Bayesian Generative AI for Power Systems and Beyond

Misha Chertkov

Applied Math @ UArizona

Jan 8, 2025, (6-th) LANL Grid Science, Santa Fe, NM

AI – Elephant in the Room

- **Dream** New AI Models
- Explain the Models

Worth Climbing the Beanstalk

- **Grow** the Models from – Stochastic Calculus, Optimal Control, Non-Equil Stat Mech

From Ugly (?) Duck to Swan

AI – Elephant in the Room

- **Dream** New AI Models
- Explain the Models

- **Grow** the Models from – Stochastic Calculus, Optimal Control, Non-Equil Stat Mech

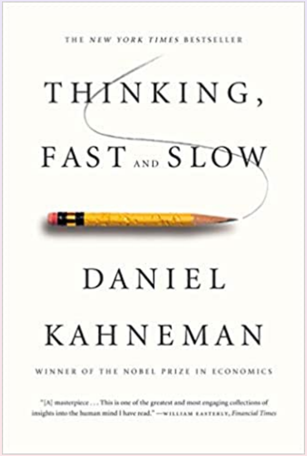
Worth Climbing the Beanstalk

From Ugly (?) Duck to Swan

Outline

1. AI ⊂ Applied Math
   - AI for Physical & Engineering Sciences
   - Key Math Topics in AI
2. Diffusion Models of AI
   - Score-Based Diffusion & Non-Eq. Stat Mech
   - Analysis of & Dynamic Phase Transitions in Diffusion
   - Diffusion & Stochastic Optimal Control
3. Down-Stream Applications
   - What Can One Do with Pre-Trained Model(s)?
   - Power/Energy Generative Desiderata

System 1 and System 2 in AI (for Sciences)

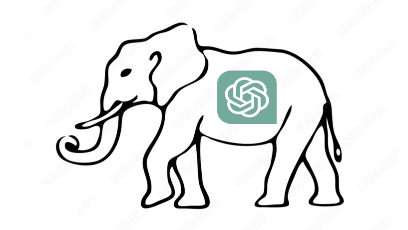


- System 1 – operates automatically & quickly
  - “black box" use of **AI**
- System 2 – allocates attention to effortfull mental activities
  - **Informed** (white box) use of **AI**
  - **Advancing AI** = building new AI with **Applied Math** (including Stat Mech)

Artificial Intelligence ⊂ Applied Math

Applied Math '21

- **Traditional Applied Math:** Driven by sciences & engineering
- **Contemporary Applied Math:** Foundations and frontiers in AI
- **Synthesis:** Integration of Traditional and Contemporary Approaches

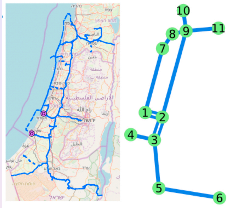


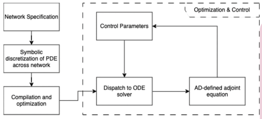
Upcoming Example of Advancing AI ⊂ Applied Math

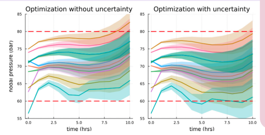
- From Stochastic ODEs to Generative Diffusion to ...

- Automatic Differentiation (AD):
  - Computes derivatives efficiently using elementary operations and the chain rule.
  - Major engine behind efficient optimization (billions of parameters) critical for "everything" AI
- Deep Learning:
- Reinforcement Learning (RL):
- Generative Models:

- Automatic Differentiation (AD):







- Focus on rare interruptions - multi-scenario optimization under uncertainty + sensitivity analysis
- Progress: **Augmenting AD with Symbolic Differentiation**
- Neural Network Free

- Academy & Industry (NOGA - energy system operator of Israel) Collaboration: Model Reduction & open-source via Julia
- Pipe Line Simulation Interest Group - best student award arXiv:2304.01955,; CDC 2024 (invited) arXiv2310.18507, 2311.08686

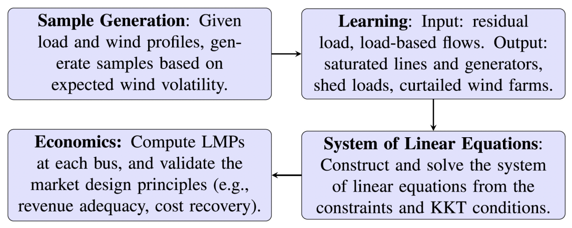
- Deep Learning:
- Reinforcement Learning (RL):
- Generative Models:

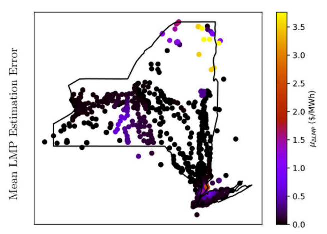
- Automatic Differentiation (AD):
- Deep Learning:
    - Represents functions through many-layered (deep), parameterized nonlinear transformations, expressed via Neural Networks (NN).
        - Even if an explicit formula for the function in questions exists – it may be costly to evaluate it (like solving OPF or UC for a given demand). Calling NN - trained to approximate the function may be cheaper.
        - NN – when trained in the supervised format (input+labels) may allow to generalize – generate samples for labels not seeing in training (used in AlphaFold to sample a stable 3D structure given a protein = label)
    - Employs Automatic Differentiation for optimizing NNs.
- Reinforcement Learning (RL):
- Generative Models:

- Automatic Differentiation (AD):
- Deep Learning:

Physics-Informed Machine Learning for Electricity Market

- IEEE Transactions on Energy Markets, 2/1 (2024)



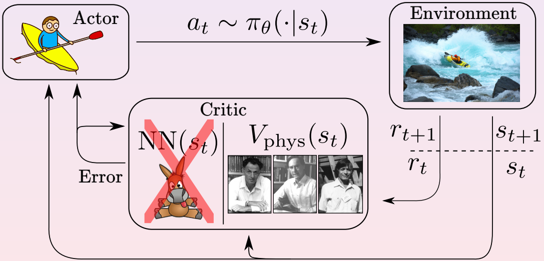


- Identify least understood bottleneck (stressed power lines)
  - Reinforcement Learning (RL):
  - Generative Models:
- Utilize Deep Learning to resolve the bottleneck

- Automatic Differentiation (AD):
- Deep Learning:
- Reinforcement Learning (RL):
  - A data-driven approach to optimal control under uncertainty using Deep Learning to predict and control actions.
  - Adaptive/dynamic approach – reinforces decisions based on the information, e.g. reward, received in the process of learning/exploration, to get better inference/exploitation.
- Generative Models:

- Automatic Differentiation (AD):
- Deep Learning:
- Reinforcement Learning (RL):

Physics-Informed RL for Swimming in Turbulence



- identify least understood bottleneck (weak NN critic)
- utilize Theory (Stochastic Hydrodynamics) to fix it
- Physical Review Reports (to appear), arXiv:2406.10242

- Generative Models:

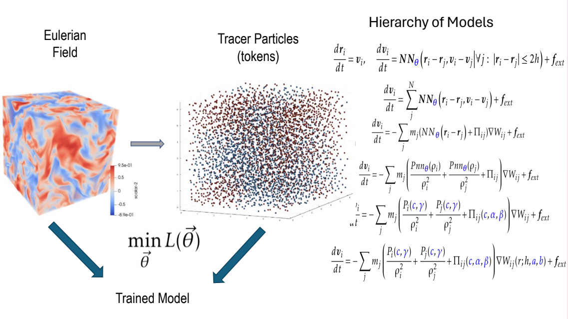
- Automatic Differentiation (AD):
- Deep Learning:
- Reinforcement Learning (RL):
- Generative Models:

- Outputs new data from existing (ground truth) data. The new data may be statistically similar to the ground truth (e.g., diffusion) or comes in response to prompts (e.g., transformers).
- Leverages Deep Learning to generate synthetic data and elements of optimal control/Reinforcement Learning, in achieving optimality

GPT-4 prompt: Show UArizona bobcat swimming butterfly style in a turbulent river.

- Automatic Differentiation (AD):
- Deep Learning:
- Reinforcement Learning (RL):
- Generative Models:

Neural Smooth Particle Hydrodynamics = GPT of Turbulence

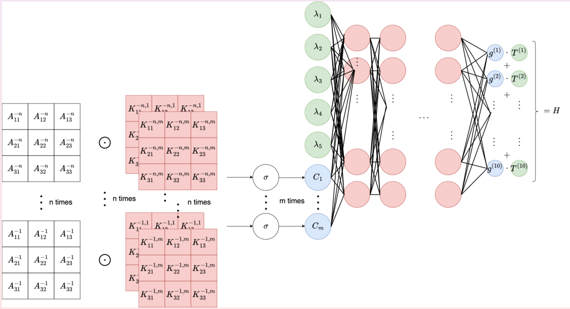


- quasi-particles = tokens
- "physical" attentions via particles & fields
- Lagrangian Large Eddy Simulations = New Reduced Order Model of Turbulence – 10.1103/PhysRevFluids.8.054602 & 10.1073/pnas.2213638120

- Automatic Differentiation (AD):
- Deep Learning:
- Reinforcement Learning (RL):
- Generative Models:

Lagrangian Deformation Neural Transformer

- for velocity gradient in turbulence



- attention (auto-regressive) module to represent Lagrangian memory
- C. Hyett et al (work in progress)

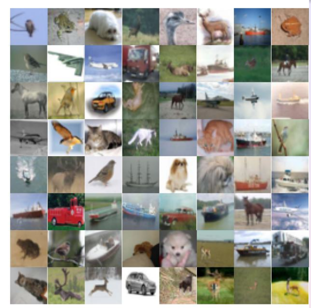
Outline

Hamidreza Behjoo (UArizona)

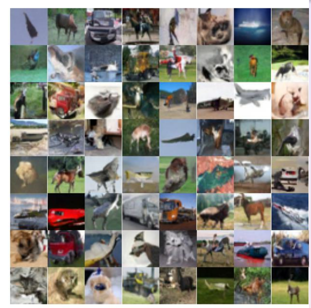
HB & MC

- U-Turn Diffusion, arXiv:2308.07421
- Space-Time Diffusion Bridges, arXiv:2402.08847 (MTNS '24)
- Harmonic Path Integral Diffusion, arXiv:2409.15166

Score-Based Diffusion



Train on Ensemble of Samples



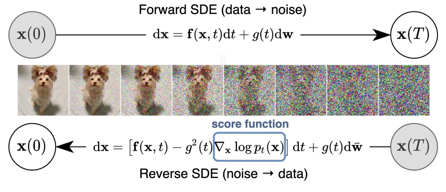
Generate Synthetic Samples

Generate Synthetic Samples which are i.i.d. from a target probability distribution, $p_{\text{target}}(\cdot)$, represented

- implicitly via Ground Truth samples
- or explicitly via energy function, $p_{\text{target}}(\cdot) \propto \exp(-E(\boldsymbol{x}))$

Diffusion Models = Non-Autonomous Stochastic ODE

Score Based Diffusion (SBD)



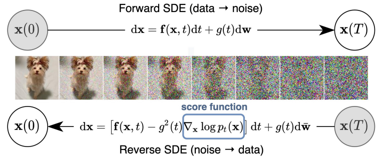
- Reverse-Time Diffusion [1,2]
- Reference (vague) to Stat. Thermodynamics [3]

- Continuous time SBD – state-of-the-art in GenAI [4]

---

[1] B. Anderson, Reverse-time diffusion equation models, 1982
[2] H. Fölmer, Time Reversal on Wiener Spaces, 1986
[3] J. Sohl-Dickstein, et al, Deep Unsupervised Learning using Nonequilibrium Thermodynamics, 2015
[4] Y. Song, et.al., Score-Based Generative Modeling through Stochastic Differential Equations, 2021

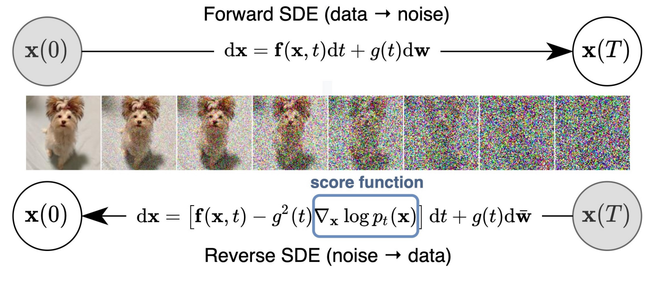
Score-Based Diffusion – Matched with NNs



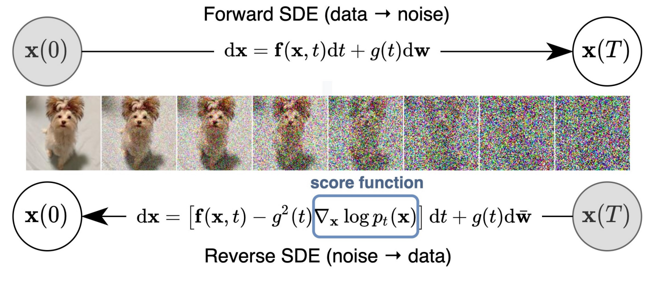
$$\partial_t p + \nabla_i (f_i p) = \frac{g^2(t)}{2} \nabla^2 p$$

$$\partial_t p + \nabla_i (f_i p) - g^2(t) \nabla_i s_i p = -\frac{g^2(t)}{2} \nabla^2 p$$

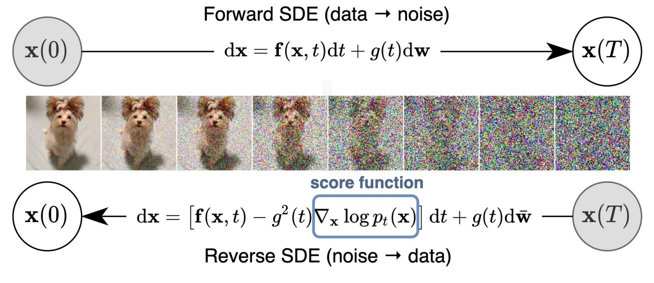
- Tractable Forward = Computations not Simulations
  - e.g. $\boldsymbol{f} = 0$: $\boldsymbol{s}(\boldsymbol{x}_t, t) = \nabla_{\boldsymbol{x}} \log \left(\sum_{n=1}^{N} \mathcal{N}\left(\boldsymbol{x}_t | \boldsymbol{x}^{(n)}; 2\hat{\boldsymbol{I}} \int_0^t dt' g(t') \right) \right)$
  - many choices, e.g. space-time & bridges arXiv:2402.08847
- Detailed Balance built in
  - can also brake it, e.g. deterministic reverse process
- Simulate Reverse. "Match" Score with NN:
  $\min_\theta \mathbb{E}_{t \sim U(0,T), \boldsymbol{x_0} \sim p_0(\cdot), \boldsymbol{x}_t \sim p_t(\cdot|\boldsymbol{x_0})} \left[\frac{\lambda(t)}{2} \| \mathbf{NN}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{s}(\boldsymbol{x}_t, t) \|_2^2 \right]$
  - avoids memorization + efficiency of inference



- $T = \infty$ ... Can we start the reverse process earlier?



- $T = \infty$ ... Can we start the reverse process earlier?
- How to initialize the reverse process?

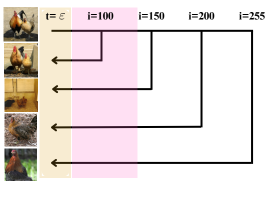


- $T = \infty$ ... Can we start the reverse process earlier?
- How to initialize the reverse process?

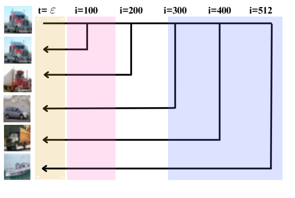
U-Turn at some finite T_u

- At T_u – Initialize the reverse with outcome of the forward
  - Initialize forward process with a Ground Truth sample
  - Compute (not simulate !) the outcome of the forward process

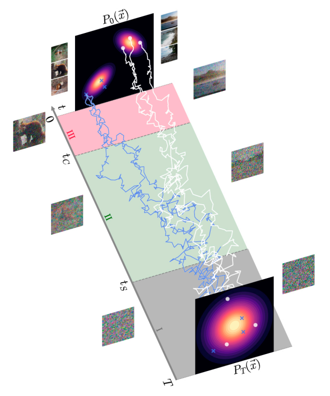
Dynamic Phase Transitions in Score Based Diffusion



ImageNet (Single Class)



CIFAR-10 (one score)



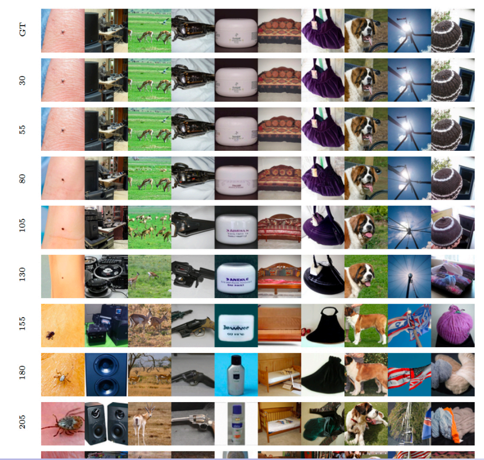
- When to U-turn? – Memorization/Collapse Transition [1,2]
- Spontaneous Choice of the Specie/Class – Speciation Transition [2,1]

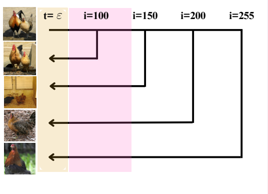
[1] HB, MC, U-turn Diffusion, 2023

[2] G. Biroli, et al, Dynamical Regimes of Diffusion Models, 2024

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from https://www.image-net.org/ & https://www.cs.toronto.edu/~kriz/cifar.html – 50,000 images each



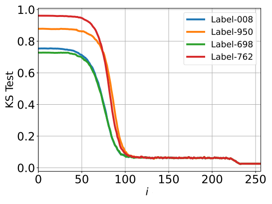


ImageNet-64 (class constrained)

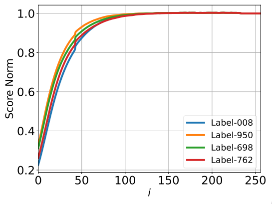
- T_m - detection
- T_m fluctuates

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

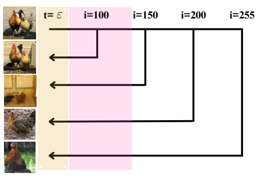
- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each



Kolmogorov-Smirnov (KS) Test



Score Norm (SN) Test



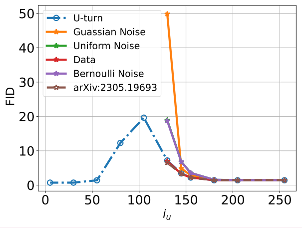
ImageNet-64 (class constrained)

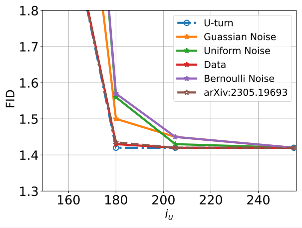
- T_m - detection
- T_m fluctuates

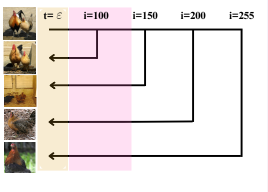
- 008 (hen), 950 (orange), 698 (palace), 762 (restaurant, eating house, eatery)
- Smooth, yet significant memorization transition in both tests
- Weak sensitivity to labels in the SN, more variations in KS at short T_u

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each







ImageNet-64 (class constrained)

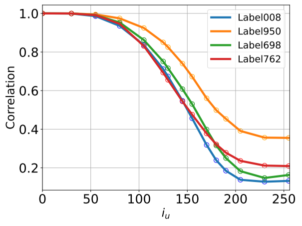
- T_m - detection
- T_m fluctuates

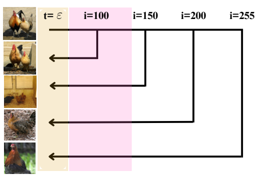
- Fréchet Inception Distance (FID) – standard comparison of multi-variate Gaussian proxies for real and generated data

- Useful if U-Turn is compared to other initializations of the reverse process
- FID – a reasonable indicator of memorization transition

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each





ImageNet-64 (class constrained)

- T_m - detection
- T_m fluctuates

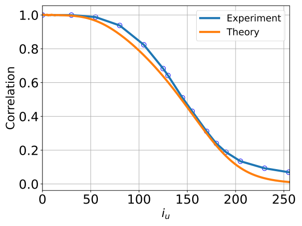
- U-Turn Auto-Correlation (AC) Function:

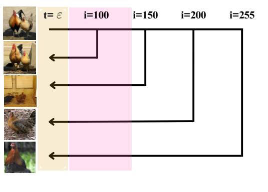
$$C_{UT}(T_u) = \frac{1}{N}\sum_{n=1}^{N} \frac{(\mathbf{x}^{(n)}(0))^T \mathbf{y}^{(n)}(0)}{(\mathbf{x}^{(n)}(0))^2}$$

- See in U-Turn AC too – smooth, yet significant memorization transition
- Strongest Variability (than in other tests) with labels – best indicator of the memorization transition we saw so-far

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeurIPS'22; $\beta(t) = 2t$; data from https://www.image-net.org/ & https://www.cs.toronto.edu/~kriz/cifar.html – 50,000 images each





ImageNet-64 (class constrained)

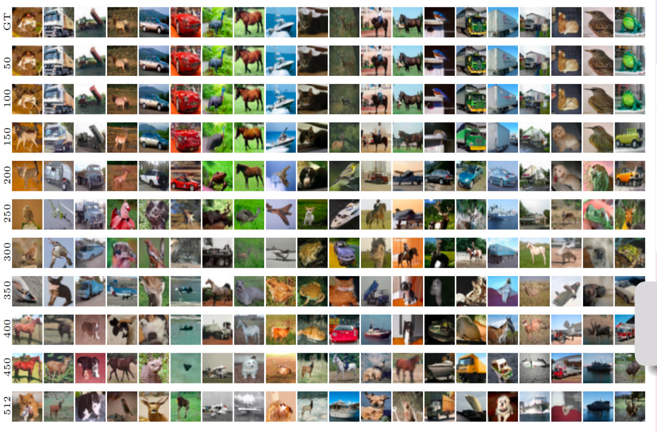
- T_m - detection
- T_m fluctuates

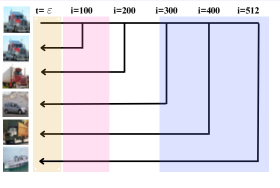
- Fully averaged over classes – $C_{UT}(T_u) = \frac{1}{N}\sum_{n=1}^{N} \frac{(x^{(n)}(0))^T y^{(n)}(0)}{(x^{(n)}(0))^2}$

- Empirical vs Gaussian Theory (score function is linear)
- Good Match – dependence on classes averages out

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeurIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each



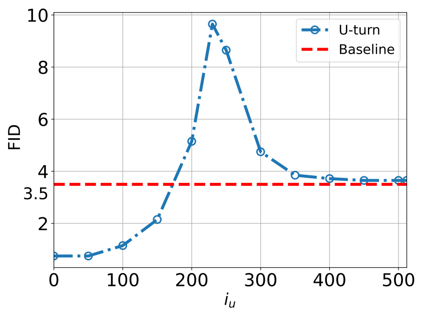


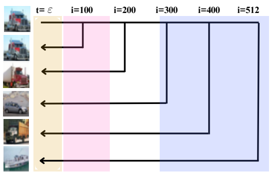
CIFAR-10 (multi-class)

- T_m, T_u - detection
- T_m, T_u fluctuate

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeurolPS'22; $\beta(t) = 2t$; data from https://www.image-net.org/ & https://www.cs.toronto.edu/~kriz/cifar.html – 50,000 images each





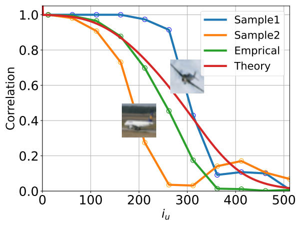
CIFAR-10 (multi-class)

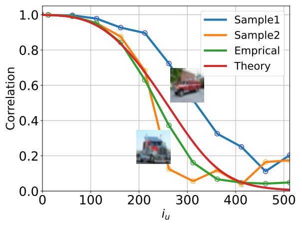
- T_m, T_u - detection
- T_m, T_u fluctuate

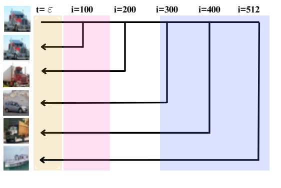
- FID vs T_u (U-Turn) against baseline (standard SBD)

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each







CIFAR-10 (multi-class)

- U-Turn Auto-Correlator conditioned on individual GT sample(s)
- Strong Variability between classes and within a class
- Quite far from Gaussian (affine score function) theory

- T_m, T_u - detection
- T_m, T_u fluctuate

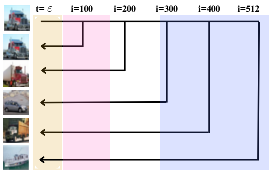
U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from https://www.image-net.org/ & https://www.cs.toronto.edu/~kriz/cifar.html – 50,000 images each

How Nonlinear (in x) the score function is?

G(aussian)-Turn

- Split Reverse Dynamics
  - $[T_g \leftarrow T \approx \infty]$ - Linear Score Function = simulations
  - $[0 \leftarrow T_g]$ - Nonlinear Score Function = computations
- Study Inference – Scanning different T_g

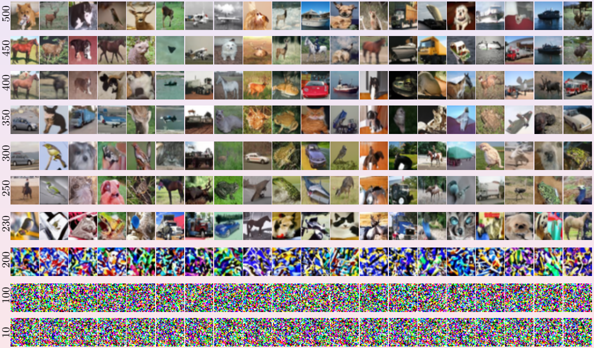


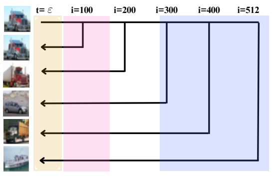
CIFAR-10 (multi-class)

- T_m, T_u - detection
- T_m, T_u fluctuate

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeurIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each





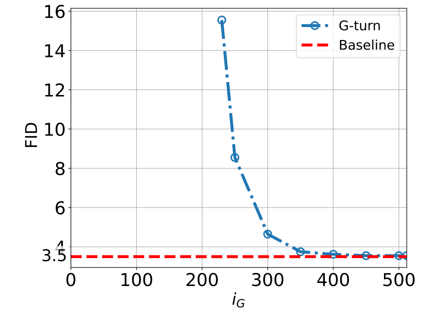
CIFAR-10 (multi-class)

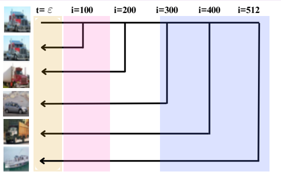
- T_m, T_u - detection
- T_m, T_u fluctuate

- Visual Examination ⇒ G-Turn is sucessful at sufficiently large $T_g \approx T_s$

U-Turn Diffusion (HB+MC, arXiv:2308.07421)

- Utilize Trained Models – T. Karras, et al – NVIDIA-Finland, NeuroIPS'22; $\beta(t) = 2t$; data from `https://www.image-net.org/` & `https://www.cs.toronto.edu/~kriz/cifar.html` – 50,000 images each





CIFAR-10 (multi-class)

- T_m, T_u - detection
- T_m, T_u fluctuate

- FID is the lowest at $T_g \approx T_s$
- FID is satisfactory at $T_g \in [T_m, T_s]$

U-Turn Diffusion: Summary & Path Forward

Summary

- Strong Tool to analyze Dynamic Phase Transitions in T_u in pre-trained models
  - Memorization, T_m
  - Speciation, T_s
- Strong sensitivity of T_m and T_s to classes to initial GT image
- Score-function (drift) in the reversed (de-noising) process is strongly non-affine at $T_u < T_m$, weakly non-affine at $T_u \in [T_m, T_s]$, affine at $T_u > T_s$

(Future) Extensions & Applications

- Self-classification – discovery of classes/clusters and their hierarchy
- Cleaning corrupted images (DJ talk) – "cleaning" transition?
- Other degrees of freedom (than T_u) to experiment with:
  - Spatio-temporal mixing – drift in forward equation
  - Breaking Detailed Ballance, e.g. deterministic de-noising

- We have discussed (some) ways to CONTROL DIFFUSION?

Can we re-state DIFFUSION

- as a (STOCHASTIC OPTIMAL) CONTROL?

Score Based Diffusion & Stochastic Optimal Control

Score Based Diffusion as "Integrable" Stochastic Optimal Control

$$\min_{\boldsymbol{u}(0\to 1;\boldsymbol{x}(0\to 1))} \mathbb{E}\left[\int_0^1 dt \frac{|\boldsymbol{u}(t;\boldsymbol{x}(t))|^2}{2} \Big| \text{Eqs. } (*,**)\right]$$

$$\text{s.t. } t\in[0,1]: \ d\boldsymbol{x}(t) = \boldsymbol{u}(t;\boldsymbol{x}(t))dt + d\boldsymbol{W}(t), \ \boldsymbol{x}(0)=0 \ (*)$$

$$p(\boldsymbol{x}(1)) = p_{\text{target}}(\boldsymbol{x}(1)) \qquad (**)$$

- Grow target distribution from point source
- Theory for sampling from $p_{\text{target}}(\cdot)$ [1], based on
  - "Integrability": Nonlinear HJB ⇒ Hopf '50 -Cole '51 ⇒ Diffusion (Mitter '81, Pavon '89)

[1] M.Tzen, M.Ragynsky, Theoretical guarantees .. with latent diffusions, 2019

Score Based Diffusion & Stochastic Optimal Control

Score Based Diffusion as "Integrable" Stochastic Optimal Control

$$\min_{\boldsymbol{u}(0\to 1;\boldsymbol{x}(0\to 1))} \mathbb{E}\left[\int_0^1 dt \frac{|\boldsymbol{u}(t;\boldsymbol{x}(t))|^2}{2} \middle| \text{Eqs. } (*, **)\right]$$

$$\text{s.t. } t \in [0,1]: \ d\boldsymbol{x}(t) = \boldsymbol{u}(t;\boldsymbol{x}(t))dt + d\boldsymbol{W}(t), \ \boldsymbol{x}(0) = 0 \ (*)$$

$$p(\boldsymbol{x}(1)) = p_{\text{target}}(\boldsymbol{x}(1)) \qquad (**)$$

- Grow target distribution from point source
- Efficient Algorithms
  - Path Integral Sampling – Fitting Control with NN, Expansive (repetitive forward propagation of SDE) [3]
  - Iterative Denoising Energy Matching – sampling to estimate score-function [4]

[3] Q.Zhang, Y.Chen, Path Integral Sampler, 2022
[4] T. Akhound-Sadegh, et al, Iterative Denoising Energy Matching, 2024

Integrable SOC with Potential, Forced & Gauged

Integrable SOC with a "Potential"

$$\min_{\boldsymbol{u}(0\to 1;\boldsymbol{x}(0\to 1))} \mathbb{E}\left[\int_0^1 dt \left(\frac{|\boldsymbol{u}(t;\boldsymbol{x}(t))|^2}{2} + V(t;\boldsymbol{x}(t))\right) \Big| \text{Eqs. } (*), (**)\right]$$

$$\text{s.t. } t\in[0,1]: \quad d\boldsymbol{x}(t) = \boldsymbol{u}(t;\boldsymbol{x}(t))dt + d\boldsymbol{W}(t), \quad \boldsymbol{x}(0)=0 \ (*)$$

$$p(\boldsymbol{x}(1)) = p_{\text{target}}(\boldsymbol{x}(1)) \qquad (**)$$

Path Integral Diffusion – Our Contribution reported today

- "Integrable" SOC in a Potential, Forced and Gauged
  - Grow target distribution from a point source
  - Based on Path Integral Control (PIC) [1]
    - control & diffusion are co-dimensional

[1] H. J. Kappen, *Path integrals ... for optimal control theory*, 2005.

Integrable SOC with Potential, Forced & Gauged

(Top Level) Integrable SOC in a Potential. Forced (& Gauged)

$$\min_{\boldsymbol{u}(\cdot;\boldsymbol{x}(\cdot))} \mathbb{E}\left[\int_0^1 dt\left(\frac{|\boldsymbol{u}(t;\boldsymbol{x}(t))|^2}{2}+V(t;\boldsymbol{x}(t))+\dot{\boldsymbol{x}}^T(t)\boldsymbol{A}(t;\boldsymbol{x}(t))\right)\Big|\text{Eqs. }(*,**)\right]$$

$$\text{s.t. } t\in[0,1]:\ d\boldsymbol{x}(t)=\boldsymbol{f}(t;\boldsymbol{x}(t))+\boldsymbol{u}(t;\boldsymbol{x}(t))dt+d\boldsymbol{W}(t),\ \boldsymbol{x}(0)=0\ (*)$$

$$p(\boldsymbol{x}(1))=p_{\text{target}}(\boldsymbol{x}(1))\qquad (**)$$

Path Integral Diffusion – Our Contribution reported today

- "Integrable" SOC in a Potential, Forced and Gauged
  - Grow target distribution from a point source
  - Based on Path Integral Control (PIC) [1]
  - Field & Gauge Extension of PIC [2]

[1] H. J. Kappen, *Path integrals ... for optimal control theory*, 2005.

[2] V. Chernyak, MC, J. Bierkens, H.J. Kappen, *SOC as Non-Eq Stat Mech: Calculus of Variations over Density and Current*, 2013

Path Integral Diffusion – (Top Level) Integrability

Hamilton-Jacobi Bellman for Cost-to-go (from t to 1)

$-\partial_t J = V + \frac{1}{2}\left(\nabla^T(\nabla J + \boldsymbol{A}) - |\nabla J + \boldsymbol{A}|^2\right) + \boldsymbol{f}^T(\nabla J + \boldsymbol{A})$
Optimal Control: $\boldsymbol{u}^* = -\nabla J - \boldsymbol{A}$

Hopf-Cole: $J(t;\boldsymbol{x}) = -\log\psi(t;\boldsymbol{x})$

$-\partial_t\psi + \tilde{V}\psi + \tilde{\boldsymbol{A}}^T\nabla\psi = \frac{1}{2}\Delta\psi, \quad \psi(1;\boldsymbol{x}) = \exp\left(-\phi(\boldsymbol{x})\right)$, $\phi(\cdot)$ is the terminal cost correspondent to $p_{\text{target}}(\cdot)$
$\tilde{V} \doteq V + \frac{1}{2}\nabla^T\boldsymbol{A} + \boldsymbol{f}^T\boldsymbol{A} - \frac{1}{2}|\boldsymbol{A}|^2, \quad \tilde{\boldsymbol{A}} \doteq \boldsymbol{A} - \boldsymbol{f}$
$\partial_t p^* + \nabla^T\left(p^*(\nabla\log\psi - \tilde{\boldsymbol{A}})\right) = \frac{1}{2}\Delta p^*, \quad p^*(0;\boldsymbol{x}) = \delta(\boldsymbol{x}), \quad p^*(1;\boldsymbol{x}) = p_{\text{target}}(\boldsymbol{x})$

Optimal Control via Green Functions:
$u^*(t;\boldsymbol{x}(t)) = \nabla_x \log\left(\int d\boldsymbol{y}\ p_{\text{target}}(\boldsymbol{y})\frac{G_-(t;\boldsymbol{x}(t);\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)}\right)$

$t \in [1\to 0]: \quad -\partial_t G_- + \tilde{V}(\boldsymbol{x};t)G_- + \tilde{\boldsymbol{A}}^T\nabla G_- = \frac{1}{2}\Delta G_-, \quad G_-(1;\boldsymbol{x}) = \delta(\boldsymbol{x}-\boldsymbol{y})$
$t \in [0\to 1]: \quad \partial_t G_+ + \tilde{V}(\boldsymbol{x};t)G_+ - \nabla^T(\tilde{\boldsymbol{A}}G_+) = \frac{1}{2}\Delta G_+, \quad G_+(0;\boldsymbol{x}) = \delta(\boldsymbol{x}-\boldsymbol{y})$

Harmonic Path Integral Diffusion (H-PID) – Integrability

H-PID – Mid Level Integrability

- **Green Functions are Gaussian** when
  - Potential is Quadratic: $V(t;\boldsymbol{x}(t)) = \boldsymbol{x}^T\hat{\beta}(t)\boldsymbol{x}/2 +$ linear and const terms
  - Force and Gauge are Affine in $\boldsymbol{x}$
- Akin to Quant Mech (in imag. time) in Harmonic Potential
- $u^*(t;\boldsymbol{x}(t)) = \nabla_{\boldsymbol{x}} \log\left(\int d\boldsymbol{y}\ p_{\text{target}}(\boldsymbol{y})\frac{G_-(t;\boldsymbol{x}(t);\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)}\right) =$ $a(t)\boldsymbol{x}(t) - b(t)\hat{\boldsymbol{x}}(t;\boldsymbol{x}(t))$

Special case, also discussed in [1] – Low Level Integrability

- $\boldsymbol{A}, \boldsymbol{f} = 0$ – zero gauge, zero force

[1] A. Teter, W, Wang & A. Halder, *Schrödinger bridge with quadratic state cost is exactly solvable*, 2024

Use Case – Harmonic, Uniform – Low Level Integrability

$V(t;\boldsymbol{x})=\beta|\boldsymbol{x}|^2 \Rightarrow$ Explicit Expression for the Green Functions

- $u^*(t;\boldsymbol{x}(t))=\nabla_{\boldsymbol{x}}\log\left(\int d\boldsymbol{y}\; p_{\text{target}}(\boldsymbol{y})\frac{G_-(t;\boldsymbol{x}(t);\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)}\right)=a(t)\boldsymbol{x}(t)-b(t)\hat{\boldsymbol{x}}(t;\boldsymbol{x}(t))$
- $u^*(t;\boldsymbol{x})=\frac{\sqrt{\beta}}{\sinh((1-t)\sqrt{\beta})}\left(\hat{\boldsymbol{x}}(t;\boldsymbol{x})-\boldsymbol{x}\cosh((1-t)\sqrt{\beta})\right)$
- **Weighted State:** $\hat{\boldsymbol{x}}(t;\boldsymbol{x})\doteq\int d\boldsymbol{y}\boldsymbol{y}w(\boldsymbol{y}|t;\boldsymbol{x})=\mathbb{E}_{\boldsymbol{y}\sim w(\cdot|t;\boldsymbol{x})}[\boldsymbol{y}]$
- **Weight (probability):** $w(\boldsymbol{y}|t;\boldsymbol{x})\propto p_{\text{target}}(\boldsymbol{y})\frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)}$

$$\frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(\boldsymbol{1};\boldsymbol{y};\boldsymbol{0})}=\sqrt{\frac{\sinh(\sqrt{\beta})}{\sinh((\boldsymbol{1}-t)\sqrt{\beta})}}\exp\left(-\frac{\sqrt{\beta}}{\boldsymbol{2}}\left((\boldsymbol{x}^{\boldsymbol{2}}+\boldsymbol{y}^{\boldsymbol{2}})\coth\left((\boldsymbol{1}-t)\sqrt{\beta}\right)-\boldsymbol{y}^{\boldsymbol{2}}\coth\left(\sqrt{\beta}\right)-\frac{\boldsymbol{2}(\boldsymbol{x}^T\boldsymbol{y})}{\sinh((\boldsymbol{1}-t)\sqrt{\beta})}\right)\right)$$

Next – Experiments & Analysis

- Dependence on β – **strength of the potential**
- What is the meaning/significance of the **Weighted State**

How to estimate the optimal control?

$p_{\text{target}}(\boldsymbol{x}) \propto \exp(-E(\boldsymbol{x}))$ – the Energy function is known explicitly

- $u^*(t;\boldsymbol{x}) = \nabla_{\boldsymbol{x}} \log \left(\int d\boldsymbol{y}\ \exp(-E(\boldsymbol{y})) \frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)} \right)$
- How to Estimate the Integral?

Importance Sampling !!

$$\int d\boldsymbol{y}\ \exp(-E(\boldsymbol{y})) \frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(\boldsymbol{1};\boldsymbol{y};\boldsymbol{0})} = \mathbb{E}_{\boldsymbol{y}\sim\mathcal{N}\left(\cdot;\boldsymbol{y}^*;\hat{\boldsymbol{H}}^{-\boldsymbol{1}}\right)} \left[\frac{\exp(-E(\boldsymbol{y}))}{\mathcal{N}\left(\boldsymbol{y};\boldsymbol{y}^*;\hat{\boldsymbol{H}}^{-\boldsymbol{1}}\right)} \frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(\boldsymbol{1};\boldsymbol{y};\boldsymbol{0})} \right]$$

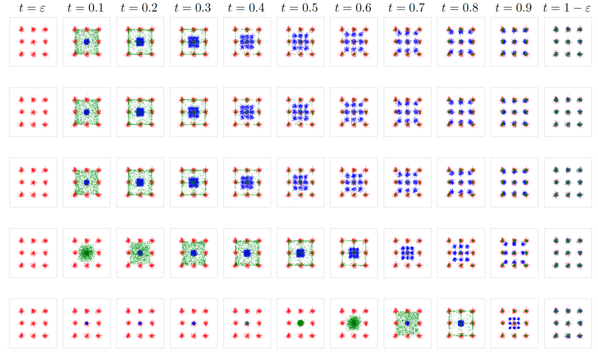
$$\nabla_{\boldsymbol{y}} \log \left(\frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(\boldsymbol{1};\boldsymbol{y};\boldsymbol{0})} \right) = \sqrt{\beta} \left(\boldsymbol{y} \left(\coth(\sqrt{\beta}) - \coth\left((\boldsymbol{1}-t)\sqrt{\beta}\right)\right) + \boldsymbol{x} \frac{\boldsymbol{1}}{\sinh((\boldsymbol{1}-t)\sqrt{\beta})} \right)$$

$$H_{ij} = -\partial_{y_i}\partial_{y_j} \log \left(\frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(\boldsymbol{1};\boldsymbol{y};\boldsymbol{0})} \right) \Bigg|_{\boldsymbol{y}\to\boldsymbol{y}_*} = \delta_{ij}\sqrt{\beta} \left(\coth\left((\boldsymbol{1}-t)\sqrt{\beta}\right) - \coth(\sqrt{\beta})\right)$$

$$\boldsymbol{y}_* = \frac{\boldsymbol{x}}{\cosh((\boldsymbol{1}-t)\sqrt{\beta}) - \sinh((\boldsymbol{1}-t)\sqrt{\beta})\coth(\sqrt{\beta})}$$

- Rely on stationary-point approximation
- Exact asymptotically at $t \to 1$
- Importance Samples are Universal – do not depend on $E(\boldsymbol{x})$

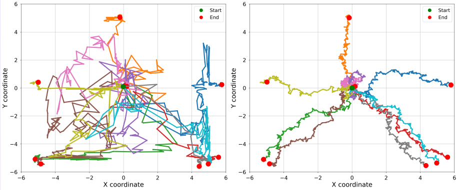
Universal Harmonic Importance Sampling (UHIC)

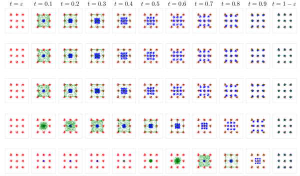


- Gaussian Mixture over (3×3) grid
- Difficult for PIS Alg.
- $s = 1, \cdots, 1000$ - samples of UHIC:
  Red - exact
  Blue - $\boldsymbol{x}^{(s)}(t)$
  Green $\hat{\boldsymbol{x}}^{(s)}(t; \boldsymbol{x}(t))$
- Rows:
  $\beta = 0, 0.1, 1, 10, 100$

Energy Function Sampling

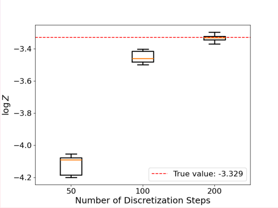
- **Speciation Transition**: 9 Gaussians = 9 species
- Seen earlier in $\hat{\boldsymbol{x}}^{(s)}(t; \boldsymbol{x}(t))$ – **the order parameter**
- Transition time depends on β. Fastest at $\beta \approx 0.1$

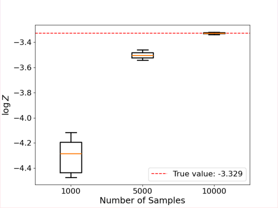




Space-Time Evolution of Samples [1]

- Dynamics is "direct" in $\boldsymbol{x}(t)$
- Much more of "exploration" meandering in $\hat{\boldsymbol{x}}(t; \boldsymbol{x}(t))$





- Good **Convergence** – in # t-steps & # of samples

[1] HB, MC, Space-Time Bridge Diffusion, 2024

Sampling from Ground Truth Samples

$$u^*(t;\boldsymbol{x}) = \nabla_{\boldsymbol{x}} \log \left(\int d\boldsymbol{y}\, p_{\text{target}}(\boldsymbol{y}) \frac{G_-(t;\boldsymbol{x};\boldsymbol{y})}{G_+(1;\boldsymbol{y};0)} \right)$$
$$\approx \nabla_{\boldsymbol{x}} \log \left(\frac{1}{S} \sum_{s=1}^{S} \frac{G_-(t;\boldsymbol{x};\boldsymbol{y}^{(s)})}{G_+(1;\boldsymbol{y}^{(s)};0)} \right)$$
$$\approx \frac{\sqrt{\beta}\left(\hat{\boldsymbol{x}}(t;\boldsymbol{x}) - \boldsymbol{x}\cosh((1-t)\sqrt{\beta})\right)}{\sinh((1-t)\sqrt{\beta})}$$

- Memorization Regime ⇒ Ground Truth samples
- Focus on **Analysis** of **Memorization transition** [1,2]

$$\hat{\boldsymbol{x}}(t;\boldsymbol{x}(t)) \doteq \sum_s \boldsymbol{y}^{(s)} w(\boldsymbol{y}^{(s)}|t;\boldsymbol{x}(t))$$
$$w(\boldsymbol{y}|t;\boldsymbol{x}) \propto \exp\left(-\frac{\sqrt{\beta}}{2}\left((\boldsymbol{x}^2+\boldsymbol{y}^2)\coth\left((1-t)\sqrt{\beta}\right) - \boldsymbol{y}^2\coth\left(\sqrt{\beta}\right) - \frac{2(\boldsymbol{x}^T\boldsymbol{y})}{\sinh\left((1-t)\sqrt{\beta}\right)}\right)\right)$$

[1] HB, MC, U-turn Diffusion, 2023
[2] G. Biroli, et al, Dynamical Regimes of Diffusion Models, 2024

Sampling from CIFAR-10

$\beta = 0.1$

$\hat{\boldsymbol{x}}(t;\boldsymbol{x}(t)) \doteq \sum_s \boldsymbol{y}^{(s)} w(\boldsymbol{y}^{(s)}|t;\boldsymbol{x}(t))$

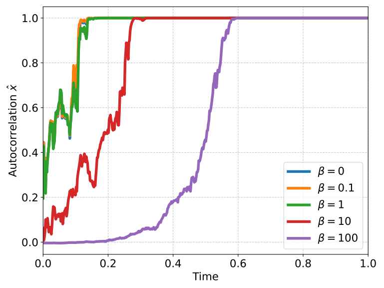
$\boldsymbol{x}(t)$

Analysis of the Memorization Transition

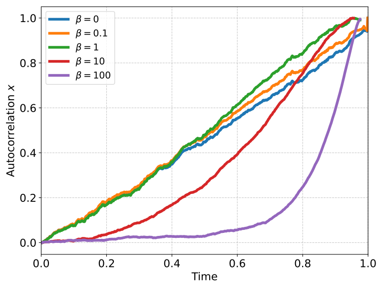
- Emergence of Two Phases
- See it earlier in the **Weighted State = Order Parameter**

Sampling from CIFAR-10

Auto-Correlations in Dynamics of a Single Sample



$(\hat{\boldsymbol{x}}^T(t; \boldsymbol{x}(t))\boldsymbol{x}(1))$



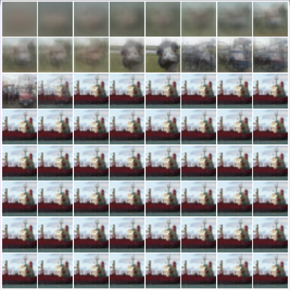
$(\boldsymbol{x}^T(t)\boldsymbol{x}(1))$

Analysis of the Memorization Transition

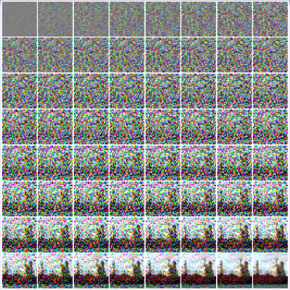
- Emergence of Two Phases
- See it earlier in the **Weighted State = Order Parameter**
- Transition time increases with β

Sampling from CIFAR-10

$\beta = 10$



$\hat{\boldsymbol{x}}(t; \boldsymbol{x}(t)) \doteq \sum_s \boldsymbol{y}^{(s)} w(\boldsymbol{y}^{(s)} | t; \boldsymbol{x}(t))$



$\boldsymbol{x}(t)$

Analysis of the Memorization Transition

- Emergence of Two Phases
- See it earlier in the **Weighted State = Order Parameter**
- Transition time increases with β
- Much more of "exploration" meandering in $\hat{\boldsymbol{x}}(t; \boldsymbol{x}(t))$

Summary – Harmonic Path Integral Diffusion (H-PID) framework

- Expressive **Stochastic Optimal Control** for Bridge Diffusion
- **"Integrable"** – Three Levels
  - Top **Potential** + **Force** + **Gauge** ⇒ Linearly Solvable = log-ratio-of backward & forward Green Functions
  - Mid Potential - quadratic, Force + Gauge are affine ⇒ **Green Functions are Gaussian** = akin **Quantum Harmonic Oscillators**
  - Low Uniform Quadratic Potential ⇒ control is a **convolution** of the **Target Distribution** with a kernel expressed via **elementary functions**
- H-PID Algorithms is **Neural Networks – FREE**, works better on **CPUs**
- Experiments on Gaussian mixtures and CIFAR-10: **Weighted State** is **Order Parameter** of a **Dynamic Phase Transition** - early pre-cursor of the resulting sample

Outline

From Pre-Trained Generative Models to Applications

Real Power of Generative Models

- Once trained (expensive), they can be adapted to various applications without retraining

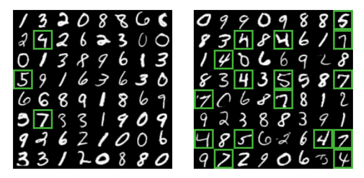
Key Applications of Pre-Trained Diffusion Models

- Constrained Inference
- Self-Labeling
- Inference of Structured Data

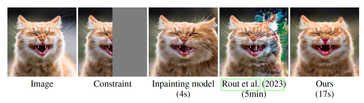
Constrained Inference

- **Constrained Inference** involves generating samples under specific constraints, such as fixing part of the state or ensuring the sample satisfies known conditions.

- **Constrained Diffusion Models via Dual Training** Introduces dual training for diffusion models to handle tasks like fair sampling and conditional generation. [arXiv:2408.15094]



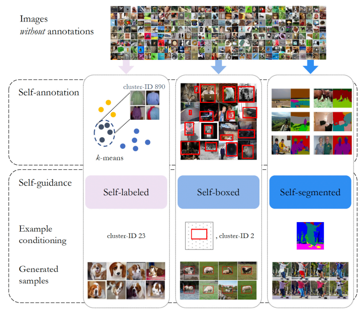
- **Fast Constrained Sampling in Pre-Trained Diffusion Models** Proposes efficient constrained sampling algorithms for pre-trained models without requiring fine-tuning. [arXiv:2410.18804]



Self-Labeling

- Use pre-trained diffusion models to uncover inherent **clusters** and **assign labels** as a downstream application.
- This involves generating samples conditioned on implicit structures in the data and discovering clusters through the learned representations.

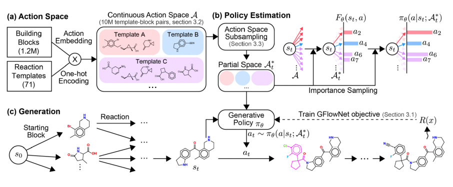
- **Diffusion Models for Clustering and Label Discovery** This paper demonstrates how pre-trained diffusion models can be leveraged for unsupervised clustering and label discovery. By conditioning the diffusion process on inferred data clusters, it enables the generation of samples aligned with these clusters, facilitating downstream tasks such as labeling. [arXiv:2210.06462]



Inference of Structured Data

- Inference of Structured Data involves working with discrete structures like graphs, time sequences, and similar datasets.
- Diffusion models combined with stochastic optimal control are particularly effective for these tasks.
- Operates in discrete space (e.g., molecular building blocks) and discrete time (e.g., steps in molecular design).
- Incorporates auto-regression/memory/transformers into diffusion

- **Graph Flow Network ...**, Introduces a framework for generative modeling in discrete spaces, focusing on graph-based structures like molecules. Combines diffusion models with stochastic optimal control to address complex design problems in discrete space and time [arXiv:2111.09266 – Bengio group] → RXNFLOW [arXiv:2410.04542 – KAIST team] (drug design)



Why Generative Models for Power (and other) Systems?

Uncertainty ⇒ Statistics ⇒ Generative Models

- Renewables
- Consumers
- Operators
- Dependencies
- . . .

Start Broad and Build ONE Generative Model

- Solve Many Downstream Problems
- Update the Generative Model as we go

Controls & Optimizations

- ... can be down-stream tasks too

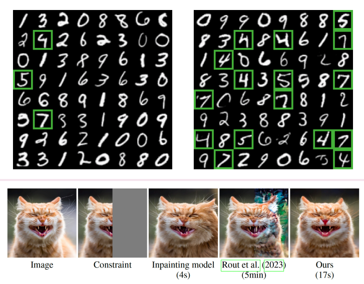
What is the State Space?

- What is the State Space of the most general (Power System) Generative Model?

- Grid Layout/Type, Weather, Stress, Season, etc — features/labels, may be part of the state
- State: Instantaneous (or time evolving) configuration of load and generation, generation status.

Possible Application(s) in Power Systems

Constrained Inference – Re-Biasing, Completing, Cleaning

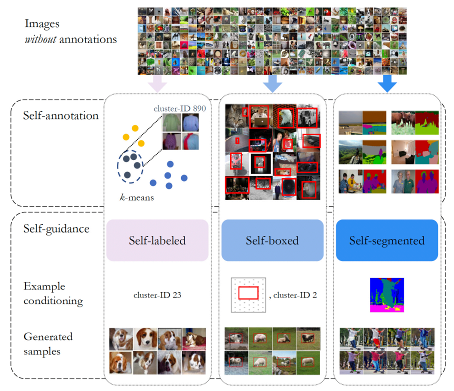


Discovery of Rare Events

- Sampling "Typical" (once a day) Contingencies, possibly conditioned to type of date, weather event, etc
- Sampling Dynamic $N-1$ events, e.g. single-phase faults
- Cyber-security of power system – correct/identify/detect "small" but "dangerous" intrusions/modifications (DJ lecture on privacy of AI – extended from black box to physical box)

Possible Application(s) in Power Systems

Self-Labeling

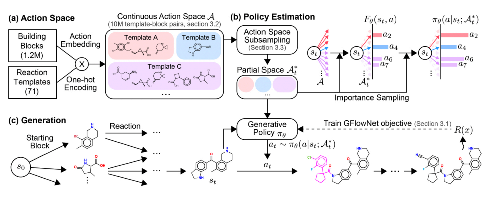


Power System Forensics

- Discovering/labeling hidden regimes, e.g. associated with actions of other participants of the energy market(s)

Possible Application(s) in Power Systems

Inference of Structured Data (Graph Flow Network)



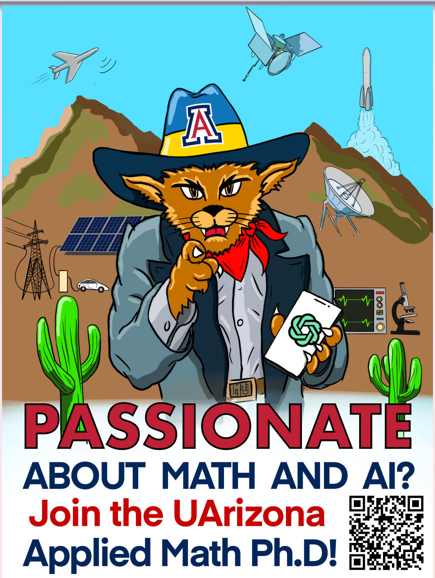
Stochastic Unit Commitment

- Task: Unit Commitment Decisions = Sampled from a generative model, where probability of UC = its reward (stochastic average over multiple load/renewable forecasts)
- Built Sequentially with Auto-Regression (switching on/off memory)
- Evaluation of a UC reward is with a state-of-the art power-system solver

Accelerate Bender Decomposition (in response to Pascal's tutorial)

- May be ...

Applied Math @ UArizona



- **Research** focused, since 1976, one of the US first in Applied Mathematics
- **Interdisciplinary**: 130+ professors/ 27 departments / 8 colleges across UA campus (Science & Engineering & Optics – top 3)
- **Mixing** traditional contemporary AM
- **65** PhD students (12/12/12/13/16/10 enrolled in 2024/23/22/21/20/19)
- 3 Core Courses to Qualify (Methods, Analysis, Algorithms) - **re-designed** to incorporate **AI** in 2019-20; + three **research rotations** in the first 3 semesters with at least two professors
- Strong collaborations with National – **DOE** – & Industrial – **DOD+** – Labs, e.g. via **NSF** (Graduate Innovation in Education) support – **pipeline**: recruitment, internerships, co-advising (triads), partial employment

Is (Bayesian) Generative = Foundational ?

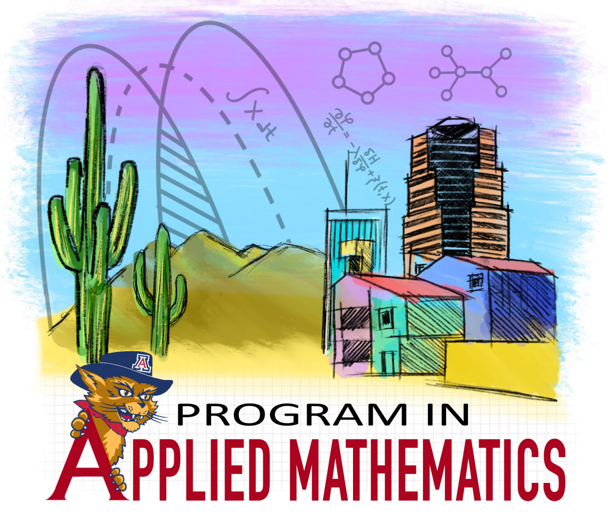
Is (Bayesian) Generative = Foundational ?

- It is UMBRELLA ⇒ for many (if not all) downstream applications

Is (Bayesian) Generative = Foundational ?

- It is UMBRELLA ⇒ for many (if not all) downstream applications

- What kind of foundational/generative AI models (for power systems and beyond) shall we ... government/DOE and industry ... train = invest in?



Thank You!