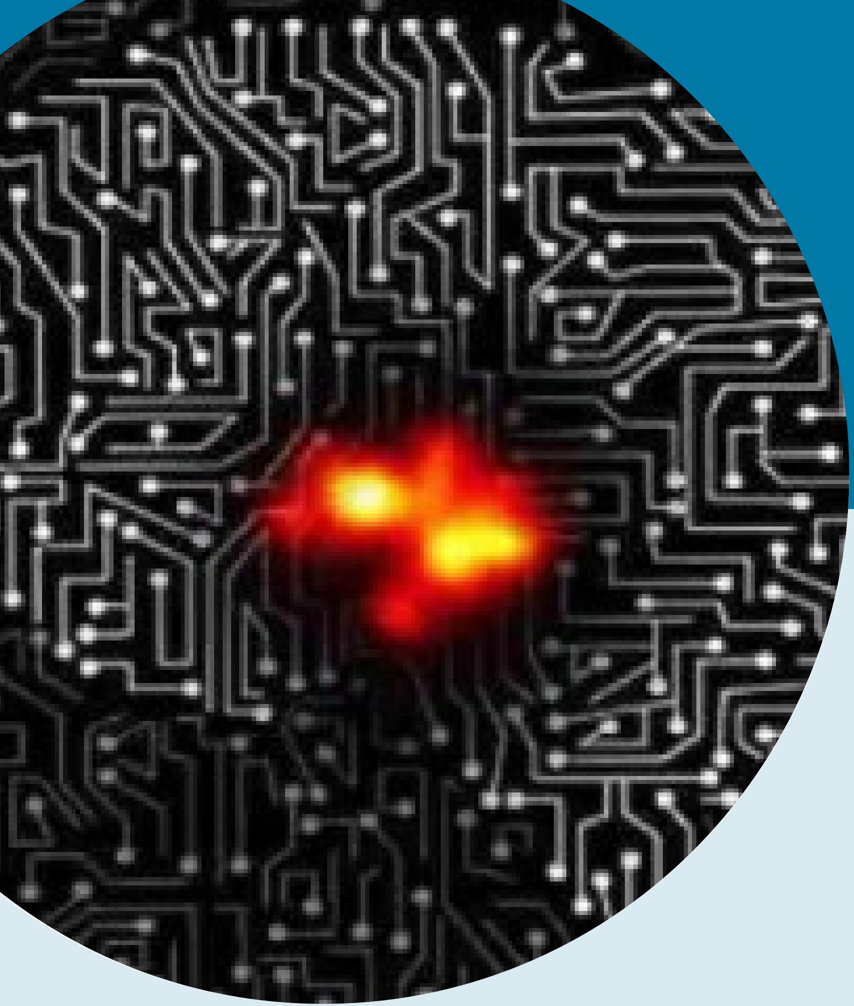


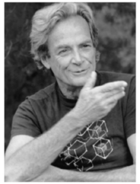
香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

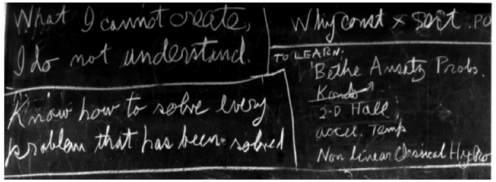
Generative AI meets Physics

Kai Zhou (CUHK-Shenzhen)

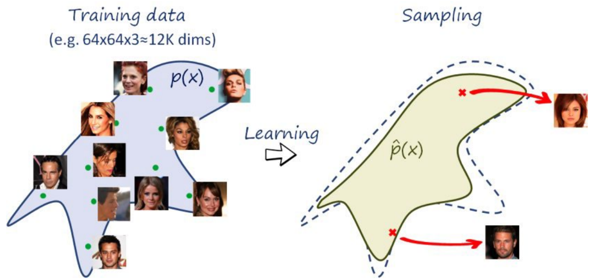
12.Oct , Lattice QCD 2025 , Huizhou

Generative Models





"What I can not create, I do not understand"



Want to **model** the observed data's underlying but unknown **distribution**, to further :

- Understand/Inference the data (inherent structure, properties, features...)
- Sample according to the distribution

Suppose observation dataset :

$$\mathbf{X} = \{x^{(1)}, x^{(2)}, ..., x^{(N)}\} \overset{i.i.d}{\sim} p_{data}(x)$$

We use parametric model to approach the data distribution :

$$p_\theta(x) \to p_{data}(x)$$

Often use NN to parametrize transformation $\log p_\theta(\mathbf{x}) = \log p_\theta(f_\theta(\mathbf{z}_0))$

- Maximize Likelihood Estimation : **(given training samples)**

$$\theta^* = \arg\max_\theta \log p_\theta(\mathbf{X}) = \arg\max_\theta \frac{1}{N}\sum_{i=1}^{N} \log p_\theta(x^{(i)})$$

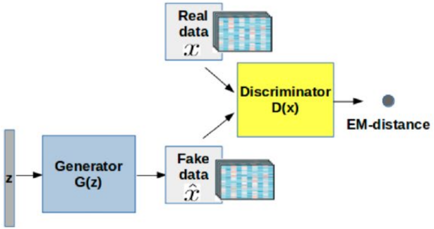
Reverse KL Divergence : Sample many $\mathbf{z}_0 \sim p_0(\mathbf{z}_0)$

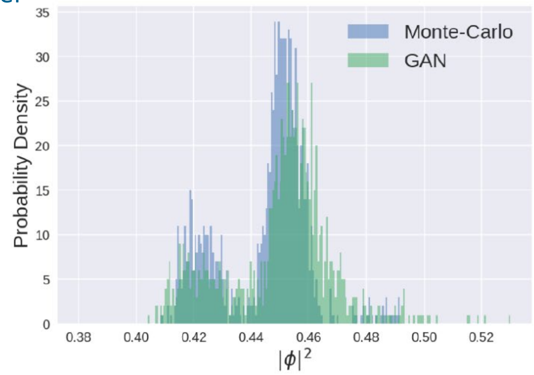
(given unnormalized target distribution, e.g., Action)

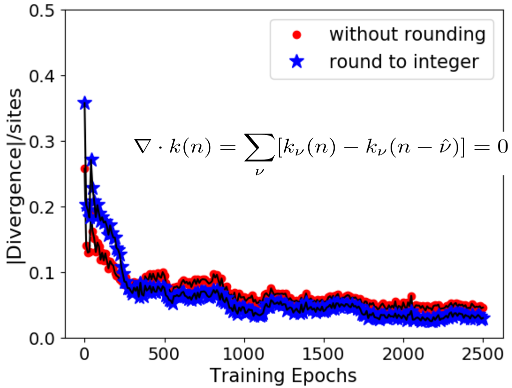
$$\theta^* = \arg\min_\theta \frac{1}{N}\sum_{i=1}^{N} [\log p_\theta(f_\theta(\mathbf{z}_0)) - \log \tilde{p}_{target}(f_\theta(\mathbf{z}_0))]$$

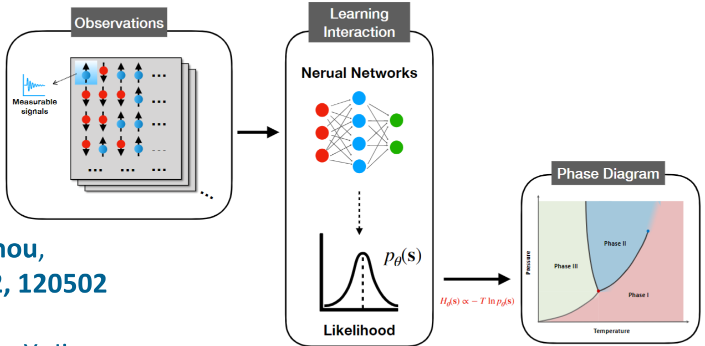
Given an ensemble of data from the target distribution

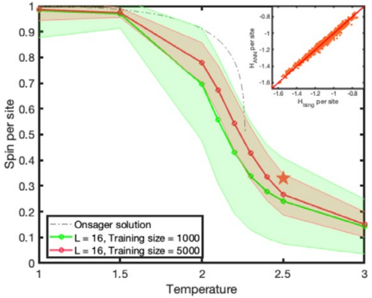
K. Zhou, G. Endrődi, L.-G. Pang, and H. Stöcker
PRD 100, 011501 (2019)

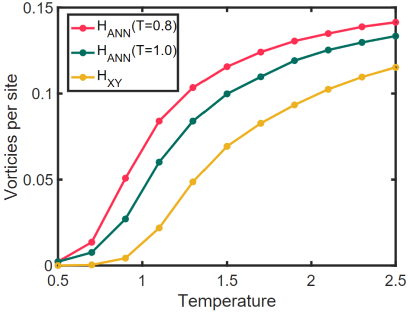












L. Wang, L. He, Y. Jiang, **K. Zhou**,
Chin.Phys.Lett. 39 (2022) 12, 120502

T. Xu, L. Wang, L. He, **K. Zhou**, Y. Jiang,
Chin.Phys.C 48 (2024) 10, 103101

With unnormalized probability distribution – Hamiltonian/Action known

$$p(\mathbf{s}) = \frac{e^{-\beta E(\mathbf{s})}}{Z}$$

- Reverse KL divergence

$$D_{\mathrm{KL}}(q_\theta \,\|\, p) = \sum_{\mathbf{s}} q_\theta(\mathbf{s}) \ln\left(\frac{q_\theta(\mathbf{s})}{p(\mathbf{s})}\right) = \beta(F_q - F)$$

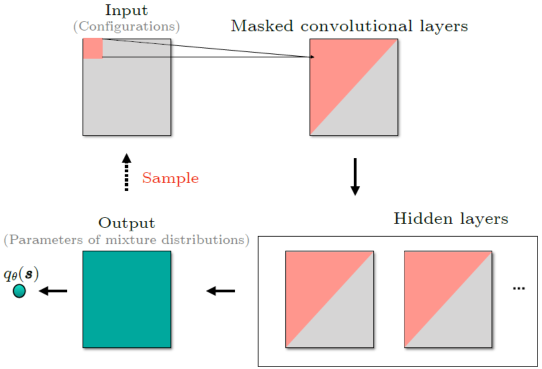
$$F_q = \frac{1}{\beta}\sum_{\mathbf{s}} q_\theta(\mathbf{s})\left[\beta E(\mathbf{s}) + \ln q_\theta(\mathbf{s})\right]$$

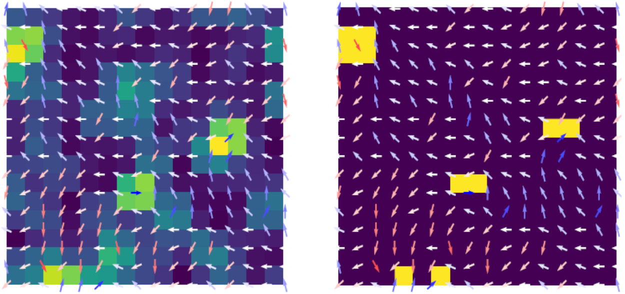
- Autoregressive Net $q_\theta(\mathbf{s}) = \prod_{i=1}^{N} q_\theta(s_i \mid s_1, \ldots, s_{i-1})$

D. Wu, Lei Wang and P. Zhang, PRL122,080602(2019)

- **Continuous Autoregressive Net** for XY model

L. Wang, Y. Jiang, L. He, K. Zhou, CPL 39, 120502 (2022)





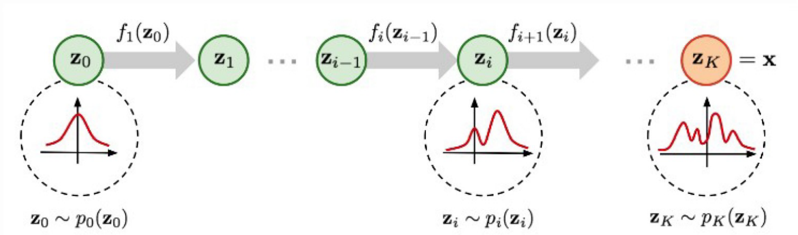
Probability distributions from CANs

Vortices

Flow based generative model given unnormalized distribution

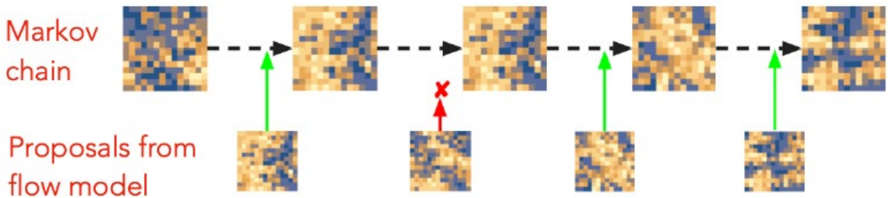
A series **(Flow)** of invertible/bijective transformations for p(z)

<u>compose several invertible transformations to form the flow :</u>



$$p_i(\mathbf{z}_i) = p_{i-1}(f_i^{-1}(\mathbf{z}_i))|\det J_{f_i^{-1}}| = p_{i-1}(\mathbf{z}_{i-1})|\det J_{f_i}|^{-1}$$

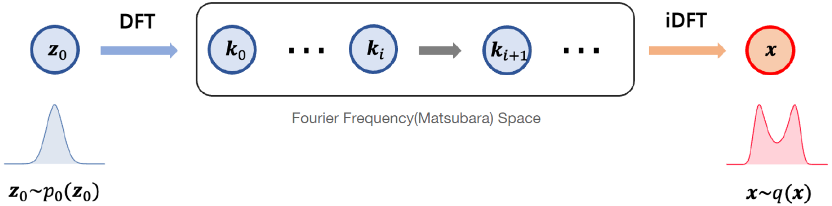
$$\rightarrow \log p(\mathbf{x}) = \log p_0(f^{-1}(\mathbf{x})) + \sum^{K} \log|\det J_{f_i^{-1}}| = \log p_0(\mathbf{z}_0) - \sum^{K} \log|\det J_{f_i}|$$

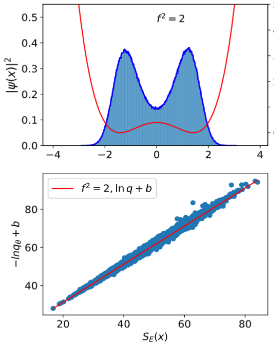


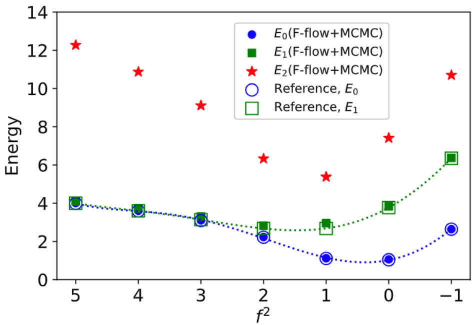
Albergo +, 1904.12072; Boyda +, 2008.05456; Favoni +, 2012.12901; Abbott +, 2208.03832; Abbott +, 2211.07541; Abbott +, 2305.02402; Bulgarelli+ 2412.00200 (SU(3)); Abbott +, arXiv:2502.00263

K.C, G. K., S. R., D. R., P. S., **Nature Reviews Physics** 5, 526-535 (2023)

Fourier Flow Model





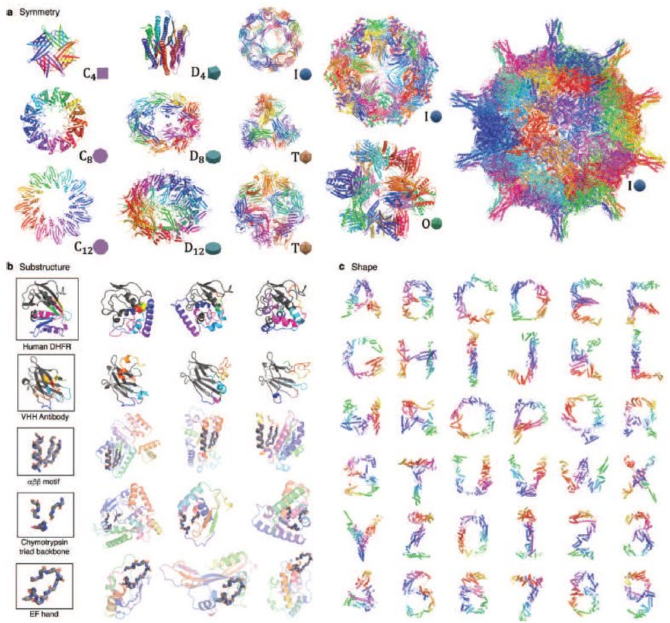


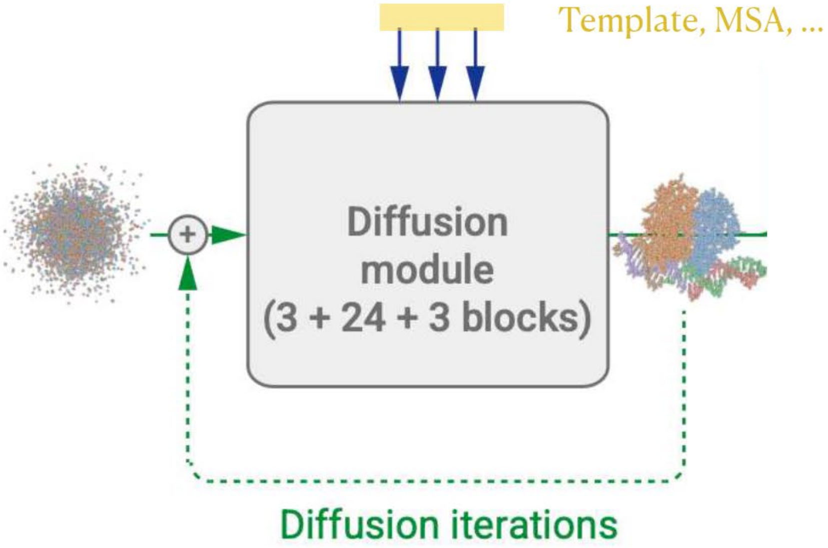
S.C, O. S, S. Z, B. C, H. S, L. W, **K. Zhou**, **PRD107, 056001(2023)**

Diffusion Model

"A heavy quark move inside quark-gluon plasma"

Diffusion Model for protein structure prediction and design





Ingraham et al, Chroma, Nature 2023
https://generatebiomedicines.com/chroma

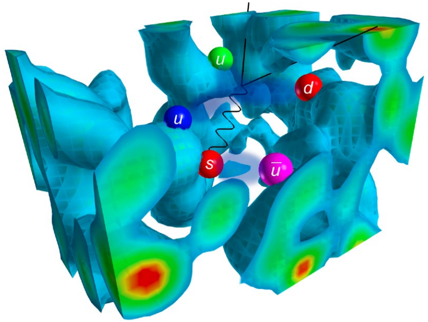
Abramson et al, AlphaFold3, Nature 2024
https://deepmind.google/technologies/alphafold/

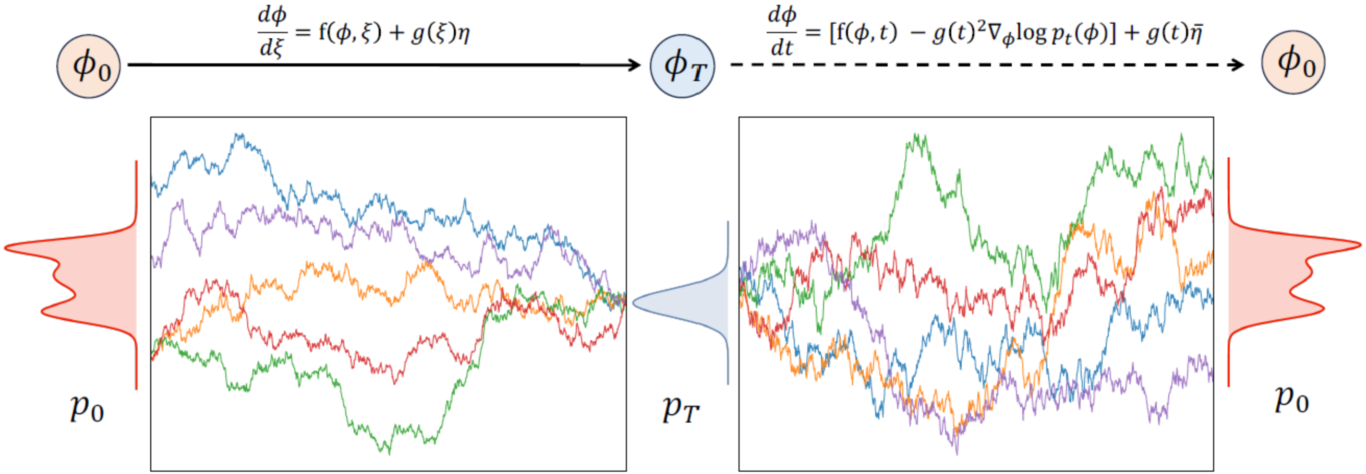
Diffusion Model on lattice QFT configurations

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

$$p(\phi) = e^{-S(\phi)}/Z$$

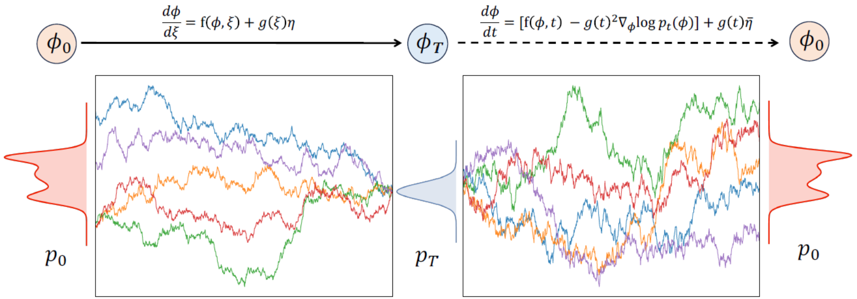
$$\langle O \rangle \approx \frac{1}{N} \sum_i O_i$$





L. Wang, G. Aarts, **K. Zhou**, **JHEP 05 (2024) 060**
L. Wang, G. Aarts, **K. Zhou**, **arXiv:2311.03578** (NeurIPS 2023 workshop “ML&Physical Sciences”)
G. A, D. E. H, L. W, **K. Zhou**, **arXiv:2410:21212** (NeurIPS 2024 workshop “ML&Physical Sciences) → **“Best Physics for AI Paper” Award**
Q. Zhu, G. Aarts, W. Wang, **K. Zhou**, L. Wang, **arXiv:2410.19602** (NeurIPS 2024 workshop “ML&Physical Sciences)
G.Aarts, D.E.H, L.W, **K. Zhou**, **arXiv:2510.01328**

Diffusion Model for field configurations



- Forward diffusion SDE

$$\frac{d\phi}{d\xi} = f(\phi,\xi) + g(\xi)\eta(\xi) \qquad \langle \eta(\xi)\eta(\xi')\rangle = 2\alpha\delta(\xi-\xi')$$

- Backward diffusion SDE

$$\frac{d\phi}{dt} = [f(\phi,t) - g^2(t)\nabla_\phi \log p_t(\phi)] + g(t)\bar{\eta}(t) \qquad \underline{t \equiv T - \xi}$$

- Score matching Training $\mathcal{L}_\theta = \sum_{i=1}^{N} \sigma_i^2 \mathbb{E}_{p_0(\phi_0)} \mathbb{E}_{p_i(\phi_i|\phi_0)} \left[\| s_\theta(\phi_i,\xi) - \nabla_{\phi_i} \log p_i(\phi_i|\phi_0) \|_2^2 \right]$

- Sample generation SDE in variance exploding scheme : $\frac{d\phi}{d\tau} = g_\tau^2 \nabla_\phi \log q_\tau(\phi) + g_\tau \bar{\eta}(\tau) \qquad \tau \equiv T - t$

$$\frac{\partial p_\tau(\phi)}{\partial \tau} = \int d^n x \left\{ g_\tau^2 \frac{\delta}{\delta\phi} \left(\bar{\alpha} \frac{\delta}{\delta\phi} + \nabla_\phi S_{\mathrm{DM}} \right) \right\} p_\tau(\phi).$$

$$\boxed{p_{\mathrm{eq}}(\phi) \propto e^{-S_{\mathrm{DM}}/\bar{\alpha}}}$$

$$\nabla_\phi S_{\mathrm{DM}} \equiv -\nabla_\phi \log q_\tau(\phi)$$

- A flow of **<u>effective action</u>** will be learned in DMs

sampling from a DM is equivalent to optimizing a stochastic trajectory to approach the "equilibrium state"

L. Wang, G. Arts, **K. Zhou, JHEP 05(2024) 060**

$$\boxed{\frac{\partial \phi(x,\tau)}{\partial \tau} = -\frac{\delta S_E[\phi]}{\delta \phi(x,\tau)} + \eta(x,\tau)}$$

$$\langle \eta(x,\tau) \rangle = 0, \quad \langle \eta(x,\tau)\eta(x',\tau') \rangle = 2\alpha\delta(x-x')\delta(\tau-\tau')$$

τ: fictitious time, α: diffusion constant

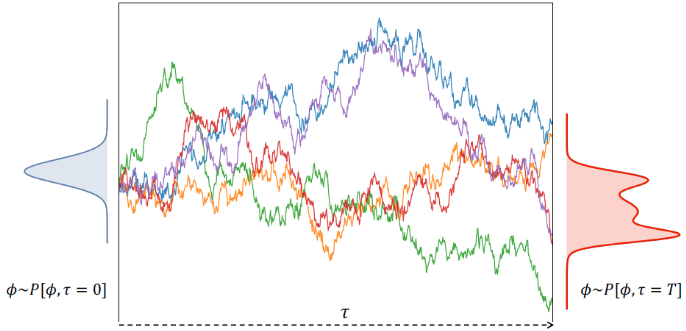
- **Fokker-Planck equation**

$$\frac{\partial P[\phi,\tau]}{\partial \tau} = \alpha \int d^n x \left\{ \frac{\delta}{\delta \phi} \left(\frac{\delta}{\delta \phi} + \frac{\delta S_E}{\delta \phi} \right) \right\} P[\phi,\tau]$$

Equilibrium solution (long-time limit),

$$P_{\text{eq}}[\phi] \propto e^{-\frac{1}{\alpha} S_E[\phi]}$$

One can construct stochastic process to reproduce the quantum path integral with its equilibrium:



Thermal equilibrium limit → Quantum distribution

- Set the diffusion constant as $\alpha = \hbar$

$$\boxed{P_{\text{eq}}[\phi] \sim e^{-\frac{1}{\hbar} S_E[\phi]} = P_{\text{quantum}}[\phi]}$$

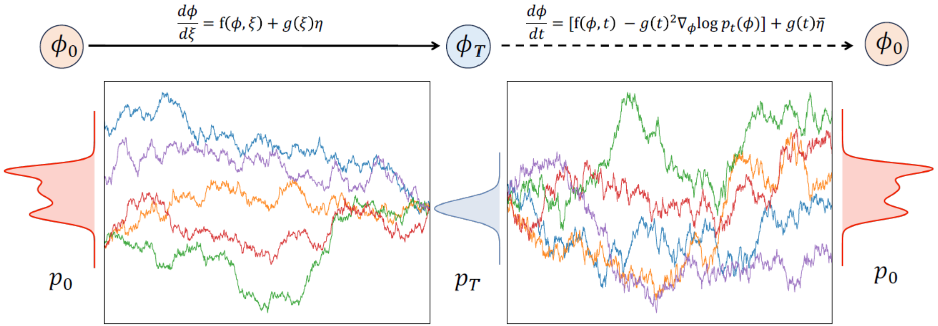
Parisi G. and Wu Y. S., **Sci. China, A 24**, ASITP-80-004 (1980).

Diffusion Model as SQ

- DM generation SDE and Stochastic Quantization :

$$\frac{\partial\phi(x,\tau)}{\partial\tau} = g^2(\tau)\nabla_\phi \log P(\phi;\tau) + g(\tau)\eta(x,\tau)$$

$$\frac{\partial\phi(x,\tau)}{\partial\tau} = -\nabla_\phi S(\phi) + \sqrt{2}\eta(x,\tau)$$



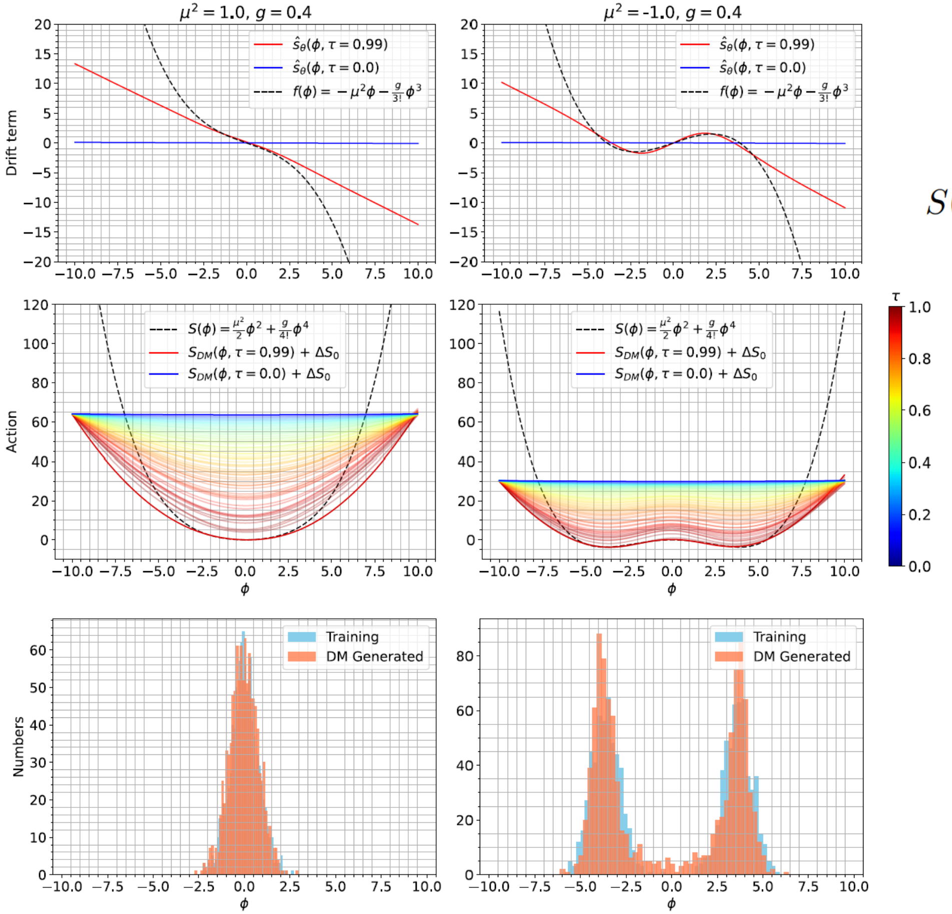
- Similarities and differences:

✓ SQ: fixed drift, determined from known action
constant noise variance (but can be generalised using kernels)
thermalisation followed by long-term evolution in equilibrium

✓ DM: drift and noise variance time-dependent, learn from data
evolution between $0 \le \tau \le T = 1$, many short runs

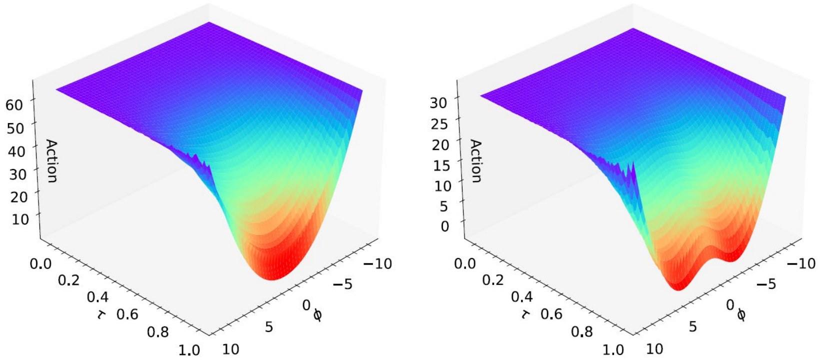
L. Wang, G. Arts, **K. Zhou, JHEP 05(2024) 060**

Effective Action on A Toy model



- Flow of the effective action

$$S(\phi) = \frac{\mu^2}{2}\phi^2 + \frac{g}{4!}\phi^4, \qquad f(\phi) = -\frac{\partial S(\phi)}{\partial \phi} = -\mu^2\phi - \frac{g}{3!}\phi^3$$



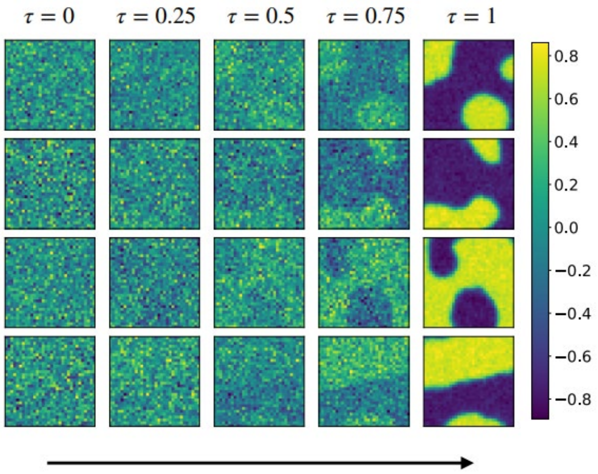
$$S_{\mathrm{DM}}(\phi, \tau) = \int^{\phi} \hat{s}_\theta(\tilde{\phi}, \tau) d\tilde{\phi}$$

DM on 2d scalar ϕ^4 model

- 32x32 lattice, HMC generated 5120 configurations as training set

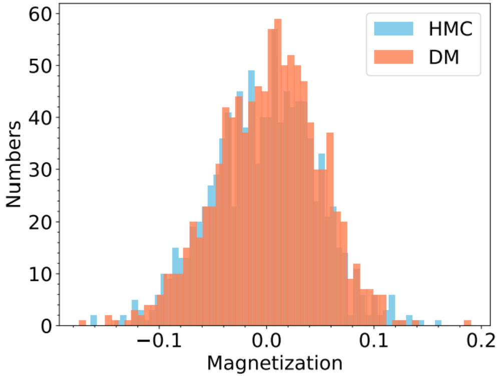
$$S_E = \sum_x [-2\kappa \sum_{\mu=1}^{d} \phi(x)\phi(x+\hat{\mu}) + (1-2\lambda)\phi(x)^2 + \lambda\phi(x)^4].$$

Broken phase :



numerous "bulk" patterns emerge

symmetric phase :



data-set	$\langle M \rangle$	χ_2	U_L
Training (HMC)	0.0012± 0.0007	2.5160 ± 0.0457	0.1042 ± 0.0367
Testing (HMC)	0.0018 ± 0.0015	2.4463 ± 0.1099	-0.0198 ± 0.1035
Generated (DM)	0.0017± 0.0015	2.4227 ± 0.1035	0.0484 ± 0.0959

Relation to (inverse) RG

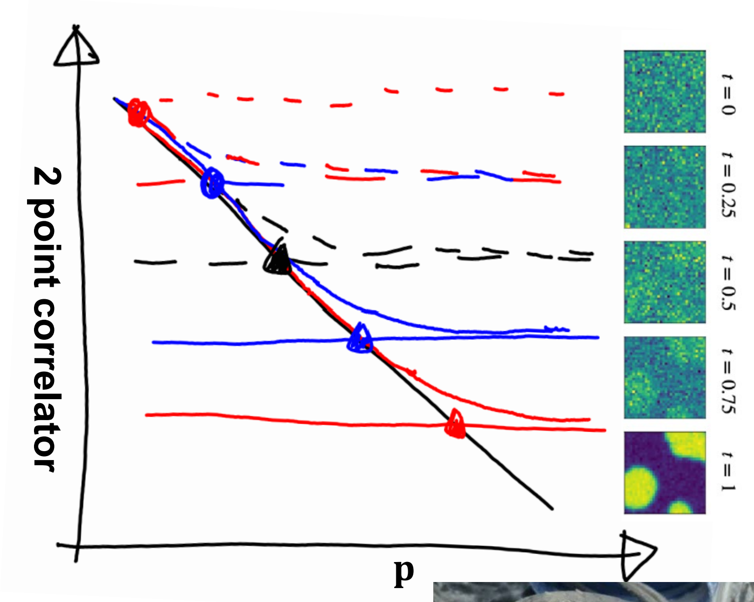
- Forward diffusion kernel: **gaussian smoothing**

$$p_\xi(\phi_\xi|\phi_0) = \mathcal{N}\left(\phi_\xi; \phi_0, \frac{1}{2\log\sigma}(\sigma^{2\xi}-1)\mathbf{I}\right)$$

$$\phi_\tau(\mathbf{x}) \;=\; \phi_0(\mathbf{x}) + \sqrt{\frac{\sigma^{2\tau}-1}{2\log\sigma}}\epsilon(\mathbf{x}) \;\text{ with }\; \epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{I})$$

- In **Fourier space**:

$$\phi_\tau(p) = \phi_0(p) + \sqrt{\frac{\sigma^{2\tau}-1}{2\log\sigma}}\epsilon(p)$$



- ! the above evolution will perturb (smear) higher momentum modes first,
- With decreasing cut scale because of the gradually increasing noise level !

In FRG, the high frequency (short-distance) degrees of freedom is progressively integrated out !

How correlations evolve (destroyed and rebuilt) in DM ?

- forward process: $\dot{x}(t) = K(x(t), t) + g(t)\eta(t)$ $\quad 0 \leq t \leq T$

noise profile $g(t) = \sigma^{t/T}$

- backward process:

$$x'(\tau) = -K(x(\tau), T-\tau) + g^2(T-\tau)\partial_x \log P(x, T-\tau) + g(T-\tau)\eta(\tau)$$

score

$\tau = T - t$

two main schemes:

- variance-expanding (VE): no drift $K(x,t) = 0$
- variance-preserving (VP) or denoising diffusion probabilistic models (DDPMs):

linear drift $K(x(t), t) = -\frac{1}{2}k(t)x(t)$

G. Arts, D. E Habib, L. Wang, **K. Zhou**, **Mach. Learn.: Sci. Tecnol. 6(2025)025004**

Formal solution for forward process in DM

- forward process: $\boxed{\dot{x}(t) = K(x(t),t) + g(t)\eta(t)}$ $\quad 0 \le t \le T$

noise profile $g(t) = \sigma^{t/T}$

- backward process:

$$x'(\tau) = -K(x(\tau), T-\tau) + g^2(T-\tau)\underbrace{\partial_x \log P(x, T-\tau)}_{\text{score}} + g(T-\tau)\eta(\tau)$$

$\tau = T - t$

two main schemes:

- variance-expanding (VE): no drift $\quad K(x,t) = 0$
- variance-preserving (VP) or denoising diffusion probabilistic models (DDPMs):

linear drift $\quad K(x(t),t) = -\frac{1}{2}k(t)x(t)$

- initial data from target ensemble $\quad x_0 \sim P_0(x_0)$
- solution: $\boxed{x(t) = x_0 f(t,0) + \int_0^t ds\, f(t,s) g(s)\eta(s)}$
- with $\quad f(t,s) = e^{-\frac{1}{2}\int_s^t ds'\, k(s')}$

G. Arts, D. E Habib, L. Wang, **K. Zhou**,
Mach. Learn.: Sci. Tecnol. 6(2025)025004

Correlations evolution in forward process in DM

- moments $\mu_n(t) = \mathbb{E}[x^n(t)]$ and cumulants or connected n-point functions $\kappa_n(t)$ $\kappa_n = \mu_n - \sum_{m=2}^{n-2} \binom{n-1}{m-1} \kappa_m \mu_{n-m}$

- second moment/cumulant: (assume: first moment vanishes: $x_0 \to x_0 - \mathbb{E}_{P_0}[x_0]$)

$$\kappa_2(t) = \mu_2(t) = \mu_2(0) f^2(t,0) + \Xi(t)$$

$$\Xi(t) = \int_0^t ds \int_0^t ds' \, f(t,s) f(t,s') g(s) g(s') \mathbb{E}_\eta[\eta(s)\eta(s')] = \int_0^t ds \, f^2(t,s) g^2(s)$$

- higher-order moment and cumulants:

$$\kappa_3(t) = \mu_3(t) = \kappa_3(0) f^3(t,0)$$

$$\mu_4(t) = \mu_4(0) f^4(t,0) + 6\mu_2(0) f^2(t,0)\Xi(t) + 3\Xi^2(t)$$

$$\kappa_4(t) = \mu_4(t) - 3\mu_2^2(t)$$

$$\kappa_4(t) = \left[\mu_4(0) - 3\mu_2^2(0)\right] f^4(t,0) = \kappa_4(0) f^4(t,0)$$

variance-expanding scheme: no drift

$$f(t,0) = 1$$

higher cumulants conserved!

$$\boxed{\kappa_{n>2}(t) = \kappa_n(0) f^n(t,0)}$$

- in variance-expanding scheme ($f(t,0) = 1$, no drift): distribution at end of forward process as correlated as target distribution

Correlations evolution in forward process in DM

- moments $\mu_n(t) = \mathbb{E}[x^n(t)]$ and cumulants or connected n-point functions $\kappa_n(t)$ $\kappa_n = \mu_n - \sum_{m=2}^{n-2} \binom{n-1}{m-1} \kappa_m \mu_{n-m}$

- second moment/cumulant: (assume: first moment vanishes: $x_0 \to x_0 - \mathbb{E}_{P_0}[x_0]$)

$$\kappa_2(t) = \mu_2(t) = \mu_2(0) f^2(t,0) + \Xi(t)$$

$$\Xi(t) = \int_0^t ds \int_0^t ds'\, f(t,s) f(t,s') g(s) g(s') \mathbb{E}_\eta[\eta(s)\eta(s')] = \int_0^t ds\, f^2(t,s) g^2(s)$$

- higher-order moment and cumulants:

$$\kappa_3(t) = \mu_3(t) = \kappa_3(0) f^3(t,0)$$

$$\mu_4(t) = \mu_4(0) f^4(t,0) + 6\mu_2(0) f^2(t,0)\Xi(t) + 3\Xi^2(t)$$

$$\kappa_4(t) = \mu_4(t) - 3\mu_2^2(t)$$

$$\kappa_4(t) = \left[\mu_4(0) - 3\mu_2^2(0)\right] f^4(t,0) = \kappa_4(0) f^4(t,0)$$

variance-expanding scheme: no drift

$$f(t,0) = 1$$

higher cumulants conserved!

$$\boxed{\kappa_{n>2}(t) = \kappa_n(0) f^n(t,0)}$$

- $\kappa_{n>2}(t) = \kappa_n(0)\, e^{-\frac{n}{2}kt}$ (constant drift) $k(t) = k$, such that $f(t,0) = \exp(-kt/2)$

Generating functions – simple structures

- proof to all orders: generating functionals $Z[J]=\mathbb{E}[e^{J(t)x(t)}]$ $\quad W[J]=\log Z[J]$

- average over both noise and target distribution $Z_\eta[J]=\mathbb{E}_\eta[e^{J(t)x(t)}]=\dfrac{\int D\eta\, e^{-\frac{1}{2}\int_0^t ds\,\eta^2(s)+J(t)\left[x_0 f(t,0)+\int_0^t ds\, f(t,s)g(s)\eta(s)\right]}}{\int D\eta\, e^{-\frac{1}{2}\int_0^t ds\,\eta^2(s)}}$

- noise average: $Z_\eta[J]=e^{J(t)x_0 f(t,0)+\frac{1}{2}J^2(t)\Xi(t)}$

- total average: $Z[J]=\mathbb{E}[e^{J(t)x(t)}]=e^{\frac{1}{2}J^2(t)\Xi(t)}\displaystyle\int dx_0\, P_0(x_0)e^{J(t)x_0 f(t,0)}$

- cumulants: $W[J]=\log Z[J]=\dfrac{1}{2}J^2(t)\Xi(t)+\log\displaystyle\int dx_0\, P_0(x_0)e^{J(t)x_0 f(t,0)}$

- 2nd cumulant: $\kappa_2(t)=\dfrac{d^2W[J]}{dJ(t)^2}\Big|_{J=0}=\Xi(t)+\mathbb{E}_{P_0}[x_0^2]f^2(t,0)$ ✓

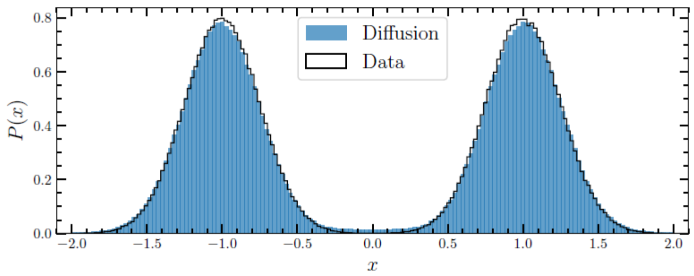
- higher-order cumulants: $\kappa_{n>2}(t)=\dfrac{d^nW[J]}{dJ(t)^n}\Big|_{J=0}=\dfrac{d^n}{dJ(t)^n}\log\mathbb{E}_{P_0}[e^{J(t)x_0 f(t,0)}]\Big|_{J=0}=\kappa_n(0)f^n(t,0)$ ✓

G. Arts, D. E Habib, L. Wang, **K. Zhou**, **Mach. Learn.: Sci. Tecnol. 6(2025)025004**

Toy model: double Gaussians

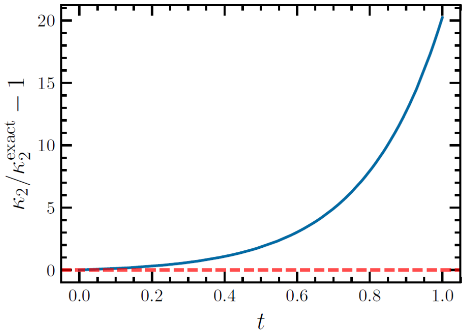
- sum of two Gaussians: $P_0(x) = \frac{1}{2}\left[\mathcal{N}(x;\mu_0,\sigma_0^2) + \mathcal{N}(x;-\mu_0,\sigma_0^2)\right]$
- moment-generating function: $Z[j] = \mathbb{E}\left[e^{jx}\right] = e^{\frac{1}{2}\sigma_0^2 j^2}\cosh(\mu_0 j)$
- cumulant-generating function: $W[j] = \frac{1}{2}\sigma_0^2 j^2 + \log\cosh(\mu_0 j)$
- only second cumulant depends on σ_0^2 :

$$\kappa_2 = \mu_0^2 + \sigma_0^2, \qquad \kappa_4 = -2\mu_0^4, \qquad \kappa_6 = 16\mu_0^6, \qquad \kappa_8 = -272\mu_0^8 \qquad \text{etc}$$

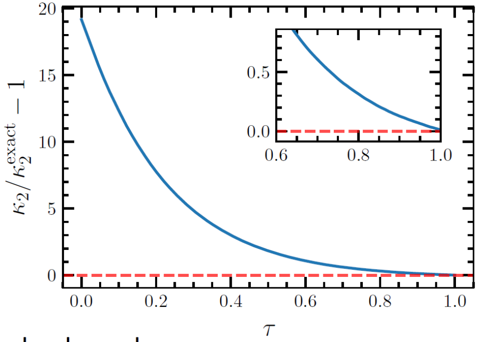


- variance-expanding scheme $\kappa_2(t) = \kappa_2(0) + \Xi(t)$ $\qquad \Xi(t) = \int_0^t ds\, g^2(s) \sim \sigma^{2t/T}$

$$\kappa_2(t) = \kappa_2^{\text{target}} + \frac{T}{\log\sigma^2}\left[\sigma^{2t/T} - 1\right] \sim \frac{T}{\log\sigma^2}\sigma^{2t/T}$$



forward



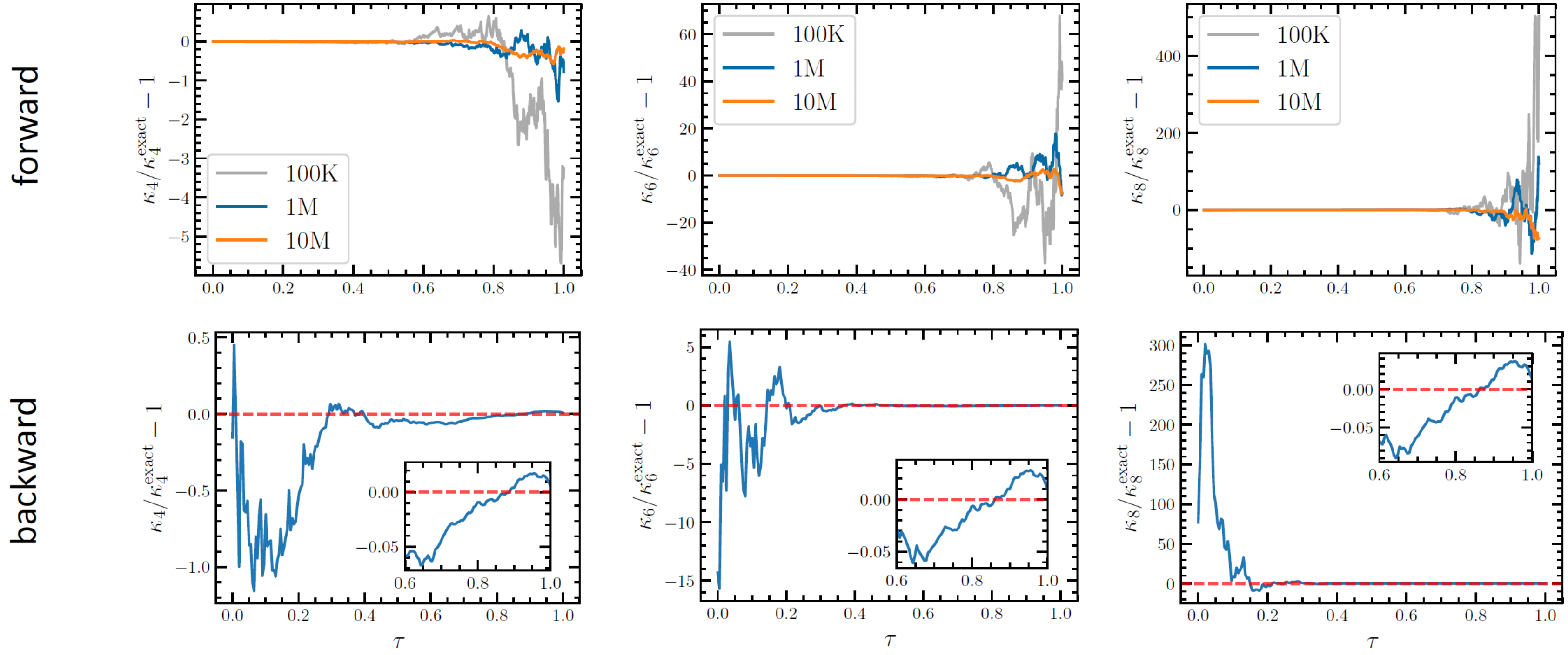
backward

G. Arts, D. E Habib, L. Wang, **K. Zhou**, **Mach. Learn.: Sci. Tecnol. 6(2025)025004**

Higher-order cumulants – approximately constant (incomplete cancelation)

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

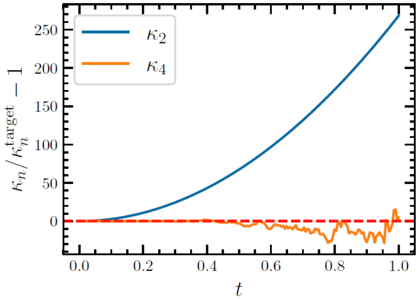
$$\kappa_{n>2}(t) = \kappa_n(0)$$



- score has higher-order cumulants encoded: cumulants are reconstructed

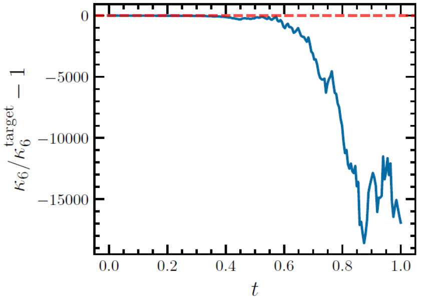
2d scalar field theory in Variance Expanding scheme

forward

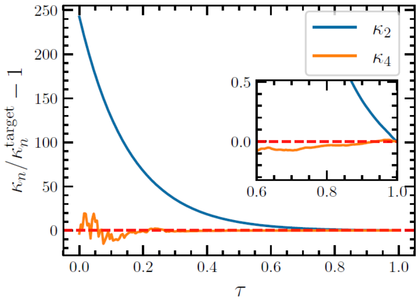


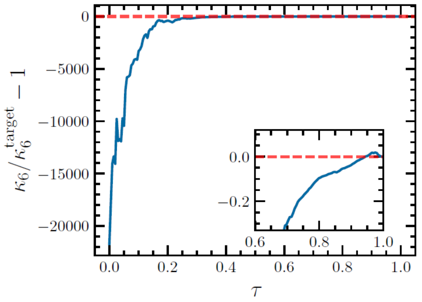
κ_2, κ_4

κ_6



backward





	κ_2	κ_4	κ_6	κ_8
HMC (normalised)	0.39597(4)	−0.29453(6)	0.90108(28)	−5.8689(25)
Diffusion model	0.39598(4)	−0.29454(7)	0.90113(32)	−5.8694(28)

$\phi^4: 32^2, \kappa = 0.4, \lambda = 0.022, 10^5$ configurations

Distribution from **Complex Langevin** samplings

- SQ works to complex Boltzmann weight one just extend $x \to z = x + iy$

$$\dot{z}(t) = K[z(t)] + \sqrt{2}\eta(t), \qquad K(z) = \frac{d}{dz}\log\rho(z) = -\frac{dS(z)}{dz}$$

- Take Real and Imaginary part and consider Complex Langevin process

$$\dot{x}(t) = K_x[x(t), y(t)] + \sqrt{2N_x}\eta_x(t), \qquad K_x(x,y) = \mathrm{Re}\frac{d}{dz}\log\rho(z)\Big|_{z\to x+iy},$$

$$\dot{y}(t) = K_y[x(t), y(t)] + \sqrt{2N_y}\eta_y(t), \qquad K_y(x,y) = \mathrm{Im}\frac{d}{dz}\log\rho(z)\Big|_{z\to x+iy},$$

- With the constraint $N_x - N_y = 1$ the F-P Eq. reads

$$\partial_t p(x,y;t) = \left[\partial_x\left(N_x\partial_x - K_x\right) + \partial_y\left(N_y\partial_y - K_y\right)\right]p(x,y;t)$$

- Thus $\langle O[x(t) + iy(t)]\rangle_\eta = \int dxdy\, p(x,y;t)O(x+iy)$

- Get a real and semi-positive definite distribution $\int dxdy\, p(x,y)O(x+iy) = \int dx\, \rho(x)O(x)$

G. Arts, D. Habibi, L. Wang, **K. Zhou, arXiv: 2510.01328**

Score and Energy-Based DM to learn the distribution

- Network represented score is essential for DM generating samples

$$\mathbf{s}_\theta(\mathbf{x},t) \approx \nabla \log p(\mathbf{x},t) \qquad \dot{\mathbf{x}}(t) = \frac{1}{2}\mathbf{K}[\mathbf{x}(t),t] - g^2(t)\mathbf{s}_{\theta*}(\mathbf{x},t) + g(t)\boldsymbol{\eta}(t),$$

- No guarantee that the score is conservative (to be integrable) numerically

$$\mathbf{s}_\theta(\mathbf{x},t) = \nabla\Phi_\theta(\mathbf{x},t) + \mathbf{r}_\theta(\mathbf{x},t), \qquad \nabla\cdot\mathbf{r}_\theta(\mathbf{x},t) = 0,$$

- The non-conservative component induce path-dependent integrated-score

$$\Phi_\theta(\mathbf{x},t) = \Phi_\theta(\mathbf{0},t) + \int_{\gamma:\mathbf{0}\to\mathbf{x}} d\mathbf{x}'\cdot\mathbf{s}_\theta(\mathbf{x}',t)$$

- For 2d case: decompose the score in a gradient and a curl

$$\mathbf{s}_\theta(\mathbf{x},t) = \nabla\Phi_\theta(\mathbf{x},t) + \nabla\times\mathbf{A}_\theta(\mathbf{x},t) \qquad \mathbf{A}_\theta(\mathbf{x},t) = (0,0,A_\theta(\mathbf{x},t)),$$

or

$$s_{x,\theta}(\mathbf{x},t) = \partial_x\Phi_\theta(\mathbf{x},t) + \partial_y A_\theta(\mathbf{x},t), \qquad s_{y,\theta}(\mathbf{x},t) = \partial_y\Phi_\theta(\mathbf{x},t) - \partial_x A_\theta(\mathbf{x},t).$$

- Poisson Eq.: $\nabla^2\Phi_\theta(\mathbf{x},t) = \nabla\cdot\mathbf{s}_\theta(\mathbf{x},t), \qquad \nabla^2 A_\theta(\mathbf{x},t) = -\left(\nabla\times\mathbf{s}_\theta(\mathbf{x},t)\right)_z$

G. Arts, D. Habibi, L. Wang, **K. Zhou**, **arXiv: 2510.01328**

Score and Energy-Based DM to learn the distribution

- Turn to **energy-based models** and impose directly $E_\theta(\mathbf{x},t) \approx -\log p(\mathbf{x},t)$

- Take **energy function** $E_\theta(\mathbf{x},t) = \frac{1}{2}||\mathbf{v}_\theta(\mathbf{x},t)||^2 \approx -\log p(\mathbf{x},t)$

- Leading to the approx. score which is conservative by construction

$$-\partial_i E_\theta(\mathbf{x},t) = -\mathbf{v}_\theta(\mathbf{x},t)\cdot\partial_i\mathbf{v}_\theta(\mathbf{x},t) \approx \partial_i \log p(\mathbf{x},t)$$

- Since requirement for differentiable energy function, choose smooth enough activation functions (e.g., SiLU/Mish).

- The need for additional derivatives makes it <u>more expensive</u>.

- End point for score in Backward Process

$$\begin{cases} \text{SBM:} \quad \mathbf{s}_\theta(\mathbf{x}) = \lim\limits_{t\to 0}\mathbf{v}_\theta(\mathbf{x},t), \\ \text{EBM:} \quad \mathbf{s}_\theta(\mathbf{x}) = -\nabla E_\theta(\mathbf{x}) = -\lim\limits_{t\to 0}\nabla E_\theta(\mathbf{x},t) = -\lim\limits_{t\to 0}\mathbf{v}_\theta(\mathbf{x},t)\cdot\nabla\mathbf{v}_\theta(\mathbf{x},t). \end{cases}$$

G. Arts, D. Habibi, L. Wang, **K. Zhou**, **arXiv: 2510.01328**

Complex-valued quartic model

- Quartic model with a complex mass $S = \frac{1}{2}\sigma_0 x^2 + \frac{1}{4}\lambda x^4, \qquad \sigma_0 = A + iB.$
- Exact results for partition function $Z = \int dx\, e^{-S(x)} = \sqrt{\frac{4\xi}{\sigma_0}} e^{\xi} K_{-\frac{1}{4}}(\xi) \qquad \xi = \sigma_0^2/(8\lambda)$
- Sampling from CL process yields empirical **histogram 2d-distribution**, with no analytical expression

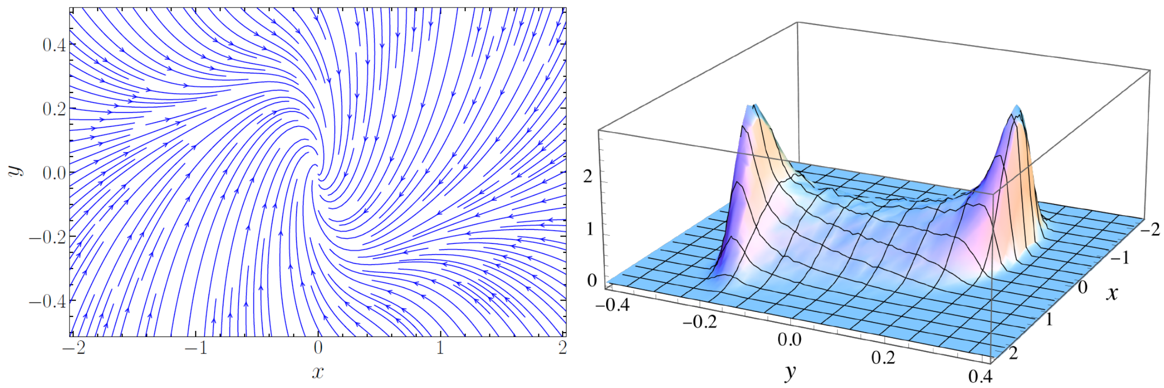


Figure 1: Complex-valued quartic model with parameters $\sigma_0 = 1 + i$ and $\lambda = 1$: CL drift in the complex plane (left) and histogram $P(x, y)$ obtained by sampling the CL process (right)

G. Arts, D. Habibi, L. Wang, **K. Zhou**, **arXiv: 2510.01328**

Complex-valued quartic model

- **Train DMs with CL sampled data**, the obtained score at the end of Backward process

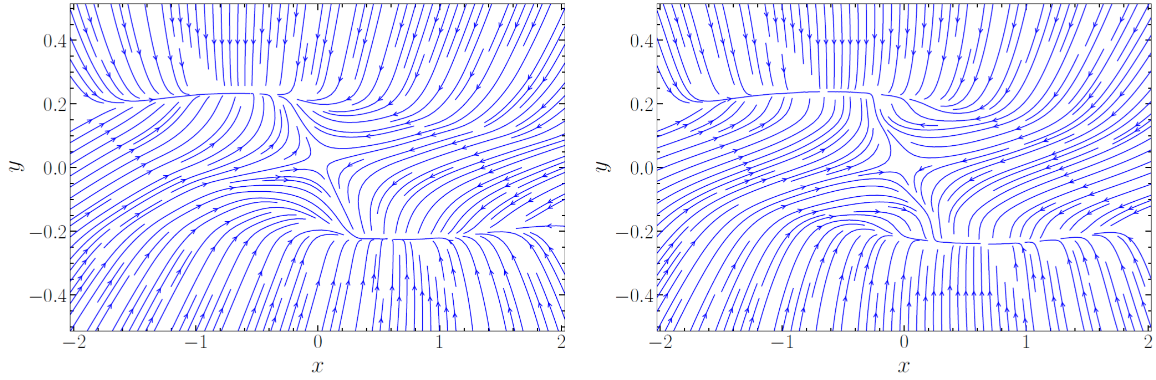


Figure 2: Learned scores in the quartic model at the end of the backward process, using a score-based (left) and energy-based (right) diffusion model.

- CL drift used in CL process with noise in x direction only, while score in DMs with noise in both directions
- CL drift is not integrable, and has attractive fixed point at the origin
- DM score shows saddle point at the origin, with two peaks position attractive

G. Arts, D. Habibi, L. Wang, **K. Zhou**, **arXiv: 2510.01328**

Complex-valued quartic model

- Drift in SBM is not conservative by analyzing the curl of the score, thus can not integrate the score directly

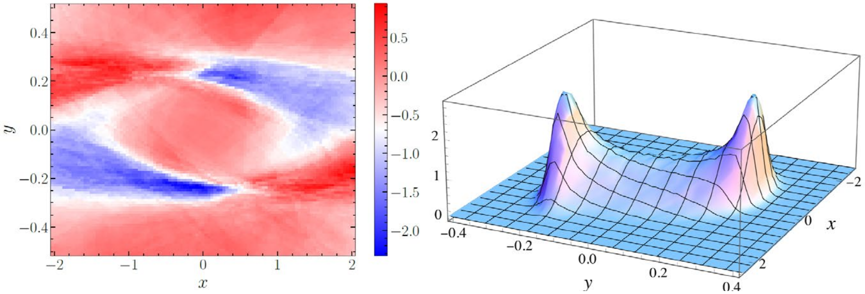


Figure 3: Quartic model using the score-based formulation: Curl of the score, averaged over 10 independently trained models (left) and histogram obtained by sampling data using the process learnt by the score-based model (right).

- EBM learned the Energy directly, the score (from gradient of energy) is conservative, with direct distribution

first time that a parametrization of distribution from a CL process is obtained in non-trivial case wo histrogram way

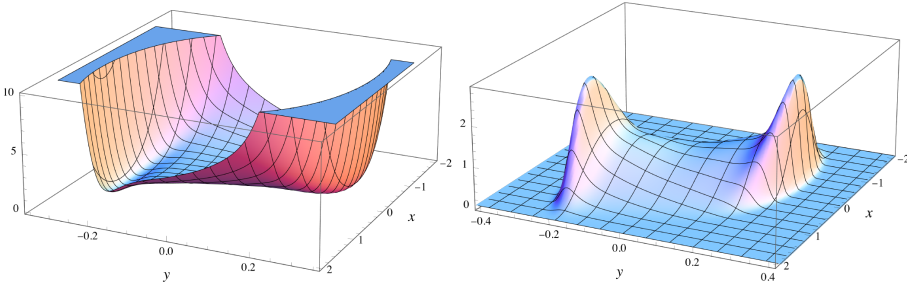
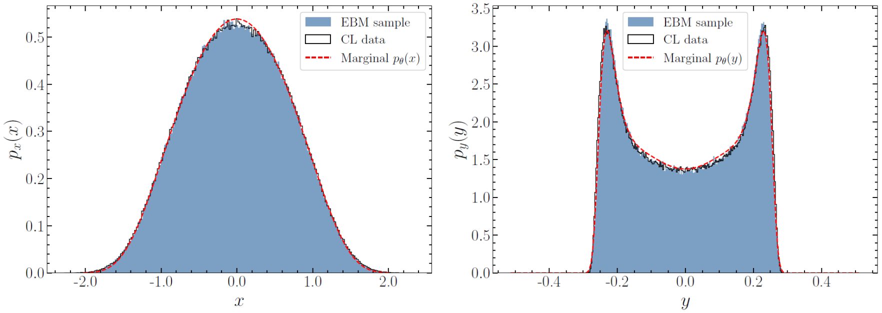


Figure 4: Quartic model: Energy $E_\theta(\mathbf{x})$ learned in the energy-based diffusion model (left) and the corresponding distribution $p_\theta(\mathbf{x}) \sim \exp[-E_\theta(\mathbf{x})]$ (right).

More quantitative comparison

- marginal distribution $p_x(x) = \int dy\, p_\theta(\mathbf{x}), \quad p_y(y) = \int dx\, p_\theta(\mathbf{x}), \quad p_\theta(\mathbf{x}) = \dfrac{\exp[-E_\theta(\mathbf{x})]}{\int dxdy\, \exp[-E_\theta(\mathbf{x})]}$



- Higher order moments and cumulants

	Re μ_2	Im μ_2	Re μ_4	Im μ_4
Exact	0.428142	-0.148010	0.423848	-0.280132
CL	0.4281(5)	$-0.1481(2)$	0.4232(11)	$-0.2798(6)$
SBM	0.4259(2)(5)	$-0.1473(1)(3)$	0.4237(4)(14)	$-0.2777(2)(9)$
EBM	0.4264(1)(37)	$-0.1487(1)(15)$	0.4192(2)(61)	$-0.2795(1)(39)$
MCMC-EBM	0.4254(2)(75)	$-0.1497(1)(31)$	0.4169(3)(122)	$-0.2802(2)(80)$
	Re μ_6	Im μ_6	Re μ_8	Im μ_8
Exact	0.580445	-0.587746	0.95105	-1.39336
CL	0.5787(26)	$-0.5866(18)$	0.9482(87)	$-1.3901(84)$
SBM	0.594(1)(4)	$-0.5882(1)(3)$	1.031(2)(14)	$-1.435(3)(14)$
EBM	0.569(1)(12)	$-0.5834(4)(11)$	0.918(2)(26)	$-1.374(2)(32)$
MCMC-EBM	0.565(1)(24)	$-0.584(1)(22)$	0.912(2)(49)	$-1.377(2)(61)$

MCMC-EBM: take energy function from EBM into MCMC

	Re κ_2	Im κ_2	Re κ_4	Im κ_4
Exact	0.428142	-0.148010	-0.060347	0.100083
CL	0.4280(5)	$-0.1480(2)$	$-0.0606(6)$	0.1003(5)
SBM	0.4259(2)(6)	$-0.1473(1)(3)$	$-0.0554(2)(4)$	0.0986(1)(4)
EBM	0.4273(2)(9)	$-0.1478(1)(2)$	$-0.0607(2)(3)$	0.1001(1)(4)
MCMC-EBM	0.4254(2)(76)	$-0.1497(1)(31)$	$-0.0602(2)(44)$	0.1030(1)(62)
	Re κ_6	Im κ_6	Re κ_8	Im κ_8
Exact	-0.00934	-0.19222	0.41578	0.5923
CL	$-0.009(1)$	$-0.194(2)$	0.414(5)	0.60(1)
SBM	$-0.0131(4)(7)$	$-0.1863(6)(11)$	0.423(2)(6)	0.557(3)(4)
EBM	$-0.0102(4)(6)$	$-0.193(1)(2)$	0.422(2)(5)	0.594(3)(8)
MCMC-EBM	$-0.0124(4)(32)$	$-0.205(1)(20)$	0.468(2)(51)	0.661(4)(81)

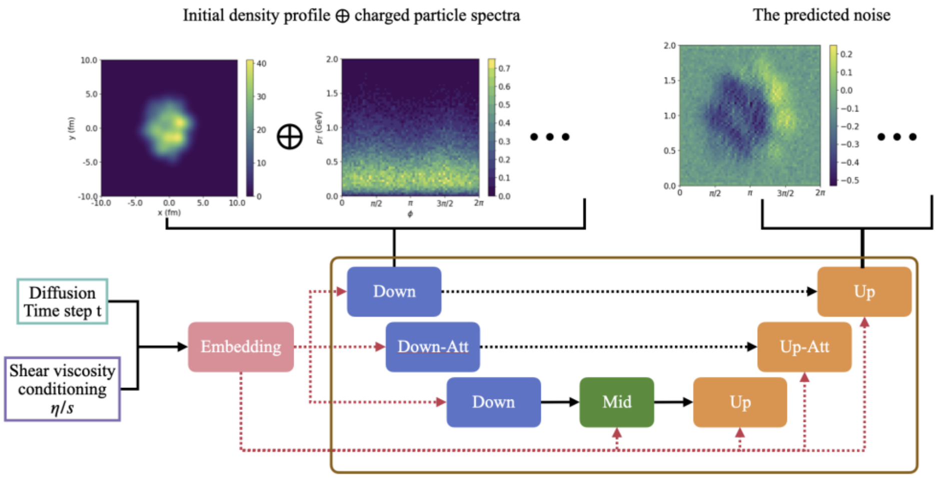
Generative diffusion model to heavy-ion collisions

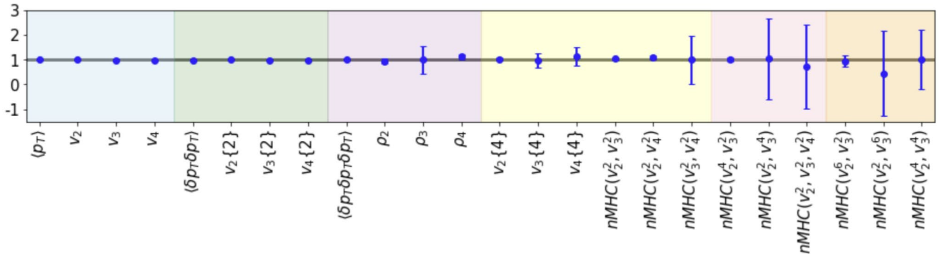
香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

An end-to-end generative diffusion model for heavy-ion collisions

arXiv:2410.13069, PRC(Letter)2025

Jing-An Sun,[1,2] Li Yan,[1,3] Charles Gale,[2] and Sangyong Jeon[2]





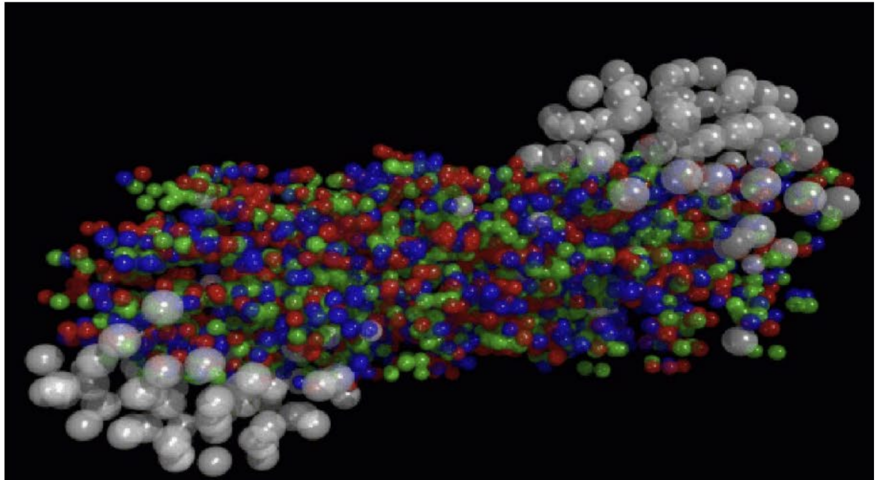
tor. We carried out (2+1)D minimum bias simulations of Pb-Pb collisions at 5.02 TeV, choosing the shear viscosity η/s to be one of three distinct values: 0.0, 0.1, and 0.2. For each value of η/s, we generate 12,000 pairs of initial entropy density profiles and final particle spectra, corresponding to 12,000 simulated events, as the training dataset. 70% of the total events are used for training and the rest are used for validation.

Considering that the spectra $\boldsymbol{S}_0$ depend on the initial entropy density profiles $\boldsymbol{I}$ and the shear viscosity η/s, we train a conditional reverse diffusion process $p(\boldsymbol{S}_0|\boldsymbol{I}, \eta/s)$ without modifying the forward process.

one single central collision event in just 10^{-1} seconds on a GeForce GTX 4090 GPU.

ble precision, as the traditional numerical simulation of hydrodynamics for one central event typically takes approximately 120 minutes (10^4 seconds) on a single CPU.

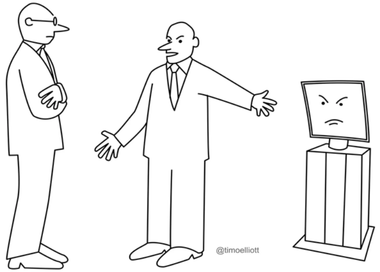
- Event-by-event collision output
- Microscopic non-equilibrium description
- hadrons on classical trajectories
  - stochastic binary scatterings
  - color string formation
  - resonance excitation and decays


https://itp.uni-frankfurt.de/~bleicher/urqmd-cern.gif

- interactions based on scattering cross sections
- default setup effective EoS: Hadron Resonance Gas
- Non-trivial interactions can be added through QMD approach

Can we emulate UrQMD with DL?

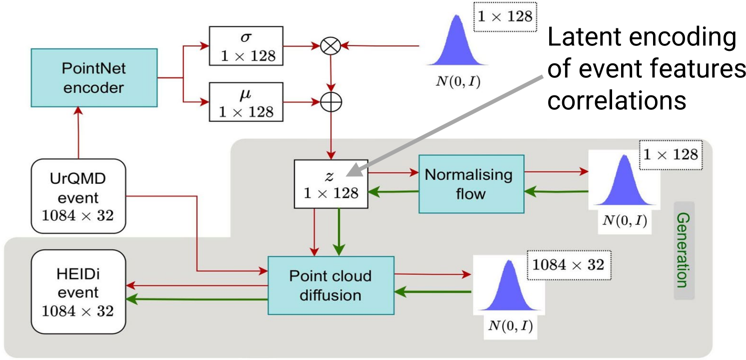
M. O.K, **K. Zhou**, J. S, H. S, **arXiv:2412.10352 , arXiv:2502.16330**

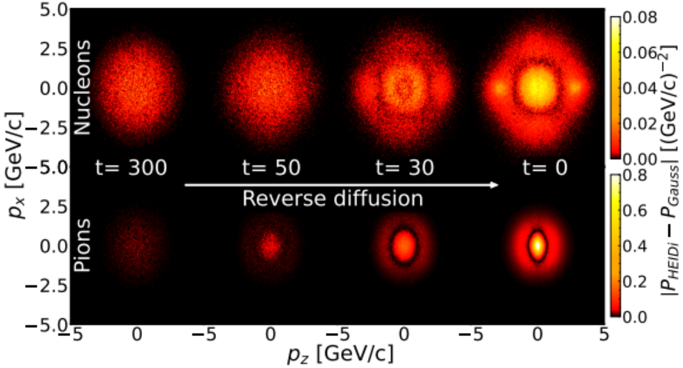


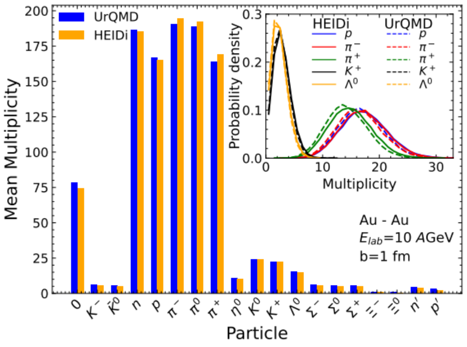
His decisions aren't any better than yours — but they're WAY faster...

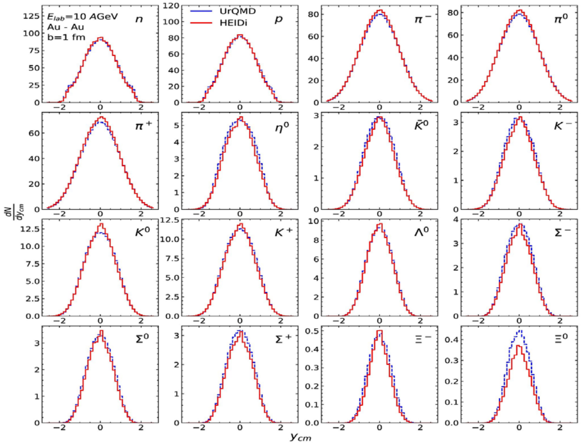
- 18k UrQMD simulation events for central Au-Au@10 AGeV collisions
- HEIDi: Heavy-ion Events through Intelligent Diffusion

PointNet encoder + Normalizing flow decoder + Pointcloud diffusion →



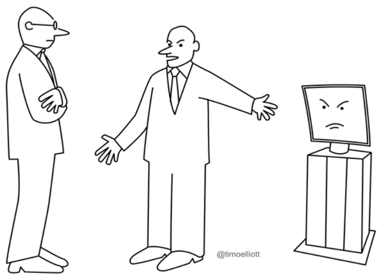






M. O.K, **K. Zhou**, J. S, H. S, arXiv:**2502.16330**, arXiv:**2412.10352**

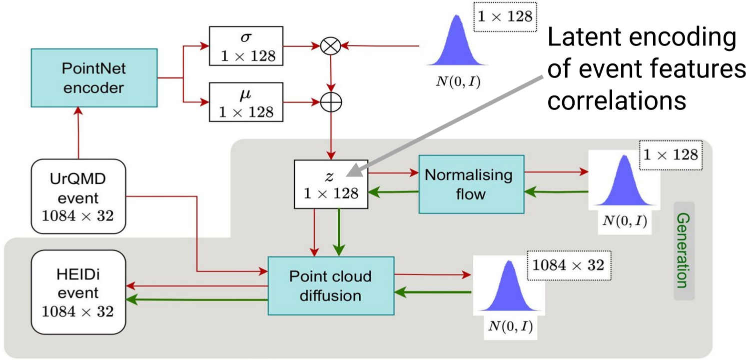
Point Cloud Diffusion Model for HICs – AI clone of simulation

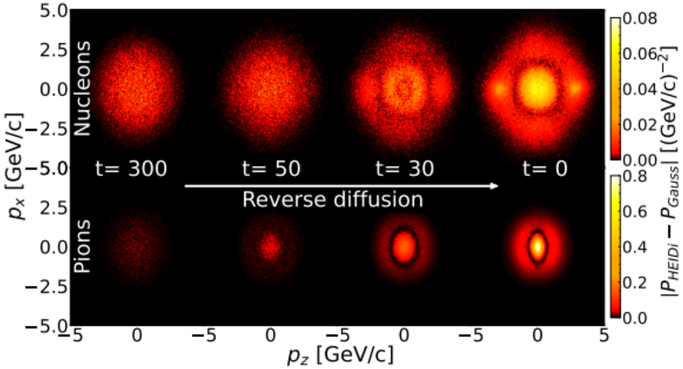


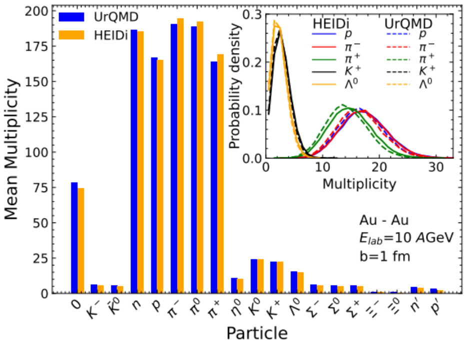
His decisions aren't any better than yours — but they're WAY faster...

- 18k UrQMD simulation events for central Au-Au@10 AGeV collisions
- HEIDi:
  Heavy-ion Events through Intelligent Diffusion

PointNet encoder + Normalizing flow decoder + Pointcloud diffusion →







- **Running time of** UrQMD simulation
  cascade : ~ 3 sec/event;
  with potential : ~ 3 min/event;
  hybrid : ~ 1 hour/event
- HEIDI on A100: ~ 30 ms/event
- Speedup **2 ~ 5 orders** of magnitude

M. O.K, **K. Zhou**, J. S, H. S, arXiv:**2502.16330**, arXiv:**2412.10352**

Summary

- Learn from Data: Generative AI can **learn underlying Hamiltonian/Action**
- Learn from Hamiltonian/Action: Generative AI can do **variational simulation**
- Diffusion model is with similarity to **Stochastic Quantization**,
  and learn a flow of effective action – RG
- **Higher-order cumulants** is preserved in variance-exploding scheme DM
- DM (SBM, EBM) can study the **distribution from complex Langevin process**
- Generative AI, e.g., DM is also helping Heavy-Ion Collisions **Fast Simulations**

Thanks !

A series **(Flow)** of invertible/bijective transformations for p(z)

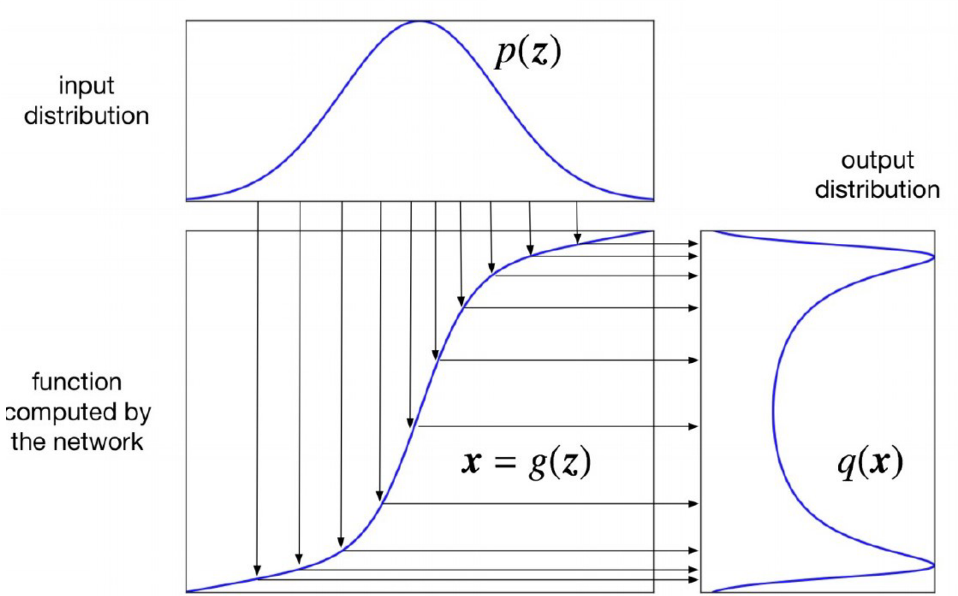
Normalizing : keep the probability to be normalized →

- Change of variable theorem

$$z \sim p(z)$$

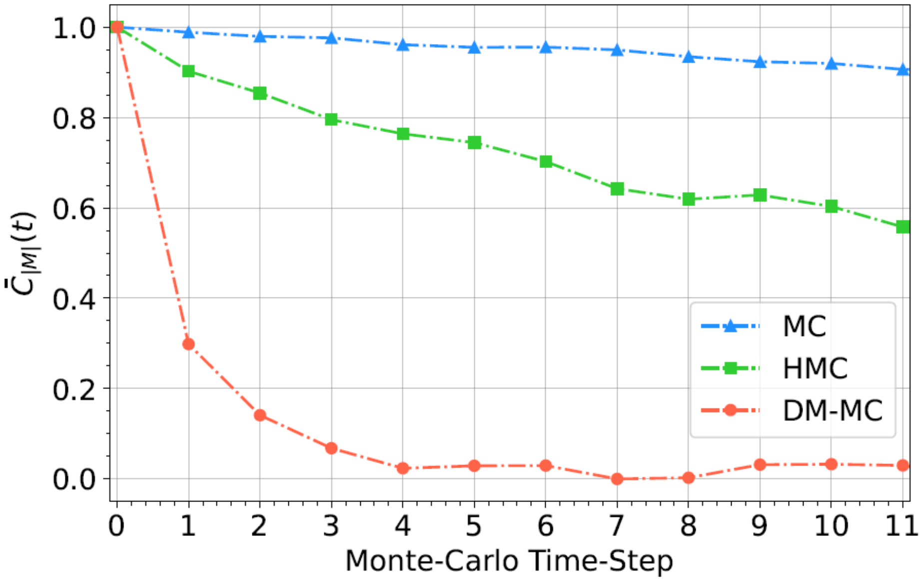
$$x = g(z)$$

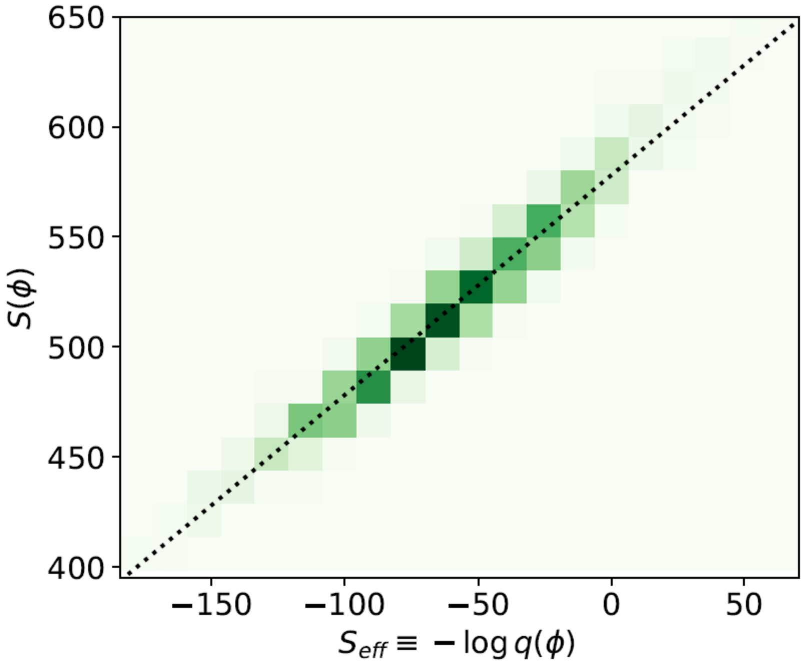
$$q(x) = p(z)\left|\frac{dz}{dx}\right|$$
$$= p(g^{-1}(x))\left|\frac{dg^{-1}(x)}{dx}\right|$$



Autocorrelation time and finally captured effective Action

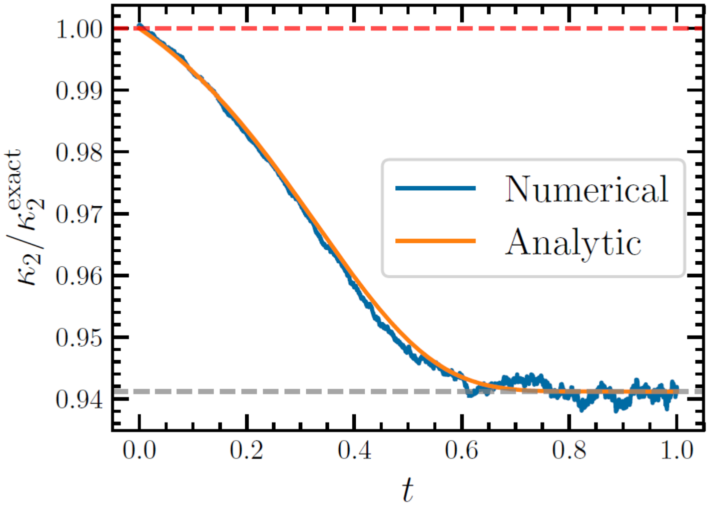
validation R2 ~0.96



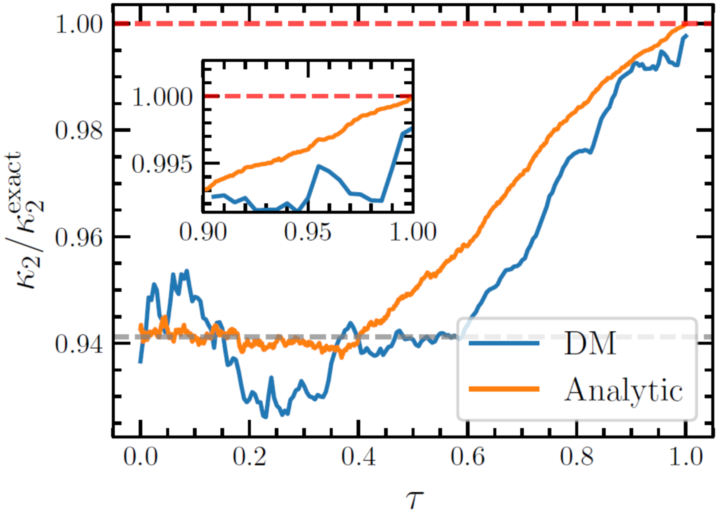


$$C_O(t) = \langle (O_{t_0} - \langle O_{t_0} \rangle)(O_{t_0+t} - \langle O_{t_0+t} \rangle) \rangle = \langle O_{t_0} O_{t_0+t} \rangle - \langle O_{t_0} \rangle \langle O_{t_0+t} \rangle$$

- variance-preserving scheme $\kappa_2(t) = \mu^2(t) + \sigma^2(t) = \left(\mu_0^2 + \sigma_0^2 - 1\right) f^2(t,0) + 1$



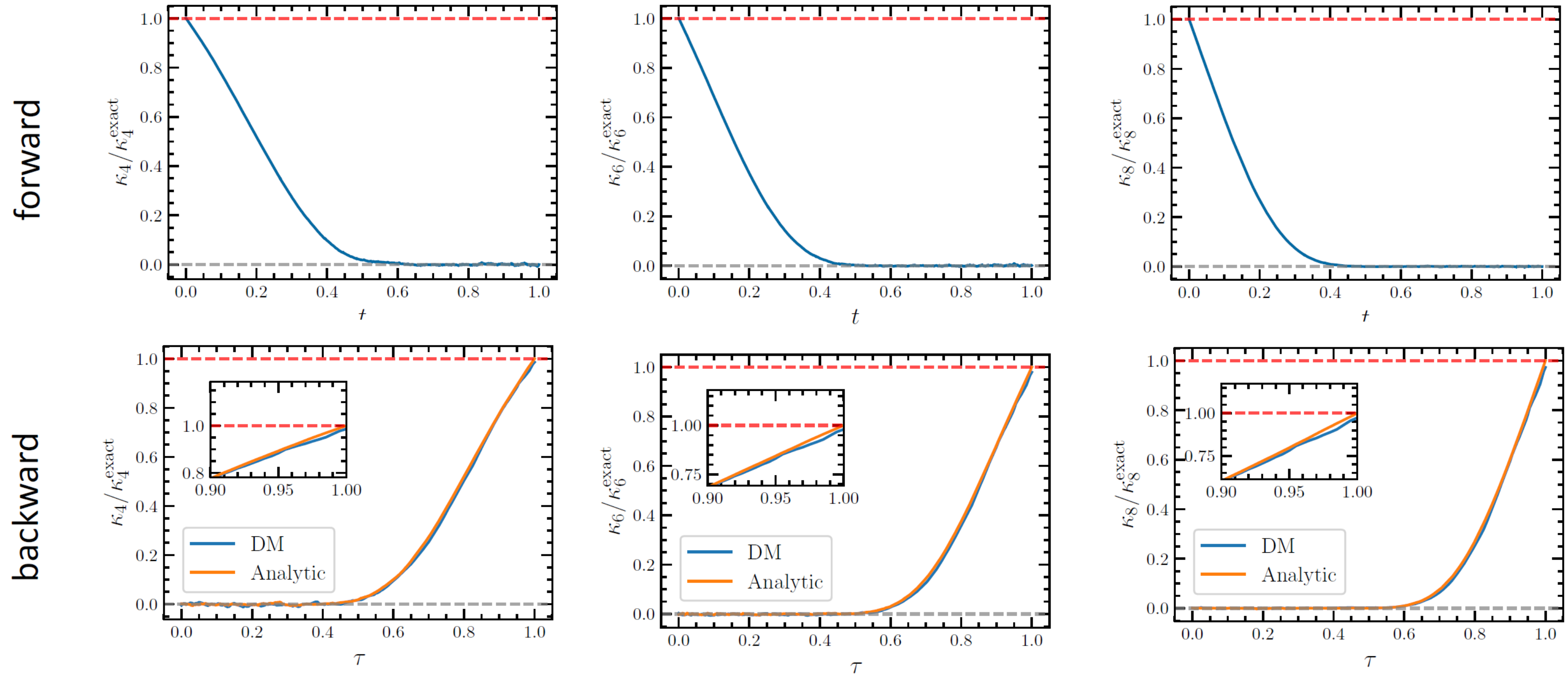
forward



backward

Higher-order cumulants in DDPM

$$\kappa_{n>2}(t) = \kappa_n(0) f^n(t,0)$$



Comparison of the two schemes

	κ_2	κ_4	κ_6	κ_8
Exact	1.0625	-2	16	-272
Data	1.0624(5)	$-2.000(2)$	16.00(2)	$-272.0(6)$
Variance expanding	1.0692(6)	$-2.001(2)$	16.03(3)	$-272.7(6)$
Variance preserving (DDPM)	1.0609(5)	$-1.976(2)$	15.72(2)	$-265.6(6)$

expectation values at the end of the backward process

✓ variance-expanding scheme slightly outperforms variance-preserving

A collision event output

- UrQMD outputs a list of final state hadrons along their momentum info
- Pointclouds: ideal representation

- Consider Au-Au 10 AGeV, impact parameter b=1 fm
  - An event= 1084 X 32
  - Empty rows=0,0,0,0,...
  - p_x, p_y, p_z, One hot encoded PID
  - 26 hadron species, spectator nucleons, empty particles

$$\mathbf{X}^{(0)} = \{\mathbf{x}_i^{(0)}\}_{i=1}^{1084}$$

$$\mathbf{x}_i^{(0)} = \{\mathbf{p}_i^{(0)}, \mathrm{ID}_i^{(0)}\},$$

$$\mathbf{p}_i^{(0)} = (p_{x_i}^{(0)}, p_{y_i}^{(0)}, p_{z_i}^{(0)})$$

M. O.K, **K. Zhou**, J. S, H. S, **arXiv:2412.10352, arXiv:2502.16330**