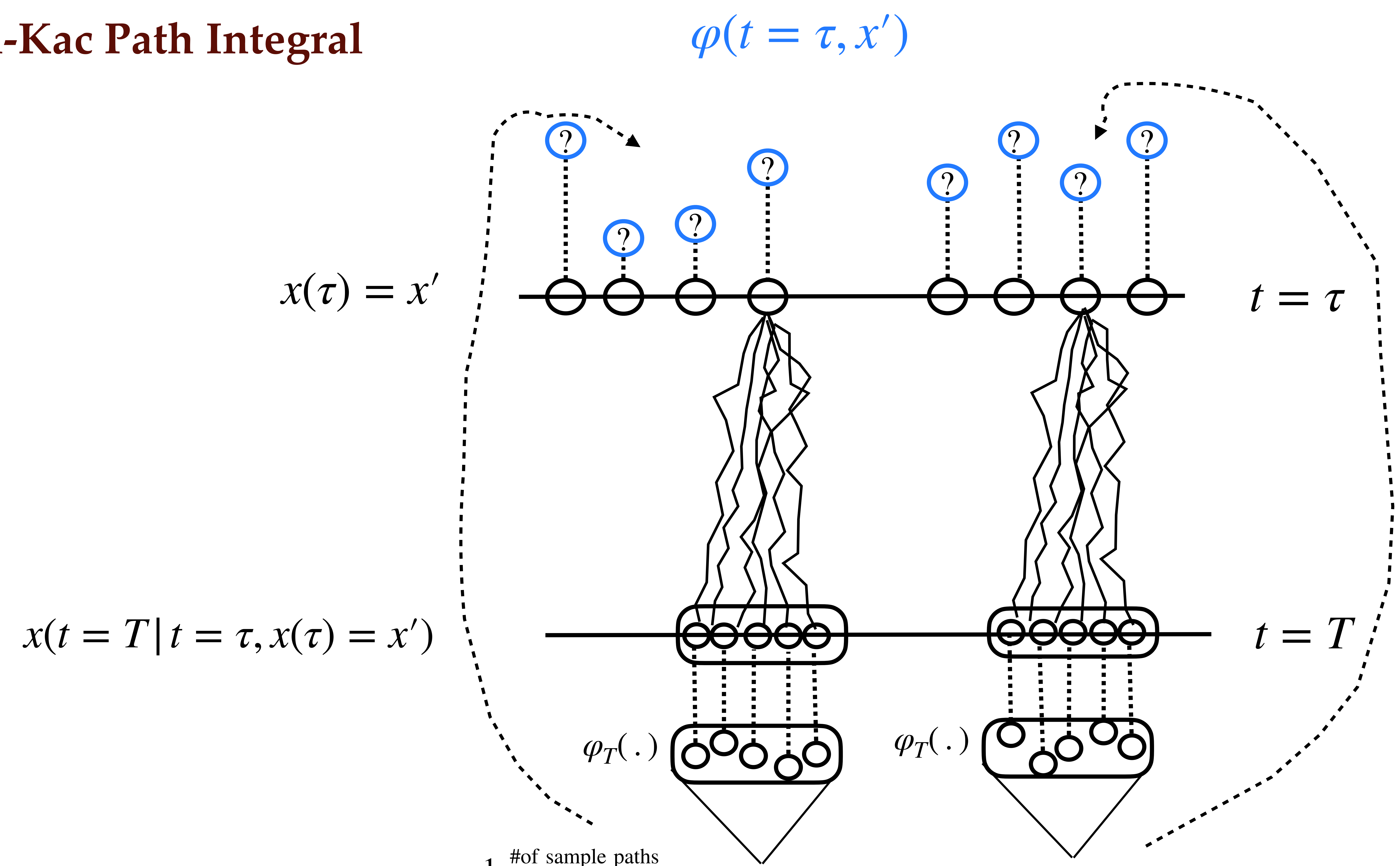
Feynman-Kac Path Integral

$$\frac{\partial \varphi}{\partial t} = L_{\text{Backward}} \varphi$$



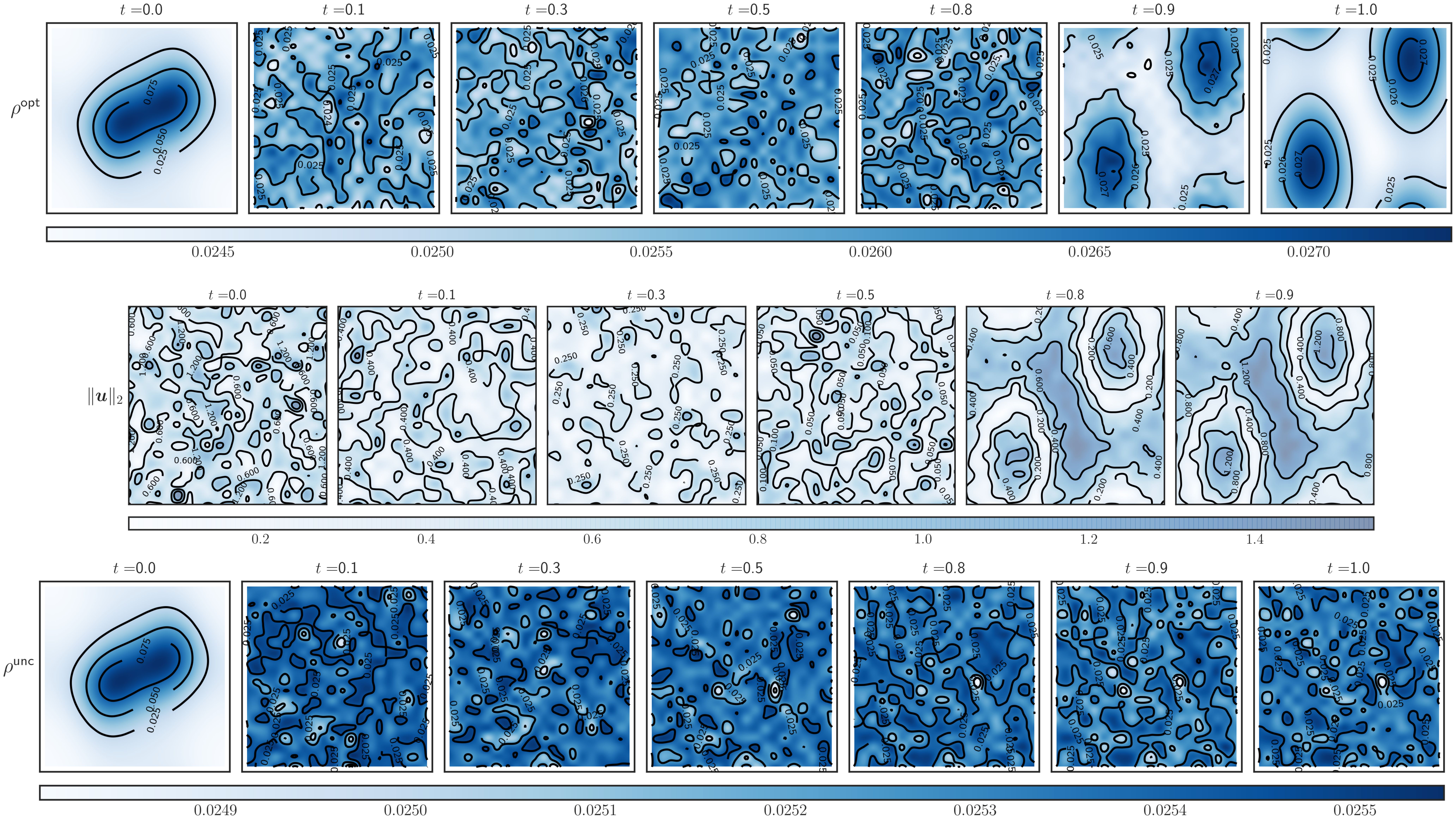
Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators

Feynman-Kac Path Integral

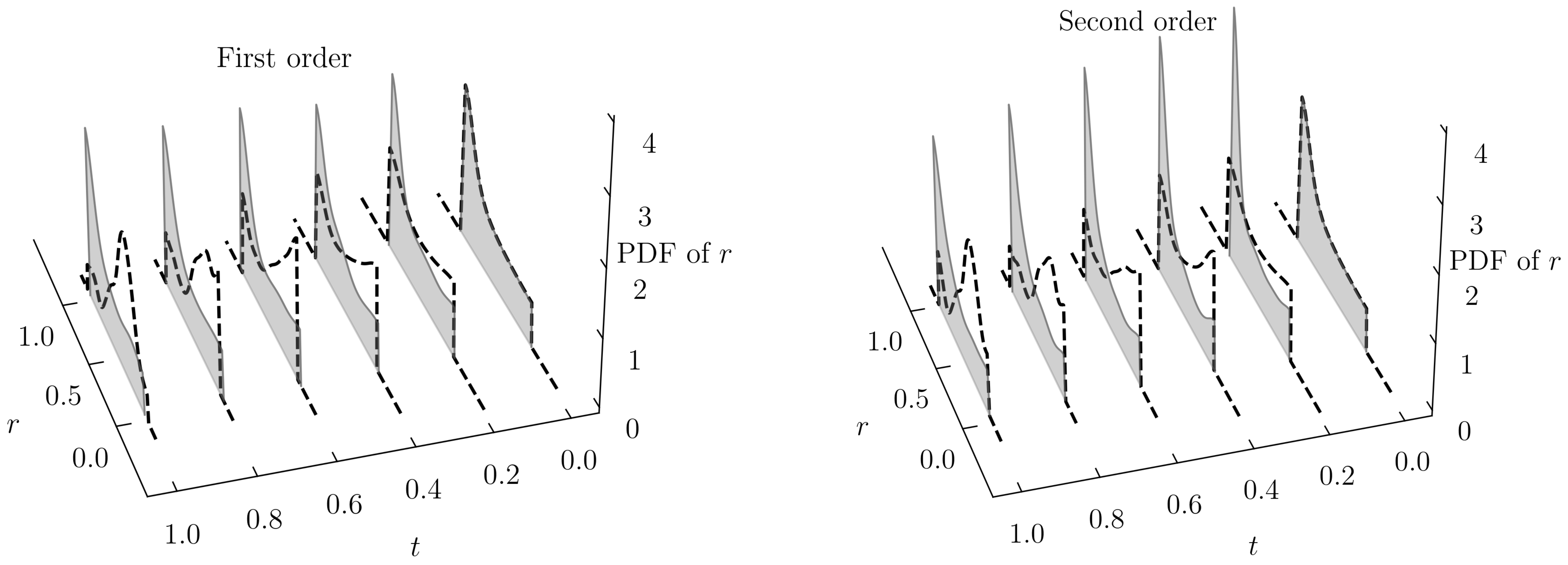


$$\mathbb{E}[\varphi_T(\,.\,)] = \frac{1}{N} \sum_{i=1}^{\#\text{of sample paths}} \varphi_i(\,.\,)$$

Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators



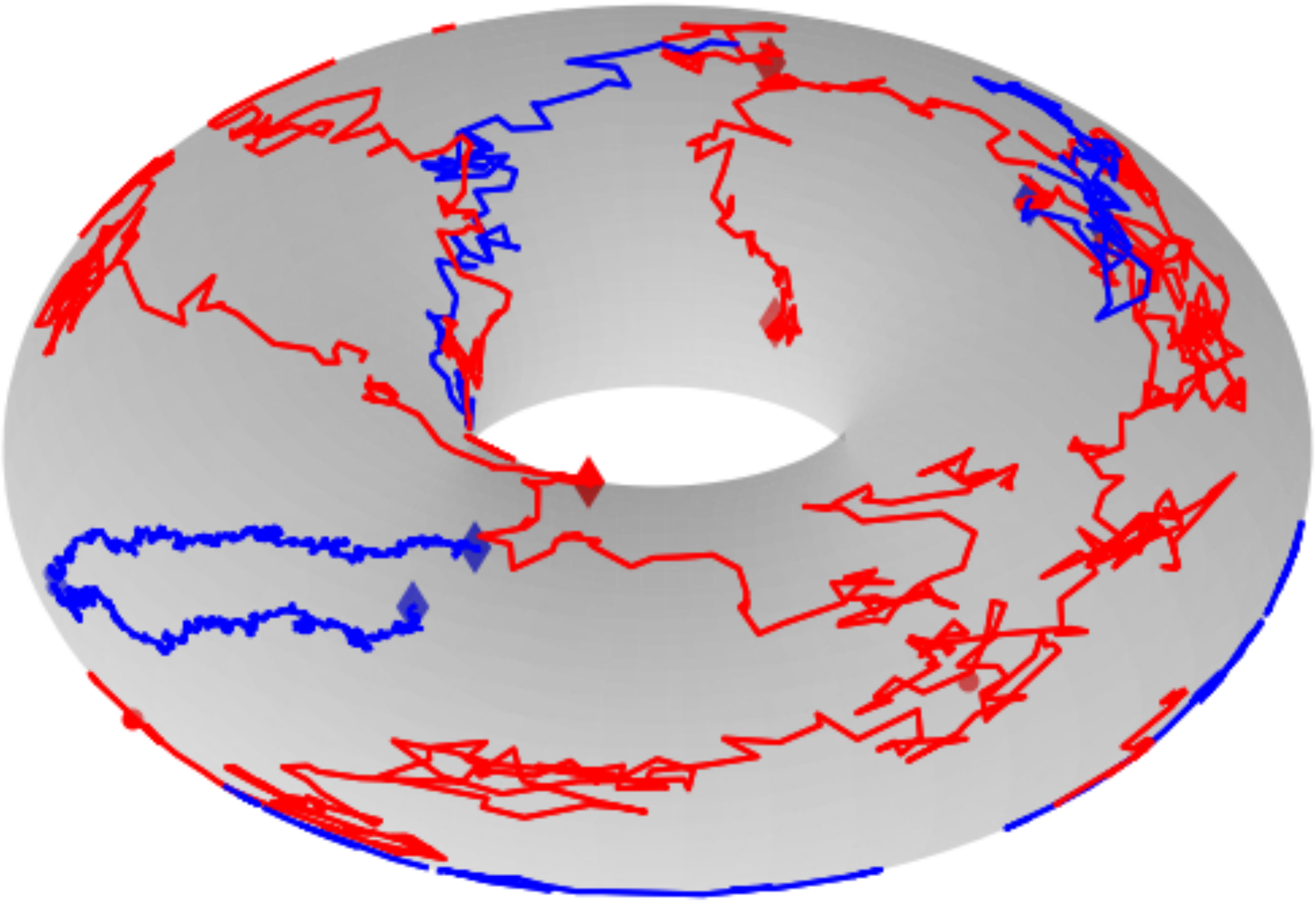
Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators



PDF of order parameter $r := \frac{1}{n}\sqrt{\left(\sum_{i=1}^{n}\cos\theta_i\right)^2 + \left(\sum_{i=1}^{n}\sin\theta_i\right)^2}$

Stochastic Control/ Control-affine/ Nonuniform Noisy Kuramoto Oscillators

Optimally Controlled Sample Paths



Stochastic Control/ Control Non-affine

Case1: Solve via PINN

Loss term for HJB PDE

$$\mathcal{L}_{\psi} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\partial \psi}{\partial t}\bigg|_{\boldsymbol{x}_i} - \frac{1}{2}(\pi^{\text{opt}})^2\bigg|_{\boldsymbol{x^u}_i} - + \frac{\partial \psi}{\partial x^u}D_1\bigg|_{\boldsymbol{x}_i^u} - + \frac{\partial^2 \psi}{\partial x^{u2}}D_2\bigg|_{\boldsymbol{x}_i^u}\right)^2$$

Loss term for FPK PDE

$$\mathcal{L}_{\rho^u} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\partial \rho^u}{\partial t}\bigg|_{\boldsymbol{x}_i^u} + \frac{\partial}{\partial x^u}\left(D_1\rho^u\right)\bigg|_{\boldsymbol{x}_i^u} - \frac{\partial^2}{\partial x^{u2}}\left(D_2\rho^u\right)\bigg|_{\boldsymbol{x}_i^u}\right)^2$$

Loss term for policy equation

$$\mathcal{L}_{\pi^{\text{opt}}} = \frac{1}{n}\sum_{i=1}^{n}\left(\pi^{\text{opt}}\Big|_{\boldsymbol{x}_i^u} - \frac{\partial \psi}{\partial x^u}\frac{\partial D_1}{\partial u}\bigg|_{\boldsymbol{x}_i^u} - \frac{\partial^2 \psi}{\partial x^{u2}}\frac{\partial D_2}{\partial u}\bigg|_{\boldsymbol{x}_i^u}\right)^2$$

Loss term for initial condition

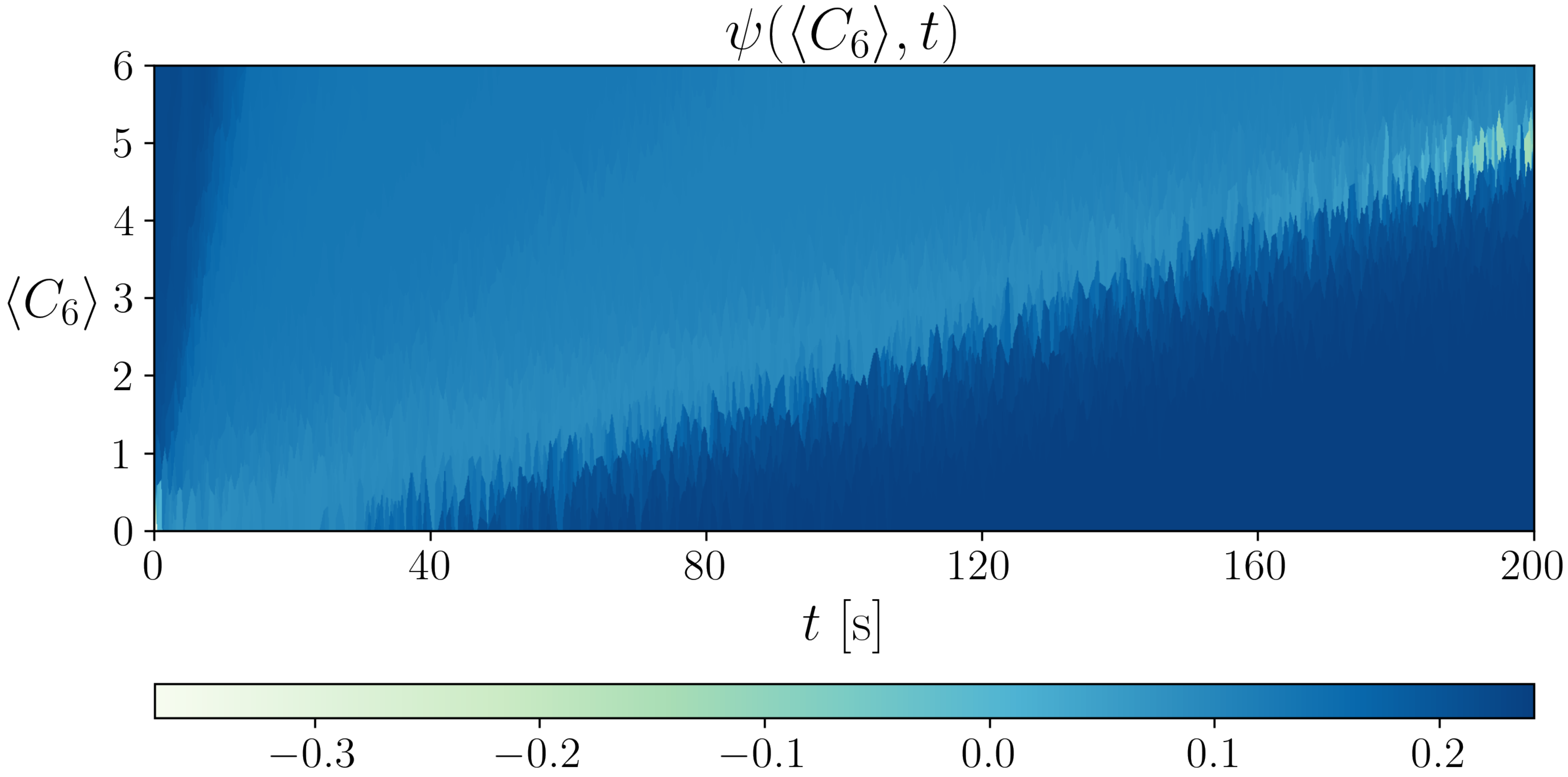
$$\mathcal{L}_{\rho_0^u} = \frac{1}{n}\sum_{i=1}^{n}\left(\rho^u\Big|_{t=0} - \rho_0^u(x)\right)^2$$

Loss term for terminal condition

$$\mathcal{L}_{\rho_T^u} = \frac{1}{n}\sum_{i=1}^{n}\left(\rho^u\Big|_{t=T} - \rho_T^u(x)\right)^2$$

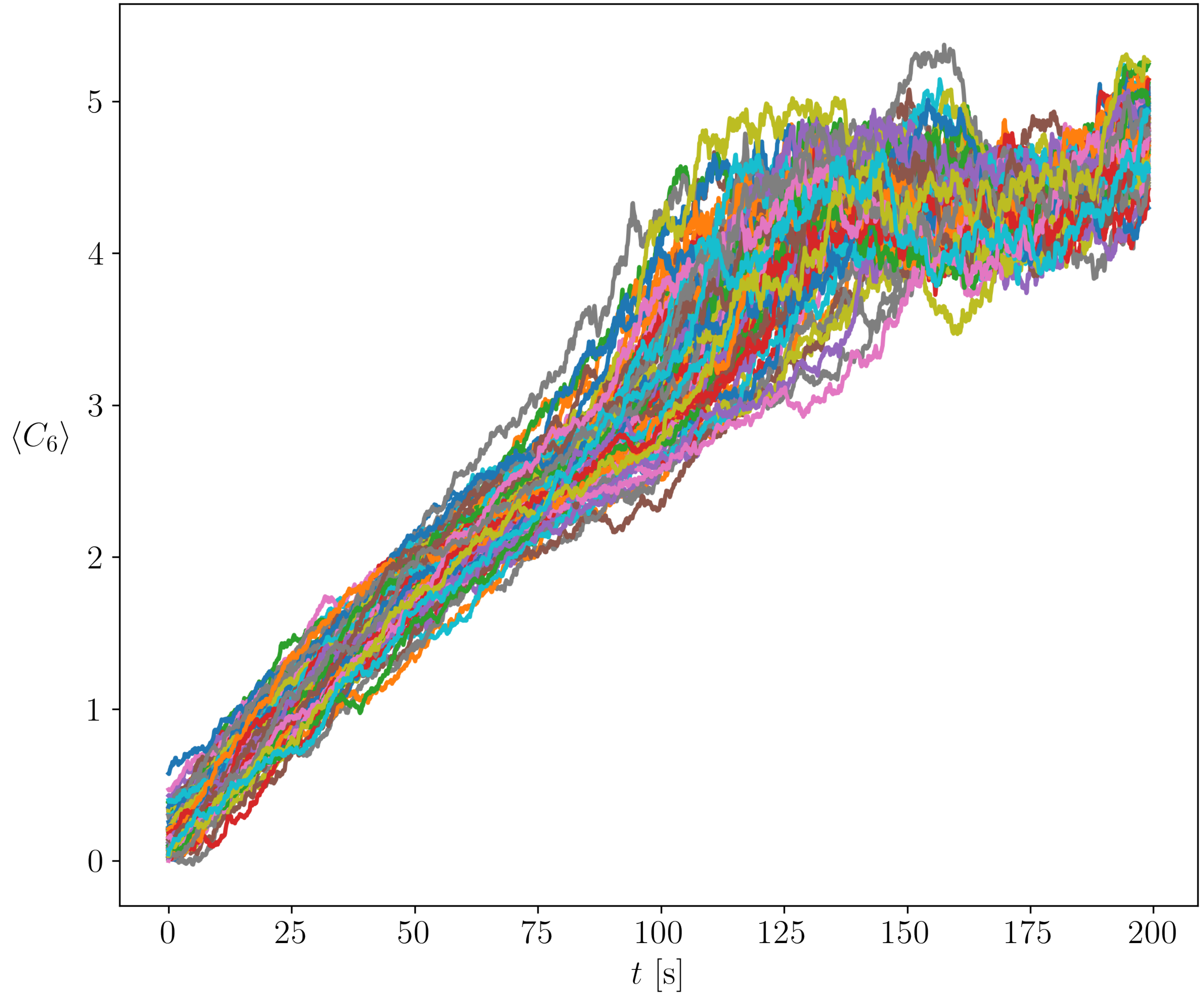
Stochastic Control/ Control Non-affine

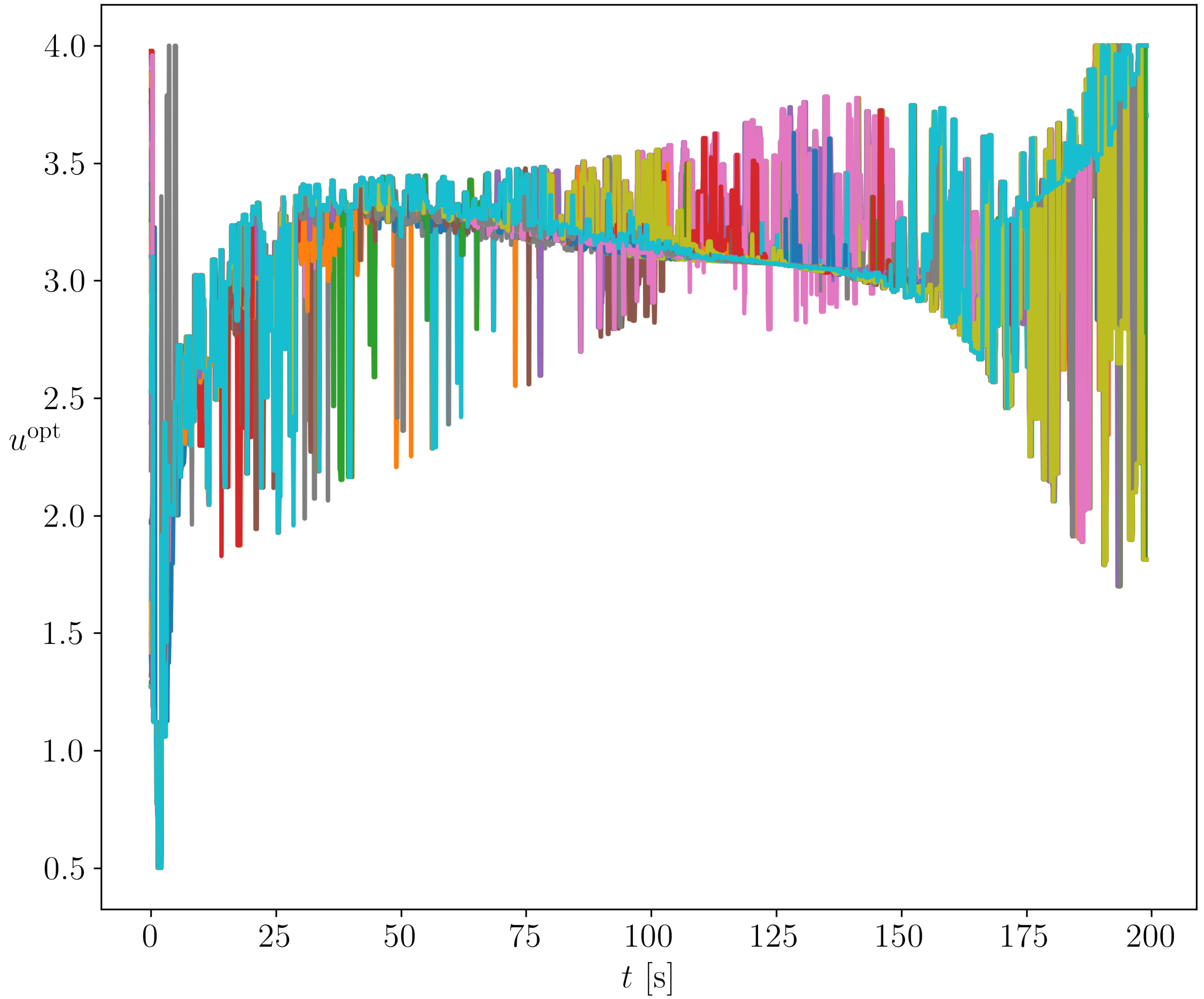
Case 1: Value Function



Stochastic Control/ Control Non-affine

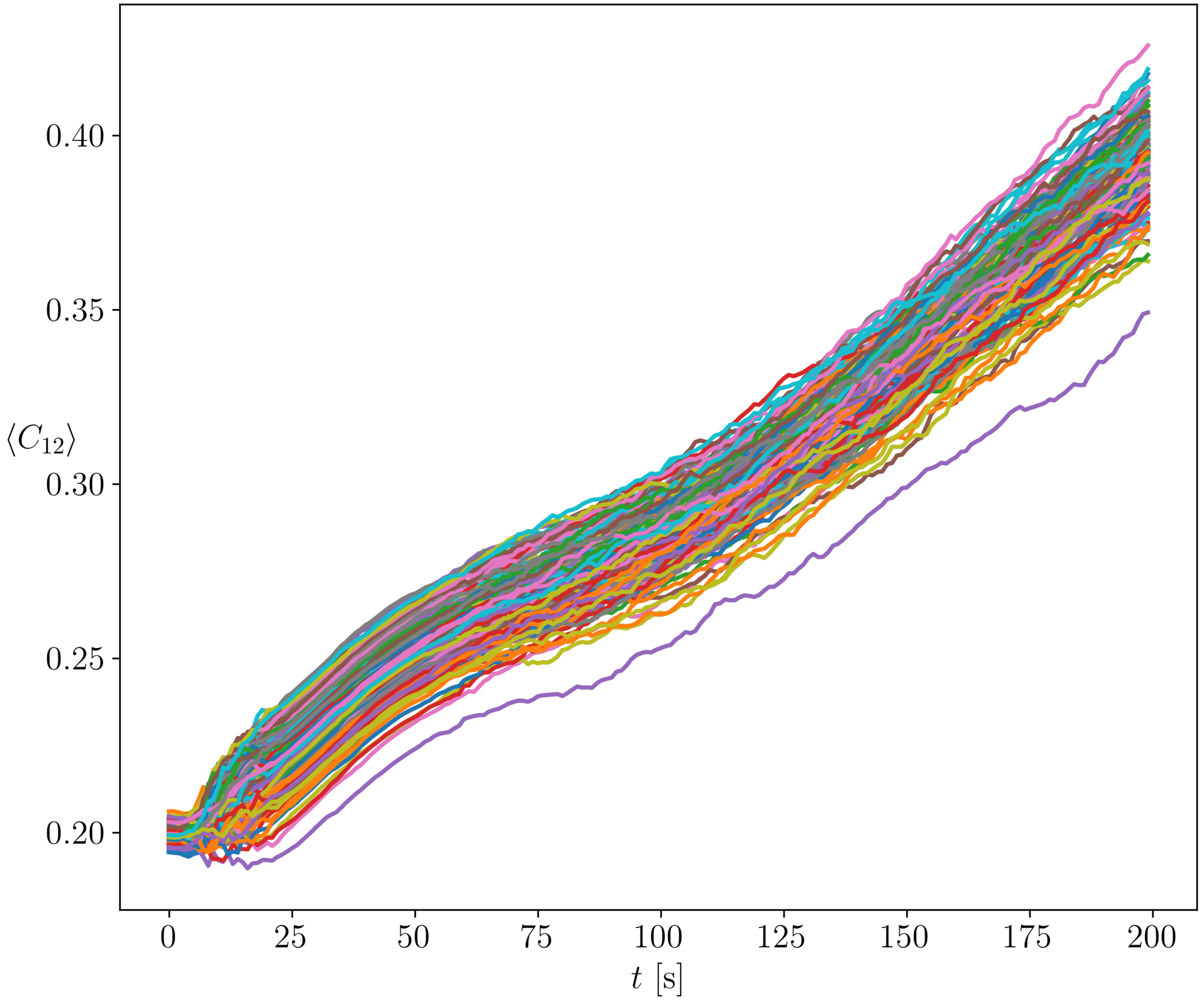
Case 1: Optimal State and Optimal Control Sample Paths

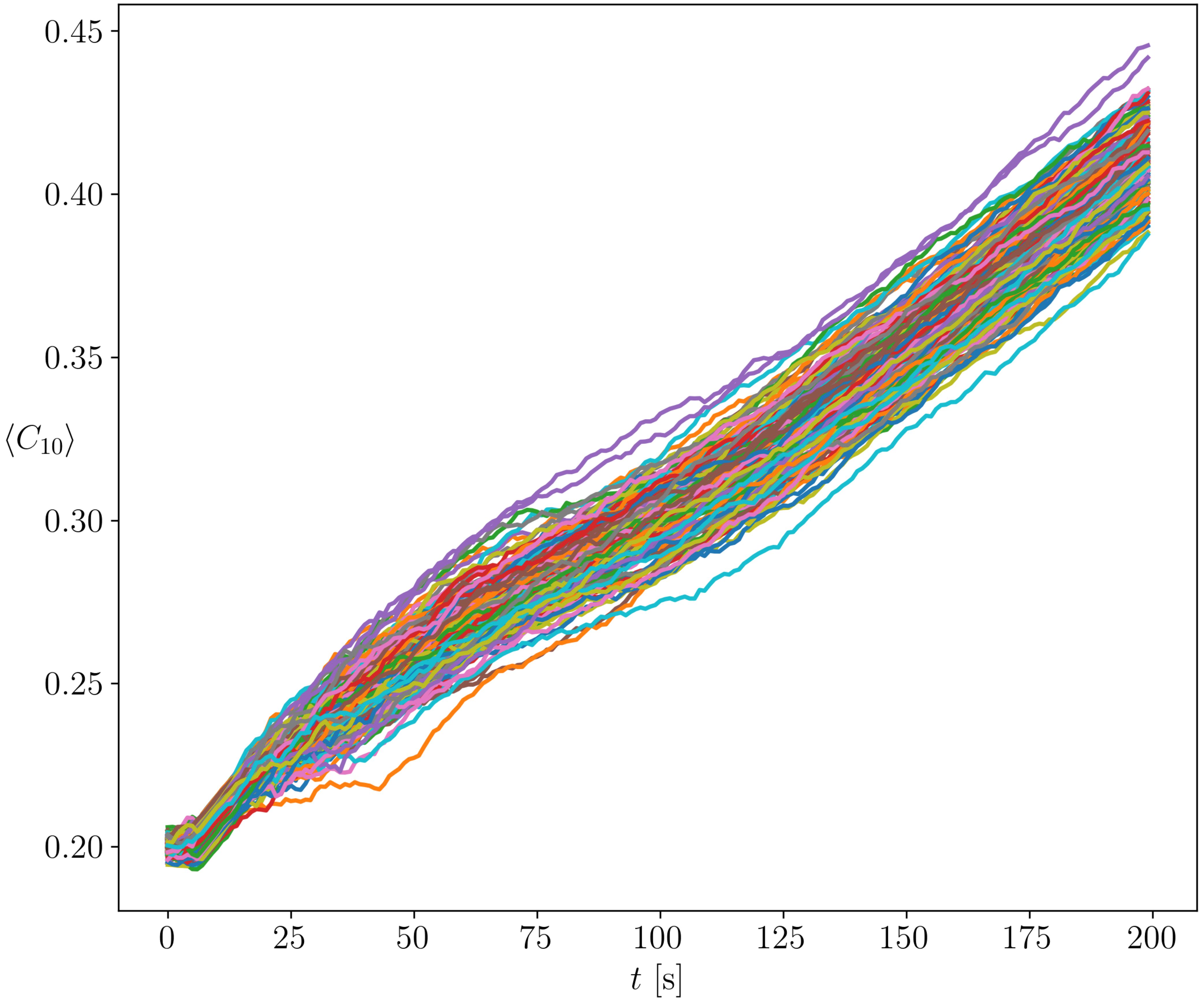




Stochastic Control/ Control Non-affine

Case 2: Closed Loop State Sample Paths





Desired transport from mean (0.2, 0.2) to (0.40,0.37) for BCC structure

Stochastic Modeling

Existing state-of-the-art

Several works on modeling the finite population:

[Lu et. al., Appl. Phys. Lett., 2014] [Edward and Bevan, Langmuir, 2014] [Matei et. al., CDC, 2020]

[Matei et. al., CDC, 2021]

[Lefevre et. al., IEEE/ASME Trans. on Mechatronics, 2022]

How to steer the large finite population toward desired pattern:

Vectorize the positions of all chiplets, then apply MPC [Matei et. al., US Patent 17121411]

Computation does not scale ... need new ideas

Discrete Version of the Proposed ADMM

Euclidean distance matrix

Outer layer ADMM

$$\boldsymbol{\mu}_i^{k+1} = \operatorname{prox}^{W}_{\frac{1}{\alpha}(F_i(\boldsymbol{\mu}_i)+\langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i\rangle)}(\boldsymbol{\zeta}^k)$$

$$= \operatorname*{arg\,inf}_{\boldsymbol{\mu}_i \in \Delta^{N-1}} \left\{ \min_{\boldsymbol{M} \in \Pi_N(\boldsymbol{\mu}_i, \zeta^k)} \frac{1}{2} \langle \boldsymbol{C}, \boldsymbol{M} \rangle + \frac{1}{\alpha} \left(F_i(\boldsymbol{\mu}_i) + \langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i \rangle \right) \right\}$$

$$\boldsymbol{\zeta}^{k+1} = \operatorname*{arg\,inf}_{\boldsymbol{\zeta} \in \Delta^{N-1}} \left\{ \left(\sum_{i=1}^{n} \min_{\boldsymbol{M}_i \in \Pi_N(\boldsymbol{\mu}_i^{k+1}, \boldsymbol{\zeta})} \frac{1}{2} \langle \boldsymbol{C}, \boldsymbol{M}_i \rangle \right) - \frac{2}{\alpha} \langle \boldsymbol{\nu}_{\text{sum}}^k, \boldsymbol{\zeta} \rangle \right\}$$

Inner layer ADMM

$$\boldsymbol{\nu}_i^{k+1} = \boldsymbol{\nu}_i^k + \alpha(\boldsymbol{\mu}_i^{k+1} - \boldsymbol{\zeta}^{k+1})$$

where N is the number of samples

With Sinkhorn regularization:

Discrete Sinkhorn divergence

Outer layer ADMM

$$\boldsymbol{\mu}_i^{k+1} = \operatorname{prox}^{W_\varepsilon}_{\frac{1}{\alpha}(F_i(\boldsymbol{\mu}_i)+\langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i\rangle)}(\boldsymbol{\zeta}^k)$$

$$= \operatorname*{arg\,inf}_{\boldsymbol{\mu}_i \in \Delta^{N-1}} \left\{ \min_{M \in \Pi_N(\boldsymbol{\mu}_i, \zeta^k)} \left\langle \frac{1}{2} \boldsymbol{C} + \varepsilon \log \boldsymbol{M}, \boldsymbol{M} \right\rangle + \frac{1}{\alpha} \left(F_i(\boldsymbol{\mu}_i) + \langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i \rangle \right) \right\}$$

$$\boldsymbol{\zeta}^{k+1} = \operatorname*{arg\,inf}_{\boldsymbol{\zeta} \in \Delta^{N-1}} \left\{ \left(\sum_{i=1}^{n} \min_{\boldsymbol{M}_i \in \Pi_N(\boldsymbol{\mu}_i^{k+1}, \boldsymbol{\zeta})} \left\langle \frac{1}{2} \boldsymbol{C} + \varepsilon \log \boldsymbol{M}_i, \boldsymbol{M}_i \right\rangle \right) - \frac{2}{\alpha} \langle \boldsymbol{\nu}_{\text{sum}}^k, \boldsymbol{\zeta} \rangle \right\}$$

Inner layer ADMM

$$\boldsymbol{\nu}_i^{k+1} = \boldsymbol{\nu}_i^k + \alpha(\boldsymbol{\mu}_i^{k+1} - \boldsymbol{\zeta}^{k+1})$$

Stochastic Learning/ Distributed Computing

$\boldsymbol{\mu}_i$ update $\rightsquigarrow$ Outer Consensus (Sinkhorn) ADMM

Example: $G_i(\boldsymbol{\mu_i}) := F_i(\boldsymbol{\mu}_i) + \langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i \rangle, \boldsymbol{\zeta}^k \in \Delta^{N-1}, \boldsymbol{k} \in \mathbb{N}_0$

$$\boldsymbol{\mu}_i^{k+1} = \operatorname{prox}^{W_\varepsilon}_{\frac{1}{\alpha}(F_i(\boldsymbol{\mu}_i)+\langle \boldsymbol{\nu}_i^k, \boldsymbol{\mu}_i\rangle)}(\boldsymbol{\zeta}^k) = \exp\left(\frac{\boldsymbol{\lambda}_{1i}^{\text{opt}}}{\alpha\varepsilon}\right) \odot \left(\exp\left(-\frac{\boldsymbol{C}^\top}{2\varepsilon}\right)\exp\left(\frac{\boldsymbol{\lambda}_{0i}^{\text{opt}}}{\alpha\varepsilon}\right)\right)$$

where $\boldsymbol{\lambda}_{0i}^{\text{opt}}, \boldsymbol{\lambda}_{1i}^{\text{opt}} \in \mathbb{R}^N$ solve

$$\exp\left(\frac{\boldsymbol{\lambda}_{0i}^{\text{opt}}}{\alpha\varepsilon}\right) \odot \left(\exp\left(-\frac{\boldsymbol{C}}{2\varepsilon}\right)\exp\left(\frac{\boldsymbol{\lambda}_{1i}^{\text{opt}}}{\alpha\varepsilon}\right)\right) = \boldsymbol{\zeta}_k$$

$$\boldsymbol{0} \in \partial_{\boldsymbol{\lambda}_{1i}^{\text{opt}}} G_i^*\left(-\boldsymbol{\lambda}_{1i}^{\text{opt}}\right) - \exp\left(\frac{\boldsymbol{\lambda}_{1i}^{\text{opt}}}{\alpha\varepsilon}\right) \odot \left(\exp\left(-\frac{\boldsymbol{C}^\top}{2\varepsilon}\right)\exp\left(\frac{\boldsymbol{\lambda}_{0i}^{\text{opt}}}{\alpha\varepsilon}\right)\right)$$

Stochastic Learning / Distributed Computing

ζ update $\rightsquigarrow$ Inner (Euclidean) ADMM

Theorem. Let $f_i(\boldsymbol{u}_i) := \langle \boldsymbol{\mu}_i^{k+1}, \log(\boldsymbol{\Gamma} \exp(\boldsymbol{u}_i/\varepsilon)) \rangle, \quad \boldsymbol{u}_i \in \mathbb{R}^N, \quad$ for all $i \in [n]$,

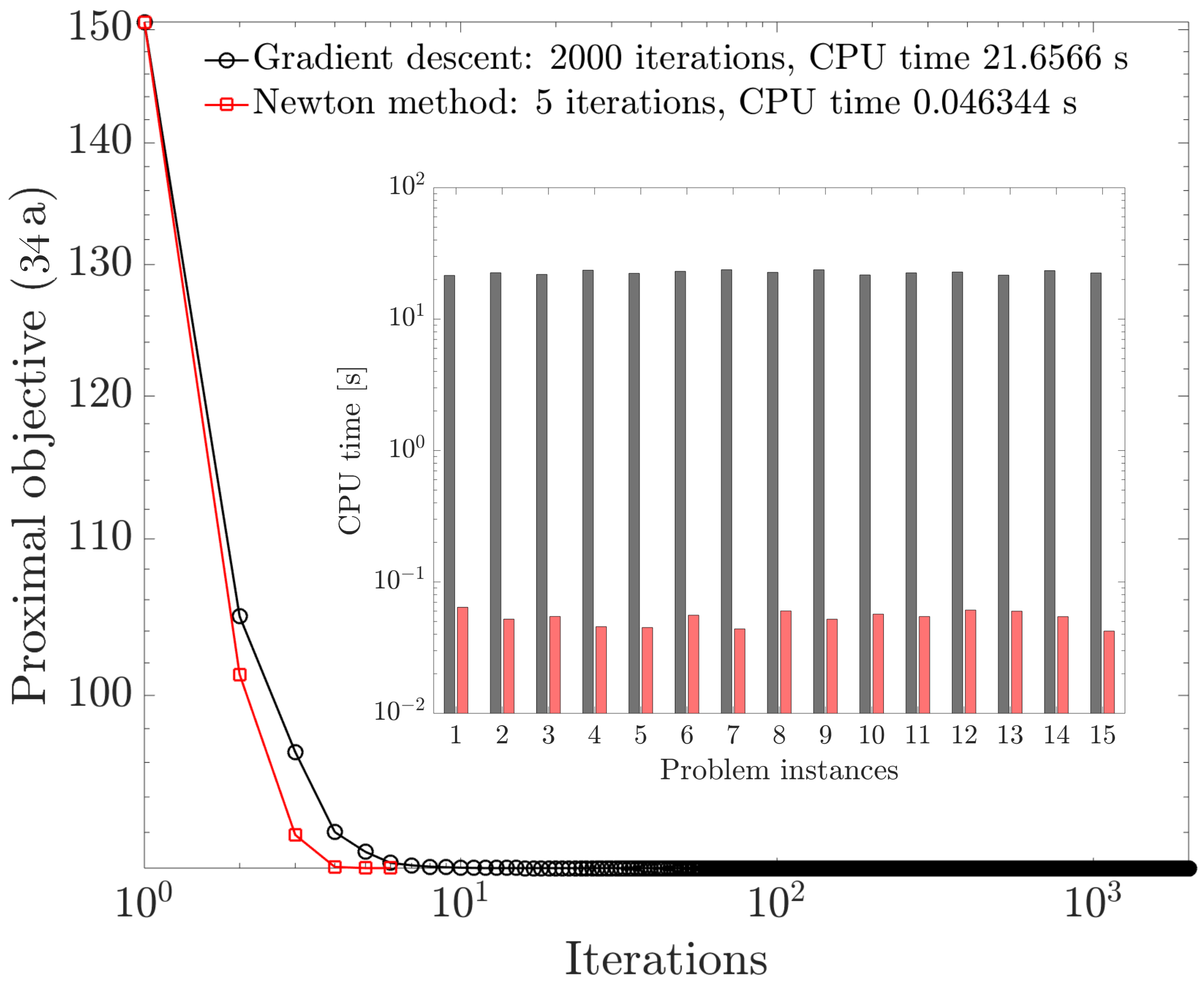
Then the following Euclidean ADMM solves

$$\boldsymbol{u}_i^{\ell+1} = \mathrm{prox}_{\frac{1}{\tau} f_i}^{\|\cdot\|_2}\left(\boldsymbol{z}_i^{\ell} - \tilde{\boldsymbol{\nu}}_i^{\ell}\right)$$

$$\boldsymbol{z}_i^{\ell+1} = \left(\boldsymbol{u}_i^{\ell+1} - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{u}_i^{\ell+1}\right) + \left(\tilde{\boldsymbol{\nu}}_i^{\ell} - \frac{1}{n}\sum_{i=1}^{n}\tilde{\boldsymbol{\nu}}_i^{\ell}\right) + \frac{2}{n\alpha}\boldsymbol{\nu}_{\mathrm{sum}}^{k}$$

$$\tilde{\boldsymbol{\nu}}_i^{\ell+1} = \tilde{\boldsymbol{\nu}}_i^{\ell} + \left(\boldsymbol{u}_i^{\ell+1} - \boldsymbol{z}_i^{\ell+1}\right)$$

No analytical solution, use e.g., Newton's method (has structured Hess)



Theorem.
Guaranteed convergence for inner layer ADMM under some constraints on hyper-parameters

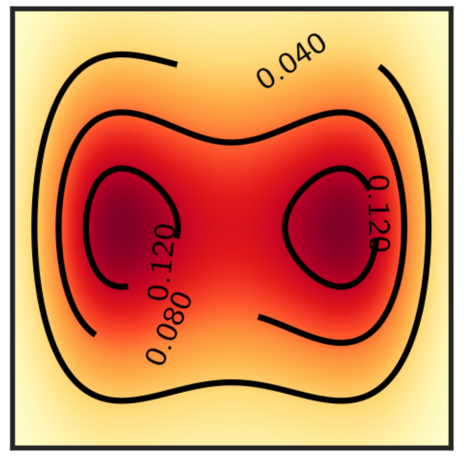
Stochastic Learning/ Distributed Computing

Experiment # 3 Linear Fokker-Planck-Kolmogorov PDE

$$\frac{\partial \mu}{\partial t} = \nabla \cdot (\mu \nabla V) + \beta^{-1} \Delta \mu$$

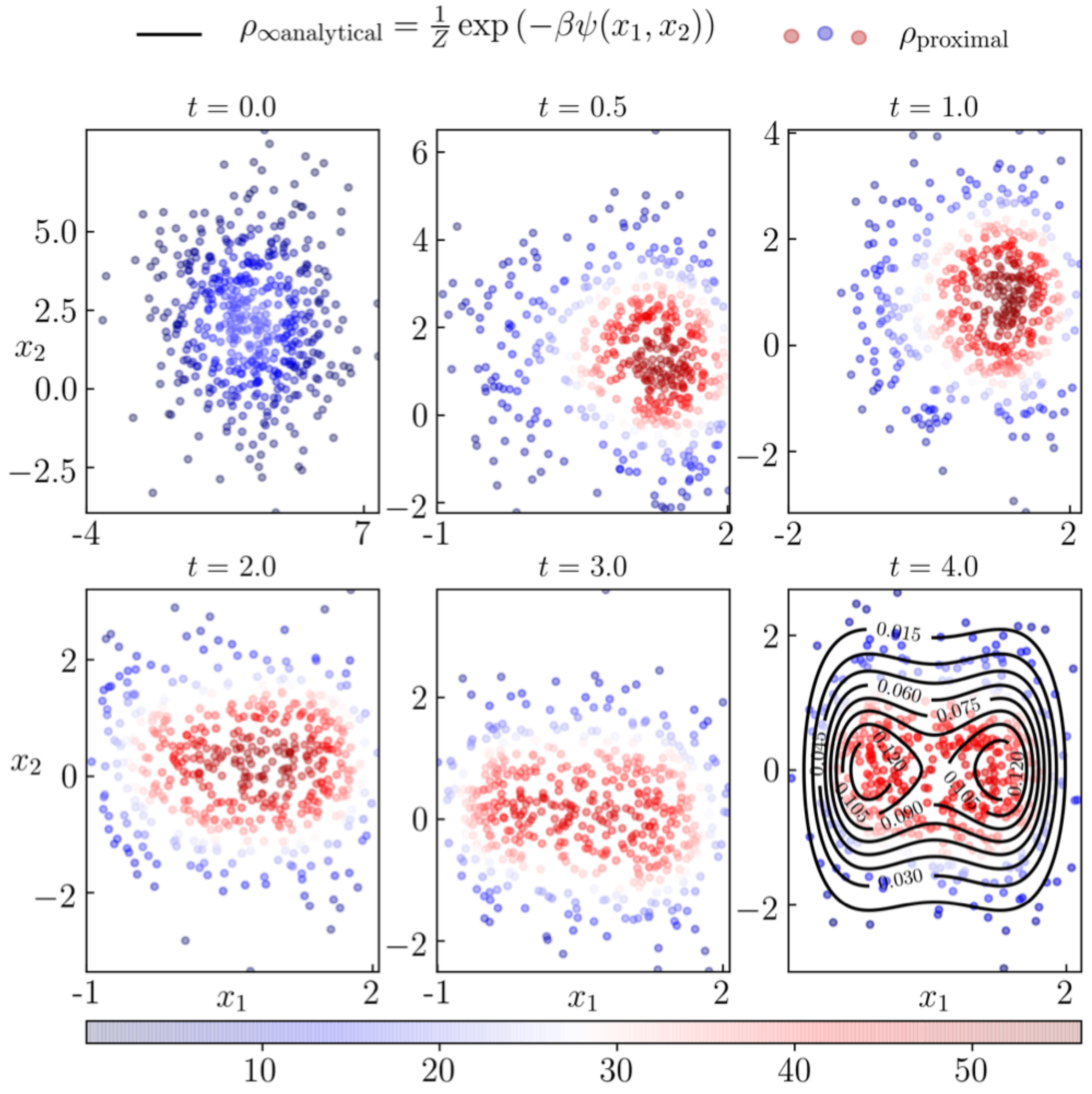
$$V(x_1, x_2) = \frac{1}{4}(1 + x_1^4) + \frac{1}{2}(x_2^2 - x_1^2)$$

$$\mu_\infty \propto \exp(-\beta V(x_1, x_2)) \mathrm{d}x_1 \, \mathrm{d}x_2$$

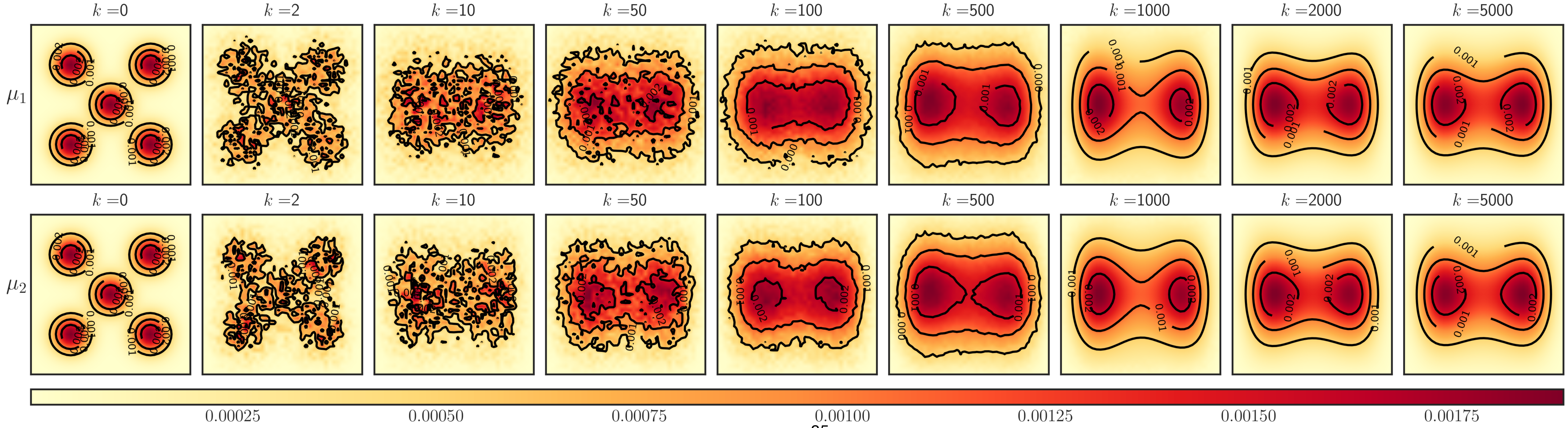


Centralized computation:

Caluya and Halder, *IEEE Trans. Automatic Control*, 2019



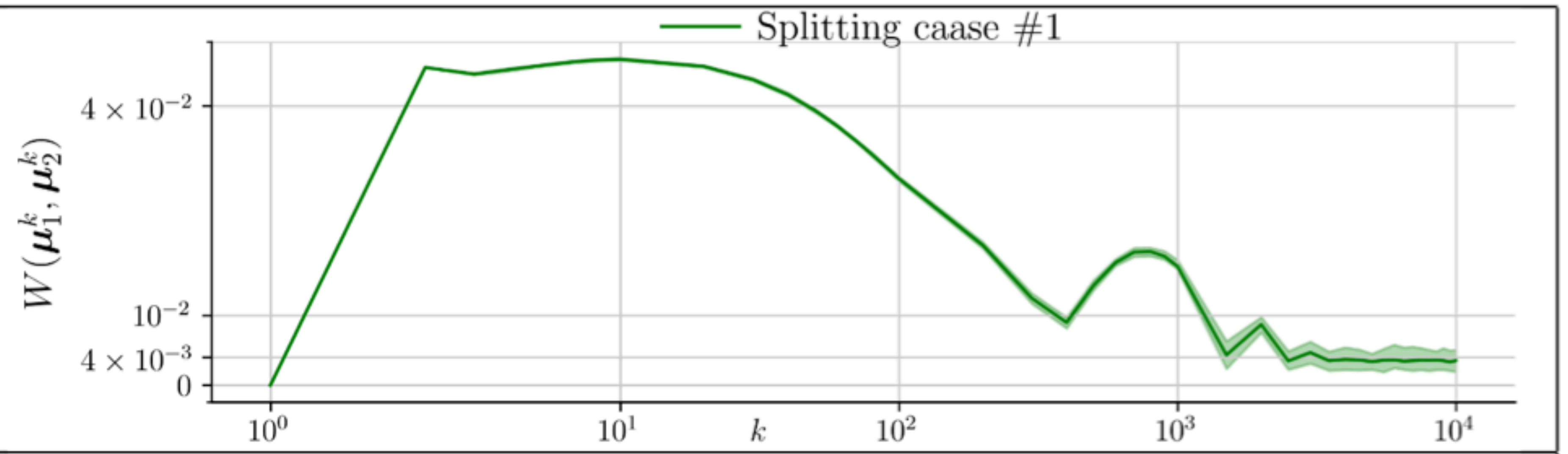
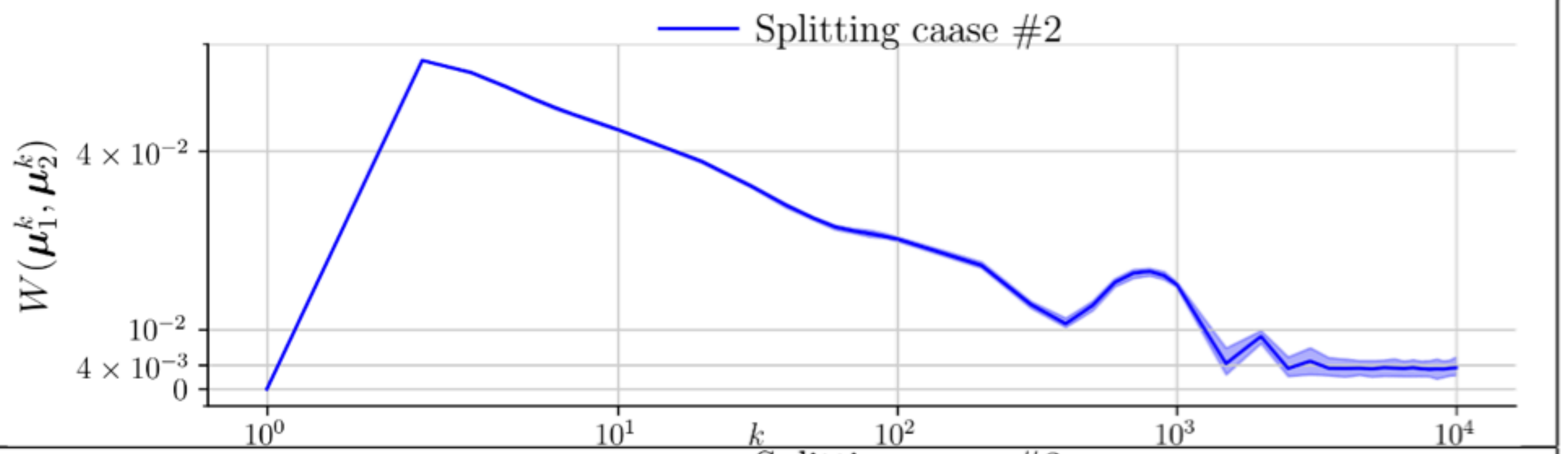
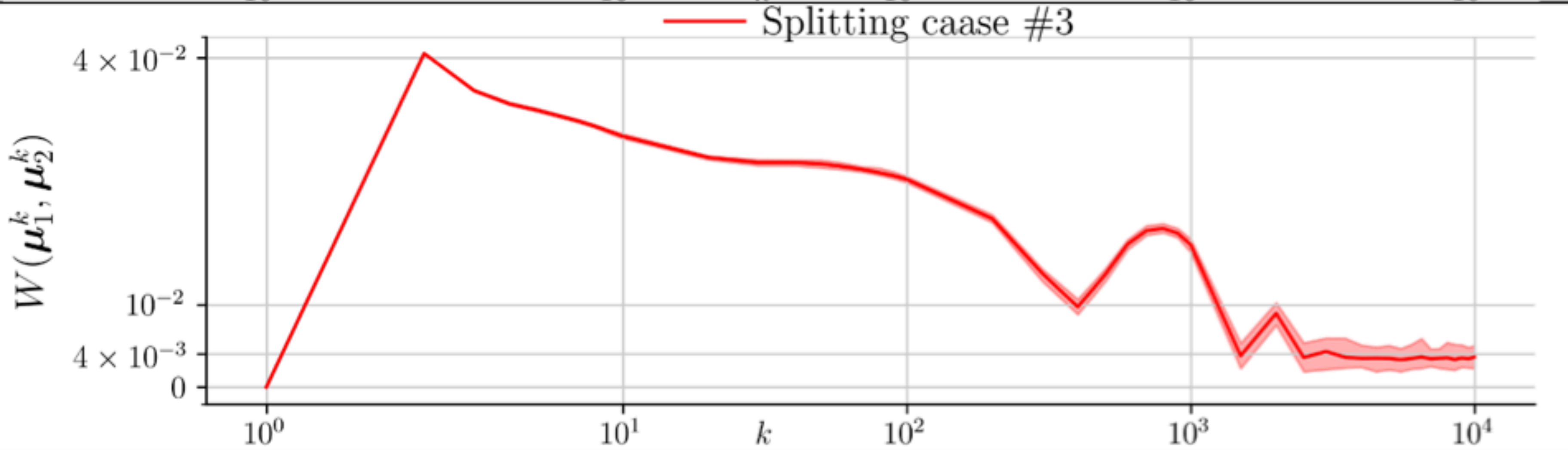
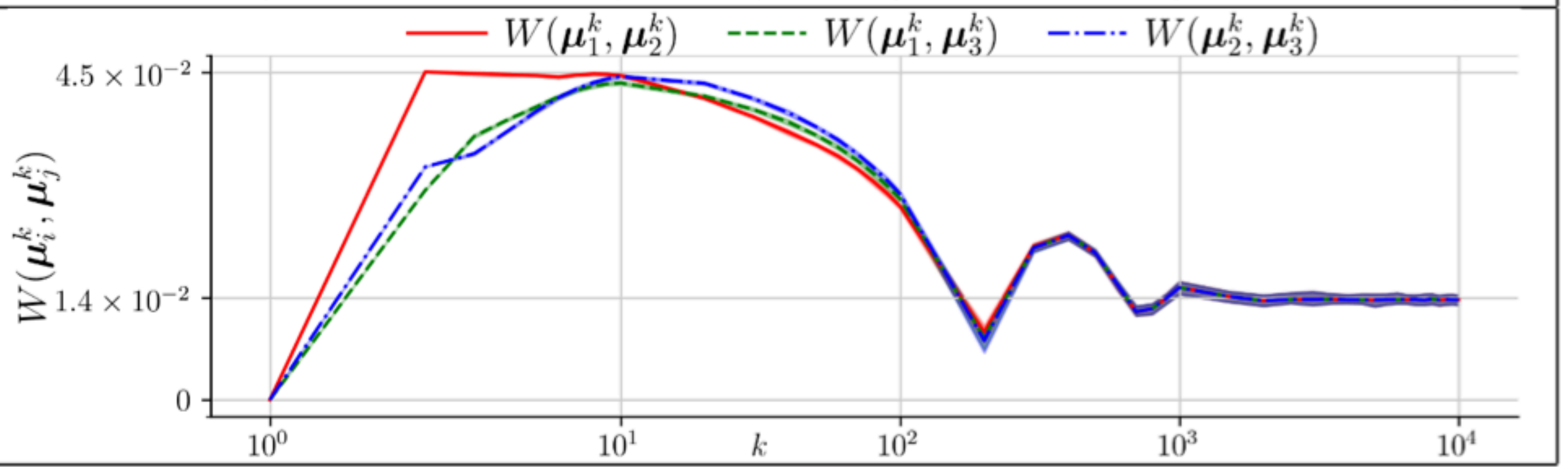
Distributed computation: $F_1(\boldsymbol{\mu}) = \langle \boldsymbol{V}_k, \boldsymbol{\mu} \rangle \quad F_2(\boldsymbol{\mu}) = \langle \beta^{-1} \log \boldsymbol{\mu}, \boldsymbol{\mu} \rangle$



Stochastic Learning/ Distributed Computing

Centralized av. runtime = 310.21 s

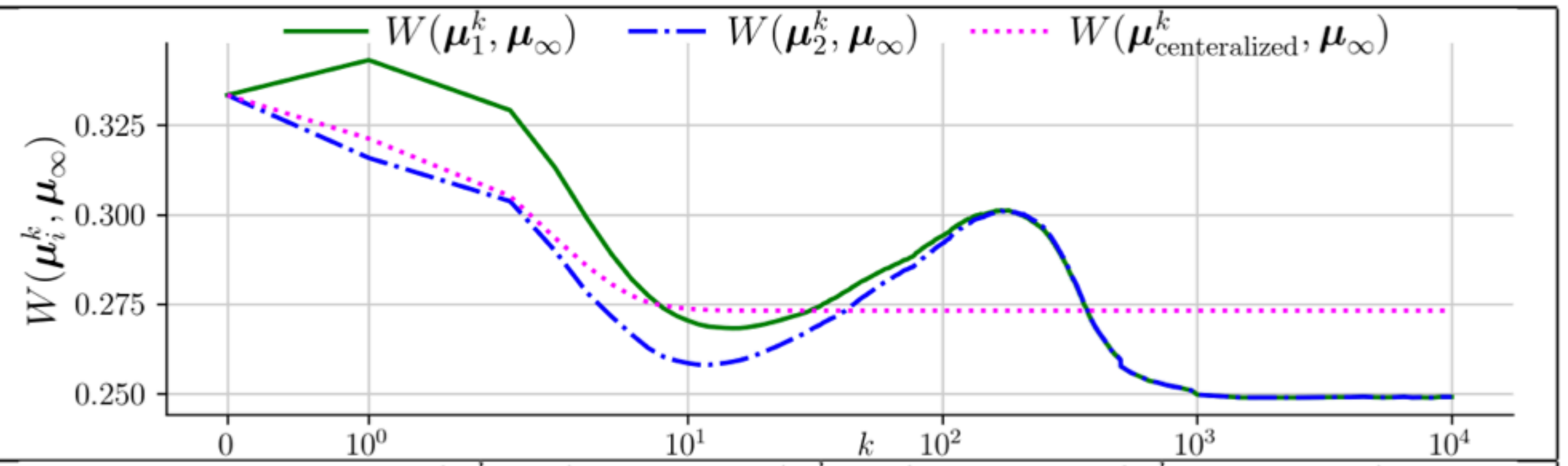
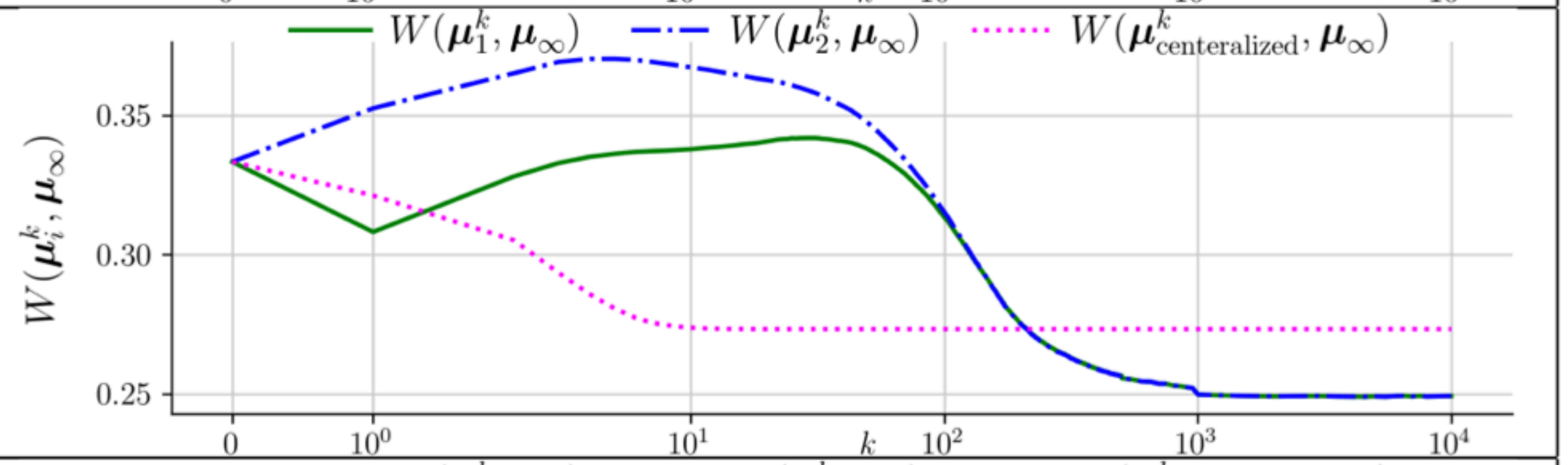
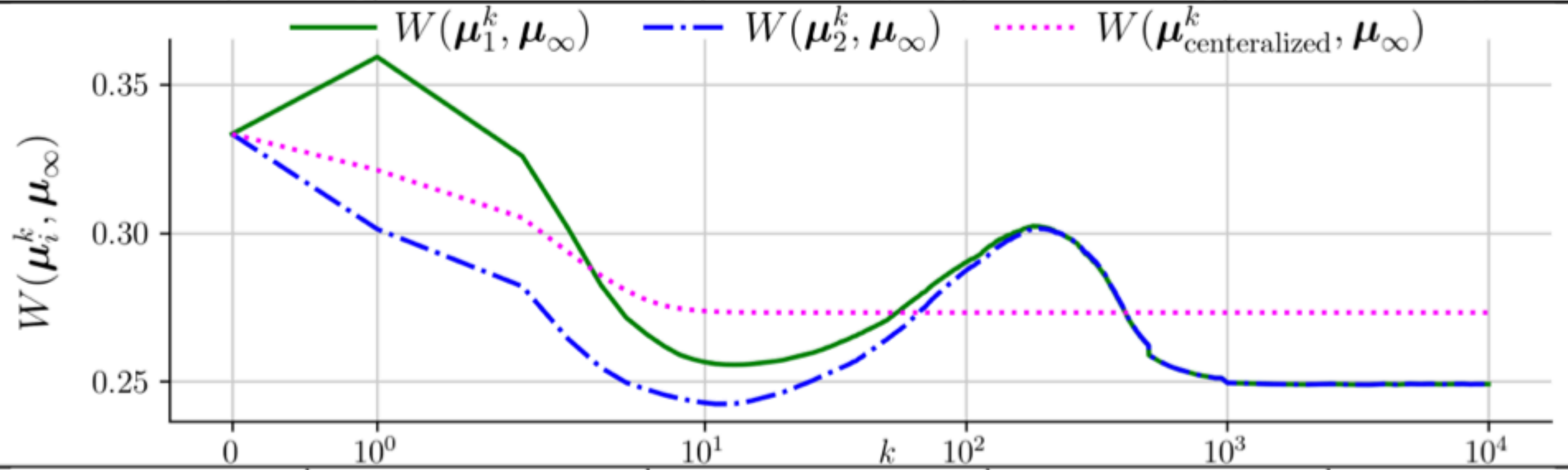
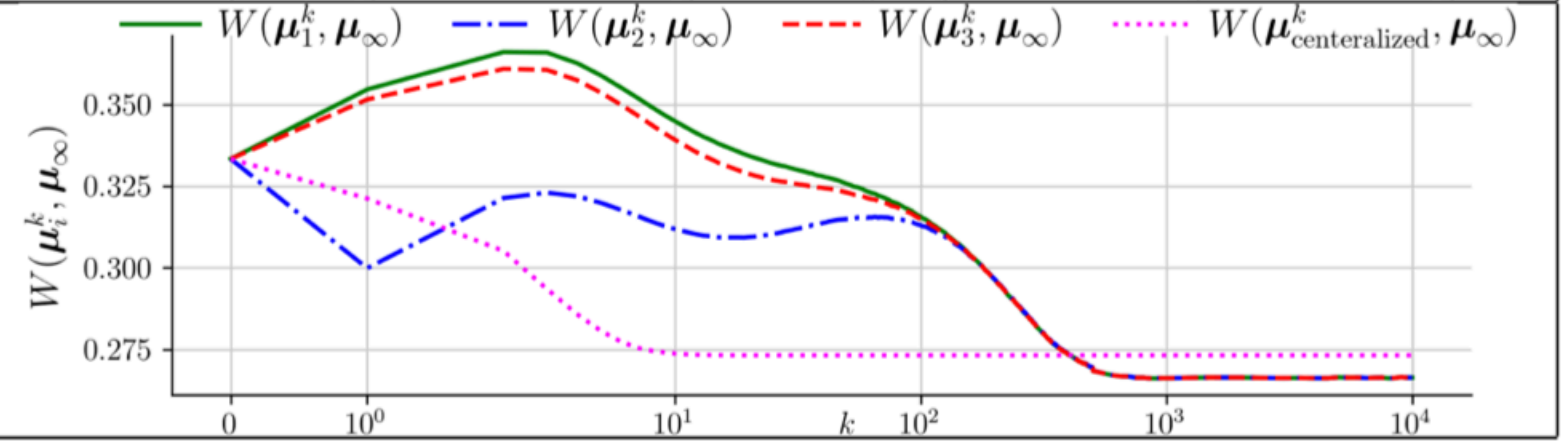
100 run statistics for each of the 4 ways of splitting: ($B_n - 1$ ways in general)

Splitting case	Functionals	Wasserstein distance
#1	$F_1(\boldsymbol{\mu}) = \left\langle \boldsymbol{V}_k + \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$, $F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{U}_k \boldsymbol{\mu}^k, \boldsymbol{\mu} \right\rangle$ av. runtime = 294.06 s	
#2	$F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}^k + \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \rangle$, $F_2(\boldsymbol{\mu}) = \langle \boldsymbol{V}_k, \boldsymbol{\mu} \rangle$ av. runtime = 285.32 s	
#3	$F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}^k + \boldsymbol{V}_k, \boldsymbol{\mu} \rangle$, $F_2(\boldsymbol{\mu}) = \left\langle \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$ av. runtime = 289.87 s	
#4	$F_1(\boldsymbol{\mu}) = \left\langle \boldsymbol{V}_k, \boldsymbol{\mu} \right\rangle$, $F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{U}_k \boldsymbol{\mu}^k \right\rangle$, $F_3(\boldsymbol{\mu}) = \left\langle \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$ av. runtime = 108.99 s	

Stochastic Learning/ Distributed Computing

Centralized is pink dotted (repeated in subplots)

100 run statistics for each of the 4 ways of splitting: ($B_n - 1$ ways in general)

Case	Functionals	Wasserstein distances
#1	$F_1(\boldsymbol{\mu}) = \left\langle \boldsymbol{V}_k + \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle,$ $F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{U}_k \boldsymbol{\mu}^k, \boldsymbol{\mu} \right\rangle$	
#2	$F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}^k + \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \rangle,$ $F_2(\boldsymbol{\mu}) = \langle \boldsymbol{V}_k, \boldsymbol{\mu} \rangle$	
#3	$F_1(\boldsymbol{\mu}) = \langle \boldsymbol{U}_k \boldsymbol{\mu}^k + \boldsymbol{V}_k, \boldsymbol{\mu} \rangle,$ $F_2(\boldsymbol{\mu}) = \left\langle \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$	
#4	$F_1(\boldsymbol{\mu}) = \langle \boldsymbol{V}_k, \boldsymbol{\mu} \rangle,$ $F_2(\boldsymbol{\mu}) = \left\langle \boldsymbol{U}_k \boldsymbol{\mu}^k, \boldsymbol{\mu} \right\rangle,$ $F_3(\boldsymbol{\mu}) = \left\langle \beta^{-1}\boldsymbol{\mu}, \boldsymbol{\mu} \right\rangle$	

Publications

- Alexis Teter, **Iman Nodozi,** and Abhishek Halder. "Proximal Mean Field Learning in Shallow Neural Networks." Transactions on Machine Learning Research, 2023.
- **Iman Nodozi**, Charlie Yan, Mira Khare, Abhishek Halder, and Ali Mesbah. "Neural Schrödinger Bridge with Sinkhorn Losses: Application to Data-driven Minimum Effort Control of Colloidal Self-assembly." IEEE Transactions on Control Systems Technology, 2023.
- **Iman Nodozi**, Abhishek Halder, and Ion Matei. "A Controlled Mean Field Model for Chiplet Population Dynamics." IEEE Control Systems Letters, 2023. Also in 62nd IEEE Conference on Decision and Control (CDC), Singapore, 2023.
- Charlie Yan, **Iman Nodozi**, and Abhishek Halder. "Optimal Mass Transport over the Euler Equation." Proceedings of the 62nd IEEE Conference on Decision and Control (CDC), Singapore, 2023. Invited paper in Session 'Optimal Transport'.
- **Iman Nodozi**, Jared O'Leary, Abhishek Halder, and Ali Mesbah. "A Physics-informed Deep Learning Approach for Minimum Effort Stochastic Control of Colloidal Self-Assembly." Proceedings of American Control Conference (ACC), San Diego, California, USA, 2023. Invited paper in Session 'Learning and Stochastic Optimal Control'.
- **Iman Nodozi**, and Abhishek Halder. "Schrödinger Meets Kuramoto via Feynman-Kac: Minimum Effort Distribution Steering for Noisy Nonuniform Kuramoto Oscillators." Proceedings of the 61st IEEE Conference on Decision and Control (CDC), Cancún, Mexico, 2022.
- **Iman Nodozi**, and Abhishek Halder. "A Distributed Algorithm for Measure-valued Optimization with Additive Objective." 25th International Symposium on Mathematical Theory of Networks and Systems (MTNS), Bayreuth, Germany, 2022. Invited paper in Session 'Optimal transport: Theory and applications in networks and systems'.
- Alexis Teter, **Iman Nodozi**, and Abhishek Halder. "Solution of the Probabilistic Lambert's Problem: Optimal Transport Approach."
- Alexis Teter, **Iman Nodozi**, and Abhishek Halder. "Solution of the Probabilistic Lambert Problem: Connections with Optimal Mass Transport, Schrödinger Bridge, and Reaction-Diffusion PDEs."
- **Iman Nodozi**, and Abhishek Halder. "Wasserstein Consensus ADMM."